MULTI-BEHAVIOR DYNAMIC CONTRASTIVE LEARNING FOR RECOMMENDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Dynamic behavior modeling has become an essential task in personalized recommender systems for learning the time-evolving user preference in online platforms. However, most next-item recommendation methods follow the single type behavior learning manner, which notably limits their user representation performance in reality, since the user-item relationships are often multi-typed in real-life applications (e.g., click, tag-as-favorite, review and purchase). To offer better recommendations, this work proposes Evolving Graph Contrastive Memory Network (EGCM) to model dynamic interaction heterogeneity for multi-behavior sequential recommendation. Specifically, we first develop a multi-behavior graph encoder to capture the short-term preference heterogeneity, and preserve the dedicated relation semantics for different types of user-item interactions. In addition, we design a dynamic cross-relational memory network, empowering EGCM to distill the long-term multi-behavior preference of users and the underlying evolving cross-type behavior dependencies over time. To enhance the user representation with multi-behavior commonality and diversity, we design a multi-behavior contrastive learning paradigm with heterogeneous short- and long-term interest modeling. Experiments on several real-world datasets show the superiority of our recommender system over various state-of-the-art baselines.

1 INTRODUCTION

Learning user's dynamic preference plays a vital role in recommender systems to predict the next items that users may be interested in Wang et al. (2019). The recent advances of neural network architectures has inspired many efforts to model the transitions between temporally-ordered items, due to the strong representation capability of deep learning techniques, *e.g.*, recurrent neural encoder Hidasi et al. (2016), convolution-based model Tang & Wang (2018) and attention mechanism Kang & McAuley (2018). More recent sequential recommender systems are built upon the Transformer Sun et al. (2019); Liu et al. (2021b) or Graph Neural Networks (GNNs) Wu et al. (2019); Ma et al. (2020); Wang et al. (2020c) to provide state-of-the-art recommendation performance. Despite their effectiveness, most of existing next-item recommendation approaches rely on only single type of user-item interaction (*e.g.*, click or purchase data), and thus are limited to capture the item-level multi-behavior interaction patterns.

In real-life recommendation scenarios, users often interact with items in various ways, based on their interests which are intrinsically time-evolving and diverse. For instance, different types of user behaviors (*e.g.*, page view, add-to-favorite, purchase) in online retailers may reflect diverse user intentions and heterogeneous user-item relationships Guo et al. (2019); Jin et al. (2020b). Leaving this fact untouched, single type of behavior modeling in previous user embedding functions is insufficient to comprehensively capture diverse user intents with behavior heterogeneity Xia et al. (2021a). Hence, multi-behavior representations can characterize the various latent factors behind user-item interactions, and maintain dedicated embedding space for different types of user behaviors in recommender systems.

While having realized the importance of modeling behavior-aware user-item relationships in recommendation, some key challenges remain to be carefully tackled. Specifically, (1) How to explicitly preserve the behavior-specific semantics pertinent to each type of user-item interactions in a dynamic manner, is not trivial in multi-behavior sequential recommendation. It is critical for the recommender to distill such heterogeneous item-level dependencies with the jointly modeling of short-term and

long-term user interests. (2) Learning informative and robust representations of multiplex user-item interactions requires a tailored modeling with a performant recommendation paradigm, which towards the encoding of users' multi-behavior commonality and diversity. While we can embed behavior-specific semantics into individual latent vectors, the understanding of multi-behavior commonality underlying global view of user-specific dynamic preference is critical to multi-behavior modeling.

Contributions. In light of these challenges, we propose a Evolving Graph Contrastive Memory (EGCM) framework that can effectively distill the heterogeneous user intentions from multi-behavior data in recommendation. Specifically, we first introduce a multi-behavior graph encoder equipped with temporal context embedding for modeling the behavior-aware short-term interests of users. Furthermore, a dynamic cross-relational memory network is proposed to incorporate heterogeneous item-item dependencies into learning user dynamic preferences with cross-behavior relational transitions. In a nutshell, this dynamic multi-behavior modeling allows us to characterize diverse user intents from the long-term perspective behind the interacted item sequence in next-item recommendation. To enhance the generalizability and robustness of our recommender, we design our multi-behavior contrastive learning paradigm to endow our EGCM with the capability of encoding multi-behavior commonality and differentiating the behavior-aware preference of various users.

To summarize, the key contributions of this work are presented as follows:

- Emphasizing the importance of learning the dynamics of the user-item relation heterogeneity from users' multi-behavior data in making next-item recommendations.
- Proposing a new model EGCM, which integrates the dynamic cross-relational dependency modeling with the multi-behavior contrastive learning paradigm, so as to distill the evolving user-item relationships at the fine-grained level of user preferences. In addition, we perform the theoretical analysis of our proposed EGCM model as presented in the Supplementary Section.
- Conducting experiments on three real-world datasets to demonstrate the superiority of EGCM. Further ablation studies and in-depth model analysis justify the rationality of our model design. To support the reproducibility of our experimental results, the model implementations can be found at the anonymous link: https://anonymous.4open.science/r/EGCM.

2 Related Work

Next-item/Sequential Recommendation. Early studies (*e.g.*, FPMC Rendle et al. (2010)) rely on the Markov chain to tackle the sequential recommendation problem. Many recent efforts have been devoted to learning users' dynamic interests with various neural network encodes, such as the RNN-based method GRU4Rec Hidasi et al. (2016) and CNN-based approach Caser Tang & Wang (2018). In addition, several self-attention relational learning models are introduced to estimate the item correlations, *e.g.*, SASRec Kang & McAuley (2018), BERT4Rec Sun et al. (2019) and TiSASRec Li et al. (2020). Inspired by the strength of graph neural networks, some recent sequential recommender systems are built over the graph-based message passing scheme to encode the multi-order dependencies among items, including MA-GNN Ma et al. (2020), SURGE Chang et al. (2021), and GCE-GNN Wang et al. (2020c). Furthermore, self-supervised learning has been used in recent sequential recommendation methods for data augmentation, like COTREC Xia et al. (2021c) and DHCN Xia et al. (2021c). However, most of existing methods are built on single type of interactions and ignore the heterogeneous behavior-aware user preferences.

Multi-Behavior Recommender System. Recently, multi-behavior recommendation has gained considerable attention due to the effectiveness of considering multi-typed user behaviors in boosting the recommendation performance Chen et al. (2021). For example, NMTR Gao et al. (2019) and DIPN Guo et al. (2019) differentiating behavior semantics with multi-task learning schemes. To encode diverse relationships between users and items, some recent studies (*e.g.*, MBGCN Jin et al. (2020b), KHGT Xia et al. (2021a) and MBGMN Xia et al. (2021b)) attempt to leverage graph neural networks for encoding the multi-behavior patterns, based on their constructed relation-aware heterogeneous user-item interaction graph. One major drawback of existing multi-behavior recommender systems is that they mostly focus on the stationary scenarios, while neglect the time-evolving multi-behavior dependencies from diverse user interest representation.

Contrastive Representation Learning. As a prevalent technique, contrastive learning has shown great potential for data augmentation with auxiliary self-supervised signals Liu et al. (2021a); Ho & Nvasconcelos (2020). For image analysis, many contrastive learning methods have been proposed to advance the modeling image data with different augmentation techniques Aberdam et al. (2021);

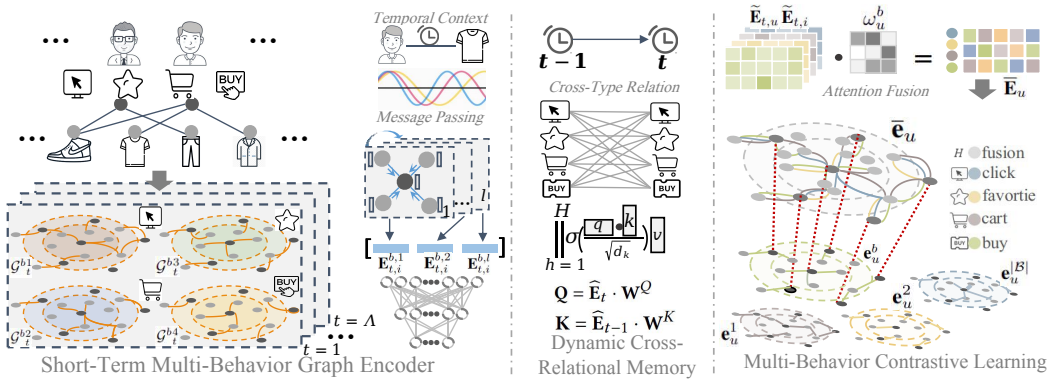


Figure 1: The model flow of EGCM framework. The multi-behavior contrastive learning module (right side) augments the graph-enhanced dynamic memory network (left & middle side) with the auxiliary behavior-aware self-supervision signals from both short-term and long-term user interests.

Verma et al. (2021); Tian et al. (2020b), such as cropping, horizontal translations and rotations. In addition, cross-view contrastive learning has provided the state-of-the-art graph representation performance with various augmentation operators, *e.g.*, node shuffles in DGI Velickovic et al. (2019), centrality-aware edge dropout in GCA Zhu et al. (2021) and subgraph sampling in MVGLR Hassani & Khasahmadi (2020). Our work is inspired by the above work and designs a new multi-behavior contrastive learning paradigm to simultaneously capture the user-specific behavior commonalities and cross-user behavior diversity.

3 METHODOLOGY

We present the technical details of our EGCM model which consists of three key components: i) Short-term multi-behavior graph encoder that captures the user's short-term interests with multi-relational graph neural networks; ii) Long-term user interest modeling which learns time-evolving multi-behavior preferences across different time slots; iii) Multi-behavior self-supervised learning which enhances the user representation with the behavior-aware instance self-discrimination.

3.1 SHORT-TERM MULTI-BEHAVIOR GRAPH ENCODER

To capture the dedicated behavior semantics for underlying user's short-term interests, we propose a multi-relational graph encoder to handle the heterogeneous item dependencies within each time slot.

3.1.1 Short-Term Multi-Behavior Graph

We define the short-term multi-behavior graph $\mathcal{G}_t^b = (\mathcal{V}_t^b, \mathcal{E}_t^b, \mathbf{M}_t^b)$ to represent the behavior-aware user-item interactions within the t-th time slot (*e.g.*, day, week, or month). In graph $\mathcal{G}_t^b$, nodes $\mathcal{V}_t^b = \mathcal{V}_{t,u}^b \cup \mathcal{V}_{t,i}^b$, and edges $\mathcal{E}_t^b$ represent the interactions between user nodes $\mathcal{V}_{t,u}^b$ and item nodes $\mathcal{V}_{t,i}^b$. Here, Λ_u denotes the total number of time slots for the multi-behavior interaction sequence S_u of user u. Given the set of edges $\mathcal{E}_t^b$, we define the user-item interaction matrix $\mathbf{M}_t^b$, where each entry $M_{t,(u,i)}^b = 1$ if user u has adopted item i under the behavior type of b within the t-th time slot.

3.1.2 Behavior-aware Message Passing

Inspired by recent efforts of utilizing graph neural network to model relational user data He et al. (2020); You et al. (2020), we design a behavior-aware message passing schema to capture the high-order dependencies among items through the recursively embedding propagation across graph layers. Formally, our behavior-aware message passing function from the (l-1)-th layer to the (l)-th layer is defined with the following form:

$$\begin{aligned} \mathbf{E}_{t,i}^{b,(l)} &= \sigma\left(\Gamma(\mathbf{M}_t^b \cdot \mathbf{M}_t^{b^T}) \cdot \mathbf{E}_{t,i}^{b,(l-1)} \cdot \mathbf{W}_{t,i}^{b,(l)}\right) \\ \Gamma(\mathbf{M}_t^b \cdot \mathbf{M}_t^{b^T}) &= (\mathbf{D}_t^b)^{-\frac{1}{2}} \cdot \mathbf{M}_t^b \cdot (\mathbf{B}_t^b)^{-1} \cdot \mathbf{M}_t^{b^T} \cdot (\mathbf{D}_t^b)^{-\frac{1}{2}} \end{aligned} \tag{1}$$

where $\mathbf{E}_{t,i}^{b,(l)}$ is the representation of item i at the l-th graph propagation layer at the t-th time slot. Here, we define M and N to denote the number of users and items, respectively, *i.e.*, $M = |\mathcal{V}_u|$, $N = |\mathcal{V}_i|$. $\mathbf{W}_{t,i}^{b,(l)} \in \mathbb{R}^{d\times d}$ is the learnable transformation matrix. During the message passing process, we adopt the PReLU He et al. (2015) as the activation function $\sigma(\cdot)$ to inject non-linearities. By stacking multiple embedding propagation layers, the behavior-specific high-order item dependencies can be preserved in the generated representations $\mathbf{E}_{t,i}^{b,(L)}$, where L denotes the number of propagation layers.

Additionally, to alleviate the large value effects of embeddings during the recursive propagation Wang et al. (2020b), we incorporate the graph Laplacian normalized function of eigenvectors Kipf & Welling (2016) $\Gamma(\cdot)$ into the message passing function. In particular, we generate two diagonal degree matrices $\mathbf{D}_t^b \in \mathbb{R}^{N\times M}$ and $\mathbf{B}_t^b \in \mathbb{R}^{M\times N}$ based on the interaction matrix $\mathbf{M}_t^b$. Entries in $\mathbf{D}_t^b$ and $\mathbf{B}_t^b$ matrices are generated as follows:

$$D_{t,(n,n)}^b = \sum_{n=1}^{|\mathcal{V}_{t,i}^b|} \mathbf{M}_{t,(n)}^b; \qquad B_{t,(m,m)}^b = \sum_{m=1}^{|\mathcal{V}_{t,u}^b|} (\mathbf{M}_{t,(m)}^b)^T \tag{2}$$

where $D_{t,(n,n)}^b$, $B_{t,(m,m)}^b$ denotes the elements of n-th, m-th row of these diagonal matrices.

Cross-Layer Aggregation. To aggregate the embeddings encoded from each graph layer, we first conduct the concatenation operation over all layer-specific embeddings, and then feed the concatenated representation into a feed-forward network with the trainable projection matrix $\mathbf{W}_{t,cat}^b \in \mathbb{R}^{(l\times d)\times d}$ to generate the aggregated item representations $\mathbf{E}_{t,i}^b$ as:

$$\mathbf{E}_{t,i}^b = \left[\mathbf{E}_{t,i}^{b,1} : \mathbf{E}_{t,i}^{b,2} : ... : \mathbf{E}_{t,i}^{b,l}\right] \cdot \mathbf{W}_{t,cat}^b; \quad \mathbf{E}_{t,i}^b = \frac{\mathbf{E}_{t,i}^b}{||\mathbf{E}_{t,i}^b||} \tag{3}$$

With the above cross-layer aggregation, the fused emebddings $\mathbf{E}_{t,i}^b$ preserves the behavior-aware short-term user preference at the t-th time slot under the behavior type of b.

3.1.3 User Short-Term Interest Representation

After encoding the item embeddings by exploring the high-order dependencies among items, we generate the user representations which maintain the behavior-specific short-term interests. Given the short-term multi-behavior graph $\mathcal{G}_t^b$, user representations can be generated by aggregating embeddings propagated from the interacted items. The embedding propagation between item i to user u is formally presented with the following form:

$$\mathbf{m}_{u_t\leftarrow i_t}^b = f(\mathbf{e}_{i_t}^b, \mathbf{e}_{u_t}^b) = \frac{1}{\sqrt{|\mathbf{B}_{t,(u)}^b||\mathbf{D}_{t,(i)}^b|}} \cdot \mathbf{W}_{t,u}^b \cdot \mathbf{e}_{i_t}^b \tag{4}$$

where $\mathbf{e}_{i_t}^b \in \mathbb{R}^{d\times 1}$ represents the item embedding within the t-th time slot under the behavior type of b. $\mathbf{W}_{t,u}^b$ is the transformation matrix with the dimension of $\mathbb{R}^{d\times d}$. For the short-term interest representation of all users, the message passing scheme can be presented with the matrix form as:

$$\mathbf{E}_{t,u}^b = \sigma(\Gamma(\mathbf{M}_t^{b^T}) \cdot \mathbf{E}_{t,i}^b \cdot \mathbf{W}_{t,u}^b); \quad \mathbf{E}_{t,u}^b = \frac{\mathbf{E}_{t,u}^b}{||\mathbf{E}_{t,u}^b||}; \;\; \Gamma(\mathbf{M}_t^{b^T}) = (\mathbf{B}_t^b)^{-\frac{1}{2}} \cdot \mathbf{M}_t^{b^T} \cdot (\mathbf{D}_t^b)^{-\frac{1}{2}} \tag{5}$$

We also leverage the $\Gamma(\cdot)$ as the normalization function for user representation $\mathbf{E}_{t,u}^b$ corresponding to users' short-term interests at the t-th time slot under the behavior type of b.

3.1.4 Temporal Context Injection

To model the dynamic nature of user-item interactions, we propose to inject the temporal contextual signals into the user representation, so as to capture the dynamic item dependencies in the short-term multi-behavior graph $\mathcal{G}_t^b$. In light of this, our temporal context encoder is built over the positional encoding technique in Transformer architecture Vaswani et al. (2017). Specifically, we introduce the sinusoid functions to generate the relative time embedding for each user-item interaction edge. By doing so, we enhance our short-term multi-behavior graph encoder with the capability of capturing behavior-aware interaction dynamics of users with the temporal context injection.

3.1.5 Priori-aware Embedding Initialization

In our short-term multi-behavior graph encoder, we rely on the time slot-specific user-item interaction data, which may lead to overfitting issue. In light of this limitation, we develop a priori-aware embedding initialization strategy which injects the representations of previous time slot $(t-1)$ into the embedding initialization of current time slot t, in order to reduce the training difficulty and deterioration Glorot & Bengio (2010). Formally, the priori-aware embedding initialization is defined as follows:

$$\mathbf{E}_{t,i}^{b,0} = \left(\zeta * \mathbf{E}_{t,i}^{b,0} + (1-\zeta) * \mathbf{E}_{t-1,i}^{b}\right) \cdot \mathbf{W}_{\zeta}^{b} \tag{6}$$

where $\mathbf{E}_{t,i}^{b,0}$ denotes the original input embeddings for the short-term multi-behavior graph $\mathcal{G}_t^b$ at the t-th time slot. $\mathbf{E}_{t-1,i}^{b}$ represents the representations encoded from the previous $(t-1)$ time slot. Here, ζ and $\mathbf{W}_{\zeta}^{b}$ are learnable aggregation weight and projection matrix, respectively.

3.2 Dynamic Cross-Relational Memory Network

In the multi-behavior sequential recommender system, long-term multi-behavior dynamics reflect a holistic view of diverse preferences of users across different time slots. To capture such global multi-behavior interests of users, we propose a dynamic cross-relational memory network to learn the evolving cross-type behavior dependencies over time in a long-term manner.

3.2.1 Multi-Behavior Embedding Tensor

After encoding users' behavior-aware short-term interest with the user ($\mathbf{E}_{t,u}^{b}$) and item ($\mathbf{E}_{t,i}^{b}$) representations, we construct a three-way tensor $\widehat{\mathbf{E}}_{t,u} \in \mathbb{R}^{M \times d \times B}$ and $\widehat{\mathbf{E}}_{t,i} \in \mathbb{R}^{N \times d \times B}$ to represent the multi-behavior short-term interest by stacking type-specific user and item embeddings as follows:

$$\widehat{\mathbf{E}}_{t,u} = stack\left(\mathbf{E}_{t,u}^{1}, \mathbf{E}_{t,u}^{2}, ..., \mathbf{E}_{t,u}^{B}\right); \;\; \widehat{\mathbf{E}}_{t,i} = stack\left(\mathbf{E}_{t,i}^{1}, \mathbf{E}_{t,i}^{2}, ..., \mathbf{E}_{t,i}^{B}\right) \tag{7}$$

where B denotes the number of behavior types $B = |\mathcal{B}|$. Hence, we can associate each time slot t with the a multi-behavior embedding tensor $\widehat{\mathbf{E}}_{t,u}$ and $\widehat{\mathbf{E}}_{t,i}$ for users and items, respectively.

3.2.2 Time-Evolving Cross-Type Behavior Dependencies

In a dynamic multi-behavior scenario, different types of user behaviors are inter-dependent among time slots in a long-term perspective. For example, customers may first view some products in online retail sites, and make their purchase decisions one day after. To capture such evolving cross-type behavior dependencies across time slots, we develop a dynamic cross-relational memory network, to explicitly learn the influence weights between time slot-specific behavior-aware representations.

Towards this end, we design a self-attention-based memory network to model the embedding correlations (*e.g.*, $\mathbf{E}_{(t-1),u}$–$\mathbf{E}_{t,u}$) between adjacent time slots $(t-1)$ and t. In particular, we take the embedding $\mathbf{E}_{t,u}$ as the input for query embeddings $\mathbf{Q}_t$, and $\mathbf{E}_{(t-1),u}$ as the inputs corresponding to the key and value embeddings $\mathbf{K}_{t-1}$ and $\mathbf{V}_{t-1}$. Formally, our designed dynamic cross-relational memory encoder $\gamma(\cdot)$ is defined with the following scaled dot-product attention function over different time slot-specific behavior representations as follows:

$$\mathbf{Z}_{t,t-1} = \gamma(\widehat{\mathbf{E}}_t \cdot \mathbf{W}_t^Q, \widehat{\mathbf{E}}_{t-1} \cdot \mathbf{W}_t^K, \widehat{\mathbf{E}}_{t-1}); \;\; \mathbf{Q}_t = \widehat{\mathbf{E}}_t \cdot \mathbf{W}_t^Q; \;\; \mathbf{K}_{t-1} = \widehat{\mathbf{E}}_{t-1} \cdot \mathbf{W}_t^K \tag{8}$$

where $\mathbf{Z}_{t,t-1}$ represents the aggregated multi-behavior embeddings of current time slot t with the incorporation of cross-type behavior dependencies from the previous $(t-1)$-th time slot. $\mathbf{W}_t^Q \in \mathbb{R}^{d \times d}$, $\mathbf{W}_t^K \in \mathbb{R}^{d \times d}$ represent the linear transformation matrices corresponding to the query and key dimension, respectively. The learned cross-type behavior dependency matrix $\Phi \in \mathbb{R}^{B \times B}$ which can be derived as $\Phi = \mathbf{Q}_t \cdot \mathbf{K}_{t-1}^T$. Each entry $\phi_{b,b'} \in \Phi$ indicates the estimated correlation between time slot- and behavior-specific representations $\mathbf{E}_{t,u}^{b}$ and $\mathbf{E}_{t-1,u}^{b'}$. Therefore, our dynamic cross-relational memory network $\gamma(\cdot)$ can be rewrote with the following form.

$$\gamma(\mathbf{Q}_t, \mathbf{K}_{t-1}, \mathbf{V}_{t-1}) = softmax\left(\frac{\mathbf{Q}_t \cdot \mathbf{K}_{t-1}^T}{\sqrt{d/h}}\right) \cdot \mathbf{V}_{t-1} \tag{9}$$

The scale $\sqrt{d/h}$ is introduced to produce a softer attention distribution for avoiding gradient vanishing Vaswani et al. (2017), and h represents the number of head representations. With the design of our dynamic cross-relational memory network, our EGCM can preserve the dedicated time-evolving behavior dependencies across different types of user interactions. In our model implementation, we utilize the Broadcast Mechanism Van Der Walt et al. (2011) to improve the computational efficiency of tensor multiplication. With the learned evolving cross-type behavior dependencies, we refine the time slot-specific user and item representations as: $\widetilde{\mathbf{E}}_{t,u} = \mu(\widehat{\mathbf{E}}_{t,u} + \mathbf{Z}_{t,t-1,u})$ and $\widetilde{\mathbf{E}}_{t,i} = \mu(\widehat{\mathbf{E}}_{t,i} \oplus \mathbf{Z}_{t,t-1,i})$, where $\mu(\cdot)$ indicates the adopted mean pooling operation.

3.3 Multi-Behavior Contrastive Learning

Having encoding the heterogeneous user interest representations corresponding to the short-term ($\widehat{\mathbf{E}}_{t,u}, \widehat{\mathbf{E}}_{t,i}$) and long-term ($\widetilde{\mathbf{E}}_{t,u}, \widetilde{\mathbf{E}}_{t,i}$) perspectives, we propose to jointly capture the diverse multi-behavior dependencies and commonality in the multi-behavior sequential recommendation scenario.

3.3.1 Heterogeneous Behavior Aggregation

We first generate the multi-behavior representation ($\overline{\mathbf{E}}_u, \overline{\mathbf{E}}_i$) by aggregating type-specific behavior representations with the gating mechanism which is formally presented as follow. Without loss of generality, $\overline{\mathbf{E}}_u$ denote the aggregated user representation for both short-term and long-term interests. Item embeddings $\overline{\mathbf{E}}_i$ can be derived in an analogical way.

$$\omega_u^b = \frac{exp(\mathbf{E}_u^b\ \mathbf{W}_f)}{\sum_{b=1}^{|\mathcal{B}|} exp(\mathbf{E}_u^b \mathbf{W}_f)}; \qquad \overline{\mathbf{E}}_u = \sum_{b=1}^{|\mathcal{B}|} \omega_u^b \mathbf{E}_u^b \tag{10}$$

where $\mathbf{W}_f \in \mathbb{R}^d$ is the transformation matrix. ω_u^b represents the learned dependency weight of b-th type of user behaviors. With the behavior aggregation gating mechanism, both short-term and long-term multi-behavior preferences of users can be preserved by the fused representations.

3.3.2 Cross-Behavior Contrastive Self-Supervision

In our EGCM model, we aim to simultaneously capture the multi-behavior commonality of individual user, and the multi-behavior distinction of different users. To this end, we design the cross-behavior contrastive learning module to enhance the multi-behavior dependency modeling with the augmented self-supervision signals. Specifically, we propose to generate our contrasting views with the type-specific behavior semantics and the fused multi-behaviour pattern. Motivated by the property of contrastive learning, the cross-behavior contrastive self-supervision signals are incorporated into our recommendation framework as auxiliary regularization, so as to jointly capture the commonality of individual user and the diversity of different users with respect to their multi-behavior preferences. Our EGCM performs the behavior-level augmentation by pulling the type-specific behavior embedding ($\mathbf{e}_i^b \in \mathbb{R}^{d\times 1}$) and multi-behavior representation ($\mathbf{e}_i \in \mathbb{R}^{d\times 1}$) of the same user u closer as positive pairs, pushing the behavior embeddings of different users away as negative pairs. Formally, our cross-behavior contrastive objective is defined as:

$$\mathcal{L}_{cl}^b = \log \frac{exp(s(\mathbf{e}_u^b, \overline{\mathbf{e}}_u)/\tau)}{exp(s(\mathbf{e}_u^b, \overline{\mathbf{e}}_u)/\tau) + \sum\limits_{u\neq u'} \big(exp(s(\mathbf{e}_u^b, \overline{\mathbf{e}}_{u'})/\tau) + exp(s(\mathbf{e}_u^b, \mathbf{e}_{u'}^b)/\tau)\big)} \tag{11}$$

where τ represents temperature parameter to control the effect of mutual information estimation Chen et al. (2020b). Here, we define the InfoNCE-based similarity function $s(\mathbf{e}_u^b, \overline{\mathbf{e}}_u) = \mathbf{e}_u^b \cdot \overline{\mathbf{e}}_u / \|\mathbf{e}_u^b\| \|\overline{\mathbf{e}}_u\|$ which is measured by the doc product between ℓ_2 normalized $\mathbf{e}_u^b$ and $\overline{\mathbf{e}}_u$.

3.4 Model Inference Phase

We define our optimized objective with the Bayesian Personalized Ranking (BPR) loss as follows:

$$\mathcal{L}_{BPR} = \sum_{(u,i^+,i^-)\in O} -\log(\text{sigmoid}(\overline{e}_u \cdot \overline{e}_{i^+} - \overline{e}_u \cdot \overline{e}_{i^-})) + \lambda ||\Theta||^2 \tag{12}$$

where O represents the pairwise training samples, *i.e.*, $O = \{(u, i^+, i^-)|(u, i^+) \in \mathcal{E}^+, (u, i^-) \in \mathcal{E}^-\}$. Here, $\mathcal{E}^+$ and $\mathcal{E}^-$ denotes the corresponding observed and unobserved interaction of user u. Θ denotes the learnable hyperparameters. To alleviate the overfitting issue, we apply the L_2

Table 1: Statistics of experimented datasets

Dataset	User #	Item #	Interaction #	Sparsity	Interaction Behavior Types
Taobao	31882	31232	167862	99.98%	{View, Favorite, Cart, Purchase}
IJCAI	22438	35573	199654	99.98%	{View, Favorite, Cart, Purchase}
E-commerce	31021	1827	370386	99.35%	{Browse, Review, Purchase}

regularization in our BPR loss. Our joint optimization objective is given below by integrating BPR loss ($\mathcal{L}_{BPR}$) with short-term ($\sum_{t}^{|\Lambda|}\sum_{b}^{|\mathcal{B}|}\mathcal{L}_{cl}^{short}$) and long-term ($\sum_{b}^{|\mathcal{B}|}\mathcal{L}_{cl}^{long}$) contrastive objectives:

$$\mathcal{L} = \sum_{b}^{|\mathcal{B}|}\mathcal{L}_{BPR} + \alpha * \sum_{b}^{|\mathcal{B}|}\mathcal{L}_{cl}^{long} + \beta * \sum_{t}^{|\Lambda|}\sum_{b}^{|\mathcal{B}|}\mathcal{L}_{cl}^{short} \tag{13}$$

where $|\Lambda|$ represents the number of time slots. α and β are regularization strength of contrastive objectives. We further provide the in-depth analysis of our EGCM model with respect to the time complexity analysis and theoretical discussion in the submitted Supplementary material.

4 Evaluation

4.1 Experimental Setup

Datasets. We perform the model evaluation on three real-world datasets with the statistics shown in Table 1. i) **Taobao** is a benchmark dataset which collects four types of user behaviors from Taobao's recommender system. ii) **IJCAI** is a released online retailing dataset by IJCAI competition with four types of user online activities. iii) **E-commerce** contains browse, review and purchase behaviors of users in a real-life online retailer. Following the same settings in Jin et al. (2020a); Xia et al. (2021b), our multi-behavior sequential recommender system regard the purchase as the target behaviors and other types of user behaviors as auxiliary behaviors.

Evaluation Protocols. In our evaluation, two representative metrics, *i.e.*, Hit Ratio (HR@N) and Normalized Discounted Cumulative Gain (NDCG@N) ($N = 10$ by default), are adopted to measure the recommendation accuracy. To be consistent with the settings in existing sequential recommender systems Sun et al. (2019), the leave-one-out evaluation strategy is used for evaluation. Additionally, for each user, the last interacted item under the target behavior type is considered as the positive samples in the test set and 99 non-interacted items are randomly sampled as negative instances.

Baselines. We compare our EGCM with various types of state-of-the-art recommendation methods.

- (i) CNN/Attention-based Sequential Recommendation Approaches: **Caser** Tang & Wang (2018) uses the convolutional filters to encode local item dependencies of user sequences. **SASRec** Kang & McAuley (2018), **TiSASRec** Li et al. (2020), **AttRec** Zhang et al. (2018) are built on self-attention mechanism to capture correlations between temporally-ordered items. **Bert4Rec** Sun et al. (2019) trains the bidirectional Transformer model with the cloze task.
- (ii) Hybrid Sequential Recommender Systems: **HGN** Ma et al. (2019) is a hierarchical gating network to learn the item feature relevance. **Chorus** Wang et al. (2020a) considers both knowledge-aware item relations and temporal context in sequential recommendation.
- (iii) GNN-based Sequential Recommendation Models: **SR-GNN** Wu et al. (2019) first introduces the graph neural networks to model the sequential behavior patterns over short item sequences. **MA-GNN** Ma et al. (2020) leverages GNNs to encode both short-term and long-term item dependencies. Furthermore, **HyperRec** Wang et al. (2020b) proposes to capture dynamic triadic item relationships using the hypergraph structures. **COTREC** Xia et al. (2021c) and **DHCN** Xia et al. (2021d) are two state-of-the-art sequential recommender systems based on self-supervised learning paradigms.
- (iv) Multi-Behavior Recommender Systems: **NMTR** Gao et al. (2019) and **DIPN** Guo et al. (2019) formalize the multi-behavior recommendation task with multi-task learning frameworks. **MBGCN** Jin et al. (2020b), **KHGT** Xia et al. (2021a) and **MBGMN** Xia et al. (2021b) design multi-behavior graph message passing schemes to model heterogeneous user-item interactions. **EHCF** Chen et al. (2020a) and **CML** Wei et al. (2022) generate additional supervision signals from auxiliary behaviors to boost the recommendation performance.

Hyperparameter Settings and Implementation Details. We implement our model in Pytorch and adopt Xavier initializer Glorot & Bengio (2010) for parameter initialization. The Cyclical Learning

Table 2: Performance comparison of all methods on different datasets in terms of *HR* & *NDCG*.

Data	Tmall		IJCAI		E-commerce	
Metric	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
Caser	0.321	0.195	0.257	0.146	0.627	0.381
AttRec	0.328	0.197	0.261	0.147	0.628	0.387
SASRec	0.319	0.184	0.263	0.149	0.604	0.363
TiSASRec	0.322	0.185	0.262	0.148	0.599	0.368
BERT4Rec	0.329	0.197	0.281	0.155	0.641	0.392
HGN	0.283	0.172	0.251	0.136	0.586	0.359
Chorus	0.335	0.201	0.270	0.149	0.639	0.402
SR-GNN	0.318	0.189	0.280	0.151	0.620	0.373
MA-GNN	0.331	0.202	0.259	0.149	0.644	0.411
HyperRec	0.333	0.204	0.265	0.151	0.648	0.413
DHCN	0.321	0.193	0.271	0.148	0.638	0.391
COTREC	0.330	0.201	0.278	0.153	0.647	0.405
NMTR	0.362	0.215	0.269	0.156	0.651	0.408
DIPN	0.325	0.193	0.276	0.151	0.655	0.397
MBGCN	0.381	0.213	0.270	0.138	0.679	0.414
KHGT	0.391	0.232	0.278	0.145	0.689	0.434
MBGMN	0.419	0.246	0.329	0.176	0.690	0.432
EHCF	0.433	0.260	0.362	0.207	0.661	0.413
CML	0.543	0.327	0.410	0.235	0.719	0.427
EGCM	**0.597**	**0.363**	**0.510**	**0.312**	**0.763**	**0.470**
Imprv.	9.94%	11.01%	24.39%	32.77%	6.12%	10.07%
p-value	$5.2e^{-5}$	$4.9e^{-6}$	$9.6e^{-6}$	$2.3e^{-5}$	$8.7e^{-3}$	$3.4e^{-4}$

Rate strategy Smith (2017) is used during the model training phase with the AdamW Loshchilov & Hutter (2017) optimizer. For fair comparison, the number of graph layers in all GNN-based models are selected from the range {1,2,3,4} to achieve the best performance. The temperature coefficient τ is tuned from {0.02, 0.035, 0.05, 0.07, 0.1, 0.3, 0.5, 0.7} in our multi-behavior contrastive learning component. We further study the influence of key hyperparameters in our model and report the results in Supplementary materials.

4.2 Recommendation Performance

We report the performance comparison results in Table 2 and summarize the findings: (1) EGCM outperforms various baselines in all cases by achieving significant performance improvements. The "imprv" indicates the relative performance improvement between EGCM and the best performed baseline CML. Through the encoding of the evolving dependencies across different types of behaviors, EGCM is able to simultaneously capture the dynamics of users' short-term and long-term interests, by distilling the underlying heterogeneous interaction patterns. (2) EGCM outperforms the compared sequential recommender systems by a large margin, which indicates that the incorporation of multi-behavior context is beneficial for disentangling behavior heterogeneity of users. (3) Conducting dynamic contrastive learning with cross-behavior dependencies, EGCM learns better multi-behavior representations compared with state-of-the-art multi-behavior recommendation approaches, by preserving both the behavior commonality and diversity of users. (4) The consistent performance improvements of our method on datasets with different sparsity degrees, benefits from the incorporation of effective self-supervision signals generated by our contrastive learning paradigm.

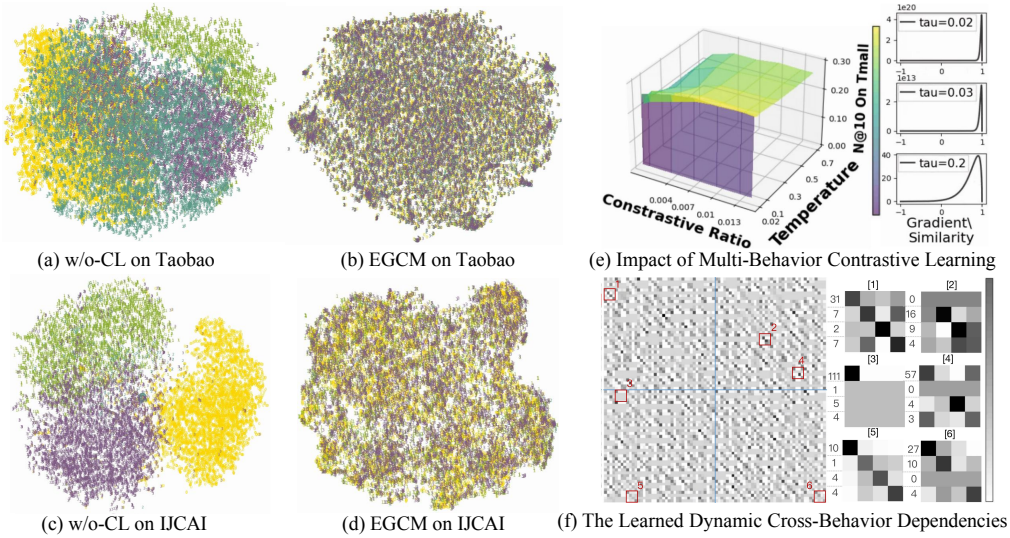
4.3 Ablation Study

We evaluate the efficacy of key components in EGCM with different variants: (1) *w/o-JBL*: Instead of performing jointly learning with multi-typed behavior supervision labels, we directly incorporate auxiliary behavior data as contextual features for user representation. (2) *w/o-CL*: We further disable our multi-behavior contrastive learning component to capture the behavior commonality and diversity. (3) *r/w-GRU*: We replace our dynamic cross-relational memory network with gated recurrent unit (GRU) to encode behavior-specific embedding sequences. (4) *w/o-MBG*: On the basis of *w/o-CL*, we remove the behavior-aware graph neural encoder, to model the high-order connectivity over the multi-behavior user-item interaction graph within a specific time slot.

From evaluation results shown in Table 3, we can observe that our multi-behavior learning with augmented short- and long-term contrastive objectives improves the generalization of multi-behavior recommender system. In addition, our cross-relational memory network is more effective than GRU

Table 3: Ablation study on the effectiveness of components in EGCM.

Data	Tmall		IJCAI		E-commerce	
Metrics	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
w/o-MBG	0.338	0.219	0.261	0.139	0.630	0.372
r/w-GRU	0.361	0.210	0.276	0.148	0.675	0.417
w/o-CL	0.391	0.231	0.300	0.166	0.734	0.461
w/o-JBL	0.473	0.291	0.367	0.205	0.747	0.469
EGCM	**0.597**	**0.363**	**0.510**	**0.312**	**0.763**	**0.470**



(a) w/o-CL on Taobao (b) EGCM on Taobao (e) Impact of Multi-Behavior Contrastive Learning

(c) w/o-CL on IJCAI (d) EGCM on IJCAI (f) The Learned Dynamic Cross-Behavior Dependencies

Figure 2: (a)-(b): Visualized behavior-aware user representations which preserve multi-behavior commonality and cross-user behavior diversity. (e): Impact study of contrastive learning in terms of temperature parameter τ(see Sec. A.1.2 for details). (f): Learned cross-behavior dependency weights across time slots.

to encode long-term multi-behavior user interests. Furthermore, the multi-behavior graph encoder brings positive effects to learn short-term heterogeneous user preferences, compared with directly information aggregation over id-corresponding embeddings for collaborative effect modeling.

4.4 In-Depth Analysis of EGCM Model

This part investigates the benefits of our EGCM model from three perspectives. In particular, i) In Figure 2 (a)-(d), we visualize the encoded behavior-specific user embeddings of our EGCM and the variant w/o-CL (without the multi-behavior contrastive learning). The results suggest that our cross-behavior data augmentation scheme is effective to preserve user's multi-behavior commonality and better differentiate diverse user preference. ii) As shown in Figure 2 (e), we can notice that a smaller value of temperature value τ may bring larger gradients for better identifying hard negatives, so as to enhance the model discrimination ability in learning personalized user interests. iii) In Figure 2 (f), we visualize the learned cross-behavior dependencies between time slots with 4×4 weight matrices for 400 sampled users on Taobao dataset and highlight 6 ones on the right side. This suggests that our EGCM can offer global relation interpretability across different types of user behaviors in a dynamic recommendation scenario.

5 Conclusion

This work studies the problem of multi-behavior sequential recommendation and propose our model that captures the interaction heterogeneity of each user at both the short-term and long-term level interests. In addition, a multi-behavior contrastive learning paradigm is presented to not only model the multi-behavior commonalities of individual user, but also enhances the distinction of behavior-aware preference of different users. Experiments on three real-world datasets demonstrate that EGCM significantly advances the recommendation performance compared with many strong baselines. In the future, we plan to extend EGCM to adapt to the cross-domain recommendation to tackle the cold-start problem in multi-behavior recommender systems.

REFERENCES

Aviad Aberdam, Ron Litman, Shahar Tsiper, Oron Anschel, Ron Slossberg, Shai Mazor, R Manmatha, and Pietro Perona. Sequence-to-sequence contrastive learning for text recognition. In *CVPR*, pp. 15302–15312, 2021.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *NIPS*, 32, 2019.

Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. Sequential recommendation with graph neural networks. In *SIGIR*, pp. 378–387, 2021.

Chong Chen, Min Zhang, Yongfeng Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. Efficient heterogeneous collaborative filtering without negative sampling for recommendation. In *AAAI*, volume 34, pp. 19–26, 2020a.

Chong Chen, Weizhi Ma, Min Zhang, Zhaowei Wang, Xiuqiang He, Chenyang Wang, Yiqun Liu, and Shaoping Ma. Graph heterogeneous multi-relational recommendation. In *AAAI*, volume 35, pp. 3958–3966, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pp. 1597–1607. PMLR, 2020b.

Chen Gao, Xiangnan He, Dahua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, and Depeng Jin. Neural multi-task recommendation from multi-behavior data. In *ICDE*, pp. 1554–1557. IEEE, 2019.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, pp. 249–256. JMLR Workshop and Conference Proceedings, 2010.

Long Guo, Lifeng Hua, Rongfei Jia, Binqiang Zhao, Xiaobo Wang, and Bin Cui. Buying or browsing?: Predicting real-time purchasing intent using attention-based deep network with multiple behavior. In *KDD*, pp. 1984–1992, 2019.

Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *ICML*, pp. 4116–4126. PMLR, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *CVPR*, pp. 1026–1034, 2015.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *SIGIR*. ACM, 2020.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *ICLR*, 2016.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Chih-Hui Ho and Nuno Nvasconcelos. Contrastive learning with adversarial examples. In *NIPS*, volume 33, pp. 17081–17093, 2020.

Bowen Jin, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. Multi-behavior recommendation with graph convolutional networks. In *SIGIR*, 2020a.

Bowen Jin, Chen Gao, Xiangnan He, Depeng Jin, et al. Multi-behavior recommendation with graph convolutional networks. In *SIGIR*, pp. 659–668, 2020b.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *ICDM*, pp. 197–206. IEEE, 2018.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

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

Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *AAAI*, volume 32, 2018.

Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *Transactions on Knowledge and Data Engineering (TKDE)*, 2021a.

Zhiwei Liu, Ziwei Fan, Yu Wang, and Philip S Yu. Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer. In *SIGIR*, pp. 1608–1612, 2021b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2017.

Chen Ma, Peng Kang, and Xue Liu. Hierarchical gating networks for sequential recommendation. In *KDD*, pp. 825–833, 2019.

Chen Ma, Liheng Ma, Yingxue Zhang, Jianing Sun, Xue Liu, and Mark Coates. Memory augmented graph neural networks for sequential recommendation. In *AAAI*, volume 34, pp. 5045–5052, 2020.

Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, pp. 811–820, 2010.

Leslie N Smith. Cyclical learning rates for training neural networks. In *WACV*, pp. 464–472. IEEE, 2017.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *CIKM*, pp. 1441–1450, 2019.

Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *WSDM*, pp. 565–573, 2018.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, pp. 776–794. Springer, 2020a.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *NIPS*, volume 33, pp. 6827–6839, 2020b.

Aaron Van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv e-prints*, pp. arXiv–1807, 2018.

Stefan Van Der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in science & engineering*, 13(2):22–30, 2011.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems (NIPS)*, 30, 2017.

Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *ICLR*, 2(3):4, 2019.

Vikas Verma, Thang Luong, Kenji Kawaguchi, Hieu Pham, and Quoc Le. Towards domain-agnostic contrastive learning. In *ICML*, pp. 10530–10541. PMLR, 2021.

Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. Make it a chorus: knowledge- and time-aware item modeling for sequential recommendation. In *SIGIR*, pp. 109–118, 2020a.

Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *CVPR*, pp. 2495–2504, 2021.

Jianling Wang, Kaize Ding, Liangjie Hong, Huan Liu, and James Caverlee. Next-item recommendation with sequential hypergraphs. In *SIGIR*, pp. 1101–1110, 2020b.

Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z Sheng, and Mehmet Orgun. Sequential recommender systems: challenges, progress and prospects. In *IJCAI*, 2019.

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*, pp. 169–178, 2020c.

Wei Wei, Chao Huang, Lianghao Xia, Yong Xu, Jiashu Zhao, and Dawei Yin. Contrastive meta learning with behavior multiplicity for recommendation. In *WSDM*, pp. 1120–1128, 2022.

Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. Self-supervised graph learning for recommendation. In *SIGIR*, pp. 726–735, 2021.

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

Lianghao Xia, Chao Huang, Yong Xu, Peng Dai, Xiyue Zhang, Hongsheng Yang, Jian Pei, and Liefeng Bo. Knowledge-enhanced hierarchical graph transformer network for multi-behavior recommendation. In *AAAI*, volume 35, pp. 4486–4493, 2021a.

Lianghao Xia, Yong Xu, Chao Huang, Peng Dai, and Liefeng Bo. Graph meta network for multi-behavior recommendation. In *SIGIR*, pp. 757–766, 2021b.

Xin Xia, Hongzhi Yin, Junliang Yu, Yingxia Shao, and Lizhen Cui. Self-supervised graph co-training for session-based recommendation. In *CIKM*, pp. 2180–2190, 2021c.

Xin Xia, Hongzhi Yin, Junliang Yu, Qinyong Wang, Lizhen Cui, and Xiangliang Zhang. Self-supervised hypergraph convolutional networks for session-based recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 4503–4511, 2021d.

Jiaxuan You, Zhitao Ying, and Jure Leskovec. Design space for graph neural networks. *NIPS*, 33: 17009–17021, 2020.

Shuai Zhang, Yi Tay, Lina Yao, and Aixin Sun. Next item recommendation with self-attention. *arXiv preprint arXiv:1808.06414*, 2018.

Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Graph contrastive learning with adaptive augmentation. In *WWW*, pp. 2069–2080, 2021.

A APPENDIX

A.1 Theoretical Discussion of EGCM Model

In this section, we conduct in-depth analysis for our multi-behavior contrastive learning paradigm. Specifically, we discuss the benefits brought by the augmented behavior-aware self-supervised learning tasks from two dimensions: i) Cross-behavior mutual information maximization; ii) User interest representation discrimination. The detailed theoretical discussion is presented as follows:

A.1.1 Cross-Behavior Mutual Information Maximization

In our multi-behavior contrastive learning framework, we propose to capture the multi-behavior commonality, by maximizing the mutual information between type-specific behavior embedding ($\mathbf{e}_i^b \in \mathbb{R}^{d\times 1}$) and multi-behavior representation ($\mathbf{e}_i \in \mathbb{R}^{d\times 1}$). In particular, we define our information maximization function as $I(\mathbf{z}_i, \mathbf{z}_p)$, where $\mathbf{z}_i, \mathbf{z}_p$ represents the anchor and positive instance in the same hypersphere, respectively. Without loss of generality, the embedding $\mathbf{z}$ is formally defined as: $\mathbf{z} = \mathbf{e}/ \parallel \mathbf{e} \parallel$. In our multi-behavior contrasting scenario, the mutual information estimation is performed between the fused representation $\overline{\mathbf{E}}_u$ and behavior-specific embdding in $\{\mathbf{E}_u^1, \mathbf{E}_u^2, ..., \mathbf{E}_u^b\}$. Such contrastive-based self-supervision signals are integrated into the BPR loss function to enhance the robustness of user representation paradigm.

Motivated by the research in Van den Oord et al. (2018); Hjelm et al. (2018); Bachman et al. (2019); Tian et al. (2020a), our augmented contrastive loss can form the lower bound of the information maximization function $I(\cdot)$ as:

$$I(\mathbf{z}_i, \mathbf{z}_p) \geq log(k) - \mathcal{L}_{cl} \tag{14}$$

where k is the number of negative samples. The inequality suggests that smaller $\mathcal{L}_{cl}$ results in larger $I(\cdot)$. In other words, minimizing $\mathcal{L}_{cl}$ is equivalent to maximizing the lower bound of mutual information in $I(\cdot)$.

The above situation can be extended to the multi-behavior modeling process. In addition to the equivalence between the contrastive objective and lower bound of information maximization, the self-supervised $\mathcal{L}_{cl}$ is closely related to function $I(\cdot)$. It can provide a theoretical basis for our multi-behavior recommender system as follows:

$$h_\theta^*(\mathbf{z}_i, \mathbf{z}_p) \propto \frac{p(\mathbf{z}_i, \mathbf{z}_p)}{p(\mathbf{z}_i)p(\mathbf{z}_p)} \propto \frac{p(\mathbf{z}_i|\mathbf{z}_p)}{p(\mathbf{z}_i)} \tag{15}$$

where h^* is the optimal point of $h_\theta = \exp(s(\mathbf{e}_u^b, \overline{\mathbf{e}}_u)/\tau)$ that is proportional to the density ratio between the joint distribution $p(\mathbf{z}_i, \mathbf{z}_p)$ and the product of marginals $p(\mathbf{z}_i)p(\mathbf{z}_p)$. This quantity is the point-wise mutual information, and the extended multi-behavior form can be implemented by optimizing the sum of a set of pair-wise objectives Tian et al. (2020a).

A.1.2 User Interest Representation Discrimination

Contrastive loss is a hardness-aware loss function Wang & Liu (2021); Wu et al. (2021). It can push away the hard negative samples a lot from the anchor by giving them greater gradients under contrastive training framework. This property is beneficial to our multi-behavior graph neural architecture. One of the most important challenges in existing GNN architecture lies in how to achieve a nice trade-off between high-order connectivity modeling and over-smoothing issue Li et al. (2018). Stacking more graph-based information propagation layers is more like to involve over-smoothing issue for encoding collaborative effects. Hence, enhance the discrimination ability of user interest representation paradigm is necessary and challenging for recommender system. To tackle the above challenge, our multi-behavior contrastive learning framework will assign larger gradients to hard negative samples so as to enhance the discrimination of user representations.

Embedding Normalization. To map the embeddings with arbitrary value distributions into the same hyperspace, we perform the embedding normalization as $\mathbf{z}_i = \mathbf{e}_i/ \parallel \mathbf{e}_i \parallel$, where $\mathbf{e}_i$ denote the output *prior* to normalization. The gradient of the loss with respect to $\mathbf{e}_i$ is related to that with respect to $\mathbf{z}_i$ via the chain rule presented as follows:

$$\frac{\partial \mathcal{L}_i(\mathbf{z}_i)}{\partial \mathbf{e}_i} = \frac{\partial \mathcal{L}_i(\mathbf{z}_i)}{\partial \mathbf{z}_i}\frac{\partial \mathbf{z}_i}{\partial \mathbf{e}_i} \tag{16}$$

$$\frac{\partial \mathbf{z}_i}{\partial \mathbf{e}_i} = \frac{\partial}{\partial \mathbf{e}_i}\left(\frac{\mathbf{e}_i}{\parallel \mathbf{e}_i \parallel}\right) = \frac{1}{\parallel \mathbf{e}_i \parallel} I - \mathbf{e}_i \left(\frac{\partial(1/\parallel \mathbf{e}_i \parallel)}{\partial \mathbf{e}_i}\right)^T$$
$$= \frac{1}{\parallel \mathbf{e}_i \parallel}\left(I - \frac{\mathbf{e}_i \mathbf{e}_i^T}{\parallel \mathbf{e}_i \parallel^2}\right) = \frac{1}{\parallel \mathbf{e}_i \parallel}\left(I - \mathbf{z}_i \mathbf{z}_i^T\right) \tag{17}$$

Gradients of Negative Pairs. We use the loss function $\mathcal{L}_i$ to calculate the partial derivative of the anchor point to analyze the influence of different samples over the gradients:

$$\frac{\partial L_i}{\partial \mathbf{z}_i} = \frac{\partial}{\partial \mathbf{z}_i}\left(-log\frac{\exp\left(\mathbf{z}_i \cdot \mathbf{z}_P/\tau\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)}\right) = -\frac{1}{\tau}\cdot \mathbf{z}_P + \frac{1}{\tau}\frac{\sum_{\mathcal{V}_A}\mathbf{z}_A \cdot \exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)}$$
$$= \frac{1}{\tau}\left(\frac{\sum_{\mathcal{V}_N}\mathbf{z}_N \cdot \exp\left(\mathbf{z}_i \cdot \mathbf{z}_N/\tau\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)} + \frac{\mathbf{z}_P \cdot \left(\exp\left(\mathbf{z}_i \cdot \mathbf{z}_P/\tau\right) - \sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)}\right) \tag{18}$$

The two terms of the formula represent the gradient of positive samples and negative samples respectively. Here, $\mathbf{z}_N$ represents the instance from the set of negative samples and $\mathbf{z}_P, \mathbf{z}_A$ are from the positive sample set and the entire set. We then mainly focus on the negative part, which can be :

$$\frac{1}{\tau}\cdot\frac{\sum_{\mathcal{V}_N}\mathbf{z}_N \cdot \exp\left(\mathbf{z}_i \cdot \mathbf{z}_N/\tau\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)} = \frac{1}{\tau\cdot \parallel \mathbf{e}_i \parallel}\cdot\frac{\sum_{\mathcal{V}_N}\left(\mathbf{z}_N - \left(\mathbf{z}_i \cdot \mathbf{z}_N\right)\cdot \mathbf{z}_i\right)\cdot \exp\left(\mathbf{z}_i \cdot \mathbf{z}_N/\tau\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)} \tag{19}$$

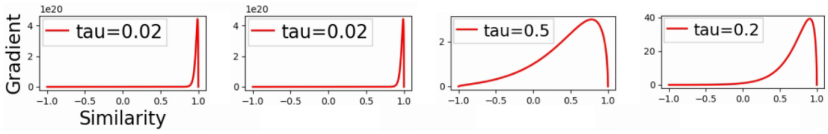
Temperature and Gradient. The proportional term of the norm of the gradient of each term in the sum formula is as follow:

$$\parallel \mathbf{z}_N - \left(\mathbf{z}_i \cdot \mathbf{z}_N\right)\cdot \mathbf{z}_i \parallel \left|\frac{\exp\left(\mathbf{z}_i \cdot \mathbf{z}_N/\tau\right)}{\sum_{\mathcal{V}_A}\exp\left(\mathbf{z}_i \cdot \mathbf{z}_A/\tau\right)}\right|$$
$$\propto \sqrt{1-\left(\mathbf{z}_i \cdot \mathbf{z}_N\right)^2}\cdot \exp\left(\mathbf{z}_i \cdot \mathbf{z}_N/\tau\right) \tag{20}$$

As z_i and z_N are both unit vectors, we introduce another variable x with the definition of $x = z_i \cdot z_N \in [-1, 1]$ to abbreviate the final result of Eq. 20:

$$c(x) \propto \sqrt{1-(x)^2}\cdot \exp\left(x/\tau\right) \tag{21}$$

where $c(x)$ is the relationship function of the gradient from the negative samples. We plot the function in Equation 21 in Figure A.1.2. The independent variable is similarity x and the dependent variable is proportional to the negative sample gradient. With the increase of x, the gradient of negative samples will increase. Moreover, as the temperature coefficient τ (tau) decreases, the gradient of negative samples given by contrastive learning will also increase significantly.



In our recommended scenario, hard negative samples $\mathbf{z}_{N-hard}$ represents the users other than the anchor user. If $\mathbf{z}_{N-hard}$ is very close to the anchor point $\mathbf{z}_i$, the value of x of the hard negative samples approaches 1, which results in more indistinguishable user representations. The contrastive loss gives a larger gradient, and as relative negative samples, the pairs will be pushed farther away from each other. In this way, EGCM enhance the user representations with multi-behavior diversity. Therefore, the experiments in Sec. 4.4 shows that as the temperature τ decreases which, the more distinguishable the user representation is, which brings the better effect. However, when the gradient is over large, gradient explosion will be observed.

A.2 **Alleviating the Data Sparsity Issue.**

Our above theoretical discussion analyzes the benefits of our multi-behavior contrastive learning paradigm in capturing the multi-behavior commonality and diversity. To be specific, the mutual information maximization between positive samples is helpful to preserve the common characteristics among different types of behaviors. Additionally, contrastive learning with negative samples can push the hard negative samples away to get distinguishable user embedding, so as to encode the behavior diversity of different users and alleviate the over-smoothing problem of our graph neural model.

Therefore, contrastive learning can improve the quality of representation and alleviate the problem caused by the scarcity of data. To this end, we selected users whose interaction number on the IJCAI dataset in {<5, <15, <35, <60} for training and testing. In order to eliminate the influence of training data and simulate a real sparse data scenario, we carry out the evaluation on our model under the setting of w/o-JBL, and compare it with two best-performed baselines (HyperRec and KHGT) from the lines of sequence-based models and multi-behavior recommender systems, respectively.

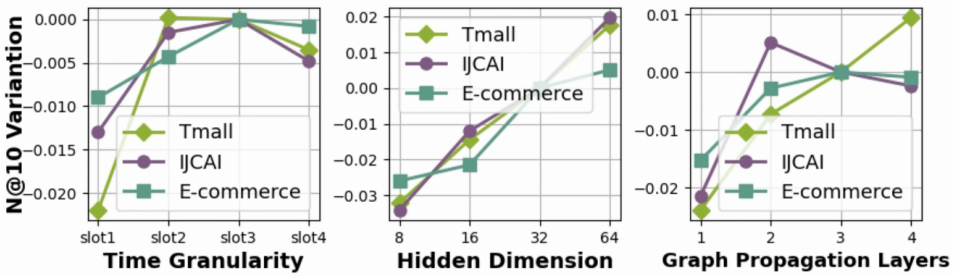
Table 4: Performance with different number of interactions in terms of HR@5@20 & NDCG@5@20.

		<5		<15		<35		<60	
		@5	@20	@5	@20	@5	@20	@5	@20
HyperRec	HR	0.0934	0.3052	0.1309	0.3465	0.1616	0.4017	0.3182	0.5909
	NDCG	0.0728	0.1108	0.0804	0.1401	0.0930	0.1655	0.2073	0.2872
KHGT	HR	0.1136	0.3371	0.1763	0.4174	0.2198	0.4702	0.4091	0.6364
	NDCG	0.0753	0.1367	0.1161	0.1715	0.1463	0.2101	0.2374	0.3171
w/o-cl	HR	0.1326	0.3674	0.1856	0.4153	0.2271	0.4920	0.4091	0.7727
	NDCG	0.0824	0.1482	0.1238	0.1883	0.1514	0.2256	0.2911	0.3895
w/o-JBL	HR	0.1806	0.4268	0.2168	0.4579	0.2547	0.5560	0.3636	0.7727
	NDCG	0.1174	0.1865	0.1460	0.2138	0.1786	0.26210	0.2010	0.3150

As shown in Table 4, it can be observed that under different sparsity degrees of user interaction data, w/o-JBL with contrastive learning task will get better results than w/o-CL. Moreover, the performance gap between our contrastive learning method and other baselines become larger with the higher sparsity degrees of interaction data, which again justfies the effectiveness of our EGCM in addressing the data scarcity for recommender system.

A.3 Hyperparameter Sensitivity Analysis

To explore the sensitivity of our proposed EGCM framework, we perform experiments to evaluate the influence of key hyperparameters, including short-term time granularity, # of graph propagation layers L, the regularization strength of contrastive learning. Figure A.3 presents the evaluation results.



Short-Term Time Granularity. To construct the short-term multi-behavior graphs, we tune the parameter of the time granularity from different time ranges due to the time span of different experimental datasets. In particular, the time granularity of Tmall, IJCAI, E-commerce dataset is selected from {1,1,5,3,9} days, {1,3,6,12} months, {1,2,4,12} weeks, respectively. From the evaluation results in Figure A.3, we can observe that shorter time period (with the higher time granularity) will leads to the overfitting problem in modeling the short-term behavior-aware user preference.

Representation Hidden Dimensionality. To investigate the effects of hidden dimensionality in our representation performance, the embedding size is chosen from the range of {8, 16, 32, 64} for parameter sensitivity evaluation. It can be seen that the results on Tmall and IJCAI datasets improves significantly with the increase of dimension size, due to the stronger representation power of larger embedding size.

# of Graph Propagation Layers. To capture the short-term multi-behavior user interests, we design the behavior-aware message passing scheme to refine user/item embeddings with the injection of multi-typed behavior context. We select the number of graph propagation layers from the range of {1, 2, 3, 4} to explore the effect of model depth. We can observe that deeper graph models may bring benefits to model the high-order collaborative effects on Tmall and E-commerce datasets. By stacking more embedding propagation layers may involve noise in representation refinement on E-commerce dataset.

A.4 Model Complexity Analysis

The main time consumption in our EGCM framework are from several key components: i) Short-term multi-behavior graph encoder: the computational cost of our graph neural architecture for item representation is $O(|\mathcal{B}| \times \Lambda \times L \times |\mathcal{E}^b_{t,M^TM}| \times d)$ for performing message passing across graph layers. Then for user, it is $O(|\mathcal{B}| \times \Lambda \times |\mathcal{E}^b_{t,M}| \times d)$. $|\mathcal{E}^b_{t,M^TM}|$, $|\mathcal{E}^b_{t,M}|$ respectively represents the number of non-zero elements in the incidence matrix $\Gamma(\mathbf{M}_t^{b^T})$ and $\Gamma(\mathbf{M}_t^{b^T}\mathbf{M}_t^b)$, under the behavior type of b during the time slot t. Here, L denotes the number of graph propagation layers of item. The operations of concatenation and linear transformations for layer aggregation takes $O(|\mathcal{B}| \times \Lambda \times L \times |\mathcal{V}_i^b| \times d)$. ii) Dynamic cross-relational memory network: The most computational cost for the cross-relational memory component comes from the self-attention operation with the time complexity of $O(\Lambda \times |\mathcal{V}^b| \times |\mathcal{B}|^2 \times d)$ quadratic with the behavior number $|\mathcal{B}|$. iii) Multi-behavior contrastive learning: The cost of InfoNCE-based mutual information calculation is $O(d)$ and $O(batch \times d)$ for the numerator and denominator (in Eq.11), respectively. Thus, our multi-behavior contrastive learning paradigm takes $O(\Lambda \times |\mathcal{B}| \times |\mathcal{E}^b| \times d)$ per epoch. Given the smaller values of L, $|\mathcal{B}|$ and Λ, our EGCM model can achieve comparable time complexity as compared to state-of-the-art GNN-based recommendation techniques.

A.5 Notation and Framework Algorithm

Notation	Description
$\mathcal{V}_u, \mathcal{V}_i, \mathcal{B}, \mathcal{E}$,	user set, item set, behavior set, interactive edge set
$u \in \mathcal{V}_u$, $i \in \mathcal{V}_i$, $b \in \mathcal{B}$, t	a specific user/item/behavior/time slot
Λ, L, d, h	number of time slot/layer, hidden dim, head num
$\mathbf{M}_t^b, \mathbf{D}_t^b, \mathbf{B}_t^b$	incidence matrix/diagonal matrix of u/i during t under b
$\mathbf{E}_{t,i}^{b,0}, \mathbf{E}_{t,i}^{b,(l)}, \mathbf{E}_{t,i}^b, \mathbf{E}_{t,u}^b$	embeddings of GNNs layer $0/l$, short term u/i during t under b
$\widehat{\mathbf{E}}_t, \widetilde{\mathbf{E}}_t$	short/long term high dimensional multi-behavior embeddings
$\overline{\mathbf{E}}_{t,u}, \overline{\mathbf{E}}_u$	short/long term aggregated embeddings
$\delta(\cdot), \Gamma(\cdot), \gamma(\cdot)$	PReLU/Laplacian normalized/cross-relational memory function
$\alpha, \beta, \zeta, \tau$	short/long contrastive strength, combined weight, temperature

Algorithm 1: The Learning Process of EGCM Framework

Input: Behavior-aware interaction sequence $S_u = \{(i_1, b_1), (i_2, b_2), ..., (i_{|S_u|}, b_{|S_u|})\}$.
Short-term multi-behavior graph $\mathcal{G}_t^b = (\mathcal{V}_t^b, \mathcal{E}_t^b, \mathbf{M}_t^b)$.

Output: Aggregated long term user/item representations $\overline{\mathbf{E}}_i$, $\overline{\mathbf{E}}_u$. The probability of the most likely next item $i_{|S_u|+1}$.

Initialize: Xavier initialized behavior-specific short term item embeddings $\mathbf{E}_{t,i}^{b,0}$. Parameters: i)Short-term multi-behavior graph encoder $\{\mathbf{W}_{t,i}^{b,(l)}, \mathbf{W}_{t,u}^b, \mathbf{W}_\zeta^b, \mathbf{W}_{t,cat}^b, \zeta\}$. ii)Dynamic cross-relational memory network $\{\mathbf{W}_t^Q, \mathbf{W}_t^K\}$. iii)Memory cross behavior self-attention and behavior fusion attention $\{\mathbf{W}_t^Q, \mathbf{W}_t^K\}$, $\{\mathbf{W}_f\}$.

```
for epoch ← 0, 1, ... do
    Update learning rate scheduler.
    for step ← 0, 1, ... do
        // Short-Term Multi-Behavior Graph:
        for t ← 0, 1, ..., |Λ| do
            Get short term embeddings: E^b_{t,u}, E^b_{t,i} ⟵ G^b_t, Eq.1, Eq.3, Eq.5, Eq.6
            Prepare behavior aggregated embeddings for L^{short}_{cl}: Ē_{t,u} ⟵ Eq.10
        end
        // Dynamic Cross-Relational Memory:
        for t ← 1, ..., |Λ| do
            Modeling time-evolving cross-type dependencies: Ẽ^b_{t,u}, Ẽ^b_{t,i} ⟵ Eq.8, Eq.9
            Aggregate Ẽ^b_{t,u}, Ẽ^b_{t,i} convey across t and b for L^{long}_{cl}, L_{BPR} : Ē_u, Ē_i ⟵ Eq.10
        end
        // Cross-Behavior Contrastive Task & Recommendation Task:
        Get final multi-task objective L ⟵ L^b_{cl} ← Eq.11 + L_{BPR} ← Eq.12
        Gradient descent back propagation.
    end
end
```