

Multi-modal Relational Item Representation Learning for Inferring Substitutable and Complementary Items

Junting Wang
junting3@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

Yanhui Guo
yanhuig@amazon.com
Amazon
Seattle, Washington, USA

Chenghuan Guo
chenghg@amazon.com
Amazon
Seattle, Washington, USA

Yan Gao
yanngao@amazon.com
Amazon
Seattle, Washington, USA

Jiao Yang
jaoyan@amazon.com
Amazon
Seattle, Washington, USA

Hari Sundaram
hs1@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

ABSTRACT

We introduce a novel self-supervised multi-modal relational item representation learning framework designed to infer substitutable and complementary items. Existing approaches primarily focus on modeling item-item associations deduced from user behaviors using graph neural networks (GNNs) or leveraging item content information. However, these methods often overlook critical challenges, such as noisy user behavior data and data sparsity due to the long-tailed distribution of these behaviors. In this paper, we propose MMSC, a self-supervised multi-modal relational item representation learning framework to address these challenges. Specifically, MMSC consists of three main components: (1) a multi-modal item representation learning module that leverages a multi-modal foundational model and learns from item metadata, (2) a self-supervised behavior-based representation learning module that denoises and learns from user behavior data, and (3) a hierarchical representation aggregation mechanism that integrates item representations at both the semantic and task levels. Additionally, we leverage LLMs to generate augmented training data, further enhancing the denoising process during training. We conduct extensive experiments on five real-world datasets, showing that MMSC outperforms existing baselines by 26.1% for substitutable recommendation and 39.2% for complementary recommendation. In addition, we empirically show that MMSC is effective in modeling cold-start items.

KEYWORDS

Recommender System, Multi-modal Recommendation, Substitute and Complementary Recommendation

1 INTRODUCTION

This paper addresses the challenge of identifying substitutable and complementary items in e-commerce services. Understanding these relationships is vital for improving e-commerce services. Identifying substitutable items can enhance delivery efficiency and suggest alternatives for out-of-stock products, while recognizing complementary items can assist in recommending potential follow-up purchases to users, boosting company's revenue.

Modeling substitutable and complementary relationships between items poses two key challenges. First, these relationships lack explicit labels and are often inferred from user behaviors, with



Figure 1: Examples of user-behavior data.

co-viewed items considered substitutable and *co-purchased* items deemed complementary [5, 6, 12, 21, 22, 26, 30, 38–40]. However, user behavior data is often noisy (Figure 1), introducing significant noise to the training process and making it difficult to evaluate performance effectively. Second, user behaviors tend to follow heavy-tailed distributions (Figure 2), where a small subset of items accounts for most behaviors, leaving the majority of items with sparse behavior data. This combination of noisy data and data sparsity further amplifies the difficulty of modeling substitutable and complementary relationships effectively.

Existing studies have explored substitutable and complementary recommendation [6, 21, 22, 27, 38–40]. GNN-based methods [4, 5, 13–15, 21, 28–30, 39, 40] learn item representations by exploring the topological structure of item-item associations derived from user behavior data. Other approaches focus on modeling item content information [6, 12, 22, 38], employing methods such as Variational Autoencoders (VAEs) [26]. These methods overlook the fundamental challenges of noisy user behaviors and data sparsity.

Our Insight: User behavior data provides valuable implicit associations between items. However, noisy behavior data and heavy-tailed distributions make it difficult to accurately model substitutable and complementary relationships. Conversely, item metadata, which serves as ground truth descriptors of items, is more robust to noisy user behaviors and data sparsity but lacks the ability to effectively capture item-item associations. Therefore, denoising user behavior data while incorporating item metadata is crucial for modeling substitutable and complementary relationships.

Present Work: We propose Multi-Modal Relational Item Representation Learning for Substitutable and Complementary Recommendation (MMSC), a novel item representation learning framework that simultaneously denoise the user behavior data and leverage item metadata to model substitutable and complementary relationships effectively. Specifically, MMSC has two key components: a multi-modal item representation learning module that leverages a multi-modal foundational model to learn from item metadata and a denoising self-supervised representation module that learns to represent items by leveraging the noisy user behaviors. We also introduce a hierarchical representation aggregation mechanism to integrate the learned item representations from these modules. For model optimization, inspired by recent advancements in large language models (LLMs) [2, 9], we augment the training data using LLMs to further denoise the user behaviors used in training. Additionally, we adopt a multi-task learning paradigm to jointly denoise relationships and infer substitutable and complementary relationships. MMSC outperforms existing baselines by 26.1% for substitutable and 39.2% for complementary recommendation on five real-world datasets. We investigate the effectiveness of each model component through an ablation study. We empirically show that MMSC also excels in modeling relationships for cold-start items. Our **key contributions** are as follows:

Integrative Content-Relational Item Representation: We propose a novel framework that explicitly models both item-item associations derived from user behaviors and item metadata to learn robust item representations. Previous works focus on either item content [6, 12, 19, 22, 26, 38] or user behaviors [21, 30, 39, 40], neglecting the complementary nature of these data sources. Our approach fuses content representations (learned by adapting a multi-modal foundational model) and behavior-based representations (captured through a meta-path encoder) via a hierarchical representation aggregator. Extensive experiments demonstrate that combining item metadata and user behaviors is crucial for modeling substitutable and complementary relationships.

Noise-aware Item Representation Learning: To the best of our knowledge, we are the first to explicitly address noisy user behavior data in substitutable and complementary recommendations. Previous methods [5, 6, 12, 21, 22, 26, 30, 38–40] typically assume reliable behavior data and neglect the noise in user behaviors. In contrast, we propose a self-supervised learning paradigm to denoise user behavior data. Additionally, we utilize large language models (LLMs) to generate augmented training data, further enhancing the denoising process. Our results show that denoising is critical for accurately modeling item-item relationships, with our mechanisms significantly improving the performance of substitutable and complementary recommendations.

2 RELATED WORK

We briefly introduce several lines of related work on substitutable and complementary recommendation.

2.1 Substitutable and Complementary Recommeniders

We categorize the related works into two types: GNN-based methods [5, 21, 30, 39, 40] and content-based [6, 12, 22, 38].

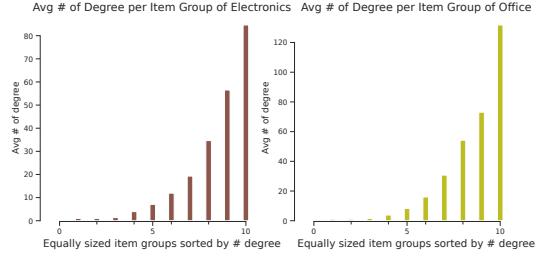


Figure 2: Degree distribution of the item-item relationship graph.

2.1.1 GNN-based Methods. GNN-based methods leverage user-behavior data to model item-item relationships by treating items as nodes and user-behavior data as edges. DecGCN [21] and DHGAN [40] are two recent representative GNN-based works. DecGCN [21] decouples the representation of items based on relationships and transform the original heterogeneous graph into multiple homogeneous graphs, paired with a co-attention mechanism to fuse the different representations of the same item. DHGAN [40], built on top of DecGCN, models item representations in hyperbolic space. Other works, such as HetaSAGE [39] and TransGAT [30] target either substitutable or complementary relationships. EMRIGCN [5] considers mutual influence between different types of relationships and proposes a two-level integration mechanism to capture shared information and relationship specific information.

While GNN-based methods are effective, they fail to address the fundamental challenges in modeling substitutable and complementary relationships, *i.e.*, noisy user-behavior data, and are generally ineffective for cold-start items. Additionally, GCN and GAT-based models [21, 30, 39, 40] decouple the original graphs into separate homogeneous graphs, and they ignore valuable connectivity patterns of items (*i.e.*, through which relationships are items connected), which are crucial in modeling item-item relationships.

2.1.2 Content-based Methods. Content-based methods [6, 12, 22, 26, 38] leverage item content information to model substitutable and complementary relationships. Sceptre [22] learns topic distributions from user reviews using Latent Dirichlet Allocation (LDA [1]). LVA [26] leverages Variational Autoencoders (VAEs [16]) to model item content information and provide personalized relationship inference. Other works [6, 12, 38] models either substitutable or complementary relationships. A2CF [6] leverages user reviews to extract item attributes and provide personalized substitutable recommendations. [38] uses transformers on item textual content to model substitutable relationships. P-companion [12] an encoder-decoder network to predict multiple complementary item types.

Content-based methods, compared to GNN-based methods, are more robust to noisy user-behavior data and are effective for cold-start items. However, they are limited in capturing the complex relationships of items, which are crucial in real-world applications.

2.2 Multi-modal Foundational Models

Multi-modal foundational models shows promising results in various tasks, such as image captioning and action recognition [25] and zero-shot image-to-text generation [17, 18]. These models leverage both textual and visual information and can be effectively used or transferred in many other tasks with little or no retraining. Recently, researchers have started to leverage multi-modal foundational models in recommendation tasks [20] and have shown promising results.

However, to the best of our knowledge, no work has explored multi-modal foundational models in substitutable and complementary recommendation tasks.

3 PRELIMINARIES

In this section, we formally define the research problems this paper address (*i.e.*, substitutable and complementary recommendation) and introduce the notations used throughout the paper.

Substitutable and Complementary Recommendation: Denote a set of items as $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$, where N is the number of items. We define \mathcal{X} as the set of attributes of items, where each item $v_i \in \mathcal{V}$ is associated with a set of attributes $x_i \in \mathcal{X}$. We denote $\mathcal{E} = \mathcal{E}^c \cup \mathcal{E}^s$ as the set of relationships between items, where \mathcal{E}^c and \mathcal{E}^s indicate complementary and substitutable relationships, respectively. We can easily form a item-item relationship graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where nodes represent items and edges represent relationships between items. The goal of substitutable and complementary recommenders is to learn scoring functions $\mathcal{F}^s(v_j|v_i)$ and $\mathcal{F}^c(v_j|v_i)$ that predict the substitutable and complementary relationships between items, respectively.

4 METHODOLOGY

We outline the architecture of MMSC, which is composed of three main components: multi-modal item representation learning (§ 4.1), self-supervised learning for behavior-based item relationships (§ 4.2), and a hierarchical embedding aggregation mechanism (§ 4.3). Furthermore, we detail the training objectives in § 4.4.

4.1 Multi-modal Item Representation Learning

Multi-modal information, such as item descriptions and images, is abundant in e-commerce and provides valuable insights for modeling item-item relationships. For instance, substitutable items often share similar images or descriptions. Recent advancements in multi-modal foundational models [17, 25] enable leveraging such content to learn more informative item representations. However, these pre-trained models are typically trained on large-scale multi-modal datasets and lack relational knowledge specific to inferring substitutable and complementary items, making trivial adaptation ineffective. To address this limitation, we propose a multi-modal item representation learning module that integrates a base multi-modal foundational model with a *relational fine-tuning layer*, aligning it for substitutable and complementary recommendations.

Specifically, let \mathbf{x}_i denote the multi-modal metadata of item v_i , and let \mathcal{M} represent the multi-modal foundational model. The model \mathcal{M} processes the metadata \mathbf{x}_i to generate the item representation $\mathbf{h}_i = \mathcal{M}(\mathbf{x}_i)$. Notably, \mathcal{M} can be any pre-trained multi-modal foundational model, such as CLIP [25] or BLIP-2 [17], with its parameters kept fixed during training.

In this work, we select BLIP-2 [17] as the base multi-modal foundational model due to its superior performance (Table 2). BLIP-2 generates sequential data as output, which we adapt for the substitutable and complementary recommendation task using a multi-head self-attention layer [32] as the *relational fine-tuning layer*. We

denote the output of this fine-tuning layer as \mathbf{q} , defined as:

$$\begin{aligned}\mathbf{q}_i &= \text{MHAttn}(\mathbf{h}_i) \\ \text{MHAttn}(\mathbf{h}_i) &= \text{Concat}(\text{head}_1, \dots, \text{head}_L) \mathbf{W}^O \\ \text{head}_l &= \text{Attn}(\mathbf{h}_i \mathbf{W}_l^Q, \mathbf{h}_i \mathbf{W}_l^K, \mathbf{h}_i \mathbf{W}_l^V) \\ \text{Attn}(Q, K, V) &= \text{softmax}\left(\frac{QK^T}{\sqrt{d/L}}\right)V\end{aligned}\quad (1)$$

where $\text{MHAttn}(\cdot)$ is the multi-head self-attention layers, \mathbf{h}_i is the metadata-based item representation obtained directly from the multi-modal foundational model, $\mathbf{W}_l^Q, \mathbf{W}_l^K, \mathbf{W}_l^V \in \mathbb{R}^{d \times d/L}$, and $\mathbf{W}^O \in \mathbb{R}^{L \times L}$ are learnable parameters, d is the representation dimension, and L is the number of heads.

Inspired by previous works [21, 40], we decouple the item representations for these two tasks. We employ two separate multi-head attention layers with different attention weights, $\text{MHAttn}^s(\cdot)$ and $\text{MHAttn}^c(\cdot)$, for fine-tuning the model to each task. The outputs are defined as $\mathbf{q}_i^s = \text{MHAttn}^s(\mathbf{h}_i)$ and $\mathbf{q}_i^c = \text{MHAttn}^c(\mathbf{h}_i)$. Thus, the representation of an item for substitutable and complementary recommendations is given by $\mathbf{q}_i = \{\mathbf{q}_i^s, \mathbf{q}_i^c\}$.

4.2 Self-supervised Behavior-based Item Representation Learning

User behavior (*e.g.*, co-view and co-purchase) plays a critical role in modeling substitutable and complementary relationships by uncovering associations between items. Prior works leverage user behaviors to construct item-item association graphs [21, 30, 39, 40] and utilize GNNs to exploit the topological connections among items. However, user behavior data lacks ground-truth relationships between items and often contains unrelated items, introducing noise into the item-item associations. To address this, we propose a self-supervised, user behavior-based item relationship learning paradigm that denoises user behaviors and leverages meta-paths [10, 31, 34] to capture complex item associations.

4.2.1 User Behavior Encoder via Meta-paths. Prior works [21, 40] that explicitly model user behavior construct item-item graphs by connecting items based on specific behavior types (*e.g.*, co-view and co-purchase). They decouple the heterogeneous item-item graph into two homogeneous graphs and apply GNNs. However, this approach overlooks valuable transitive associations between items.

To address this, we propose to learn items representations using carefully designed meta-paths for inferring substitutable and complementary relationships. A *meta-path* defines a structured pathway connecting nodes of specific types through specific relations in a heterogeneous graph. Let s be a substitutable relationship and c be a complementary relationship. A meta-path $v_1 \xrightarrow{s} v_2 \xrightarrow{c} v_3 \xrightarrow{s} v_4$ connects v_1 and v_4 through substitutable and complementary relations. In this case, v_1 and v_4 are likely to be complementary items, as they are linked through a substitutable item v_2 and a complementary item v_3 . These transitive relationships, which are crucial for modeling substitutable and complementary connections, are lost when decoupling the item-item graph.

We denote the set of meta-paths as $\Phi = \{\phi_1, \phi_2, \dots, \phi_K\}$, where K is the number of meta-paths and ϕ_k is the k -th meta-path. The set of neighbors of item v_i through meta-path ϕ_k is denoted as $\mathcal{N}_i^{\phi_k}$. To aggregate the neighborhood information for item v_i , we employ

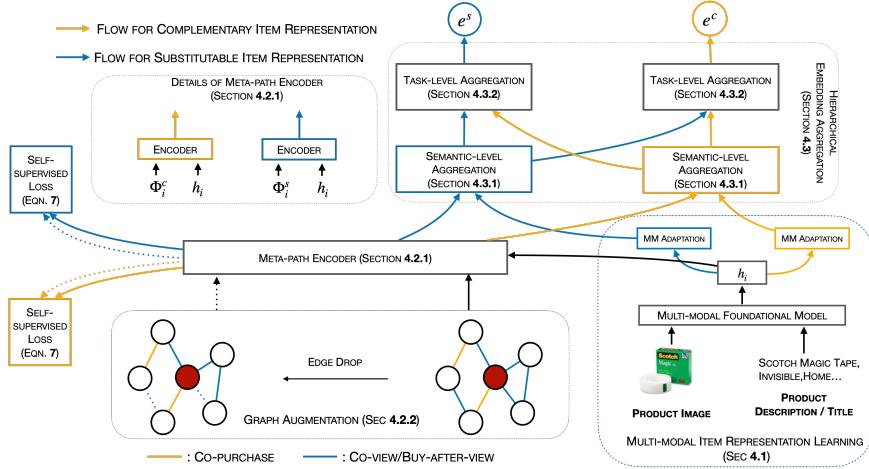


Figure 3: The model architecture of MMSC.

a multi-head node-level self-attention mechanism followed by a path-level attention mechanism.

The node level attention aggregates $\mathcal{N}_i^{\phi_k}$ of item v_i to obtain the node-level representation $z_i^{\phi_k}$. Specifically, we first compute the attention score between item v_i and its neighbors $\mathcal{N}_i^{\phi_k}$ as follows:

$$\alpha_{ij}^{\phi_k} = \frac{\exp(\text{LeakyReLU}(\mathbf{W}_{\phi_k}^\top [\mathbf{h}_i || \mathbf{h}_j]))}{\sum_{v_t \in \mathcal{N}_i^{\phi_k}} \exp(\text{LeakyReLU}(\mathbf{W}_{\phi_k}^\top [\mathbf{h}_i || \mathbf{h}_t]))} \quad (2)$$

where \mathbf{W}_{ϕ_k} is the weight matrix for meta-path ϕ_k , $||$ denotes the concatenation operation, ${}^\top$ represents transposition, and LeakyReLU [36] is the nonlinear activation. We use the output of the multi-modal foundational model \mathbf{h}_i as the input. This allows us to inject the multi-modal information of items to learn the item-item relationships. The node-level attention is then computed as:

$$z_i^{\phi_k} = \sigma \left(\sum_{v_j \in \mathcal{N}_i^{\phi_k}} \alpha_{ij}^{\phi_k} \mathbf{W}_a^\top \mathbf{h}_j \right) \quad (3)$$

where $\mathbf{W}_a \in \mathbb{R}^{d \times d}$ is the learnable weight matrix and σ is another nonlinear activation ELU [8]. To attend to information from different representation subspaces and stabilize training, we adopt the multi-head self-attention mechanism for more robust learning [32–34]. Specifically, we repeat the node-level attention for T times with T independent attention mechanisms and concatenate the learned embeddings. Therefore, the node-level attention is computed as:

$$z_i^{\phi_k} = \prod_{t=1}^T \sigma \left(\sum_{v_j \in \mathcal{N}_i^{\phi_k}} \alpha_{ij}^{\phi_k, t} (\mathbf{W}_a^\top)^\top \mathbf{h}_j \right) \quad (4)$$

where $\alpha_{ij}^{\phi_k, t}$ is the attention coefficient of the t -th attention.

Different meta-paths capture different fine-grained associations between items. Therefore, each node level representation $z_i^{\phi_k}$ are complementary to each other. To learn a more informative representation of item v_i , we aggregate the node-level representations $z_i^{\phi_k}$ through all meta-paths to obtain the semantic-level representation

p_i . We define the importance of each meta-path $\beta_i^{\phi_k}$ as:

$$\begin{aligned} w_i^{\phi_k} &= \sum_{v_j \in \mathcal{N}_i^{\phi_k}} s^\top \tanh(\mathbf{W}_b \cdot z_j^{\phi_k} + \mathbf{b}) \\ \beta_i^{\phi_k} &= \frac{\exp(w_i^{\phi_k})}{\sum_{k'=1}^K \exp(w_i^{\phi_{k'}})} \end{aligned} \quad (5)$$

where $\mathbf{W}_b \in \mathbb{R}^d$, $\mathbf{b} \in \mathbb{R}^d$ and $s \in \mathbb{R}^d$ are learnable vectors, and $\tanh(\cdot)$ is the tanh activation function. To compute the final representation p_i of item v_i , we have:

$$p_i = \sum_{k=1}^K \beta_i^{\phi_k} z_i^{\phi_k} \quad (6)$$

Similar to the multi-modal item representation, we decouple the item representations. We denote the representations for substitutable and complementary recommendations as p_i^s and p_i^c , respectively. Specifically, p_i^s is computed along a set of carefully designed meta-paths, Φ^s , which capture substitutable relationships, while p_i^c is computed along a set of meta-paths, Φ^c , that capture complementary relationships. A detailed list of the meta-paths used is provided in § 5.1.4. Separate user behavior encoders are employed for Φ^s and Φ^c . The output of the behavior-based item relationship learning module is given by $p_i = \{p_i^s, p_i^c\}$.

4.2.2 Self-supervised Behavior Denoising. In order to learn robust item representations from the item-item graph, we need to force the model to learn representations that are invariant to structural perturbations, which simulate potential noise in the item-item associations. To this end, we propose a self-supervised learning objective aimed at denoising and enhancing the robustness of item representations. We utilize graph-level dropout [35], a type of structural perturbation where a fraction of edges (representing user behaviors) in the item-item graph are randomly removed to create an alternative view of the graph. This perturbed graph is denoted as $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$. We apply the same meta-path encoder in § 4.2.1 and compute the alternative item representations p'_i .

We treat different views of the same node as positive pairs (e.g., \mathbf{p}_i and \mathbf{p}'_i) and views of different nodes as negative pairs. To enhance the robustness of item representations, we minimize the contrastive loss between the positive and negative pairs. Specifically, we adopt the InfoNCE [11] as our self-supervised learning objective:

$$\begin{aligned}\mathcal{L}_{\text{self}}^s &= -\frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \log \frac{\exp(s(\mathbf{p}_i^s, \mathbf{p}_i^{s'})/\tau)}{\sum_{j \neq i} \exp(s(\mathbf{p}_i^s, \mathbf{p}_j^s)/\tau)} \\ \mathcal{L}_{\text{self}}^c &= -\frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \log \frac{\exp(s(\mathbf{p}_i^c, \mathbf{p}_i^{c'})/\tau)}{\sum_{j \neq i} \exp(s(\mathbf{p}_i^c, \mathbf{p}_j^c)/\tau)} \\ \mathcal{L}_{\text{self}} &= \mathcal{L}_{\text{self}}^s + \mathcal{L}_{\text{self}}^c\end{aligned}\quad (7)$$

where s is the cosine similarity function, and τ is the temperature parameter. We will discuss the optimization in § 4.4.

4.3 Hierarchical Representation Aggregation

We present a hierarchical representation aggregation strategy that combines multi-modal and behavior-based item representations at the semantic level and integrates substitutable and complementary representations at the task level, leveraging neural gating mechanisms for effective fusion.

4.3.1 Semantic-Level Aggregation. The multi-modal item representation (§ 4.1) encapsulates the item's metadata, while the behavior-based item representation (§ 4.2) captures high-order associations between items. Intuitively, aggregating these two representations ensures that the aggregated representation effectively integrates both aspects, which are essential for modeling substitutable and complementary relationships.

We design a gating mechanism to select salient feature dimensions from the multi-modal \mathbf{q}_i and behavior-based \mathbf{p}_i item representations, producing the final item representation. Specifically, we use a neural gating mechanism that learns a non-linear gate, \mathbf{g} , to control the flow of information between these representations. Without loss of generality, we demonstrate the gating mechanism for learning the item representation \mathbf{a}_i^s for the substitutable recommendation task. The gating mechanism is defined as:

$$\begin{aligned}\mathbf{a}_i^s &= \text{Gating}_{\text{sem}}^s(\mathbf{q}_i^s, \mathbf{p}_i^s) \\ \text{Gating}_{\text{sem}}^s(\mathbf{q}_i^s, \mathbf{p}_i^s) &= \mathbf{g} \odot \mathbf{p}_i^s + (1 - \mathbf{g}) \odot \mathbf{q}_i^s, \\ \mathbf{g} &= \sigma(\mathbf{W}_{g_1} \mathbf{p}_i^s + \mathbf{W}_{g_2} \mathbf{q}_i^s + \mathbf{b}_g)\end{aligned}\quad (8)$$

where σ is the sigmoid function, \odot denotes the element-wise multiplication, \mathbf{W}_{g_1} , \mathbf{W}_{g_2} , and \mathbf{b}_g are learnable parameters. Note that we can compute $\mathbf{a}_i^c = \text{Gating}_{\text{sem}}^c(\mathbf{q}_i^c, \mathbf{p}_i^c)$ through another separate gating function $\text{Gating}_{\text{sem}}^c$ with separate learnable parameters for complementary recommendation.

4.3.2 Task-Level Aggregation. While we learn item representations for substitutable and complementary recommendation tasks separately, these representations can still benefit each other [21, 40]. Substitutable recommendations focus on capturing similarities between items, while complementary recommendations emphasize their co-occurrence patterns or relationships. Combining these representations enables the model to leverage shared insights, such as overlapping features or common interaction contexts, enriching the representation of each item. To achieve this, we again fuse the item representations from these two tasks using neural gating

Template of LLM Augmentation for Substitutable Items:

Answer the following question with yes or no only. I am considering two items "`{asin_x}`" and "`{asin_y}`". If one of them is out-of-stock, can I buy the other one to serve the same purpose?

Template of LLM Augmentation for Complementary Items:

Here is one example of two items that if I bought one item then I can also buy the other to serve as a complementary: The two items are Sheaffer(R) Pen Refills, Ink Cartridges, Jet Black, Pack Of 5 and Sheaffer 100 Red Fountain Pen 9307-0.

Answer the following question with yes or no only. I am considering two items `{asin_x}` and `{asin_y}`, if I bought one item, then can I buy the other to serve as a complementary?

Table 1: LLM Augmentation prompts for user behaviors (§ 4.4.1). We replace `{asin_x}` and `{asin_y}` with the metadata (e.g., description and title) of the items with corresponding ASINs.

mechanisms.

$$\mathbf{e}_i^s = \text{Gating}_{\text{task}}^s(\mathbf{a}_i^s, \mathbf{a}_i^c), \quad \mathbf{e}_i^c = \text{Gating}_{\text{task}}^c(\mathbf{a}_i^s, \mathbf{a}_i^c) \quad (9)$$

The final representation of item v_i is $\mathbf{e}_i = \{\mathbf{e}_i^s, \mathbf{e}_i^c\}$.

4.4 LLM-Augmented and Multi-Task Learning

We leverage LLMs to augment the user behavior data and jointly optimize the objective for substitutable and complementary recommendation tasks as well as the self-supervised objective.

4.4.1 LLM-augmented Learning. In prior works [12, 21, 37, 40], ground truth relationships are estimated from user behaviors, such as *co-view* for substitutable relationships and *co-purchase* for complementary relationships. However, these estimates are often noisy, as co-purchase data may include unrelated or even substitutable items (Figure 1). To address this, we leverage LLMs to augment user behaviors by filtering out noise using carefully designed prompts (Table 1) for inferring substitutable and complementary relationships. Specifically, we sample a subset of user behaviors and use LLMs to evaluate whether item pairs are substitutable or complementary. We show that LLM-augmented user behaviors are more reliable in inferring item relationships, as demonstrated in § 5.1.2.

While LLMs excel at assessing item-item relationships, their slow inference speed makes them impractical for large-scale modeling. Consequently, computing item representations remains essential. To address this, we sample a subset of user behaviors to construct the filtered, LLM-augmented training data, denoted as \mathcal{E}_{LLM} .

4.4.2 Multi-task Learning. We jointly optimize the objectives for substitutable and complementary recommendations along with the self-supervised objective (eq. (7)). For substitutable and complementary recommendations, we use the triplet loss, defined as:

$$\begin{aligned}\mathcal{L}_{\text{triplet}}^s &= \sum_{(v_i, v_j^+) \in \mathcal{E}_{\text{LLM}}^s, v_k^- \in \mathcal{E}^s} \max(0, \text{margin} + s(\mathbf{e}_i^s, \mathbf{e}_{j^+}^s) - s(\mathbf{e}_i^s, \mathbf{e}_{k^-}^s)) \\ \mathcal{L}_{\text{triplet}}^c &= \sum_{(v_i, v_j^+) \in \mathcal{E}_{\text{LLM}}^c, v_k^- \in \mathcal{E}^c} \max(0, \text{margin} + s(\mathbf{e}_i^c, \mathbf{e}_{j^+}^c) - s(\mathbf{e}_i^c, \mathbf{e}_{k^-}^c)) \\ \mathcal{L}_{\text{triplet}} &= \mathcal{L}_{\text{triplet}}^s + \mathcal{L}_{\text{triplet}}^c\end{aligned}\quad (10)$$

where $s(\cdot, \cdot)$ denotes the cosine similarity, \mathbf{e}_i , \mathbf{e}_j^+ , and \mathbf{e}_k^- are the representations of item v_i , positive item v_j^+ , and negative item v_k^- , respectively. The margin controls the distance between positives and negatives. Note that the positive pairs (v_i, v_j^+) are sampled from

DATASET	OFFICE			TOOLS			TOYS			HOME			ELECTRONICS			
	SUBSTITUTABLE	H@10	M@10	N@10	H@10	M@10	N@10									
GRAPH NEURAL NETWORKS																
GATNE-I [3]	0.629	0.384	0.443	0.634	0.367	0.430	0.659	0.435	0.489	0.555	0.324	0.379	0.611	0.348	0.410	
GAT [33]	0.758	0.490	0.554	0.755	0.494	0.557	0.764	0.511	0.572	0.728	0.455	0.520	0.728	0.466	0.528	
HAN [34]	0.796	0.500	0.571	0.765	0.460	0.532	0.759	0.492	0.556	0.779	0.500	0.567	0.775	0.469	0.542	
SUBSTITUTE AND COMPLEMENTARY RECOMMENDERS																
DecGCN [21]	0.561	0.302	0.363	0.637	0.344	0.413	0.573	0.318	0.378	0.533	0.265	0.328	0.617	0.318	0.396	
DHGAN [40]	0.866*	0.573	0.644*	0.916*	0.652	0.716*	0.901*	0.658*	0.717*	0.930*	0.727*	0.777*	0.916*	0.652*	0.716*	
MULTI-MODAL FOUNDATIONAL MODELS																
Blip2 [17]	0.779	0.519	0.581	0.889	0.658*	0.714	0.857	0.587	0.652	0.901	0.692	0.744	0.814	0.547	0.611	
CLIP [25]	0.841	0.580*	0.642	0.880	0.644	0.701	0.751	0.527	0.583	0.911	0.685	0.740	0.688	0.483	0.532	
MMSC	0.980	0.782	0.831	0.989	0.841	0.877	0.978	0.827	0.864	0.984	0.830	0.869	0.989	0.816	0.859	
% Improvement	+13.1%	+36.5%	+29.0%	+7.9%	+29.0%	+22.5%	+8.5%	+25.7%	+20.5%	+5.8%	+14.2%	+11.8%	+8.0%	+25.2%	+20.0%	
COMPLEMENTARY	H@10	M@10	N@10	H@10	M@10	N@10	H@10	M@10	N@10	H@10	M@10	N@10	H@10	M@10	N@10	
GRAPH NEURAL NETWORKS																
GATNE-I [3]	0.690	0.487	0.536	0.780	0.521	0.583	0.786	0.570	0.622	0.807	0.594	0.645	0.781	0.521	0.583	
GAT [33]	0.769*	0.567*	0.615*	0.844*	0.669*	0.712*	0.844*	0.609*	0.665*	0.875	0.713*	0.752*	0.830*	0.589*	0.647*	
HAN [34]	0.769	0.534	0.590	0.646	0.415	0.470	0.766	0.530	0.587	0.672	0.446	0.500	0.775	0.469	0.542	
SUBSTITUTE AND COMPLEMENTARY RECOMMENDERS																
DecGCN [21]	0.630	0.379	0.440	0.727	0.459	0.523	0.573	0.318	0.378	0.573	0.318	0.378	0.620	0.372	0.415	
DHGAN [40]	0.761	0.540	0.529	0.826	0.585	0.643	0.761	0.529	0.529	0.897	0.681	0.733	0.825	0.546	0.612	
MULTI-MODAL FOUNDATIONAL MODELS																
Blip2 [17]	0.727	0.482	0.541	0.834	0.621	0.672	0.727	0.482	0.541	0.854	0.628	0.682	0.632	0.428	0.476	
CLIP [25]	0.763	0.553	0.603	0.803	0.618	0.662	0.708	0.493	0.545	0.838	0.644	0.690	0.566	0.405	0.443	
MMSC	0.964	0.789	0.832	0.986	0.885	0.911	0.980	0.863	0.892	0.980	0.872	0.899	0.978	0.821	0.860	
% Improvement	+25.4%	+39.2%	+35.3%	+16.8%	+32.3%	+28.0%	+16.1%	+62.8%	+34.1%	+9.3%	+22.3%	+19.6%	+17.8%	+39.4%	+32.9%	

Table 2: Substitutable and complementary recommendation results. The best performance is highlighted in bold, and the second-best is marked with an asterisk (*). We calculate the percentage improvement relative to the second-best baseline. We observe an average improvement of approximately 26.1% in M@10 for substitutable recommendations and 39.2% in M@10 for complementary recommendations.

\mathcal{E}_{LLM} , while the negative pairs (v_i, v_k^-) are randomly sampled from the user behavior data \mathcal{E} .

Finally, the overall multi-task learning objective is:

$$\mathcal{L} = \mathcal{L}_{\text{triplet}} + \lambda \mathcal{L}_{\text{self}} \quad (11)$$

where λ is the hyperparameter that controls the weight of the self-supervised objective.

5 EXPERIMENTS

We conduct extensive experiments on five real-world datasets and introduce the following research questions to guide this section: **RQ1:** How does MMSC perform compared to state-of-the-art methods? **RQ2:** How do different components of MMSC contribute to the overall performance? **RQ3:** How effective is MMSC in modeling relationships for cold-start items? **RQ4:** How does MMSC perform on items with noisy and sparse user behaviors? **RQ5:** How do the different parameters affect the performance of MMSC?

5.1 Datasets and Experimental Details

5.1.1 Datasets and preprocessing. We conduct experiments on the Amazon review dataset [23], following previous work [21, 27, 40]. Specifically, we choose the following five categories: Office Products, Tools and Home Improvement, Electronics, Toys and Games, and Home and Kitchen. We present the statistics of the datasets here¹. To ensure the quality of items, we filter out items with no image or text information. For each item, we regard its

¹https://anonymous.4open.science/r/MMSC_Supplementary-4CD4/

title and description as textual information and its image as visual information. Following previous works [12, 21, 40], we formulate the substitutable and complementary recommendation task as the link prediction task. We approximate *co-view* and *buy-after-view* as substitutable relationships and *co-purchase* as complementary relationships. Then, for each item, we randomly sample one edge for each type of relationships (*i.e.*, substitutable and complementary) as test candidate and the rest as training. We employ large language models (LLMs) to refine the candidate test set by utilizing carefully designed prompts (Table 1), ensuring high quality in the test relationships. The refined test set is denoted as Y_{sub} and Y_{com} for substitutable and complementary relationships, respectively.

5.1.2 Dataset Analysis. We conduct a case study on the user behavior noise and LLM's effectiveness in inferring item relationships. Based on 100 samples from the Office and Electronics datasets, user behavior achieves 78% and 24.5% accuracy for inferring substitutable and complementary relationships, while LLM labels achieve 94.7% and 57.9%. We use human labels as ground truth. This shows that user behavior is particularly noisy for complementary relationships, and LLMs provide more reliable estimates for both.

5.1.3 Evaluation Protocol. Per test relationship in Y_{sub} and Y_{com} , we sample 1000 negatives uniformly. We rank the test relationship against the negatives and evaluate the performance using Hit Ratio (H@10), Mean Reciprocal Rank (M@10), and NDCG@10 (N@10).

5.1.4 Implementation Details. We sample five negative samples per train relationship. We use publicly available implementations

OFFICE	SUBSTITUTABLE			COMPLEMENTARY		
	H@10	M@10	N@10	H@10	M@10	N@10
MMSC	0.980	0.782	0.831	0.964	0.789	0.832
w/o SSL (eq. (7))	0.968	0.749	0.803	0.958	0.757	0.801
w/o TA (§ 4.3.2)	0.980	0.776*	0.827*	0.965*	0.775*	0.822*
w/o SSL & TA	0.976*	0.757	0.811	0.867	0.649	0.731
w/o MM (§ 4.1)	0.911	0.603	0.678	0.837	0.594	0.660
w/o BM (§ 4.2)	0.849	0.560	0.630	0.727	0.482	0.541
w/o \mathcal{E}_{LLM} (§ 4.4)	0.967	0.727	0.786	0.940	0.718	0.772
w/o 3rd-hop (§ 5.1.4)	0.972	0.750	0.804	0.953	0.730	0.785
ELECTRONICS	H@10	M@10	N@10	H@10	M@10	N@10
MMSC	0.989	0.816	0.859	0.978	0.821	0.860
w/o SSL (eq. (7))	0.981	0.797	0.844	0.969	0.774	0.822
w/o TA (§ 4.3.2)	0.98	0.798	0.844	0.970	0.787	0.832
w/o SSL & TA	0.983	0.789	0.837	0.959	0.749	0.800
w/o MM (§ 4.1)	0.981	0.772	0.824	0.965	0.787	0.831
w/o BM (§ 4.2)	0.891	0.622	0.687	0.732	0.487	0.545
w/o \mathcal{E}_{LLM} (§ 4.4)	0.988*	0.812*	0.856*	0.970*	0.789*	0.833*
w/o 3rd-hop (§ 5.1.4)	0.981	0.785	0.834	0.966	0.785	0.829

Table 3: Ablation Results on Amazon Office and Electronics. w/o SSL means the model was trained without the self-supervised learning objective. TA corresponds to the task-level embedding aggregation in § 4.3.2. MM denotes the multi-modal learning component in § 4.1. BM denotes the behavior-based learning component in § 4.2, and \mathcal{E}_{LLM} denotes the LLM-augmented training in § 4.4.

for the baselines. We use Adam optimizer and tune the learning rate in the range $\{10^{-4}, 10^{-3}, 10^{-2}\}$. We set dropout to 0.2 and tune α in § 4.4.2 in the range $\{10^{-3}, 5^{-2}, 10^{-2}\}$. For every dataset, we fix the size of \mathcal{E}_{LLM} to be 500K. We use Flan-T5-XXL [7] as the LLM. We train every model till convergence, repeat five times with different random seeds, and report the average performance.

We explore item-item associations within the 3-hop neighborhood of each item. The meta-paths we used are as follows: $\Phi^s = \{v_1 \xrightarrow{s} v_2, v_1 \xrightarrow{s} v_2 \xrightarrow{s} v_3, v_1 \xrightarrow{s} v_2 \xrightarrow{s} v_3 \xrightarrow{s} v_4\}$ and $\Phi^c = \{v_1 \xrightarrow{c} v_2, v_1 \xrightarrow{c} v_2 \xrightarrow{s} v_3, v_1 \xrightarrow{c} v_2 \xrightarrow{s} v_3 \xrightarrow{c} v_4, v_1 \xrightarrow{s} v_2 \xrightarrow{c} v_3 \xrightarrow{s} v_4, v_1 \xrightarrow{s} v_2 \xrightarrow{s} v_3 \xrightarrow{s} v_4\}$.

5.1.5 Baselines. We choose baselines from three categories: Graph neural networks (GATNE-I [3], GAT [33] and HAN [34]), substitutable and complementary recommenders (DecGCN [21] and DHGAN [40]), and multi-modal foundational models (CLIP [25] and Blip2 [17]). We do not include methods that are not open-sourced.

5.2 Substitutable and Complementary Recommendation Results (RQ1)

We present the results in Table 2. MMSC achieves a significant average of 26.1% improvement in M@10 for substitutable and 39.2% improvement in M@10 for complementary recommendation. DHGAN [40] is the second-best performing baseline for substitutable recommendation, and GAT [33] is the second-best performing baseline for complementary recommendation. The multi-modal foundational models, Blip2 [17] and CLIP [24], perform poorly compared to the others, showing that the multi-modal foundational models are not designed to model relationships between items.

We see more substantial performance gain in complementary recommendation. We attribute this to the noisy nature and the diversity of the complementary relationships, which poses a more

OFFICE	SUBSTITUTABLE			COMPLEMENTARY		
	H@10	M@10	N@10	H@10	M@10	N@10
GAT	0.188	0.078	0.104	0.049	0.022	0.028
DHGAN	0.247	0.100	0.135	0.143	0.064	0.082
Blip2-SA	0.834*	0.532*	0.604*	0.702*	0.435*	0.499*
MMSC	0.921	0.702	0.754	0.775	0.505	0.568
Δ	+10.4%	+32.0%	+24.8%	+10.4%	+16.1%	+13.8%
ELECTRONICS	H@10	M@10	N@10	H@10	M@10	N@10
GAT	0.161	0.066	0.088	0.080	0.025	0.038
DHGAN	0.224	0.106	0.133	0.183	0.093	0.113
Blip2-SA	0.814*	0.495*	0.571*	0.550*	0.302*	0.360*
MMSC	0.922	0.698	0.753	0.687	0.416	0.480
Δ	+13.3%	+41.0%	+31.9%	+24.9%	+37.7%	+33.3%

Table 4: Cold-start Results on Office and Electronics.

significant challenge to the baselines using only user behavior-based or content-based information. In contrast, MMSC leverages both user behavior-based and content-based information, which helps to mitigate the noise and capture diverse relationships.

5.3 Ablation Study (RQ2)

We conduct ablation studies to understand the contribution of different components of MMSC (Table 3). We empirically observe that the self-supervised learning objective significantly improves performance, *i.e.*, improving M@10 by 3.3% in substitutable and 5.1% in complementary recommendation, highlighting its effectiveness in handling noisy complementary relationships. Task-level embedding aggregation also notably enhances performance, improving M@10 by 1.4% (substitutable) and 3.1% (complementary). Multi-modal and behavior-based learning components are complementary and contribute significantly to the performance, while the behavior-based module contributes more to the performance gain. This is expected as the behavior-based module is designed to capture the fine-grained associations between items, where the multi-modal module captures general item information, underlining the importance of jointly leveraging user behavior and content-based information. We see a more significant gain on the office dataset with LLM augmentation. We suspect the reason is that we fixed the size of \mathcal{E}_{LLM} to be 500K, which might not be sufficient for the larger electronics dataset. We also observe that the model without 3rd-hop neighbors (*i.e.*, we constrain the length of meta-paths to be less than 3) performs poorly, suggesting that complex meta-paths are effective in capturing fine-grained item associations.

5.4 Cold-star Inference (RQ3)

5.4.1 Cold-star Inference Procedure. We explore the effectiveness of MMSC under cold-start settings. We define the cold-start items as \mathcal{V}' , where $\mathcal{V} \cap \mathcal{V}' = \emptyset$ and $\nexists \mathcal{E}_{ij'}, v_i \in \mathcal{V}$ and $v_{j'} \in \mathcal{V}'$, *i.e.*, items that are not seen in training. We first leverage the multi-modal foundational model to obtain the initial item representation \mathbf{h}' for cold-start items. Then, we use \mathbf{h}' as a query and search \mathbf{h} for the top-k most similar items (denoted as C) in the existing item inventory, *i.e.*, items that appeared in the training set. Then, we mean pool the final representations of the selected items to obtain the final representation of items in C . Note that all model parameters are fixed during cold-start inference. The final representation of

the cold-start item is:

$$\mathbf{e}'_j = \frac{1}{|C_{j'}|} \sum_{v_i \in C_{j'}} \mathbf{e}_i, \quad \forall v_{j'} \in \mathcal{V}' \quad (12)$$

We adapt GAT and DHGAN using the same inference procedure.

5.4.2 Cold-start Results. We present the results in Table 4. Notably, MMSC significantly outperforms baselines by an average of 36.5% (substitutable) and 26.9% (complementary). GAT and DHGAN perform poorly in cold-start scenarios since their reliance on graph homophily limits generalizability to disconnected items. While Blip2 adapts better to cold-start scenarios through multi-modal information, it still underperforms compared to MMSC, suggesting multi-modal information alone might not be sufficient for cold-start scenarios and highlighting the advantage of MMSC's combined use of user behavior and content information in cold-start settings.

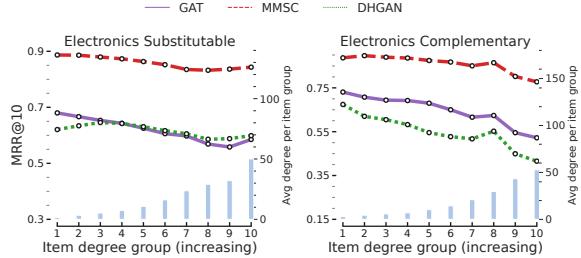


Figure 4: Performance of on Electronics w.r.t. different item degree groups. MMSC shows greater improvement on items with fewer behavior data (Group 1-3) in substitutable recommendation.

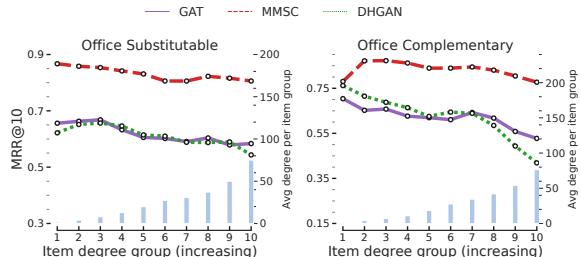


Figure 5: Performance of on Office w.r.t. different item degree groups. Similarly, MMSC shows greater improvement on items with fewer behavior data (Group 1-3) in substitutable recommendation.

5.5 Qualitative Analysis (RQ4)

5.5.1 Performance on different item degree groups. We compare MMSC against baselines in Figure 4 and Figure 5, grouping items into 10 equal-sized groups based on node degree in the item-item behavior graph. MMSC outperforms the baselines across all groups for both substitutable and complementary recommendation, showing greater improvements for items with fewer behavior data points (Groups 1-3) in substitutable recommendation and for items with more behavior data (Groups 8-10) in complementary recommendation. We attribute these improvements to the self-supervised learning objective and multi-modal component, which effectively denoise user behavior data. MMSC outperforms the baselines on items with fewer behavior data in complementary recommendation (Group 1-3). The gains are smaller compared to substitutable recommendation, suggesting the complexity and challenge of modeling complementary relationships.

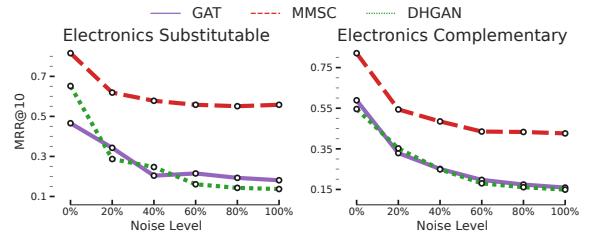


Figure 6: Performance of on Electronics w.r.t. different noise levels.

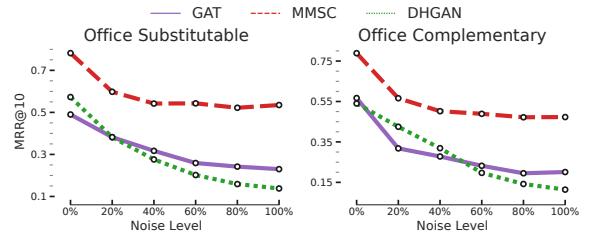


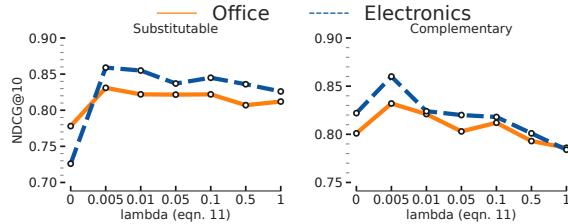
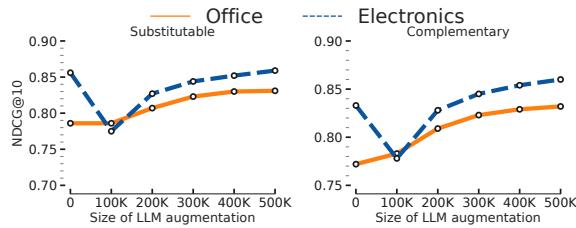
Figure 7: Performance of on Office w.r.t. different noise levels.

5.5.2 Robustness to noise. We study how robust MMSC is to noise in user behaviors (Figure 6 and Figure 7). We add noise to the user behavior data by randomly sampling non-existing behaviors (*i.e.*, edges in item-item graph). 0% noise indicates no noise, while 100% noise indicates the noisy behaviors and the original behaviors are of equal size. The performance improvement of MMSC increases as noise increases, suggesting that MMSC is more robust to noise in both recommendation tasks, and the performance stays stable even when the noise level is high (*i.e.*, $\geq 60\%$). This suggests that MMSC is more effective in learning from noisy user behavior data.

5.6 Sensitivity Analysis (RQ5)

5.6.1 Sensitivity to λ (§ 4.4.2). MMSC achieves optimal results at $\lambda = 0.005$ (Figure 8), with performance improving as λ increases up to this point. We attribute this improvement to a balanced trade-off between the self-supervised and supervised objectives; overly large λ values degrade performance by overshadowing the supervised objective. Additionally, substitutable recommendation performance is less sensitive to variations in λ than complementary recommendation, suggesting that the complementary recommendation performance can benefit more from tuning λ .

5.6.2 Sensitivity to \mathcal{E}_{LLM} (§ 4.4.1). We vary the size of \mathcal{E}_{LLM} 100K to 500K (Figure 9), 0 being no LLM augmentation, which corresponds to the performance of MMSC without LLM augmentation in Table 3. MMSC's performance improves with larger $|\mathcal{E}_{LLM}|$. LLM augmentation consistently boosts performance on the Office dataset, regardless of augmented label size. In the Electronics dataset, however, LLM augmentation's effect depends on the size: performance drops at 100K and 200K, possibly because these sizes are insufficient for the dataset's larger scale. Notably, LLM augmentation reduces the number of training samples but likely improves their quality, which may explain the maintained or improved performance. This efficiency is beneficial in large-scale online settings, where fewer but higher-quality samples can enhance training.

Figure 8: Performance of different λ in § 4.4.2.Figure 9: Performance varying size of E_{LLM} in § 4.4.1.

5.7 Discussion

MMSC outperforms baselines in both substitutable and complementary recommendation, including cold-start scenarios. We empirically demonstrate MMSC’s robustness in learning effective representations from noisy user behaviors. We expect MMSC to be particularly beneficial in real-world applications with sparse and noisy user behaviors. However, MMSC has limitations. First, it assumes static relationships and neglects temporal dynamics. Second, it does not explicitly model independent item-item relationships during training. We leave these limitations for future work.

6 CONCLUSION

We identified two critical challenges in modeling substitutable and complementary relationships: noisy user behavior data and heavy-tailed user-behavior distributions. To address these, we proposed MMSC, a multi-modal relational item representation learning framework leveraging item associations and content information. Our empirical results show that MMSC can effectively learn excels in modeling items with sparse and noisy associations.

REFERENCES

- [1] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.
- [2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.
- [3] Yukuo Cen, Xu Zou, Jianwei Zhang, Hongxian Yang, Jingren Zhou, and Jie Tang. 2019. Representation Learning for Attributed Multiplex Heterogeneous Network. *CoRR abs/1905.01669* (2019). arXiv:1905.01669 <http://arxiv.org/abs/1905.01669>
- [4] Haoming Chen, Yetian Chen, Jingjing Meng, Yang Jiao, Yikai Ni, Yan Gao, Michinari Momma, and Yi Sun. 2023. Improving product search with season-aware query-product semantic similarity. In *Companion Proceedings of the ACM Web Conference 2023*. 864–868.
- [5] Huajie Chen, Jiyuan He, Weisheng Xu, Tao Feng, Ming Liu, Tianyu Song, Runfeng Yao, and Yuanyuan Qiao. 2023. Enhanced multi-relationships integration graph convolutional network for inferring substitutable and complementary items. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 4157–4165.
- [6] Tong Chen, Hongzhi Yin, Guanhua Ye, Zi Huang, Yang Wang, and Meng Wang. 2020. Try This Instead: Personalized and Interpretable Substitute Recommendation. *CoRR abs/2005.09344* (2020). arXiv:2005.09344 <https://arxiv.org/abs/2005.09344>
- [7] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling Instruction-Finetuned Language Models. *arXiv:cs.LG/2210.11416* <https://arxiv.org/abs/2210.11416>
- [8] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. 2016. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *arXiv:cs.LG/1511.07289* <https://arxiv.org/abs/1511.07289>
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina N. Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. <https://arxiv.org/abs/1810.04805>
- [10] Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17)*. Association for Computing Machinery, New York, NY, USA, 135–144. <https://doi.org/10.1145/3097983.3098036>
- [11] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research)*, Yee Whye Teh and Mike Titterington (Eds.), Vol. 9. PMLR, Chia Laguna Resort, Sardinia, Italy, 297–304. <https://proceedings.mlr.press/v9/gutmann10a.html>
- [12] Junheng Hao, Tong Zhao, Jin Li, Xin Luna Dong, Christos Faloutsos, Yizhou Sun, and Wei Wang. 2020. P-companion: A principled framework for diversified complementary product recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2517–2524.
- [13] Xinrui He, Shuo Liu, Jacky Keung, and Jingrui He. 2024. Co-clustering for federated recommender system. In *Proceedings of the ACM Web Conference 2024*. 3821–3832.
- [14] Xinrui He, Tianxin Wei, and Jingrui He. 2023. Robust basket recommendation via noise-tolerated graph contrastive learning. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 709–719.
- [15] Yang Jiao, Fan Yang, Yetian Chen, Yan Gao, Jia Liu, and Yi Sun. 2024. Rethinking sequential relationships: Improving sequential recommenders with inter-sequence data augmentation. In *Companion Proceedings of the ACM Web Conference 2024*. 641–645.
- [16] Diederik P Kingma and Max Welling. 2022. Auto-Encoding Variational Bayes. *arXiv:stat.ML/1312.6114* <https://arxiv.org/abs/1312.6114>
- [17] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. *arXiv:cs.CV/2301.12597* <https://arxiv.org/abs/2301.12597>
- [18] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. *arXiv:cs.CV/2201.12086* <https://arxiv.org/abs/2201.12086>
- [19] Yunzhe Li, Junting Wang, Hari Sundaram, and Zhining Liu. 2025. A zero-shot generalization framework for llm-driven cross-domain sequential recommendation. *arXiv preprint arXiv:2501.19232* (2025).
- [20] Qidong Liu, Jiaxi Hu, Yutian Xiao, Xiangyu Zhao, Jingtong Gao, Wanyu Wang, Qing Li, and Jiliang Tang. 2024. Multimodal Recommender Systems: A Survey. *arXiv:cs.IR/2302.03883* <https://arxiv.org/abs/2302.03883>
- [21] Yiding Liu, Yulong Gu, Zhuoyue Ding, Junchao Gao, Ziyi Guo, Yongjun Bao, and Weipeng Yan. 2020. Decoupled Graph Convolution Network for Inferring Substitutable and Complementary Items. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 2621–2628. <https://doi.org/10.1145/3340531.3412695>
- [22] Julian McAuley, Rahul Pandey, and Jure Leskovec. 2015. Inferring Networks of Substitutable and Complementary Products. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '15)*. Association for Computing Machinery, New York, NY, USA, 785–794. <https://doi.org/10.1145/2783258.2783381>
- [23] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 43–52.
- [24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.
- [25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. *arXiv:cs.CV/2103.00020* <https://arxiv.org/abs/2103.00020>

- [26] Vineeth Rakesh, Suhang Wang, Kai Shu, and Huan Liu. 2019. Linked Variational AutoEncoders for Inferring Substitutable and Supplementary Items. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. Association for Computing Machinery, New York, NY, USA, 438–446. <https://doi.org/10.1145/3289600.3290963>
- [27] Vineeth Rakesh, Suhang Wang, Kai Shu, and Huan Liu. 2019. Linked Variational AutoEncoders for Inferring Substitutable and Supplementary Items. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. Association for Computing Machinery, New York, NY, USA, 438–446. <https://doi.org/10.1145/3289600.3290963>
- [28] Aravind Sankar, Junting Wang, Adit Krishnan, and Hari Sundaram. 2020. Beyond Localized Graph Neural Networks: An Attributed Motif Regularization Framework. In *2020 IEEE International Conference on Data Mining (ICDM)*. 472–481. <https://doi.org/10.1109/ICDM50108.2020.00056>
- [29] Aravind Sankar, Junting Wang, Adit Krishnan, and Hari Sundaram. 2021. ProtoCF: Prototypical Collaborative Filtering for Few-Shot Recommendation. In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys '21)*. Association for Computing Machinery, New York, NY, USA, 166–175. <https://doi.org/10.1145/3460231.3474268>
- [30] Jin Shang, Yang (Andrew) Jiao, Chenghuan Guo, Minghao Sun, Yan Gao, Jia (Kevin) Liu, Michinari Momma, Itetsu Taru, and Yi Sun. 2024. Transitivity-encoded graph attention networks for complementary item recommendations. In *ICDM 2024*. <https://www.amazon.science/publications/transitivity-encoded-graph-attention-networks-for-complementary-item-recommendations>
- [31] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-scale Information Network Embedding. In *Proceedings of the 24th International Conference on World Wide Web (WWW '15)*. International World Wide Web Conferences Steering Committee, 1067–1077. <https://doi.org/10.1145/2736277.2741093>
- [32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [33] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. [arXiv:stat.ML/1710.10903](https://arxiv.org/abs/1710.10903) <https://arxiv.org/abs/1710.10903>
- [34] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Peng Cui, P. Yu, and Yanfang Ye. 2021. Heterogeneous Graph Attention Network. [arXiv:cs.SI/1903.07293](https://arxiv.org/abs/1903.07293) <https://arxiv.org/abs/1903.07293>
- [35] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 726–735.
- [36] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. 2015. Empirical Evaluation of Rectified Activations in Convolutional Network. [arXiv:cs.LG/1505.00853](https://arxiv.org/abs/1505.00853) <https://arxiv.org/abs/1505.00853>
- [37] An Yan, Chaosheng Dong, Yan Gao, Jinmiao Fu, Tong Zhao, Yi Sun, and Julian McAuley. 2022. Personalized complementary product recommendation. In *Companion Proceedings of the Web Conference 2022*. 146–151.
- [38] Wenting Ye, Hongfei Yang, Shuai Zhao, Haoyang Fang, Xingjian Shi, and Naveen Nepali. 2023. A Transformer-Based Substitute Recommendation Model Incorporating Weakly Supervised Customer Behavior Data. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 3325–3329. <https://doi.org/10.1145/3539618.3591847>
- [39] Liyuan Zheng, ZUO Zhen, Wenbo Wang, Chaosheng Dong, Michinari Momma, and Yi Sun. 2021. Heterogeneous graph neural networks with neighbor-SIM attention mechanism for substitute product recommendation. (2021).
- [40] Zhiheng Zhou, Tao Wang, Linfang Hou, Xinyuan Zhou, Mian Ma, and Zhuoye Ding. 2022. Decoupled Hyperbolic Graph Attention Network for Modeling Substitutable and Complementary Item Relationships. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 2763–2772. <https://doi.org/10.1145/3511808.3557281>