Enhancing E-commerce Representation Learning via Hypergraph Contrastive Learning and Interpretable LLM-Driven Analysis

Yiyue Qian
iamyiyue@amazon.com
AWS Generative AI Innovation Center
Seattle, Washington, USA

Shinan Zhang
shinanz@amazon.com
AWS Generative AI Innovation Center
Seattle, Washington, USA

Lanhao Chen
lanhaoc@amazon.com
Amazon
Seattle, Washington, USA

Diego Socolinsky
sclinsky@amazon.com
AWS Generative AI Innovation Center
Maryland, USA

Negin Sokhandan
ngnsl@amazon.com
AWS Generative AI Innovation Center
Santa Clara, California, USA

Song Cui
songcui@amazon.com
Amazon
Santa Clara, California, USA

De Chen
dchenam@amazon.com
Amazon
Seattle, Washington, USA

Suchitra Sathyanarayana
suchisat@amazon.com
Amazon
Bengaluru, Karnataka, India

Abstract

E-commerce has experienced significant growth recently, generating vast amounts of data on user preferences, interactions, and purchase patterns. Effectively modeling and representing users and products in these online ecosystems is crucial for various applications. However, existing approaches for e-commerce representation learning face several limitations: (i) they primarily consider user behavior patterns while ignoring rich group-wise relationships; (ii) some works focus on either user or product representations, failing to learn both simultaneously; (iii) results on downstream tasks are generated by "black box" models, making it difficult to interpret prediction results. To address these challenges, we propose **RelationExpert**, a general e-commerce representation learning framework. It consists of two components: **RelationEmbed**, an e-commerce representation learning foundation model, and **TaskReport**, an interpretability-driven LLM module. RelationEmbed is a self-supervised hypergraph contrastive learning model to capture multi-modal features and rich group-wise relationships among unlabeled data, i.e., merchants, customers, and products. TaskReport generates interpretable reports that explain the results of downstream tasks utilizing RelationEmbed's learned embeddings. As a result, (i) *General*: RelationExpert is applicable to various e-commerce related tasks; (ii) *Novel and Powerful*: as the first e-commerce hypergraph contrastive learning framework, RelationEmbed significantly outperforms existing methods across eight downstream tasks on two markets; (iii) *Interpretable and Reliable*: TaskReport provides clear insights into "black box" results and delivers reliable reports with high factuality and clarity.

WWW Companion '25, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1331-6/2025/04
https://doi.org/10.1145/3701716.3717579

CCS Concepts

• **Computing methodologies** → **Unsupervised learning**; **Neural networks**; **Natural language generation**; • **Information systems** → **Online shopping**.

Keywords

E-commerce Representation Learning, Hypergraph Representation Learning, Large Language Models

ACM Reference Format:
Yiyue Qian, Shinan Zhang, Lanhao Chen, Diego Socolinsky, Negin Sokhandan, Song Cui, De Chen, and Suchitra Sathyanarayana. 2025. Enhancing E-commerce Representation Learning via Hypergraph Contrastive Learning and Interpretable LLM-Driven Analysis. In *Companion Proceedings of the ACM Web Conference 2025 (WWW Companion '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3701716.3717579

1 Introduction

Over recent decades, e-commerce has experienced significant growth and prosperity, generating vast amounts of data on user preferences, interactions, and purchase patterns. Effectively modeling and representing users and products in online stores is crucial for personalized recommendations, targeted advertising, fraud detection, and overall user experience enhancement [4, 7, 28, 29, 40, 42]. Figure 1 illustrates nine customers and two merchants involved in trading two products in an online store. These customers and merchants exhibit natural group-wise relationships, such as sharing the same information (e.g., telephone number) across different accounts and purchasing the same products among different customers. Effectively and efficiently learning representations of these natural group-wise relationships is essential.

Some methods [1, 5, 11–13, 27] have been proposed to learn representations in e-commerce recently. However, these methods still face several limitations: (i) Some existing approaches [12, 13] focus on training the foundational models based on user behavior

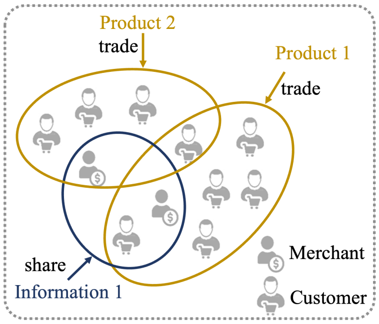

Figure 1: Showcase of group-wise relationships in markets.

patterns to learn e-commerce representations. For instance, ShopperBERT [40] learns from billions of user behaviors to produce user embeddings for downstream tasks, but it overlooks the rich group-wise connections among users, such as shared information or similar purchasing patterns. Although recent works [7, 29] design graph neural networks (GNNs) in modeling the relations among merchants, customers, and products online, these works are limited to pairwise relationships, which may not adequately capture the group-wise higher-order interactions prevalent in e-commerce scenarios. (ii) Most existing works support representation learning on either users or products online, failing to learn both simultaneously. For instance, COPE framework [7] learns product representations across different media domains, while ShopperBERT [40] focuses on user representations from user behaviors. (iii) After obtaining representations, most existing models operate as "black boxes" that leverages these representations for downstream tasks directly, making it challenging to understand and interpret results of these tasks.

To address the aforementioned limitations, we propose a framework called **RelationExpert**, which comprises two key components: **RelationEmbed**, an e-commerce foundation module designed to model rich group-wise relationships and multi-modal features over unlabeled data, and **TaskReport**, an interpretability-driven LLM module that generates interpretable reports based on the results of downstream tasks. Specifically, we first construct e-commerce hypergraphs to represent the high-order group-wise relationships among merchants, customers, and products, as well as their multi-modal information. To address the first and second challenges, we design a dual-level hypergraph contrastive learning module to train a hypergraph model on the unlabeled data. After sufficiently training, we obtain a pre-trained hypergraph foundation model and embeddings for merchants, customers, and products, which can be easily fine-tuned or leveraged for various downstream tasks (e.g., product-customer recommendation, abuser detection, and clustering). Furthermore, to tackle the third challenge, we develop TaskReport, which leverages LLM to automatically generate interpretability-driven reports by considering the group-wise relationships and the classification results in downstream tasks. In summary, the main contributions of this work are:

- **General**: We introduce RelationExpert, a general e-commerce representation learning framework, encompassing the hypergraph foundation model RelationEmbed and the interpretability-driven reports generated by TaskReport. This general framework is applicable to various e-commerce-related tasks.
- **Novel and Powerful**: To the best of our knowledge, RelationEmbed is the first e-commerce foundation model that designs hypergraph contrastive learning to learn embeddings of merchants, customers, and products over real-world unlabeled data. These embeddings achieve significant improvements in eight downstream tasks over two real-world market datasets compared to existing methods.
- **Interpretable and Reliable**: TaskReport is developed to offer clear insights to "black box" results. By accessing the quality of factuality and clarity in task-oriented LLM reports, TaskReport shows its excellent capability and reliability.

2 Related Work

E-commerce Representation Learning. Representation learning has become a crucial technique in e-commerce, enabling platforms to capture complex user behaviors, product relationships, and multi-modal data. Recent works on e-commerce representation learning can be categorized into several key areas: multi-modal representation learning [4, 23, 42, 53, 57], large-scale pre-training [8, 18, 38, 40, 41], and graph-based representation learning [7, 24, 29, 56]. As e-commerce offers various types of data (e.g., text, image, and video) in different modalities, it is an increasing need for representation learning that can bridge data in different modalities. For instance, COPE framework [7] is proposed to address the challenges posed by rich content in online markets, unifying product representations across different media domains including product pages, short videos, and live streams. The second group is related to large-scale pre-training for user representation learning, which is inspired by the promising performance of general-purpose language pre-training. For instance, ShopperBERT [40], learns from billions of user behaviors to produce user embeddings that can be applied to various downstream tasks such as user profiling, anomaly detection, and recommendations. The third group contains graph neural network methods for representation learning, which are inspired by the excellent performance of GNNs in learning complex relationship in real-world data. For instance, U-ROAD [29], designs a universal framework for multimodal heterogeneous graph neural networks to detect various types of abusers in online stores. Although these methods can either leverage complex relationships and multi-modal data in online stores, they are unable to depict the complex group-wise relations. Besides, most of existing works only support either user representation learning or product representation learning, while they are not able to learn both user and product representations. What's more, when applying the pre-trained representations to downstream tasks, it is always challenging to interpret the results of downstream tasks.

Hypergraph Contrastive Learning. Unlike regular graphs [32, 36, 37, 47, 50, 51, 51, 54, 55] where each edge connects exactly two nodes, hypergraphs allow each hyperedge connects an arbitrary number of nodes. Hypergraph neural networks (HyGNNs) [9, 10, 30, 49] have gained considerable attention in recent years with their strong ability to capture complex relationships among networks. AllDeepSets [9], a notable work that designs multi-set functions to learn the information propagation between nodes and hyperedges. Inspired by existing graph contrastive learning works [33–35, 45, 46], researchers start to explore the benefits of contrastive learning over hypergraphs [22, 26, 31, 43]. For instance, HyperGCL [44] proposes a generative method to create generative augmentations of hypergraphs. However, these works have limitations in describing group-wise behaviors and multi-modal information in e-commerce during hypergraph contrastive learning. To this end, we design a dual-level e-commerce hypergraph contrastive learning to reach agreements among representations of users, products, and groups.

3 Preliminary

In this section, we introduce the relevant definition and formally define e-commerce representation learning problem.

Definition 3.1. **Hypergraph.** Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X})$ denotes a hypergraph, where $\mathcal{V} = \{v_1, v_2, \cdots, v_N\}$ is the set of nodes with size $N = |\mathcal{V}|$, $\mathcal{E} = \{e_1, e_2, \cdots, e_M\}$ is the set of hyperedges with size $M = |\mathcal{E}|$, and $\mathcal{X} = \{x_1, x_2, \ldots x_N\}$ is the set of node attribute features with $x_i \in \mathbb{R}^d$. Unlike the pairwise edges in a graph, each hyperedge $e \in \mathcal{E}$ represents a higher-order interaction among a set of nodes. A hypergraph can be represented by an incidence matrix $H \in \mathbb{R}^{N \times M}$, where $H_{ve} = 1$ if $v \in e$ Otherwise, $H_{ve} = 0$. For each node $v \in \mathcal{V}$ and hyperedge $e \in \mathcal{E}$, we leverage $d(v) = \sum_{e \in \mathcal{E}} H_{ve}$ and $d(e) = \sum_{v \in \mathcal{V}} H_{ve}$ to denote the node degree and the hypergraph degree, respectively.

Problem 1. ***E-commerce Representation Learning***. *Given online market data, containing different types of nodes (i.e., merchant, customer, and product) along with multi-modal features X, merchant, customer, and product labels Y, as well as multiple types of group-wise relationships $\mathcal{E}$ among nodes, we build an e-commerce hypergraph $\mathcal{G}$ to model the rich multi-modal node features and group-wise relationships among nodes and further design an e-commerce hypergraph contrastive learning module to learn the merchant, customer, and product embeddings simultaneously over the unlabeled data. Then the learned embeddings of merchants, customers, products, and groups generated by the hypergraph foundation model can be directly applied to any downstream tasks. What's more, the results of downstream tasks will be presented in an interpretable report generated by LLM.*

4 Proposed Model

In this section, we present the details of RelationExpert which includes three modules: hypergraph construction (Figure 2.(a)), RelationEmbed foundation model via hypergraph contrastive learning (Figure 2. (b)), and an interpretability-driven LLM module called TaskReport (Figure 2. (c)).

4.1 E-commerce Hypergraph Construction

Unlike edges in graphs that can only connect two nodes, a hyperedge in hypergraphs can connect any number of nodes, which is flexible to capture the group-wise relationships among merchants, customers, and products in online markets. Therefore, we employ hypergraphs to depict both the high order group-wise relationships in e-commerce markets, and their informative multi-modal features.

4.1.1 Features in E-commerce Hypergraphs. **Merchant and Customer Features:** Merchant and customer features play a crucial role in analyzing and optimizing e-commerce markets. In this work, we extract some typical features to capture merchant and customer's behaviors. Specifically, for merchants, numerical features such as product pricing strategies provide insights into market positioning and competitiveness. The age of a merchant's account serves as an indicator of experience and reliability, potentially correlating with higher trust levels and performance. Sales volumes offer valuable metrics to assess a merchant's success and operational efficiency. On the customer side, for instance, purchase frequency, average order value, and customer lifetime value help segment users and understand their engagement levels. We leverage these numerical features as the feature vectors attached to each merchant or customer node in hypergraphs.

Multi-modal Product Features: For products in online stores, we consider the rich multi-modal features, integrating both textual and visual data. For text information, we leverage the pre-trained language model *Sentence-Bert* [39] to generate text embeddings with a dimension of 384. Simultaneously, we utilize the pre-trained image model *ResNet* [15] to extract image feature vectors of size 2048. Ultimately, we concatenate these text and image features into a comprehensive feature vector measuring 2432 dimensions, which is then utilized for product hyperedges in hypergraphs. We consider product as one type of our hyperedges in our e-commerce hypergraph and attach features to each product hyperedge.

4.1.2 Hyperedges in E-commerce Hypergraphs. We define two distinct types of group-wise relations, or hyperedges, among entities in online markets. These hyperedges are designed to capture and represent high-order group-wise relations within the e-commerce network. The first category, *R1*: *merchants/customers-sell/purchase-product*, encapsulates the group-wise interactions based on order history, connecting merchants, customers, and products. Specifically, it indicates that a group of customers purchased a particular product from a merchant within a defined time frame. The second category, *R2*: *merchant/customer-share-info*, identifies instances where a group of customers or merchants share identical information. To illustrate these concepts, consider three example hyperedges in Figure 3: Product 1, Product 2, and Information 1. Product 1 connects six customers to one merchant, while Product 2 links four customers to another merchant. Additionally, a third hyperedge, Information 1, connects these two merchants and a customer who share common personal information. Figure 3 shows the e-commerce hypergraph schema which effectively extracts and represents group-wise relationships. By leveraging hyperedge definitions, we can uncover intricate patterns of behavior that might remain hidden in traditional graph representation learning.

4.2 RelationEmbed: E-commerce Hypergraph Contrastive Learning

In this section, we will introduce how we design dual-level hypergraph contrastive learning over unlabeled merchants, customers, and products in online markets.

4.2.1 E-commerce Hypergraph Augmentation: Based on existing works [22, 26, 44, 52], we first summarize five types of hypergraph augmentation methods (i.e., attribute masking, hyperedge removal, node dropping, edge perturbation, and sub-hypergrpah generation).

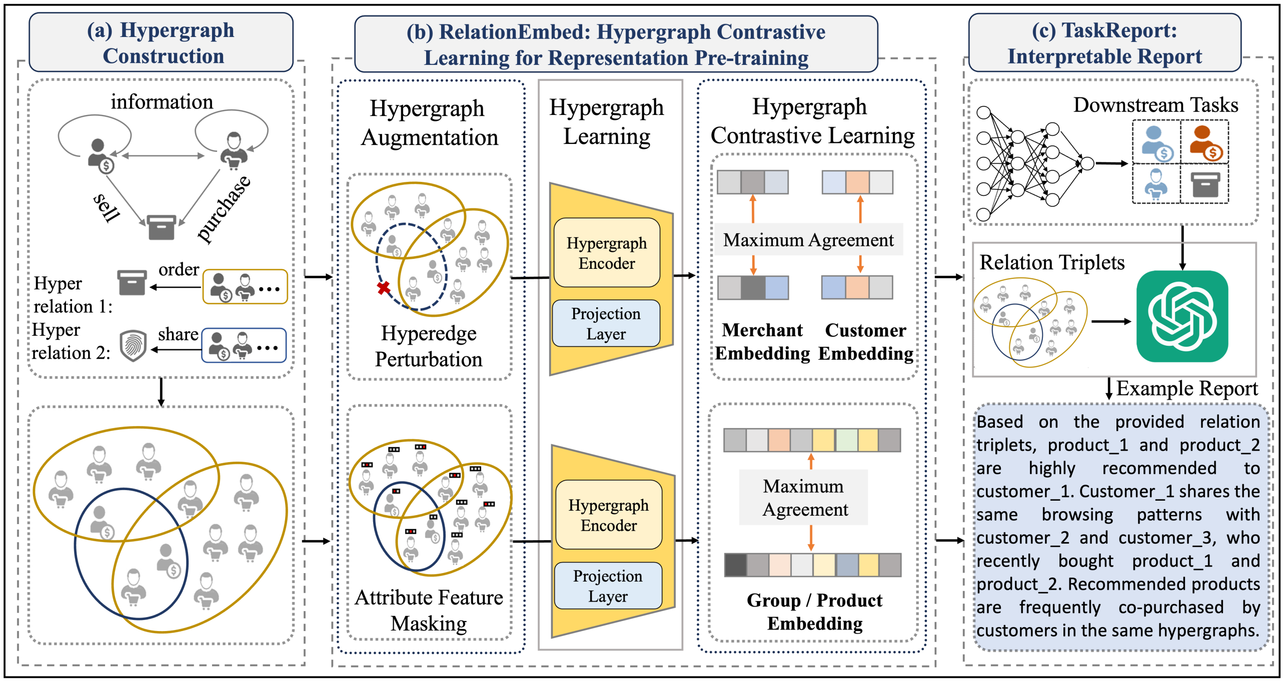

Figure 2: The framework of RelationExpert: (a) It first designs e-commerce hypergraph schema to extract rich group-wise relationships among entities (i.e., merchants, customers, and products) in online markets. (b) With two augmented hypergraphs generated via hypergraph augmentations, they are fed into the hypergraph encoder. By maximizing the agreements between positive and negative merchant/customer/group/product contrastive pairs in the embedding space, we can obtain the embeddings of merchants, customers, products, and groups learned by RelationEmbed over the unlabeled data. (c) The learned embeddings are directly utilized for various downstream tasks related e-commerce services. With the classification results, TaskReport generates interpretable and reliable reports based on the relation triplets and the predicted labels over downstream tasks.

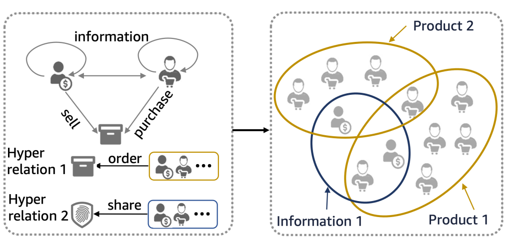

Figure 3: E-commerce hypergraph schema.

Given these augmentation methods, we select hyperedge perturbation and attribute feature masking as the pair of hypergraph augmentation methods and further obtain the augmented hypergraph pair $(\widetilde{\mathcal{G}}_1, \widetilde{\mathcal{G}}_2)$ where $[\widetilde{\mathcal{G}}_1 = (\mathcal{V}_1, \widetilde{\mathcal{E}_1}, \mathcal{X}_1), \widetilde{\mathcal{G}}_2 = (\mathcal{V}_2, \mathcal{E}_2, \widetilde{\mathcal{X}_2})]$. Here $\widetilde{\mathcal{E}_1}$ and $\widetilde{\mathcal{X}_2}$ are hyperedge set after hyperedge removal and feature set after masking in e-commerce hypergraphs, respectively.

4.2.2 E-commerce Hypergraph Representation Learning: After obtaining two augmented e-commerce hypergraphs, we feed them to HyGNNs to model rich group-wise relations and multi-modal attribute features over unlabeled data. As we have two types of nodes (i.e., merchant and customer) and two types of hyperedges (i.e., product and information) in hypergraphs, we apply a type-specific mapping layer to transform all features of nodes and hyperedges into the same space. The projected features are denoted as $\mathbf{X}$:

$$\mathbf{X} = g(\mathcal{X}) = \mathcal{X} \cdot W_o, \tag{1}$$

where W is the type-specific transformation matrix for different node and hyperedge types. In this work, $|o| = 4$.

Then we feed augmented e-commerce hypergraph $\widetilde{\mathcal{G}}$ to HyGNNs to learn the embeddings of merchants, customers, and products. In this work, we choose AllDeepSets [9] as the hypergraph encoder $f(\cdot)$ to learn the node (merchant and customer) and hyperedge (product) embeddings, which can be formulated as follows:

$$\mathbf{Z}_{e,:}^{(t+1)} = f_{\mathcal{V}\rightarrow\mathcal{E}}(V_{e,\mathbf{U}^{(t)}}; \mathbf{Z}_{e,:}^{(t)}), \quad \mathbf{U}_{v,:}^{(t+1)} = f_{\mathcal{E}\rightarrow\mathcal{V}}(E_{v,\mathbf{Z}^{(t+1)}}; \mathbf{U}_{v,:}^{(t)}), \tag{2}$$

where $\mathbf{Z}_{e,:}^{(t+1)}$ is the embedding of hyperedge e at time $t+1$, $\mathbf{U}_{v,:}^{(t+1)}$ is the embedding of node v, $f_{\mathcal{V}\rightarrow\mathcal{E}}$ and $f_{\mathcal{E}\rightarrow\mathcal{V}}$ are two multiset functions in terms of the input. $\mathbf{Z}_{e,:}^{0}$ denotes the original attribute feature of hyperedge e and $\mathbf{U}_{v,:}^{0}$ denotes the attribute feature of node v, where $\mathbf{U}_{v,:}^{0} = \mathbf{X}_{v,:}$.

4.2.3 User-level Hypergraph Contrastive Learning: After that, a dual-level hypergraph contrastive strategy is devised to align the user embeddings in a local manner and match the group-wise group embeddings from a global perspective. In order to achieve the node embedding agreements among merchants and customers, we design user-level hypergraph contrastive learning (HyGCL) to maximize the similarity between positive user pairs while minimizing the similarity between negative user pairs. Specifically, given two nodes (merchant/customer nodes) (v_i, v_j) from $(\widetilde{\mathcal{G}}_1,\widetilde{\mathcal{G}}_2)$, we obtain the user embeddings $(\mathbf{u}_i^1, \mathbf{u}_j^2)$ generated by the encoder $f_{\mathcal{V}\to\mathcal{E}\to\mathcal{V}}$ in Equation 2. We then feed $(\mathbf{u}_i^1, \mathbf{u}_j^2)$ to projection head $h(\cdot)$ to project embeddings into the same space for contrastive learning. (v_i, v_j) is a positive contrastive user pair if $i = j$. Otherwise, it is considered as a negative user pair in user-level HyGCL.

4.2.4 Group-level Hypergraph Contrastive Learning: Although user-level HyGCL maximizes the embedding agreements among users, it may not be sufficient to capture group-wise behaviors within hyperedges (i.e., products and shared information). To this end, we design a group-level HyGCL module to capture the group behaviors within the corresponding products / shared information from a global perspective. Specifically, for each hyperedge $e_i \in \mathcal{E}$, we get the hyperedge embedding $\mathbf{z}_i$ by $f_{\mathcal{V}\to\mathcal{E}}$ in Equation 2. Then group embeddings $\mathbf{H}$ integrating hyperedge (product / shared information) embeddings $\mathbf{Z}$ and node (user) embeddings $\mathbf{U}$ is formulated as:

$$\mathbf{h}_i = \mathbf{z}_i \oplus \frac{1}{d(e_i)} \sum_{m\in e_i} \mathbf{u}_m, \tag{3}$$

where $\mathbf{h}_i$ denotes group embedding distinguished by hyperedge e_i, $d(e_i)$ denotes the degree of e_i, and $\oplus$ is the concatenation operator. $\mathbf{h}_i^1$ and $\mathbf{h}_j^2$ denote the group embeddings distinguished by products or shared info e_i and e_j in $\widetilde{\mathcal{G}}_1$ and $\widetilde{\mathcal{G}}_2$, respectively. We further design a group-level contrastive strategy to reach agreements among group embeddings. In specific, $(\mathbf{h}_i^1,\mathbf{h}_j^2)$ will be viewed as positive contrastive group pair if $i = j$. Otherwise, it would be a negative contrastive group pair.

4.2.5 Contrastive Optimization: The objective of dual-level contrastive learning is to ensure that the same merchants/customers/groups from different augmented hypergraphs are encoded closely in the embedding space, while different merchants/customers/groups are embedded farther apart. Therefore, the dual-level contrastive loss $\mathcal{L}_{\text{u\_g}}$ can be formulated as:

$$\begin{aligned}
\mathcal{L}_{\text{u_g}} &= \lambda_1 * \mathcal{L}_{\text{user}} + \lambda_2 * \mathcal{L}_{\text{group}}, \ \text{where} \\
\mathcal{L}_{\text{user}} &= -\log \sum_{v_i\in\mathcal{V}} \frac{\exp(\text{sim}(\mathbf{u}_i^1, \mathbf{u}_i^2))}{\sum_{k\neq i} \exp(\text{sim}(\mathbf{u}_i^1, \mathbf{u}_k^2)) + \exp(\text{sim}(\mathbf{u}_i^1, \mathbf{u}_i^2))}, \\
\mathcal{L}_{\text{group}} &= -\log \sum_{e_i\in\mathcal{E}} \frac{\exp(\text{sim}(\mathbf{h}_i^1, \mathbf{h}_i^2))}{\sum_{k\neq i} \exp(\text{sim}(\mathbf{h}_i^1, \mathbf{h}_k^2)) + \exp(\text{sim}(\mathbf{h}_i^1, \mathbf{h}_i^2))}.
\end{aligned} \tag{4}$$

Here λ_1 and λ_2 are hyper-parameters to balance user-level and group-level HyGCL.

4.3 TaskReport: Interpretability-driven LLM Report for Downstream Tasks

4.3.1 Fine-tuning over Various Downstream Tasks: After sufficiently pre-training hypergraph encoder over unlabeled data, we obtain the pre-trained hypergraph foundation model and acquire the embeddings of merchants, customers, products, and groups. Then, we utilize these embeddings for various downstream tasks, i.e, abuser detection, product-customer recommendation, and clustering.

Abuser Classification. With learned embeddings of merchants, customers, and groups, we can detect abusive users and groups in online markets. Here we take abusive group classification task as an example. With the learned group embedding $\mathbf{H}$, we feed $\mathbf{H}$ into a MLP layer to predict the probability of being abusive groups, i.e., $\hat{Y}_i = \sigma(\mathbf{H}_\mathbf{i} W_c)$. Besides, to handle the class imbalance among abusive and benign groups, we introduce Focal Loss [25] to focus on hard mis-classified samples, which is formally defined as:

$$\mathcal{L}_{\text{focal}} = -\frac{1}{|\mathcal{E}_l|} \sum_{i\in\mathcal{E}_l} \alpha\,(1-\hat{Y}_i)^\gamma \log(\hat{Y}_i), \tag{5}$$

where $\mathcal{E}_l$ is the set of labeled groups, γ is the focusing parameter to control the rate at which easy nodes will be down-weighted, and $\alpha \in [0, 1]$ is a weighting hyper-parameter for multiple classes.

Product-Customer Recommendation. Next, we will also introduce another task about recommending products to customers, a.k.a., link prediction. With the learned embedding of customers and products, we can perform link prediction for product recommendations in online markets. Specifically, we combine the customer embedding $\mathbf{U}$ and product embedding $\mathbf{Z}$ using two methods: Weight-L2 [21] and Hadamard product [16]. Let us take Hadamard product method as an example. Similarly to abuser classification task, we calculate $\mathbf{L} = \mathbf{U} \odot \mathbf{Z}$ and the combined embedding $\mathbf{L}$ is then fed into a MLP layer to predict the probability of a link between a customer and a product, i.e., $\hat{Y}i = \sigma(\mathbf{L}_\mathbf{i} Wc)$, where σ is the sigmoid activation function. We use binary cross-entropy loss for this task:

$$\mathcal{L}_{\text{ce}} = -\frac{1}{|\mathcal{E}_l|} \sum_{i\in\mathcal{E}_l} [Y_i \log(\hat{Y}_i) + (1-Y_i)\log(1-\hat{Y}_i)], \tag{6}$$

where $\mathcal{E}_l$ is the set of labeled links, Y_i is the true label, and $\hat{Y}_i$ is the predicted probability of a link among customer and product.

4.3.2 Interpretabililty-driven LLM Report: To enhance interpretability of the "black box" model's results, we propose an LLM module called TaskReport to automatically generate interpretable reports for downstream tasks. Specifically, we utilize Claude 3 Sonnet [2] as the LLM model. For each user (merchant, customer, group associated with the product), we input these relevant 2-hop relationships as triplets (e.g., customer1, purchase, product1) along with predicted labels of involved users and products from the upstream MLP classifier into LLM. Each report can be formulated as:

$$\text{Report} = \text{LLM}\left(\{\hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_m\}, \{(v_{s1}, e_1, v_{d1}), \cdots, (v_{sk}, e_k, v_{dk})\} \mid \theta\right). \tag{7}$$

θ represents LLM parameter. (v_s, e, v_d) denotes the relation triplets, where v_s is the source node, v_d is the destination node, and e is the relation type. Note that we divide the group-wise relationships in e-commerce hypergraphs into triplets and feed them to LLM. These triplets allow LLM to comprehend complex relationships among merchants, customers, products, and shared information. $\{\hat{Y}_1, \cdots, \hat{Y}_m\}$ is the predicted label set of all entities involved in the triplet set. The pseudo-code is outlined in Alogrithm 1.

5 Experiment

In this section, we first introduce two datasets from two e-commerce markets. Then we conduct extensive experiments to evaluate RelationEmbed on eight tasks (i.e., detection tasks of abusive merchants, customers, and groups, and product-customer link prediction tasks on M1 and M2 markets). Besides, we evaluate TaskReport by judging the report in two dimensions: factuality and clarity.

Algorithm 1: Training Procedure of RelationExpert

Require Market data, Multi-modal feature $\mathcal{X}$, Labeled users and products Y, Hypergraph encoder $f(\cdot)$, Projection head $h(\cdot)$.
Build e-commerce hypergraph $\mathcal{G}$ with attribute features $\mathcal{X}$ and hyperedges $\mathcal{E}$ according to hypergraph schema in Figure 3.
Initialize hypergraph encoder $f(\cdot)$ and projection head $h(\cdot)$.
RelationEmbed:
for each epoch t **do**
  Augmentation: $\mathcal{G} \xrightarrow{A_1} \widetilde{\mathcal{G}}_1$, $\mathcal{G} \xrightarrow{A_2} \widetilde{\mathcal{G}}_2$, where $[\widetilde{\mathcal{G}}_1 = (\mathcal{V}_1, \widetilde{\mathcal{E}_1}, \mathcal{X}_1), \widetilde{\mathcal{G}}_2 = (\mathcal{V}_2, \mathcal{E}_2, \widetilde{\mathcal{X}_2})]$.
  Feed augmented hypergraphs $(\widetilde{\mathcal{G}}_1, \widetilde{\mathcal{G}}_1)$ to $f(\cdot)$ in Eq. 2 for obtaining the merchant and customer embeddings $(\mathbf{u}^1,\mathbf{u}^2)$ from the user view.
  Get the product embeddings $\mathbf{Z}$ and group embeddings $(\mathbf{h}^1,\mathbf{h}^2)$ from the group view;
  Feed the node and group embeddings into the projection head $h(\cdot)$;
  Optimize $f(\cdot)$ and $h(\cdot)$ by minimizing dual-level contrastive loss $\mathcal{L}_{\text{u\_g}}$ in Eq. 4.
end for
TaskReport:
for each user/product/group **do**
  Apply the pre-trained merchant/customer/product/group embeddings to the corresponding downstream tasks.
  With the predicted label $\hat{Y}$ over downstream tasks and the relation triplets (v_s, e_k, v_d), we feed them to LLM model (Cluade Sonnet) to generate interpretable report.
  Futher employ LLM models, e.g., Claude Sonnet3.5, to evaluate the factuality and clarity of the generated reports.
end for
Return Interpretable reports in downstream tasks.

5.1 Experiment Setup

5.1.1 Datasets. We collect two e-commerce datasets from two markets (i.e., M1 market and M2 market). Next, we will introduce the statistics of these two datasets in detail.

E-commerce Dataset 1 on M1 Market: The hypergraph constructed on M1 market has 237,833 nodes and 649,591 hyperedges, including 9,785 merchants, 228,048 customers, 26,875 product hyperedges, and 377,718 info hyperedges. A group comprises customers and the merchant trading a specific product. For both downstream tasks (i.e., abuser detection and product-customer link prediction), labeled data (i.e., merchants, customers, groups, product-customer pairs) are split into three sets according to their action time, i.e., training set, validation set, and testing set for model training, validation, and testing, respectively.

E-commerce Dataset 2 on M2 Market: To sufficiently validate RelationEmbed, we also collect another e-commerce data from M2 market. The constructed hypergraph on this market has 222,097 nodes and 829,818 hyperedges, including 53,247 merchants, 168,850 customers, 474,464 product hyperedges, and 355,354 info hyperedges. Data is split into training, validation, and testing sets following the same mechanism in E-commerce Dataset 1.

5.1.2 Baseline: To comprehensively evaluate our framework, we compare RelationEmbed with twelve baseline models in five groups:

B1: Feature-based model. For the abusive user (merchant and customer) detection, we employ Xgboost [6] to detect these abusive users. For abusive groups, we first get the average of all customer features. Then we concatenate the product features, merchant features, and the averaged customer features as the group features. We further feed the group features into xgboost for classification.

B2: Graph-based methods. We convert the hypergraph structure into a graph with attribute features and graph structures and feed it to three GNNs including GCN [20], GIN [48], and GraphSAGE [14] to learn the merchant, customer, and product embeddings. After obtaining these embeddings, we get the average of all customer features and then concatenate the product features, merchant features, and the averaged customer features as the group features. And we feed these embedding to a 2-layer MLP classifier to detect abusive users/groups and predict links between products and customers.

B3: Hypergraph-based methods. We compare RelationEmbed with three hypergraph neural networks including HyperGCN [49], UniGCNII [17], and AllDeepSets [9]. Following RelationEmbed, group embedding is the concatenation of the product hyperedge embedding and the average of all corresponding users embedding.

B4: Contrastive-based methods. We further compare RelationEmbed with three hypergraph contrastive learning methods including HyperGCL [49], CHGNN [43], and TriCL [22]. And we feed these learned embeddings to MLP classifier to handle downstream tasks.

B5: E-commerce-based methods. We further reproduce two models, i.e., U-ROAD [29], a heterogeneous graph neural networks designed for abuser detection in E-commerce, and K2M [57], a multi-modal pre-training model in E-commerce. For U-ROAD, we feed these learned embeddings to MLP classifier to handle downstream tasks, while for K2M, we utilize the pre-trained product embeddings generated by K2M as features and feed this to our hypergraph encoder AllDeepSet for downstream tasks.

5.1.3 Experimental Setting: To evaluate the performance of RelationExpert, we adopt the widely-used metrics: For RelationEmbed, we utilize Macro-F1 score for abuser detection tasks and we adopt AUC score and AP (average precision) score for product-customer link prediction tasks. For TaskReport, we design the factuality and clarity metrics to access the quality of generated reports over downstream tasks. Experiments are conducted under the AWS environment of the g5.48xlarge with GPU resources. For each GNN and HyGNN model (i.e., GraphSAGE, GIN, GCN, U-ROAD, HyperGCN, UniGCNII, and AllDeepSets), we employ two layers neural network with weight decay 1e-5 and the dimension of node and hyperedge embeddings generated by these models is 200. We use Adam [19] optimizer with a learning rate of 0.001. λ_1 and λ_2 in Eq. 4 are set as 1.0. The value of α and γ in Eq. 5 are different for different downstream tasks. All experiments are conducted in five runs and mean and stand deviation are provided.

5.2 Performance Discussion on RelationEmbed

5.2.1 Performance Comparison: Table 1 shows the Macro F1 scores of all methods for abusers (i.e., abusive merchants, abusive customers, and abusive groups) detection on M1 and M2 markets. From this table, we can conclude that: (i) Graph-based methods

Table 1: F1 Performance (mean % ± std) comparison for abuser detection on M1 market and M2 market. Best performances are highlighted in blue, while the runner-up performances are shaded in gray.

Setting		M1 Market			M2 Market		
Group	Model	Merchant	Customer	Group	Merchant	Customer	Group
B1	Xgboost [6]	39.65 ± -	14.71 ± -	18.36 ± -	32.53 ± -	40.60 ± -	23.89 ± -
B2	GCN [20]	41.43 ± 4.59	24.82 ± 4.87	33.15 ± 5.21	35.52 ± 4.74	45.75 ± 4.57	29.42 ± 4.98
	GIN [20]	40.35 ± 4.67	25.12 ± 4.46	33.45 ± 5.43	35.02 ± 4.57	46.15 ± 4.73	29.98 ± 4.87
	GraphSAGE [14]	42.54 ± 4.51	25.39 ± 4.15	35.95 ± 4.62	37.25 ± 4.87	47.45 ± 4.46	31.12 ± 4.93
B3	HyperGCN [49]	45.86 ± 3.57	21.19 ± 3.71	31.38 ± 4.43	42.15 ± 3.59	49.63 ± 3.19	33.70 ± 3.35
	UniGCNII [17]	46.74 ± 3.48	23.15 ± 3.41	33.14 ± 4.58	43.25 ± 3.47	50.15 ± 3.48	35.75 ± 4.05
	AllDeepSets [9]	59.12 ± 3.98	34.70 ± 3.72	43.34 ± 4.32	50.15 ± 3.65	54.31 ± 3.72	35.42 ± 3.92
B4	HyperGCL [44]	63.43 ± 3.46	38.42 ± 3.57	47.42 ± 4.57	53.24 ± 3.27	58.59 ± 3.45	39.45 ± 3.51
	CHGNN [43]	64.46 ± 3.31	40.35 ± 3.18	48.45 ± 4.32	54.15 ± 3.57	60.25 ± 3.01	41.53 ± 4.12
	TriCL [22]	65.45 ± 3.54	42.14 ± 3.34	49.31 ± 4.13	55.05 ± 3.87	60.12 ± 3.41	42.14 ± 3.78
B5	U-ROAD [29]	48.14 ± 3.42	30.51 ± 3.07	40.16 ± 4.05	43.15 ± 4.01	52.31 ± 3.87	34.45 ± 3.58
	K3M [57]	61.74 ± 4.72	36.15 ± 4.53	45.14 ± 4.62	52.41 ± 4.43	55.35 ± 4.51	37.15 ± 4.31
Ours	**RelationEmbed**	70.99 ± 3.76	48.37 ± 3.52	53.53 ± 4.92	57.42 ± 3.42	65.72 ± 3.78	48.17 ± 3.97

Table 2: AUC and AP Performance (mean % ± std) comparison for product-customer link prediction on M1 market and M2 market. Best performances are highlighted in blue, while the runner-up performances are shaded in gray.

Setting	M1 Market				M2 Market			
	Weight-L2		Hadamard		Weight-L2		Hadamard	
Model	AUC	AP	AUC	AP	AUC	AP	AUC	AP
Xgboost [6]	34.15 ± -	33.12 ± -	31.15 ± -	30.23 ± -	29.46 ± -	28.15 ± -	25.42 ± -	24.38 ± -
GCN [20]	37.25 ± 4.36	37.15 ± 4.42	34.42 ± 4.22	33.13 ± 4.05	31.14 ± 4.06	30.89 ± 4.57	28.05 ± 4.21	27.56 ± 4.31
GIN [48]	38.54 ± 4.13	38.11 ± 4.24	35.05 ± 4.12	34.01 ± 4.51	32.35 ± 4.21	31.45 ± 4.04	29.41 ± 4.34	28.41 ± 4.63
GraphSAGE [14]	38.43 ± 4.16	38.55 ± 4.31	35.07 ± 4.32	35.04 ± 4.45	32.98 ± 4.13	32.11 ± 4.31	30.12 ± 4.07	29.12 ± 4.67
U-ROAD [29]	45.12 ± 4.12	45.12 ± 4.09	42.15 ± 4.41	42.01 ± 4.58	37.41 ± 4.36	37.42 ± 4.31	36.12 ± 4.01	35.21 ± 3.91
HyperGCN [49]	41.14 ± 3.98	41.14 ± 3.71	38.91 ± 4.24	39.05 ± 4.12	35.54 ± 4.25	35.71 ± 4.43	34.51 ± 4.14	32.51 ± 4.71
UniGCNII [17]	43.05 ± 3.91	42.54 ± 3.89	39.24 ± 4.41	40.58 ± 4.75	36.54 ± 4.98	36.71 ± 4.47	35.15 ± 4.25	33.12 ± 4.68
AllDeepSets [9]	51.14 ± 3.75	50.14 ± 3.75	46.24± 4.41	46.58± 3.45	44.21 ± 4.41	42.32± 4.14	43.21 ± 4.41	41.41 ± 4.14
HyperGCL [44]	55.31 ± 3.42	53.14 ± 3.53	50.12 ± 4.13	50.12 ± 3.78	48.51 ± 3.31	46.25 ± 3.12	47.51 ± 3.13	45.12 ± 3.54
CHGNN [43]	57.14 ± 3.15	54.75 ± 3.54	52.15 ± 4.31	53.53 ± 3.56	50.33 ± 3.51	48.43 ± 4.31	50.31 ± 3.31	46.67 ± 3.91
TriCL [22]	58.32 ± 3.32	55.45 ± 3.41	53.26 ± 4.42	53.44 ± 3.58	51.24 ± 3.24	48.98 ± 3.43	50.15 ± 3.41	48.05 ± 3.13
U-ROAD [29]	45.12 ± 4.12	45.12 ± 4.09	42.15 ± 4.41	42.01 ± 4.58	37.41 ± 4.36	37.42 ± 4.31	36.12 ± 4.01	35.21 ± 3.91
K3M [57]	52.45 ± 4.23	52.13 ± 4.41	48.16 ± 4.13	47.41 ± 4.63	46.14 ± 4.15	44.57 ± 4.25	44.14 ± 4.15	42.57 ± 4.25
RelationEmbed	64.14 ± 3.31	61.35 ± 3.21	60.14 ± 3.51	60.12 ± 3.14	58.35 ± 3.43	56.11 ± 3.45	57.13 ± 3.15	56.14 ± 3.14

Table 3: Factuality and clarity scores (mean ± std, range [1, 3]) on 500 reports for each downstream task evaluated by Claude Sonnet 3.5. Best Factuality performances are highlighted in blue and best Clarity performances are shaded in gray.

Setting		M1 Market - **TaskReport**				M2 Market -**TaskReport**			
LLM	Metrics	Merchant	Customer	Group	Product-Customer	Merchant	Customer	Group	Product-Customer
Sonnet 3.5	Factuality	2.64 ± 0.13	2.51 ± 0.17	2.33 ± 0.41	2.42 ± 0.21	2.57 ± 0.24	2.47 ± 0.21	2.37 ± 0.47	2.40 ± 0.24
	Clarity	2.88 ± 0.08	2.86 ± 0.07	2.73 ± 0.14	2.81 ± 0.09	2.85 ± 0.12	2.84 ± 0.09	2.75 ± 0.13	2.79 ± 0.15

(B2) outperform the feature-based model (B1), demonstrating the value of incorporating graph relations and features. (ii) Hypergraph-based methods (B3) surpass graph-based methods (B2), indicating hyperedges better capture these group-wise relationships. (iii) Hypergraph contrastive learning methods in B4 outperforms hypergraph models, validating the self-supervised model's effectiveness in learning embeddings from unlabeled data. (iv) RelationEmbed achieves the excellent performance over all baseline methods including two models designed for e-commerce tasks (B5) across six detection tasks, showing the superiority of RelationEmbed.

Table 2 shows the AUC and AP of all methods for product-customer link prediction on M1 and M2 markets. Specifically, we randomly remove a certain fraction of customers within the corresponding product hyperedges as positive product-customer pairs and randomly select a certain fraction of customers to some arbitrary product hyperedges as negative product-customer pairs. For each product-customer pair, we leverage two common binary operators, Weight-L2 and Hadamard to learn the product-customer embeddings, respectively. From this table, we can conclude that: (i) Weight-L2 operation for product-customer pairs is more effective than that with Hadamard operation for link prediction tasks on both markets. (ii) Graph-based methods (B2) outperform the feature-based model (B1) and Hypergraph-based methods (B3) surpass graph-based methods (B2). (iii) Hypergraph contrastive learning methods (B4) outperforms hypergraph models (B3). AllDeepSet, as our hypergraph encoder, has the best performance. (iv) Fine-tuned RelationEmbed achieves the excellent performance over all mentioned baseline methods across two markets.

5.2.2 Embedding Visualization: In order to make comparisons in an intuitive manner, Figure 4 visualizes the embedding generated by RelationEmbed w/o labels (our hypergraph foundation model), AllDeepSets w/ labels, and RelationEmbed w/ labels (fine-tuned over our foundation model) for abusive merchant detection task on M1. Yellow nodes indicate abusive merchants, and blue nodes are benign merchants. From this figure, we find out: (i) RelationEmbed's embeddings (a) show a clear trend in distinguishing abusive and benign merchants, even without labels. (ii) RelationEmbed's fine-tuned embeddings (c) have a more distinct boundary between abusive and benign merchants compared to (b), with less overlap. (iii) This superior performance in learning merchant embeddings from unlabeled data validates the effectiveness of RelationEmbed.

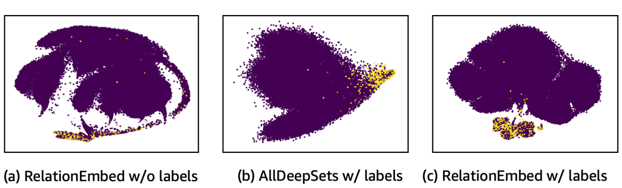

Figure 4: Embedding visualization on abusive merchant detection task on M1 market.

5.2.3 Ablation Study: We further conduct ablation experiments on two tasks (abusive merchant classification and abusive group classification) across M1 and M2 markets to analyze the contribution of each component in RelationEmbed. We remove the contrastive learning module (A1), hypergraph augmentation (A2), and group-level hypergraph contrastive learning (A3) separately. The results in Figure 5 show that removing the contrastive module (A1) causes the largest performance drop, indicating its significant contribution. Removing hypergraph augmentation (A2) also decreases, highlighting its necessity. Removing group-level contrastive learning (A3) leads to a clear decline, especially on abusive group detection tasks, showing its importance in RelationEmbed.

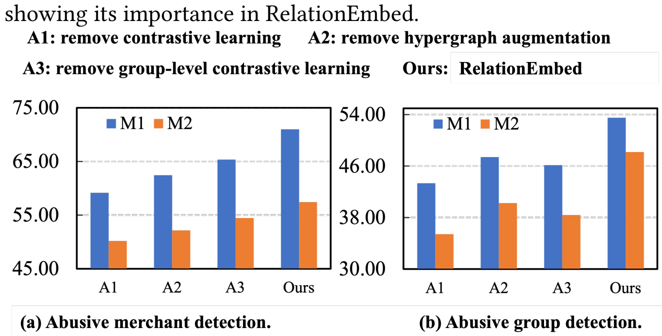

Figure 5: Performances of different model variants about RelationEmbed over two detection tasks across two markets.

5.3 Performance Discussion on TaskReport

TaskReport is a highly efficient module that generates interpretability-driven reports for downstream tasks in seconds (approximately 5 seconds), saving valuable time compared to the typical 30 minutes required to analyze a case and generate a report manually. To validate the interpretability of reports generated by TaskReport, we design and develop a LLM-as-a-Judge module to evaluate the quality of the reports in factuality (factual correct) and clarity (understandability) using a 1-3 rating scale: 1-Incorrect/Unclear, 2-Partial Correct/Clear, and 3-Correct/Clear. Mention that, for factuality evaluation, we focus on validating whether these statements in the report are correct.

In this work, we employ Claude 3 Sonnet [2] to generate the reports, while employ Claude 3 Sonnet 3.5 [3] to evaluate them. Table 3 lists the average factuality correct scores and clarity scores of the generated 500 reports for each task. We find that the evaluation results achieve relatively high performance on both factuality and clarity dimensions, demonstrating the excellent generation ability of TaskReport in providing reliable and interpretable reports.

6 Conclusion

In this paper, we propose RelationExpert, a framework for e-commerce representation learning. It consists of two main modules: RelationEmbed and TaskReport. RelationEmbed is a representation learning foundation model that learns embeddings for merchants, customers, and products from unlabeled real-world hypergraph data. These learned embeddings can be directly applied to various downstream tasks, such as abuser detection tasks and product-customer recommendation tasks. TaskReport automatically generates reports interpreting the classification results in downstream tasks. Comprehensive experiments on eight tasks across two real-world markets demonstrate RelationEmbed's superiority, achieving significant improvements over existing baseline models. Evaluations of the generated reports validate TaskReport's factual correctness and interpretability. These results highlight RelationExpert's capabilities in learning complex relations in e-commerce markets.

References

[1] Mohammad Allahbakhsh, Aleksandar Ignjatovic, Boualem Benatallah, Seyed-Mehdi-Reza Beheshti, Elisa Bertino, and Norman Foo. Collusion detection in online rating systems. In *APWeb*, 2013.

[2] Anthropic. The claude 3 model family: Opus, sonnet, haiku. https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf, 2024.

[3] Anthropic. Claude 3.5 sonnet. https://www.anthropic.com/news/claude-3-5-sonnet, 2024.

[4] Xuehan Bai, Yan Li, Yanhua Cheng, Wenjie Yang, Quan Chen, and Han Li. Cross-domain product representation learning for rich-content e-commerce. In *ICCV*, 2023.

[5] Jiuxin Cao, Rongqing Xia, Yifang Guo, and Zhuo Ma. Collusion-aware detection of review spammers in location based social networks. *World Wide Web*, 22(6):2921–2951, 2019.

[6] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *KDD*, 2016.

[7] Yankai Chen, Tuan Truong, Xin Shen, Jin Li, and Irwin King. Shopping trajectory representation learning with pre-training for e-commerce customer understanding and recommendation. 2024.

[8] Yanzhe Chen, Huasong Zhong, Xiangteng He, Yuxin Peng, and Lele Cheng. Real20m: A large-scale e-commerce dataset for cross-domain retrieval. In *MM*, 2023.

[9] Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are allset: A multiset function framework for hypergraph neural networks. In *ICLR*, 2022.

[10] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *AAAI*, 2019.

[11] Soumava Ghosh, Ravi Anand, Tanmoy Bhowmik, and Siddhanth Chandrashekhar. Gosage: Heterogeneous graph neural network using hierarchical attention for collusion fraud detection. In *ICAIF*, 2023.

[12] Ritwik Giri, Srikanth V Tenneti, Fangzhou Cheng, Karim Helwani, Umut Isik, and Arvindh Krishnaswamy. Self-supervised classification for detecting anomalous sounds. 2020.

[13] Yulong Gu, Wentian Bao, Dan Ou, Xiang Li, Baoliang Cui, Biyu Ma, Haikuan Huang, Qingwen Liu, and Xiaoyi Zeng. Self-supervised learning on users' spontaneous behaviors for multi-scenario ranking in e-commerce. In *CIKM*, 2021.

[14] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NeurIPS*, 2017.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[16] A Hedayat and Walter Dennis Wallis. Hadamard matrices and their applications. *The annals of statistics*, pages 1184–1238, 1978.

[17] Jing Huang and Jie Yang. Unignn: a unified framework for graph and hypergraph neural networks. In *IJCAI*, 2021.

[18] Yang Jin, Yongzhi Li, Zehuan Yuan, and Yadong Mu. Learning instance-level representation for large-scale multi-modal pretraining in e-commerce. In *CVPR*, 2023.

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[20] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[21] Anders Krogh and John Hertz. A simple weight decay can improve generalization. 1991.

[22] Dongjin Lee and Kijung Shin. I'm me, we're us, and i'm us: Tri-directional contrastive learning on hypergraphs. In *AAAI*, 2023.

[23] Chenyi Lei, Yong Liu, Lingzi Zhang, Guoxin Wang, Haihong Tang, Houqiang Li, and Chunyan Miao. Semi: A sequential multi-modal information transfer network for e-commerce micro-video recommendations. In *KDD*, 2021.

[24] Zhao Li, Xin Shen, Yuhang Jiao, Xuming Pan, Pengcheng Zou, Xianling Meng, Chengwei Yao, and Jiajun Bu. Hierarchical bipartite graph neural networks: Towards large-scale e-commerce applications. In *ICDE*. IEEE, 2020.

[25] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[26] Tianyi Ma, Yiyue Qian, Chuxu Zhang, and Yanfang Ye. Hypergraph contrastive learning for drug trafficking community detection. In *ICDM*, 2023.

[27] Himangshu Paul and Alexander Nikolaev. Fake review detection on online e-commerce platforms: a systematic literature review. *Data Mining and Knowledge Discovery*, 35(5):1830–1881, 2021.

[28] Yiyue Qian. *Graph Representation Learning Techniques for the Combat Against Online Abusive Activity.* PhD thesis, University of Notre Dame, 2024.

[29] Yiyue Qian, Philip Chen, Song Cui, and De Chen. Universal ring-of-abusers detection via multi-modal heterogeneous graph learning. 2023.

[30] Yiyue Qian, Tianyi Ma, Chuxu Zhang, and Yanfang Ye. Adaptive expansion for hypergraph learning. 2023.

[31] Yiyue Qian, Tianyi Ma, Chuxu Zhang, and Yanfang Ye. Dual-level hypergraph contrastive learning with adaptive temperature enhancement. In *WWW*, 2024.

[32] Yiyue Qian, Tianyi Ma, Chuxu Zhang, and Yanfang Ye. Adaptive graph enhancement for imbalanced multi-relation graph learning. In *WSDM*, 2025.

[33] Yiyue Qian, Chunhui Zhang, Yiming Zhang, Qianlong Wen, Yanfang Ye, and Chuxu Zhang. Co-modality graph contrastive learning for imbalanced node classification. In *NeurIPS*, 2022.

[34] Yiyue Qian, Yiming Zhang, Nitesh Chawla, Yanfang Ye, and Chuxu Zhang. Malicious repositories detection with adversarial heterogeneous graph contrastive learning. In *CIKM*, 2022.

[35] Yiyue Qian, Yiming Zhang, Qianlong Wen, Yanfang Ye, and Chuxu Zhang. Rep2vec: Repository embedding via heterogeneous graph adversarial contrastive learning. In *SIGKDD*, 2022.

[36] Yiyue Qian, Yiming Zhang, Yanfang Ye, and Chuxu Zhang. Adapting meta knowledge with heterogeneous information network for covid-19 themed malicious repository detection. In *IJCAI*, 2021.

[37] Yiyue Qian, Yiming Zhang, Yanfang Ye, and Chuxu Zhang. Distilling meta knowledge on heterogeneous graph for illicit drug trafficker detection on social media. In *NeurIPS*, 2021.

[38] Yiming Qiu, Chenyu Zhao, Han Zhang, Jingwei Zhuo, Tianhao Li, Xiaowei Zhang, Songlin Wang, Sulong Xu, Bo Long, and Wen-Yun Yang. Pre-training tasks for user intent detection and embedding retrieval in e-commerce search. In *CIKM*, 2022.

[39] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP-IJCNLP*, 2019.

[40] Kyuyong Shin, Hanock Kwak, Kyung-Min Kim, Minkyu Kim, Young-Jin Park, Jisu Jeong, and Seungjae Jung. One4all user representation for recommender systems in e-commerce. *arXiv preprint arXiv:2106.00573*, 2021.

[41] Kyuyong Shin, Hanock Kwak, Su Young Kim, Max Nihlén Ramström, Jisu Jeong, Jung-Woo Ha, and Kyung-Min Kim. Scaling law for recommendation models: Towards general-purpose user representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4596–4604, 2023.

[42] Wonyoung Shin, Jonghun Park, Taekang Woo, Yongwoo Cho, Kwangjin Oh, and Hwanjun Song. e-clip: Large-scale vision-language representation learning in e-commerce. In *CIKM*, 2022.

[43] Yumeng Song, Yu Gu, Tianyi Li, Jianzhong Qi, Zhenghao Liu, Christian S Jensen, and Ge Yu. Chgnn: A semi-supervised contrastive hypergraph learning network. *arXiv preprint arXiv:2303.06213*, 2023.

[44] Tianxin Wei, Yuning You, Tianlong Chen, Yang Shen, Jingrui He, and Zhangyang Wang. Augmentations in hypergraph contrastive learning: Fabricated and generative. In *NeurIPS*, 2022.

[45] Qianlong Wen, Zhongyu Ouyang, Chunhui Zhang, Yiyue Qian, Yanfang Ye, and Chuxu Zhang. Adversarial cross-view disentangled graph contrastive learning. *arXiv preprint arXiv:2209.07699*, 2022.

[46] Qianlong Wen, Zhongyu Ouyang, Chunhui Zhang, Yiyue Qian, Chuxu Zhang, and Yanfang Ye. Gcvr: Reconstruction from cross-view enable sufficient and robust graph contrastive learning. In *UAI*, 2022.

[47] Qianlong Wen, Zhongyu Ouyang, Jianfei Zhang, Yiyue Qian, Yanfang Ye, and Chuxu Zhang. Disentangled dynamic heterogeneous graph learning for opioid overdose prediction. In *KDD*, 2022.

[48] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *ICLR*, 2019.

[49] Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha Talukdar. Hypergcn: A new method for training graph convolutional networks on hypergraphs. In *NeurIPS*, 2019.

[50] Yanfang Ye, Yujie Fan, Shifu Hou, Yiming Zhang, Yiyue Qian, Shiyu Sun, Qian Peng, Mingxuan Ju, Wei Song, and Kenneth Loparo. Community mitigation: A data-driven system for covid-19 risk assessment in a hierarchical manner. In *CIKM*, 2020.

[51] Yanfang Ye, Shifu Hou, Yujie Fan, Yiming Zhang, Yiyue Qian, Shiyu Sun, Qian Peng, Mingxuan Ju, Wei Song, and Kenneth Loparo. α-satellite: An ai-driven system and benchmark datasets for dynamic covid-19 risk assessment in the united states. *IEEE Journal of Biomedical and Health Informatics*, 2020.

[52] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. In *NeurIPS*, 2020.

[53] Tom Zahavy, Abhinandan Krishnan, Alessandro Magnani, and Shie Mannor. Is a picture worth a thousand words? a deep multi-modal architecture for product classification in e-commerce. In *AAAI*, 2018.

[54] Yiming Zhang, Yiyue Qian, Yujie Fan, Yanfang Ye, Xin Li, Qi Xiong, and Fudong Shao. dstyle-gan: Generative adversarial network based on writing and photography styles for drug identification in darknet markets. In *ACSAC*, 2020.

[55] Yiming Zhang, Yiyue Qian, Yanfang Ye, and Chuxu Zhang. Adapting distilled knowledge for few-shot relation reasoning over knowledge graphs. In *SDM*, 2022.

[56] Da Zheng, Minjie Wang, Quan Gan, Zheng Zhang, and George Karypis. Learning graph neural networks with deep graph library. In *WWW*, 2020.

[57] Yushan Zhu, Huaixiao Zhao, Wen Zhang, Ganqiang Ye, Hui Chen, Ningyu Zhang, and Huajun Chen. Knowledge perceived multi-modal pretraining in e-commerce. In *MM*, 2021.