# Prohibited Item Detection via Risk Graph Structure Learning

Yugang Ji
jiyugang@bupt.edu.cn
Beijing University of Posts and Telecommunications
Beijing, China

Guanyi Chu
cgy463@bupt.edu.cn
Beijing University of Posts and Telecommunications
Beijing, China

Xiao Wang
xiaowang@bupt.edu.cn
Beijing University of Posts and Telecommunications
Peng Cheng Laboratory
China

Chuan Shi*
shichuan@bupt.edu.cn
Beijing University of Posts and Telecommunications
Peng Cheng Laboratory
China

Jianan Zhao
andy.zhaoja@gmail.com
Unviersity of Notre Dame
Notre Dame, USA

Junping Du
junpingdu@126.com
Beijing University of Posts and Telecommunications
Beijing, China

## ABSTRACT

Prohibited item detection is an important problem in e-commerce, where the goal is to detect illegal items online for evading risks and stemming crimes. Traditional solutions usually mine evidence from individual instances, while current efforts try employing advanced Graph Neural Networks (GNN) to utilize multiple risk-relevant structures of items. However, it still remains two essential challenges, including weak structure and weak supervision. This work proposes the **R**isk **G**raph **S**tructure **L**earning model (**RGSL**) for prohibited item detection. RGSL first introduces structure learning into large-scale risk graphs, to reduce noisy connections and add similar pairs. It then designs the pairwise training mechanism, which transforms the detection process as a metric learning from candidates to their similar prohibited items. Furthermore, RGSL generates risk-aware item representations and searches risk-relevant pairs for structure learning iteratively. We test RGSL on three real-world scenarios, and the improvements to baselines are up to **21.91%** in AP and **18.28%** in MAX-F1. Meanwhile, RGSL has been deployed on an e-commerce platform, and the improvements to traditional solutions are up to **23.59%** in ACC@1000 and **6.52%** in ACC@10000.

## CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; **Artificial intelligence**; • **Information systems** → **World Wide Web**.

## KEYWORDS

prohibited item detection, risk graph, graph structure learning, pairwise labeling

* Corresponding Author.


*WWW '22, April 25–29, 2022, Virtual Event, Lyon, France*

ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3512190

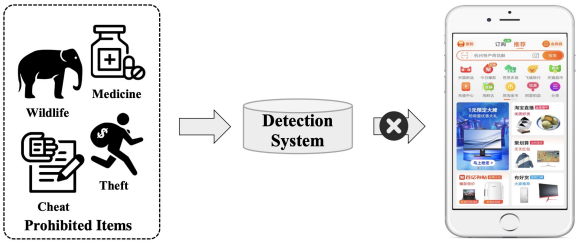



**Figure 1: Prohibited item detection in e-commerce.**

**ACM Reference Format:**
Yugang Ji, Guanyi Chu, Xiao Wang, Chuan Shi*, Jianan Zhao, and Junping Du. 2022. Prohibited Item Detection via Risk Graph Structure Learning. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3485447.3512190

## 1 INTRODUCTION

Nowadays, increasing penetration and rapid development of online shopping have significantly changed the lifestyle of humans [9, 39], due to advantages in lower prices and wide product variety. However, there are various items against laws hidden in e-commerce as well, creating prominent personal and social issues. For instance, as shown in Figure 1, millions of counterfeit medicines and wildlife products continuously tried to sell on Taobao platform[1].

In order to search and delete enormous illegal items, prohibited item detection has played an essential and fundamental role during the past decades [12, 29, 34]. Traditional industrial solutions prefer to deploy conventional machine learning or deep learning methods [8, 25] to independently make decisions for each instance based on feature engineering. Obviously, these methods suffer from heavy adversarial efforts by illicit sellers. It is easy to camouflage attributes (e.g., textural descriptions and images) of prohibited items, very similar to those of normal ones. More importantly, there are abundant relations among items to describe latent risk relevance caused by shared factors (e.g., items sold by the same seller, items clicked by the same visitor and so on). It is usually much more unaffordable

[1] https://www.taobao.com/

to disguise theses risk-aware relations, through some adversarial attacks against relevance topology. However, such valuable risk relevance cannot be directly introduced in existing solutions and have been rarely employed until now.

Graph Neural Networks (GNN) [1, 35] have become popular for graphic structure analysis, due to the ability to simultaneously model both structures and attributes via neighborhood aggregation. The powerful GNNs [11, 19, 30] have been widely introduced to handle various e-commerce scenarios such as recommendation systems [2, 6, 22], user alignment [42], review helpfulness prediction [20, 26] and so on. A naïve idea is to connect items by risk-relevant relations (e.g., "*same seller*", "*same visitor*", "*relevant seller*" and etc. ) as a risk graph and directly employ existing GNNs to learn representations of items for node classification. However, it still remains two vital problems need to be considered. On the one hand, the manually designed risk relations are often noisy and incomplete as well as heterogeneous (i.e., weak structures), while GNNs are highly sensitive to the quality of graph structures for information propagation. On the other hand, there are multiple subcategories of each risk (e.g., illegal medicine devices contain prescription drugs, red soluble fat and scar cream), while items are simply labeled as prohibited or not. This phenomenon (i.e., weak supervision) indicates that same-labeled items would contain very different attributes, violating basic assumption of node classification. To sum up, aiming at fully exploiting both structures and attributes to handle the problem of prohibited item detection, we have to face the following challenges.

First, how to learn and optimize heterogeneous structures of risk graphs? To adaptively learn high-quality graph structures for GNNs, several studies have been proposed around the central concept of graph structure learning (GSL) [4, 41, 46]. Traditional GSL prefers to revise structures by removing noisy connections from topologies and adding necessary pairs based on their attribute similarity. However, most methods deal with homogeneous and small graphs, which are inadequate in real-world heterogeneous graphs, especially risk scenarios containing abundant semantics. Moreover, existing approaches [4, 40, 46] prefer to optimize the whole graph, which would suffer from unaffordable memory cost and high computational complexity when dealing with large-scale graphs.

Second, how to handle the weak supervised information for training and inferring? Traditional GSL-based GNNs are usually designed for node-wise classification, resulting in the limitation when dealing with multiple subcategories of risks. A basic idea is to introduce metric learning to learn the distance between labeled items. However, because of limited human resources, items in risk graphs are labeled with simple binary value (i.e., prohibited or not). The constraints that same-labeled items should be closer is very difficult to satisfy. Besides, we mainly focus on detecting prohibited items, the metric learning between normal items could be meaningless while occupying much computational cost.

In this paper, we are the first to introduce graph structure learning to prohibited item detection. We first construct a risk graph to describe the risk-relevant relations among items, and then propose a novel iterative **R**isk **G**raph **S**tructure **L**earning framework (**RGSL**) for prohibited item detection. RGSL respectively designs the heterogeneous structure learning and pairwise metric learning on risk graphs to overcome both weak structures and weak supervision. Specifically, heterogeneous structure learning is to reduce noisy structures and add incomplete connections by evaluating their corresponding importance, and then construct item representations with effective heterogeneous message passing. And then, we transform the detection process as a pairwise metric learning from candidates to their corresponding similar illegal seeds. Furthermore, we design the unified framework which iteratively learns risk-relevant representations of items and generates confident label-similar pairs for GSL in turn.

In a nutshell, the contributions of this paper are:

- To our best knowledge, we are the first to introduce graph structure learning to address the problem of prohibited item detection, which can clarify the risk-relevant structures and identify prohibited items at the same time.
- We propose the novel RGSL consisting of both heterogeneous structure learning and pairwise training in an iterative manner. RGSL is able to construct risk-relevant structures and generate effective item representations by iteratively keeping the consistency of feature smoothness and label homophily during detection process.
- We evaluate RGSL by designing both various offline and online experiments. Compared to state-of-the-art alternatives, the improvements of RGSL are obvious up to 21.91%, 18.28%, 23.59% and *6.52%* in AP, Max-F1, ACC@1000 and ACC@10000 metrics.

## 2 RELATED WORK

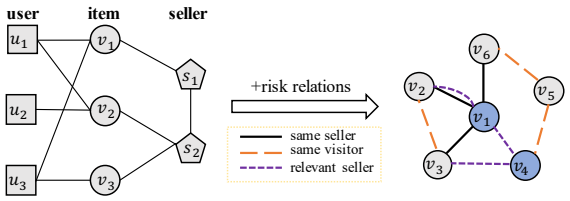
This section briefly introduces and analyzes the most related work, including graph neural networks and graph structure learning.

Graph neural networks (GNN) [35], aiming at extending deep neural networks to model structured data, have been widely used for representation learning on graphs [11, 13, 19, 30]. Kipf *et al.* [19] propose the Graph Convolutional Networks (GCN) which propagates graph information recursively to generate node representation. Hamilton *et al.* [11] further extend GNNs to an inductive setting via sampled neighborhood aggregation. Considering that neighborhoods are often noisy, researchers introduce attention mechanism [30] and importance sampling strategies [3, 14] to re-wight neighborhood aggregation by attributed and structural similarity. However, these GNNs pay no attention to types of nodes and edges, implying the limitation when modeling risk graphs. Recently, heterogeneous GNNs have become popular [13, 16, 32, 38, 42] for representation learning on heterogeneous graphs, preserving semantics of real-world graphs as much as possible. The earlier RGCN [27] directly designs multiple linear projection weights for each edge type. Wang *et al.* [32], Cen *et al.* [2] and Hu *et al.* [13] respectively introduce hierarchical heterogeneous attention, heterogeneous self-attention and heterogeneous mutual attention mechanisms to extract fine-grained semantic information propagation. Meanwhile, Ji *et al* [16] summarize the general framework of heterogeneous GNNs and propose several heterogeneous importance sampling strategies to keep effective and efficient message passing of different-typed neighborhoods. However, when dealing with risk graphs, current homogeneous and heterogeneous GNNs have to face the incomplete structures and weak supervision because of enormous items but limited human resources.

**Table 1: Notations.**

| Notation | Description |
|---|---|
| $\mathcal{G}$ | the input risk graph |
| $\mathcal{G}_T$ | the topological structures of $\mathcal{G}$ |
| $\mathcal{G}_A$ | the attribute similarity graph of $\mathcal{G}$ |
| $\mathcal{V}$ | the item set of $G$ |
| $\mathcal{V}_L, \mathcal{V}_S$ | the labeled item set and the prohibited seeds of $G$ |
| $\mathcal{E}$ | the edge set of $G$ |
| $\mathcal{R}$ | the relation type set of $\mathcal{E}$ |
| $\psi \in \mathcal{R}$ | the relation type of $\mathcal{R}$ |
| $\mathcal{N}_{\psi,i}$ | the type-$\psi$ neighbors of $v_i$ |
| $d_a$ | the dimension of attributes |
| $d$ | the dimension of representation |
| $K$ | the size of pairwise labels |
| $T$ | the number of iterative training |
| $\boldsymbol{X} \in \mathbb{R}_a^d$ | the attributes of items |
| $\boldsymbol{Y}$ | the set of labels |
| $\boldsymbol{\Gamma}_\psi(v_i, v_j)$ | the type-$\psi$ similarity of $v_i$ and $v_j$ |
| $\mathcal{G}_A^t$ | the $t^{th}$ attribute similarity graph |
| $\boldsymbol{H}^t \in \mathbb{R}^d$ | the $t^{th}$ representation of items |

As GNNs usually require high-quality graphic structures for effective message passing, Graph Structure Learning (GSL) has attached many attentions recently, targeting at jointly learning an optimized graph and corresponding representations [46]. GLNN [10] proposes to keep both sparsity and feature smoothness of graphs and incorporates them into a hybrid objective. ProGNN [17] takes low-rank prior into consideration by using nuclear norm of generated graphs. Following the assumption that graphs are generated via a sampling process from certain distributions, LDS-GNN [7] frames GSL as a bilevel programming problem and generate edges of each node with a parameterized Bernoulli distribution. GAUG-O [41] proposes to integrate GNN based edge predictor with traditional adjacency matrix as sampling probability. GEN [31] keeps label homophily within both embedding-based KNN graphs and topological graphs in all layers of GNNs. The most popular direction of GSL is directly learn a metric function between pairwise representations of nodes to derive edge weights. AM-GCN [33] learns the relevance and difference between topological graph and attribute-KNN graph at the same time. GAUG-M [41] integrates with the self-supervised graph embedding while GRCN [37] utilizes the GNN-based node representation to generate confident unconnected pairs as candidates for metric learning. IDGL [4] proposes to generate graphs and learn metrics in an iterative manner to keep both feature smoothness and label homophily. Unfortunately, these methods are all to deal with homogeneous graphs, which cannot handle the rich semantics within risk graphs. Limited work [40] has been done in heterogeneous graphs by propagating the probability between relevant nodes in views of attributes and topologies. However, the high computational complexity and expensive memory cost make it unaffordable when dealing with risk detection. Moreover, the weak supervision also makes it difficult to train an effective GNN.




(a) E-commerce Graph (b) Risk Graph

**Figure 2: The construction of risk graph. $v_1$ and $v_4$ are two prohibited items and the others are normal ones.**

## 3 PRELIMINARIES

Here we introduce the concepts of risk graphs, the problem of prohibited item detection as well as graph homophily. The main notations are summarized in Table 1.

In e-commerce platforms, graph modeling and representation learning have been widely used in many tasks such as user alignment [42] and intent recommendation [6]. In risk scenarios, we prefer to connect items by risk-relevant relations, where neighborhoods are likely to share risk.

Definition 1. ***Risk Graph.*** *A risk graph is denoted as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \boldsymbol{X}\}$ where $\mathcal{V}$ and $\mathcal{E}$ are the set of items and their edges, $\boldsymbol{X}$ is the attributes of nodes. The types of edges are all recorded as $\mathcal{R}$. Notice that $|\mathcal{R}| \geq 2$.*

There are a lot of risk-relevant relations in industry and we select the three representative relations for discussion. As shown in Figure 2(b), we connect independent items via several risk-relevant relations, including (1) *same seller* to connect items belong to the same seller, (2) *same visitor* to connect items which are visited by some same visitors, and (3) *relevant seller* to connect items sold by relevant sellers to overcome the multiple fake identifications of adversarial sellers. Compared to traditional heterogeneous graphs [28], the structures of risk graphs are often noisy and incomplete where prohibited items usually connect with normal items rather than other prohibited items due to adversarial efforts of illegal sellers and the concentration of normal items.

Definition 2. ***Prohibited Item Detection [15].*** *Given a risk graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \boldsymbol{X}\}$, label set $\boldsymbol{Y}$, prohibited item detection is to learn an effective identification function $\mathcal{H} : \mathcal{H}(\mathcal{G}) \rightarrow \boldsymbol{Y}^{|\mathcal{V}|}$. $\boldsymbol{Y}_i \in \{0, 1\}$ where 0 and 1 respectively label the normal and prohibited items.*

To quantitatively evaluate the noisy and incomplete structures, the ratio of same-labeled pairs (i.e., homophily) is often a good choice in many scenarios [5, 24, 43, 44]. However, traditional homophily metric heavily suffers from imbalanced labels while there are a large number of normal items and very few prohibited ones in risk graph. Inspired by [21], we calculate the risk homophily metric $\rho$ on risk graphs as follows

$$\rho = \frac{1}{2}\left(\left[\frac{|\mathcal{E}_{pp}|}{|\mathcal{E}_{p*}|} - \frac{Y_p}{Y}\right]_+ + \left[\frac{|\mathcal{E}_{nn}|}{|\mathcal{E}_{n*}|} - \frac{Y_n}{Y}\right]_+\right), \tag{1}$$

where $\mathcal{E}_{pp}$ and $\mathcal{E}_{nn}$ respectively denote the edges of both prohibited or normal items, $\mathcal{E}_{p*}$ and $\mathcal{E}_{n*}$ denote the edges consisting of at least

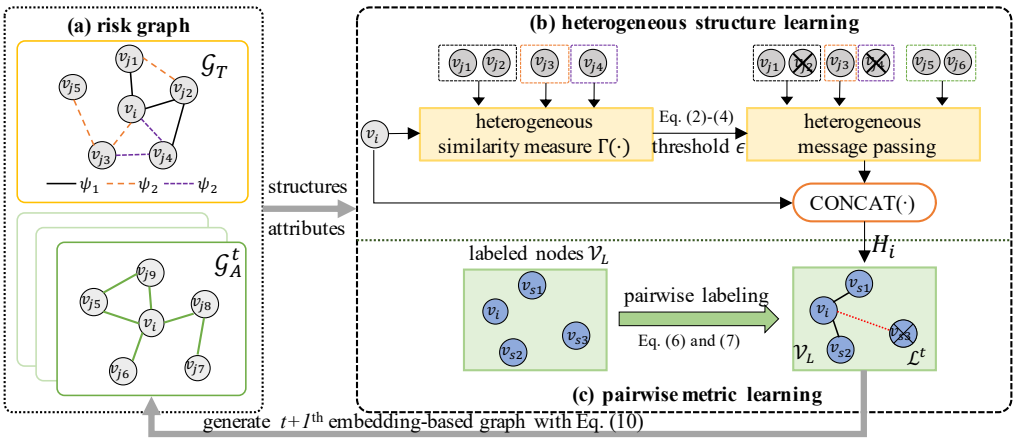



**Figure 3: The overall architecture of RGSL. (a) is the risk graph consisting of topological structures $\mathcal{G}_T$ and attribute similarity graph $\mathcal{G}_A^t$. (b) is the heterogeneous structure learning to remove noisy neighbors such as $v_{j2}$ and $v_{j4}$ and add attribute similar neighbors such as $v_{j5}$ and $v_{j6}$ to construct node representation $H_i$. (c) is the pairwise metric learning to classify edges between candidate $v_i$ and its relevant prohibited seeds $v_{s1}, v_{s2}$ except $v_{s3}$. It further iteratively generates new $\mathcal{G}_A^{t+1}$.**

one prohibited/normal items, $Y = Y_p \cup Y_n$ denotes the labels where $Y_p$ and $Y_n$ are prohibited items and norm items, $|\cdot|$ is the count option, Abs$[\cdot]$ is to extract the absolute information. Obviously, $\rho$ eases the influence from imbalance labels by considering both the ratio of prohibited and norm items. The risk homophily can not only evaluate graph structures but also guide iterative training at the same time.

# 4 METHODOLOGY

## 4.1 Overview

Figure 3 illustrates the overall architecture of RGSL. In this architecture, given a risk graph consisting of several experiential risk-relevant relations in Figure 3(a), we respectively design the heterogeneous structure learning, pairwise metric learning and iterative training mechanism to overcome the two challenges (i.e., weak structures and weak supervision). Specifically, (1) since the constructed risk graphs in e-commerce are susceptible to noise and incomplete, in Figure 3(b), we design the heterogeneous structure learning which evaluates importance of neighborhoods in not only topologies (e.g., $\mathcal{G}_T$) but also attributes (e.g., $\mathcal{G}_A$) and adaptively remove/add low-/high-quality connections for effective message passing. (2) To handle the weak supervision in expression, as shown in Figure 3(c), we design the pairwise metric learning to learn distance from candidates to their related prohibited items. Furthermore, we iteratively learn the risk-relevant representations and generate risk similarity graphs iteratively to keep the consistency of feature smoothness and label homophily indeed.

## 4.2 Heterogeneous Structure Learning

We begin with the heterogeneous structure learning to augment risk graphs for effective message passing. Although structure learning [33, 41] has been employed to handle noisy and incomplete graphs, unfortunately, most existing approaches are on homogeneous graphs which fail to evaluate the impact of semantics inside multiple risk-relevant relations when adding or removing edges. Thereby, aiming at obtaining high-quality structures, we construct the attribute similarity graph of enormous items and further design the heterogeneous similarity measure to find similar neighborhoods from both heterogeneous topologies and attribute graphs.

At first, we calculate the Euclidean distance between different items based on attributes $\boldsymbol{X}$, and construct confident attribute similarity graph $\mathcal{G}_A$ by ranking and preserving top-20 connections of nodes to ensure efficiency. Besides, we respectively randomly sample several neighborhoods for each type of risk-relevant relations to keep efficient structure learning. Notice that, the $\mathcal{G}_A$ will be iteratively optimized since the initialized structures suffer from many adversarial attacks.

To evaluate the quality of heterogeneous topological and confident attributed neighbors of items, we design the heterogeneous similarity measure in a multi-head mechanism, taking multimodal characteristics into consideration. Specifically, given an item $v_i$, we encode the basic representation $\boldsymbol{h}_i \in \mathbb{R}^{1\times d}$ of its attributes as

$$\boldsymbol{h}_i = \sigma(\boldsymbol{X}_i\boldsymbol{W} + b), \tag{2}$$

where $\boldsymbol{X}_i \in \mathbb{R}^{1\times d_a}$ where $d_a$ denotes the size of attributes, $\boldsymbol{W} \in \mathbb{R}^{d_a\times d}$ and $b$ are learnable parameters, $d$ is the dimension of item representations, $\sigma(\cdot)$ is the activation function and we choose RELU$(\cdot)$ in this work.

Then we transform $\boldsymbol{h}_i$ as a multi-head tensor, i.e., $\boldsymbol{h}_i \in \mathbb{R}^{M\times 1\times \frac{d}{M}}$ where $M$ is the total number of heads, and then calculate the similarity of $v_j$ to $v_i$ at $\psi$-typed relation as

$$\boldsymbol{\Gamma}_\psi(v_i, v_j) = f(\gamma_\psi \boldsymbol{h}_i \boldsymbol{W}_\psi \boldsymbol{h}_j^T), \tag{3}$$

where $\boldsymbol{\Gamma}_\psi(v_i, v_j) \in \mathbb{R}^{M\times 1\times 1}$ denotes the multi-head impact of $v_i$ upon type-$\psi$ connection, $\boldsymbol{W}_\psi \in \mathbb{R}^{M\times \frac{d}{M}\times \frac{d}{M}}$ is the type-$\psi$ projection tensor, $\gamma_\psi$ is the prior factor to denote the general significance of each meta relation, and here we set $\frac{1}{\sqrt{M/d}}$. $f(\cdot)$ is to keep $\Gamma_\psi(v_i, v_j) \in (0, 1)$ which is sigmoid$(\cdot)$ used in this work.

Moreover, we obtain the learned heterogeneous similarity graph by filtering noisy topological edges and adding confident unconnected pairs, namely,

$$\hat{\boldsymbol{a}}_{i,j,\psi} = \begin{cases} \boldsymbol{\Gamma}_{\psi}(v_i, v_j), & \sum_{m} \boldsymbol{\Gamma}_{\psi}^{(m)}(v_i, v_j)/M > \epsilon, \\ \mathbf{0} & \text{otherwise}, \end{cases} \tag{4}$$

where $\hat{\boldsymbol{a}}_{i,j,\psi}$ denotes the learned edge weight, $\epsilon$ denotes the threshold value for adding/removing edges, $\boldsymbol{\Gamma}_{\psi}^{(m)}(v_i, v_j)$ is the $m^{\text{th}}$ similarity.

Furthermore, by normalizing $\hat{\boldsymbol{a}}_{i,j,\psi}$ over all neighborhoods for each head as $\boldsymbol{a}_{i,j,\psi}$, we design the heterogeneous message passing to construct final representations of nodes with adaptive neighborhoods, namely

$$\boldsymbol{H}_i = \text{HeteAGG}(\boldsymbol{h}_i, \{\boldsymbol{a}_{i,j,\psi}\boldsymbol{h}_j\boldsymbol{W}_{\psi}^{H} | v_j \in \mathcal{N}_{i,\psi}, \psi \in \mathcal{R} \cup \{A\}\}), \tag{5}$$

where $\boldsymbol{W}_{\psi}^{H} \in \mathbb{R}^{m \times \frac{d}{M} \times \frac{d}{M}}$ denotes the type-$\psi$ projection tensor to consider rich semantics, the total edge types contain the attributed type $A$ besides topological $\mathcal{R}$, HeteAGG$(\cdot)$ is the heterogeneous aggregator and we choose the popular mean-pooling here for message passing. Notice that, inspired by [44], we concatenate rather than add node and neighborhood representations to ease low homophily of structures. In addition, we transform $\boldsymbol{H}_i$ with size 1-by-$d$ by concatenating all heads finally.

## 4.3 Pairwise Metric Learning

Besides good structures to construct node representation with effective message passing, GNNs also require plenty of fine-grained supervised information for parameter optimization. However, because of limited human resources, items in risk graphs are labeled with simple binary value (i.e., prohibited or not) , while prohibited items indeed belong to multiple subcategories of risks, as mentioned in Section 1.

To enrich the supervised information in both size and expression, a natural idea is to transform node classification into metric learning [36], which learns the distance of same-labeled or different-labeled items. However, there are multiple subcategories of each risk, implying the quite difference of prohibited items. Moreover, existing labeling often equally treat both prohibited and normal items, while the relevance among normal items contributes little to risk detection. Here we introduce the pairwise metric learning of multiple subcategories as follows.

Given the observed labeled items $\mathcal{V}_L \in \mathcal{V}$ with their classification $\boldsymbol{Y}$, we first select the prohibited items as seeds $\mathcal{V}_S$, namely,

$$\mathcal{V}_S = \{v_i | v_i \in \mathcal{V}_L \wedge \boldsymbol{Y}_i = 1\}. \tag{6}$$

Then we calculate the relevance from all labeled items to prohibited seeds, and select several most relevant seeds to construct pairwise instances. The attribute-based relevance $r_a(v_i, v_j)$ is calculated by cosine similarity as

$$r_a(v_i, v_s) = \frac{\boldsymbol{X}_i \boldsymbol{X}_s}{||\boldsymbol{X}_i|| \cdot ||\boldsymbol{X}_s||}, \tag{7}$$

where $v_i \in \boldsymbol{V}_L$ and $v_s \in \boldsymbol{Y}_S$. We rank pairs of $v_i$ according to $r_a(v_i, v_s)$ and select top-$n$ pairs as pairwise labels. Besides, the connected labeled pairs in multiple relations can also be chased as labels. We label the pairs $\boldsymbol{Y}_{i,j} = \boldsymbol{Y}_i$.

Naturally, the loss with pairwise labels is defined as follows,

$$\mathcal{L}_{PW} = -\sum_{<i,s>} \boldsymbol{Y}_{i,s} \cdot log(\hat{\boldsymbol{Y}}_{i,s}) + (1 - \boldsymbol{Y}_{i,s}) log(1 - \hat{\boldsymbol{Y}}_{i,s}), \tag{8}$$

where $\hat{\boldsymbol{Y}}_{i,s}$ is the prediction of pair $< v_i, v_s >$, calculated by

$$\hat{\boldsymbol{Y}}_{i,s} = \text{MLP}(\boldsymbol{H}_i || \boldsymbol{H}_s), \tag{9}$$

where $\boldsymbol{H}_i$ and $\boldsymbol{H}_s$ calculated by Eq. (5) respectively denote the representation of $v_i$ and $v_s$, MLP$(\cdot)$ denotes the Multi-Layer Perceptron and we set the number of layers to 2 in this paper.

## 4.4 The Iterative Unified Framework

Due to adversarial attacks of illegal sellers, prohibited and normal items are often very similar, and hence the attribute similarity graph generated in Section 4.2 is likely to be low homophily. Since the homophily of structures influences performances of GNNs deeply [44], we propose to learn risk-aware embedding of items and generate corresponding confident attribute graph $\mathcal{G}_A^t$ iteratively, to keep both feature smoothness and label homophily during training.

Given representations of items $\boldsymbol{H}^{t-1}$ from $(t-1)^{\text{th}}$ epoch, we generate attribute similarity graph in $t^{\text{th}}$ epoch as

$$\mathcal{G}_{A,i,j}^t = \begin{cases} 1, & \text{rank}_i(s_{i,j}^t) \le k \\ 0 & \text{otherwise}, \end{cases} \tag{10}$$

where $\mathcal{G}_{A,i,j}^t$ denotes the connection of $v_i$ and $v_j$ at $t^{\text{th}}$ epoch, $\text{rank}_i(\cdot)$ is the descending ranking operation of $v_i$, and $s_{i,j}^t$ denotes the cosine similarity of $\boldsymbol{H}_i^t$ and $\boldsymbol{H}_j^t$. We select top-k confident pairs of $v_i$.

By modeling and integrating heterogeneous structure learning and pairwise training into a iterative unified framework, both the graph structures and GNNs can be optimized at the same time. The optimized objective is

$$\mathcal{L}^t = \mathcal{L}_{PW}^t(\mathcal{G}_T, \mathcal{G}_A^t) + \xi \cdot \Omega(\boldsymbol{\Theta}), \tag{11}$$

where $\mathcal{L}_{PW}^t(\mathcal{G}, \mathcal{G}_A^t)$ denotes the iterative pairwise loss of Eq. (8) with the corresponding $\mathcal{G}_A^t$, $\mathcal{G}_A^0 = \mathcal{G}_A$, $\mathcal{L}^0(\mathcal{G}_T, \mathcal{G}_A^0) = \mathcal{L}_{PW}$, $\xi$ denotes the regularization of all learnt parameters $\boldsymbol{\Theta}$, and $\Omega$ denotes the L2 regularization to keep generalization and avoid over-fitting. We adopt Adam [18] to minimize $\mathcal{L}^t$ in a distributed batch-wise training manner to ensure both effectiveness and efficiency. In addition, when inferring, we calculate the average probability of unlabeled candidates to prohibited seeds as the final prediction.

## 4.5 Complexity Analysis

Since attribute similarity graphs are generated offline. the major computational complexity of RGSL is from heterogeneous structure learning and pairwise training. For heterogeneous structure learning, the complexity is $O((|\mathcal{R}|+1) \times n \times d \times d_a \times m \times |Y|)$ where $n$ is the size of neighbors of each meta relation, $d_a$ and $d$ are the dimension of attributes and representations. For pairwise training, the exception of complexity is $O(([k(1-\epsilon)+1]n \times |Y| \times (|\mathcal{R}|+1) \times 2d^2 \times M)$ where $k$ is the size of pairwise labels. Obviously, the computational complexity is linear with the scale of supervised items, implying the scalability of our RGSL.

**Table 2: Description of datasets.**

| Dataset | Medicine | Cheat | Theft |
|---|---|---|---|
| #Nodes | 9,268,306 | 12,838,322 | 10,991,736 |
| # Same Seller | 1,756,013 | 3,207,579 | 4,500,577 |
| # Relevant Seller | 3,546,395 | 4,461,413 | 6,077,735 |
| # Same Visitor | 2,075,251 | 4,770,026 | 6,261,145 |
| # Labels | 975,193 | 1,236,492 | 1,344,712 |
| Label Rate | 10.52% | 9.63% | 12.23% |
| Prohibition Ratio | 3.84% | 1.93% | 2.43% |
| Homophily | 0.2036 | 0.1943 | 0.1319 |

# 5 EXPERIMENTS

In this section, we conduct extensive experiments on real-world datasets, and do the ablation study and parameter analysis to showcase the advantages of our design choices. In addition, besides offline evaluation, our RGSL have been deployed on real-world e-commerce platforms and do online testing. Since traditional GSL approaches would suffer from too expensive time cost, we select MLP for comparison online.

## 5.1 Datasets

We collect the two-week web-scale datasets in three risk scenarios including illegal medical device (i.e., "*Medicine*"), cheating and counterfeiting equipment (i.e., "*Cheat*") and theft tool (i.e., "*Theft*"), from a popular second-hand commodity trading platform.

For each risk dataset, we empirically construct a risk graph to preserve the risk-relevant semantics between billions of items, and adopt word2vec [23] to embed the description of items as 512-dimensional numerical features. Next, we introduce how to construct training, validation and test instances. For offline experiments , the instances are randomly divided into training and testing with rate 9:1. The detailed statistics of these datasets are described in Table 2. Besides offline experiments, we also evaluate the performance of our method by designing online testing of one-week online dataset.

## 5.2 Experimental Settings

*5.2.1 Baselines.* We compare with eight representative baseline methods including traditional LR and current powerful MLP which have been deployed on e-commerce platforms, *six* outstanding GNNs including homogeneous GNNs ( GraphSAGE [11] and GAT [30]), GSL based homogeneous GNNs (GAUG [41] and GRCN [37]), and heterogeneous GNNs (HAN [32] and HGT [13]).

- **Logistic Regression (LR)** is a traditional machine learning algorithm with good interpretability used in industry for classification.
- **Multi-Layer Perception (MLP)** is a classic deep learning algorithm which detects the nonlinear relevance of item features via multi-layer neural networks for classification.
- **GraphSAGE** [11] is a representative GNN model which construct node representations by gathering information from neighborhoods following feature smoothness assumption.
- **GAT** [30] is an eminent GNN framework which calculates the relevance between nodes and their neighborhoods via attention mechanism to keep noise reduction.
- **GRCN** [37] is a GSL based GNN which performs edge addition and edge re-weighting based on both attributes and topologies. Notice that, the basic GNN model is GraphSAGE in this work to keep the efficiency.
- **GRUG** [41] is a GSL model as well and it integrates self-supervised GNN to reverse topological structures with embedding-based adjacency matrix. Notice that, we utilize GRUG-M as the comparison and the basic GNN model is set as GraphSAGE in this work.
- **HAN** [32] is a popular heterogeneous GNN which designs a hierarchical message passing consisting of both node-level and semantic-level attention mechanisms of heterogeneous neighbors.
- **HGT** [13] is a widely used heterogeneous graph transformer to deal with heterogeneous interaction graphs based on heterogeneous mutual attention mechanism to aggregate information considering both edge and node types.
- **RGSL**. This is our proposed model consisting of heterogeneous structure learning and pairwise training to handle the problem of low-quality structure and few simple labeling in prohibited item detection scenarios.

*5.2.2 Implementation Details.* We train and test all baselines and our RGSL with Tensorflow 1.12 on PAI platform[2] with Tesla GeForce GTX 1080 Ti Cluster. Besides, we utilize AliGraph [45] API to load the very large risk graphs and do neighborhood sampling in a distributed system. All parameters of these models are randomly initialized with Gaussian distribution and optimized with Adam [18] to keep fair comparison. We set the number of workers to 8, the batch size to 1024, the learning rate to 0.005, the feature embedding $d$ to 512, the regularization weight $\gamma$ to 0.01 and the dropout rate to 0.4. The sample size $n$ in homogeneous GNNs (i.e., GraphSAGE, GAT, GRCN and GRUG) is set to 5. For heterogeneous GNNs (i.e., HAN, HGT and RGSL), we set the sample size of each relation to 5. All methods are optimized until in convergence. We generate 5 pairs for each candidate in RGSL and set the maximum iteration up to 20 and the iterations $T$ to 3. The hyper-parameter sensitivity will be discussed in Section 5.5.

*5.2.3 Evaluation Metrics.* In offline experiments, we select the **Max-F1** (the max F1 value by varying the threshold of recalled items), Average Precision (**AP**) and Area Under Curve (**AUC**) as the metrics to evaluate global performance of all test instances. In online experiments, we choose ACC@1000 and ACC@10000 (i.e., the accuracy of top 1,000 or 10,000 recalled items by manually reviewing) to showcase the improvement of RGSL to the deployed MLP model. Notice that, the larger values of Max-F1, AP, AUC ACC@1000 or ACC@10000 indicate the better performance.

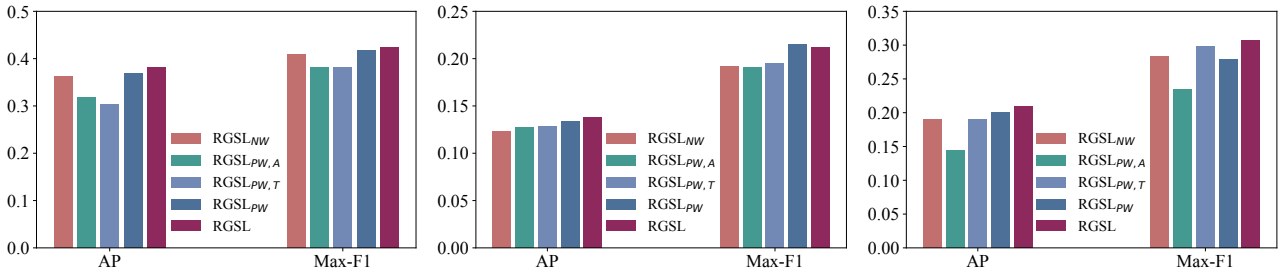
## 5.3 Performance Evaluation

In this section, we empirically compare RGSL to several state-of-the-art alternatives in three risk scenarios on e-commerce platforms.

[2]https://www.aliyun.com/product/bigdata/product/learn

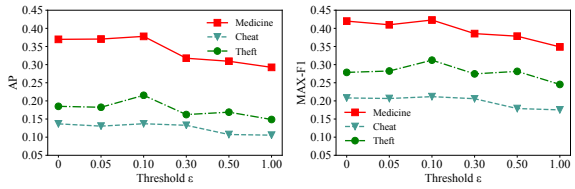
**Table 3: Performance of baselines and RGSL for risk detection on the three datasets. The best performance is in bold and the second best is underlined. Relative improvements of RGSL w.r.t. the second best are reported as well.**

| Dataset | Metric | LR | MLP | GraphSAGE | GAT | GAUG | GRCN | HAN | HGT | RGSL | Improv. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Medicine | AP | 0.1629 | 0.2925 | 0.2666 | 0.2624 | 0.2764 | 0.2792 | 0.2747 | 0.3128 | **0.3813** | 21.91% |
| | Max-F1 | 0.2454 | 0.3489 | 0.3465 | 0.3486 | 0.3566 | 0.3558 | 0.3504 | 0.3895 | **0.4230** | 8.59% |
| | AUC | 0.9497 | 0.9679 | 0.9690 | 0.9703 | 0.9690 | 0.9683 | 0.9723 | 0.9758 | **0.9781** | 0.24% |
| Cheat | AP | 0.0579 | 0.1055 | 0.1084 | 0.1123 | 0.1191 | 0.1119 | 0.1122 | 0.1168 | **0.1432** | 20.20% |
| | Max-F1 | 0.1183 | 0.1752 | 0.1708 | 0.1765 | 0.1889 | 0.1803 | 0.1844 | 0.1903 | **0.2251** | 18.28% |
| | AUC | 0.9253 | 0.9416 | 0.9479 | 0.9496 | 0.9500 | 0.9474 | 0.9498 | 0.9528 | **0.9576** | 0.50% |
| Theft | AP | 0.1292 | 0.1487 | 0.1746 | 0.1614 | 0.1448 | 0.1729 | 0.1618 | 0.1695 | **0.2106** | 20.64% |
| | Max-F1 | 0.2234 | 0.2454 | 0.2578 | 0.2552 | 0.2516 | 0.2506 | 0.2616 | 0.2623 | **0.3064** | 16.81% |
| | AUC | 0.9284 | 0.9505 | 0.9606 | 0.9615 | 0.9619 | 0.9596 | 0.9481 | 0.9655 | **0.9677** | 0.24% |



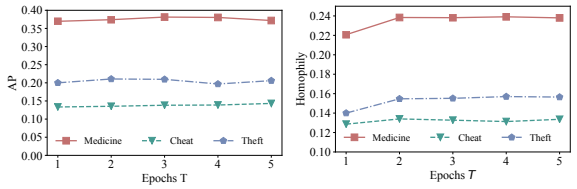

**(a) Medicine** **(b) Cheat** **(c) Thelf**

**Figure 4: Performance comparison of RGSL and its variants on the three risk scenarios.**




**(a) AP** **(b) Max-F1**

**Figure 5: Adjusting the value of threshold $\epsilon$.**




**(a) AP** **(b) Homophily of $\mathcal{G}_A^t$**

**Figure 6: Adjusting the number of epochs $T$.**

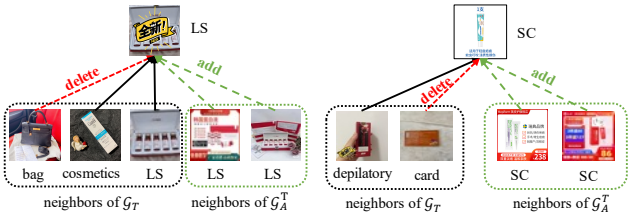
Table 3 presents the overall AP, Max-F1 and AUC results of different methods, where the following two main observations can be made.

First, RGSL achieves the best performance on the three datasets for prohibited item detection. Compared to the second best one, the improvement is prominent from 20.20% up to 21.91% in AP, from 0.24% up to 0.50% in AUC and from 8.59% up to 18.28% in Max-F1. This phenomenon is reasonable. Compared to homogeneous GNNs and heterogeneous HAN and HGT, RGSL is able to optimize graph structures for better message passing. Compared to GSL-based GAUG and GRCN, RGSL is advantaged in pairwise training to overcome weak supervised information and preserving semantics. Due to the imbalanced labeling, AUC value is relatively large than other metrics and the improvement is not very obvious.

Second, GSL-based baselines and our RGSL can perform better than others in lower-homophily datasets *Cheat* and *Theft*. The homogeneous GAUG and GRCN can even perform better than HAN, indicating the necessary to employ structure learning when dealing with complex noisy graphs.

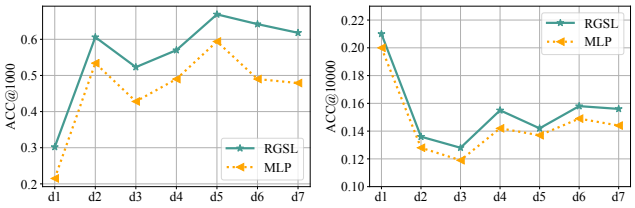
## 5.4 Variant Analysis

RGSL is to fully exploit attributes and topologies to learn graph structures for effective neighborhood aggregation and integrate with iterative pairwise labeling to mutually enrich supervised information and enhance structure learning. In this section, we analyze three kinds of RGSL variants to showcase the advantages of our design choices. (1) **RGSL**$_{NW}$ replaces iterative pairwise training with node-wise classification. (2) **RGSL**$_{PW,A}$ and **RGSL**$_{PW,T}$ respectively only consider attributed or topological graphs. (3) **RGSL**$_{PW}$ removes the iterative framework against **RGSL**. The results of RGSL




**(a) Lipolytic Solution (LS)** **(b) Scar Cream (SC)**

**Figure 7: Case study in *Medicine*. Notice that, the deleted edges are from $\mathcal{G}_T$ and the added edges are from iterative $\mathcal{G}_A^T$.**




**(a) ACC@1000** **(b) ACC@10000**

**Figure 8: Online testing of RGSL and MLP.**

and its variants in terms of AP and Max-F1 on the three datasets are shown in Figure 4.

Obviously, RGSL consistently outperforms all its variants. (1) Compared with $\text{RGSL}_{NW}$, the advantages of RGSL is in pairwise training to overcome the few size and poor expression of manual labels. (2) Compared with $\text{RGSL}_{PW,A}$ and $\text{RGSL}_{PW,T}$, the improvement of RGSL results from its ability to fully integrate with multiplex relations and attributes for structure learning. (3) Compared with $\text{RGSL}_{PW}$, our RGSL can iteratively add confident candidate connections which address the gap of label homophily and feature smoothness to a certain extent.

## 5.5 Parameter Analysis

There are two key parameters impacting the performance for prohibited item identification, namely the threshold $\epsilon$ to filter connections and the number of epochs $T$ for iterative pairwise training.

To evaluate the impact of our filtering mechanism, we vary the value of $\epsilon$ from 0 to 1 and report the AP and Max-F1 performance of RGSL on the three datasets in Figure 5. Notice that $\epsilon = 0$ and $\epsilon = 1$ are two extreme cases where the former removes none edges and the latter remove all connections. Obviously, RGSL achieves the best performance with $\epsilon = 0.1$ and too large value will leads to very poor performance. This phenomenon demonstrates that our heterogeneous structure learning has the ability to reduce noisy neighborhoods besides adding valuable unconnected similar pairs.

To evaluate the impact of iterative training framework, we adjust the number of epochs $T$ from 1 to 5, and report AP performance of RGSL and homophily of iterative attributed graph on all three datasets in Figure 6. There are two main observations. On the one hand, when $T$ initially increases, the performance of RGSL is continuously improved, until reaching a saturation at about $T = 3$, indicating the effectiveness of constructing enhanced attributed graph. On the other hand, the homophily of attributed graphs obviously increases by replacing attributes with label-wise embedding of nodes, which implies the promotion of optimized structures to better performance of tasks.

## 5.6 Case Study

In Figure 7, we showcase two representative cases in *Medicine* dataset including Lipolytic Solution (LS) and Scar Cream (SC) to visualize the advantages of our proposed RGSL. Notice that the second best baseline HGT cannot identify them correctly. We can easily find that the original neighbors from $\mathcal{G}_T$ contains several noisy information, such as the bag in Figure 7(a) and the card in Figure 7(b). Besides, there are many risk-similar items would be unconnected. Obviously, our RGSL has the ability to delete noisy neighbors in $\mathcal{G}_T$ and add similar similar items in $\mathcal{G}_A$ at the same time, making the message passing more effective.

## 5.7 Online Testing

We have deployed RGSL on real-world e-commerce platform for online prohibited medicine device detection. Since existing GSL-based methods suffer from heavy memory cost and high computational complexity, we compare RGSL with current industrial solution (i.e., MLP) via online testing. As the predicted candidates are manually reviewed, it is unaffordable to evaluate the performance of all candidates. For daily results, we rank the candidate items with $\hat{Y}$ and select top-10000 items for manually checking, and report the ACC@1000 and ACC@10000 performance in Figure 8.

There are two major observations. First, the long-term observations show that RGSL outperforms MLP in all 7 days and demonstrates the high industrial practicability of RGSL for detecting prohibited items. Second, compared to MLP, the average improvement of RGSL is up to 23.59% and 6.25% in ACC@1000 and ACC@10000. This phenomenon indicates that our RGSL can reduce obvious human resources, especially for top-1000 checking.

## 6 CONCLUSION

In this paper, we focus on prohibited item detection to guarantee the health of e-commerce. We are the first to introduce graph structure learning into risk scenarios to address this problem. To handle the essential challenges of weak structures and weak supervision, we design the novel RGSL consisting of heterogeneous structure learning and pairwise training mechanism, and optimize both structures and detection iteratively to keep the consistency of label homophily and feature smoothness. Extensive results on both offline and online experiments showcase the obvious advantages of our proposed model.

## ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China (No. U20B2045, 62192784, 62172052, 61772082, 62002029, U1936104) and and the Fundamental Research Funds for the Central Universities (No. 2021RC28).

# REFERENCES


[1] Hongyun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. 2018. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Transactions on Knowledge and Data Engineering* 30, 9 (2018), 1616–1637.

[2] Yukuo Cen, Xu Zou, Jianwei Zhang, Hongxia Yang, Jingren Zhou, and Jie Tang. 2019. Representation Learning for Attributed Multiplex Heterogeneous Network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*. 1358–1368.

[3] Jie Chen, Tengfei Ma, and Cao Xiao. 2018. FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

[4] Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2020. Iterative Deep Graph Learning for Graph Neural Networks: Better and Robust Node Embeddings. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

[5] Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. 2020. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988* (2020).

[6] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*. 2478–2486.

[7] Luca Franceschi, Mathias Niepert, Massimiliano Pontil, and Xiao He. 2019. Learning Discrete Structures for Graph Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA (Proceedings of Machine Learning Research, Vol. 97)*. 1972–1982.

[8] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189–1232.

[9] Chen Gao, Tzu-Heng Lin, Nian Li, Depeng Jin, and Yong Li. 2021. Cross-platform Item Recommendation for Online Social E-Commerce. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[10] Xiang Gao, Wei Hu, and Zongming Guo. 2020. Exploring structure-adaptive graph learning for robust semi-supervised classification. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.

[11] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. 1024–1034.

[12] Guoxiu He, Yangyang Kang, Zhe Gao, Zhuoren Jiang, Changlong Sun, Xiaozhong Liu, Wei Lu, Qiong Zhang, and Luo Si. 2019. Finding camouflaged needle in a haystack? pornographic products detection via berrypicking tree model. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 365–374.

[13] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous Graph Transformer. In *The World Wide Web Conference, WWW 2020, Taipei, Taiwan, April 20-24, 2020*. 2704–2710.

[14] Wenbing Huang, Tong Zhang, Yu Rong, and Junzhou Huang. 2018. Adaptive sampling towards fast graph representation learning. *arXiv preprint arXiv:1809.05343* (2018).

[15] Yugang Ji, Chuan Shi, and Xiao Wang. 2021. Prohibited Item Detection on Heterogeneous Risk Graphs. In *The 30th International Conference on Information and Knowledge Management, CIKM 2021, Virtual Event, Queensland, Australia, November 1 - 5, 2021*. 3867–3877.

[16] Yugang Ji, Mingyang Yin, Hongxia Yang, Jingren Zhou, Vincent W. Zheng, Chuan Shi, and Yuan Fang. 2021. Accelerating Large-Scale Heterogeneous Interaction Graph Embedding Learning via Importance Sampling. *ACM Transactions on Knowledge Discovery from Data* 15, 1 (2021), 10:1–10:23.

[17] Wei Jin, Yao Ma, Xiaorui Liu, Xianfeng Tang, Suhang Wang, and Jiliang Tang. 2020. Graph Structure Learning for Robust Graph Neural Networks. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*. 66–74.

[18] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

[19] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

[20] Ao Li, Zhou Qin, Runshi Liu, Yiqun Yang, and Dong Li. 2019. Spam review detection with graph convolutional networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2703–2711.

[21] Derek Lim, Xiuyu Li, Felix Hohne, and Ser-Nam Lim. 2021. New Benchmarks for Learning on Non-Homophilous Graphs. *arXiv preprint arXiv:2104.01404* (2021).

[22] Yuanfu Lu, Yuan Fang, and Chuan Shi. 2020. Meta-learning on heterogeneous information networks for cold-start recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1563–1573.

[23] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

[24] Hesham Mostafa, Marcel Nassar, and Somdeb Majumdar. 2021. On Local Aggregation in Heterophilic Graphs. *arXiv preprint arXiv:2106.03213* (2021).

[25] Fionn Murtagh. 1991. Multilayer perceptrons for classification and regression. *Neurocomputing* 2, 5-6 (1991), 183–197.

[26] Xiaoru Qu, Zhao Li, Jialin Wang, Zhipeng Zhang, Pengcheng Zou, Junxiao Jiang, Jiaming Huang, Rong Xiao, Ji Zhang, and Jun Gao. 2020. Category-aware Graph Neural Networks for Improving E-commerce Review Helpfulness Prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2693–2700.

[27] Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling Relational Data with Graph Convolutional Networks. In *5th International Conference, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings (Lecture Notes in Computer Science, Vol. 10843)*. 593–607.

[28] Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, and S Yu Philip. 2016. A survey of heterogeneous information network analysis. *IEEE Transactions on Knowledge and Data Engineering* 29, 1 (2016), 17–37.

[29] Kaisong Song, Yangyang Kang, Wei Gao, Zhe Gao, Changlong Sun, and Xiaozhong Liu. 2021. Evidence Aware Neural Pornographic Text Identification for Child Protection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, Virtual Event, February 2-9, 2021*. 14939–14947.

[30] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph Attention Networks. *CoRR* abs/1710.10903 (2017).

[31] Ruijia Wang, Shuai Mou, Xiao Wang, Wanpeng Xiao, Qi Ju, Chuan Shi, and Xing Xie. 2021. Graph Structure Estimation Neural Networks. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. 342–353.

[32] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S. Yu. 2019. Heterogeneous Graph Attention Network. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*. 2022–2032.

[33] Xiao Wang, Meiqi Zhu, Deyu Bo, Peng Cui, Chuan Shi, and Jian Pei. 2020. Am-gcn: Adaptive multi-channel graph convolutional networks. In *Proceedings of the 26th ACM SIGKDD International conference on knowledge discovery & data mining*. 1243–1253.

[34] Haiqin Weng, Shouling Ji, Fuzheng Duan, Zhao Li, Jianhai Chen, Qinming He, and Ting Wang. 2019. Cats: cross-platform e-commerce fraud detection. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 1874–1885.

[35] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. 2021. A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems* 32, 1 (2021), 4–24.

[36] Hong Xuan, Abby Stylianou, and Robert Pless. 2020. Improved Embeddings with Easy Positive Triplet Mining. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020*. 2463–2471.

[37] Donghan Yu, Ruohong Zhang, Zhengbao Jiang, Yuexin Wu, and Yiming Yang. 2020. Graph-Revised Convolutional Network. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2020, Ghent, Belgium, September 14-18, 2020 (Lecture Notes in Computer Science, Vol. 12459)*. 378–393.

[38] Wentao Zhang, Yuan Fang, Zemin Liu, Min Wu, and Xinming Zhang. 2020. mg2vec: Learning Relationship-Preserving Heterogeneous Graph Representations via Metagraph Embedding. *IEEE Transactions on Knowledge and Data Engineering* (2020), 1–1.

[39] Hongke Zhao, Qi Liu, Hengshu Zhu, Yong Ge, Enhong Chen, Yan Zhu, and Junping Du. 2018. A Sequential Approach to Market State Modeling and Analysis in Online P2P Lending. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 48, 1 (2018), 21–33.

[40] Jianan Zhao, Xiao Wang, Chuan Shi, Binbin Hu, Guojie Song, and Yanfang Ye. 2021. Heterogeneous Graph Structure Learning for Graph Neural Networks. In *35th AAAI Conference on Artificial Intelligence (AAAI)*.

[41] Tong Zhao, Yozen Liu, Leonardo Neves, Oliver Woodford, Meng Jiang, and Neil Shah. 2021. Data Augmentation for Graph Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11015–11023.

[42] Vincent W. Zheng, Mo Sha, Yuchen Li, Hongxia Yang, Yuan Fang, Zhenjie Zhang, Kian-Lee Tan, and Kevin Chen-Chuan Chang. 2018. Heterogeneous Embedding Propagation for Large-Scale E-Commerce User Alignment. In *IEEE International Conference on Data Mining, Singapore, November 17-20, 2018*. 1434–1439.

[43] Jiong Zhu, Ryan A Rossi, Anup Rao, Tung Mai, Nedim Lipka, Nesreen K Ahmed, and Danai Koutra. 2021. Graph Neural Networks with Heterophily. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11168–11176.

[44] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. 2020. Beyond Homophily in Graph Neural Networks: Current Limitations and Effective Designs. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

[45] Rong Zhu, Kun Zhao, Hongxia Yang, Wei Lin, Chang Zhou, Baole Ai, Yong Li, and Jingren Zhou. 2019. AliGraph: A Comprehensive Graph Neural Network Platform. *Proc. VLDB Endow.* 12, 12 (2019), 2094–2105.

[46] Yanqiao Zhu, Weizhi Xu, Jinghao Zhang, Qiang Liu, Shu Wu, and Liang Wang. 2021. Deep Graph Structure Learning for Robust Representations: A Survey. *arXiv preprint arXiv:2103.03036* (2021).