

# Learning to Explore: Policy-Guided Outlier Synthesis for Graph Out-of-Distribution Detection

Li Sun<sup>1,2\*</sup>, Lanxu Yang<sup>1</sup>, Jiayu Tian<sup>2</sup>, Bowen Fang<sup>3</sup>, Xiaoyan Yu<sup>4</sup>,  
Junda Ye<sup>2</sup>, Peng Tang<sup>5</sup>, Hao Peng<sup>6</sup>, Philip S. Yu<sup>7</sup>

<sup>1</sup>North China Electric Power University

<sup>2</sup>Beijing University of Posts and Telecommunications

<sup>3</sup>Tsinghua University

<sup>4</sup>Beijing Institute of Technology

<sup>5</sup>Shandong University

<sup>6</sup>Beihang University

<sup>7</sup>University of Illinois Chicago

{ccesunli, yanglanxu24}@ncepu.edu.cn, jytian04@outlook.com, psyu@uic.edu

## Abstract

Detecting Out-of-Distribution (OOD) graphs—those are drawn from a different distribution from the training data—is a critical task for ensuring the safety and reliability of Graph Neural Networks. The main challenge in unsupervised graph-level Out-of-Distribution detection lies in its common reliance on purely in-distribution (ID) data. This ID-only training paradigm leads to an incomplete characterization of the feature space, resulting in decision boundaries that lack the robustness needed to effectively separate ID from OOD samples. While incorporating synthesized outliers into the training process is a promising direction, existing generation methods are limited by their dependence on pre-defined, non-adaptive sampling heuristics (e.g., distance- or density-based). Such fixed strategies lack the flexibility to systematically explore the most informative OOD regions for refining decision boundaries. To overcome this limitation, we propose a novel Policy-Guided Outlier Synthesis (PGOS) framework that replaces static heuristics with a learned, adaptive exploration policy. PGOS trains a reinforcement learning agent to autonomously navigate low-density regions within a structured latent space, sampling representations that are maximally effective for regularizing the OOD decision boundary. These sampled points are then decoded into high-quality pseudo-OOD graphs to enhance the detector’s robustness. Extensive experiments demonstrate the strong performance of our method, state-of-the-art results on multiple graph OOD and anomaly detection benchmarks.

## Introduction

Graph Neural Networks (GNNs) have become a cornerstone for graph-level classification, achieving state-of-the-art results in critical domains like molecular science (Lee et al. 2025) and social network analysis (Sun et al. 2025b; Li et al. 2025; Sun et al. 2023c,a; Song et al. 2025). However, this success relies on the standard assumption that test data is drawn from the same distribution as the training data (Kipf and Welling 2017; Hamilton, Ying, and Leskovec

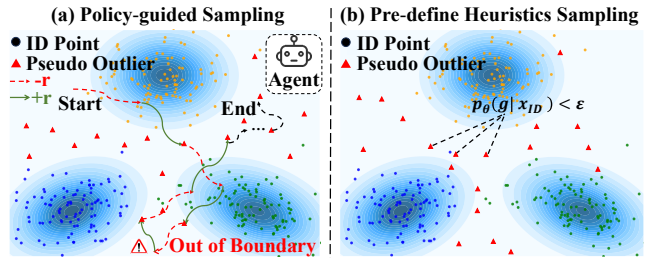


Figure 1: Comparisons between policy-guided and heuristic-based sampling. (a) Policy-guided sampling via a learned agent. (b) Heuristic sampling follows predefined rules.

2017; Veličković et al. 2018; Zhuo et al. 2025b; Yang et al. 2025; Zhuo et al. 2025a). In real-world applications, this closed-world assumption is often violated, as deployed models inevitably encounter inputs from novel or shifted distributions—commonly known as Out-of-Distribution (OOD) samples. When faced with these OOD graphs, a standard GNN can fail silently, producing erroneous predictions with high confidence. This vulnerability poses a significant risk to model reliability and safety, making the detection of OOD inputs a critical challenge for building trustworthy graph learning systems (Sun et al. 2023b, 2025a).

Existing research in unsupervised graph OOD detection has largely focused on modeling the in-distribution (ID) data manifold to indirectly identify outliers. Common strategies involve learning stable ID patterns through various means, such as contrastive learning (Liu et al. 2023a; Hou et al. 2025), generative modeling (Shen et al. 2024), or post-hoc manipulations (Wang et al. 2024; Guo et al. 2023). A unified characteristic of these methods is their exclusive reliance on in-distribution data for training. When the data distribution is complex, relying solely on ID data may fail to reveal essential cues for OOD detection.

To overcome the limitations of ID-only training, an alternative paradigm is Outlier Exposure (OE) (Hendrycks, Mazeika, and Dietterich 2019) or Outlier Synthesis, which

\*Corresponding Author: Li Sun.

has been extensively validated in computer vision (Du et al. 2022; Tao et al. 2023; Zheng et al. 2023). The core idea is to generate artificial outliers in the latent space to explicitly regularize the decision boundary. A common approach is to sample from low-likelihood regions of the in-distribution data, which can be modeled parametrically with class-conditional Gaussians (Du et al. 2022) or non-parametrically using k-NN density estimators (Tao et al. 2023). Other advanced strategies employ Hamiltonian Monte Carlo for more diverse exploration of the latent space (Li and Zhang 2025) or leverage powerful diffusion models to decode synthesized latent features into photo-realistic outlier images (Du et al. 2023). However, the application of these powerful techniques to graph-structured data remains nascent, with HGOE representing a preliminary example that uses graph-based interpolation (Junwei et al. 2024). A critical observation is that this entire diverse family of synthesis methods shares a fundamental limitation: they all rely on pre-defined heuristics—such as distance or density—to identify target regions for sampling. These fixed strategies lack the adaptability to systematically explore the most informative OOD regions for refining the decision boundaries. This raises a critical question:

*How can we move beyond fixed heuristics to systematically and adaptively discover the most informative outlier locations in the latent space?*

Tackling this question is particularly challenging in the unsupervised graph domain due to two fundamental issues. First, the latent space shaped by conventional contrastive learning is ill-suited for targeted exploration. While these methods can group similar graphs, they fail to establish explicit semantic prototypes for the clusters. This absence of learnable prototypes renders the low-density regions between clusters truly unstructured and difficult to navigate. Second, designing an adaptive exploration policy is itself a non-trivial task. To be effective, the policy must be guided by a carefully crafted reward mechanism that goes beyond simple metrics like distance or density. The core challenge lies in precisely defining what constitutes an informative outlier and translating this abstract notion into a concrete learning signal. Therefore, a principled framework should first impose a meaningful structure on the latent space and then learn a systematic policy for exploration.

To address this, we propose a novel framework Policy-Guided Outlier Synthesis (PGOS) that first imposes a meaningful structure on the latent space. Specifically, we employ prototypical contrastive learning (Li et al. 2021; Lin et al. 2022) to establish the explicit learnable prototypes that conventional methods lack. These prototypes act as semantic anchors transforming the previously unstructured regions into a navigable space defined by well-separated clusters of ID data. Building upon this structured latent space, PGOS reframes outlier synthesis as a targeted exploration problem and tackles the policy design challenge by introducing a highly specialized RL agent. This agent’s behavior is governed by three key principles: (1) a tailored reward function that compels exploration into the low-density regions between prototypes; (2) a hard boundary constraint that ensures exploration remains relevant to the data manifold; (3)

a novel spatially-aware entropy regularization method that dynamically encourages maximal exploration near the cluster boundaries. This principled guidance system enables the agent to learn an optimal policy for discovering latent representations that are maximally effective for regularizing the OOD decision boundary. These representations are then decoded into high-quality pseudo-OOD graphs to enhance robust model training. Our main contributions are as follows:

- We reconsider outlier synthesis for graph OOD detection by adaptively exploring the ID-OOD boundary with a learnable policy beyond static heuristics.
- We design a policy-guided agent whose novel exploration strategy integrates a tailored reward, boundary constraints, and adaptive entropy regularization to efficiently discover informative pseudo-outliers.
- Extensive experiments on 25 benchmarks validate our method’s superiority, where we establish new state-of-the-art performance on 12 of these datasets.

## Related Work

**Graph-level Out-of-Distribution Detection.** Unsupervised graph-level Out-of-Distribution detection aims to identify graphs that deviate from the training distribution. The current dominant paradigm is contrastive learning methods like GOOD-D (Liu et al. 2023a), which captures hierarchical semantics, and SEGO (Hou et al. 2025), which leverages structural entropy (Li and Pan 2016). Other approaches exploit intrinsic graph properties like substructures or spectral anomalies (Gu, Qiao, and Li 2025; Ding et al. 2024), rely on the reconstruction quality of generative models (Shen et al. 2024), or explicitly model the class-conditioned distributions (Zhang et al. 2025). Additionally, post-hoc methods such as AAGOD (Guo et al. 2023) enhance pre-trained detectors, while test-time strategies like GOODAT (Wang et al. 2024) perform detection without accessing training data. Orthogonal to these methods, other work provides statistical guarantees via conformal prediction (Lin et al. 2025). Recently, inspired by Outlier Exposure in computer vision, HGOE (Junwei et al. 2024) incorporates external real-world and internal synthetic outliers. However, its synthesis relies on a pre-defined interpolation heuristic. In contrast, our framework employs an adaptive policy to actively discover informative outliers beyond such fixed generation rules.

**Outlier Synthesis for OOD Detection.** Outlier synthesis improves OOD detection by generating pseudo-samples that mimic distributional shifts. Representative methods like VOS (Du et al. 2022) and HamOS (Li and Zhang 2025) create boundary-aware outliers in feature space, while diffusion-based (Du et al. 2023), flow-based (Kumar et al. 2023), and non-parametric (Tao et al. 2023) approaches enable fine-grained control over sample diversity and location. Recent advances explore diversity-aware sampling (Jiang et al. 2024; Zhu et al. 2023), noisy outlier robustness (Zheng et al. 2023), and structural awareness via energy-based memory (Hofmann et al. 2024). Yet, most methods adopt predefined generation objectives, lacking adaptive exploration of the data manifold. This hinders their ability to generate semantically aligned and decision-relevant outliers.

## Preliminaries and Problem Definition

An attributed graph is denoted as  $G = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} = \{v_1, \dots, v_n\}$  is the set of  $n = |\mathcal{V}|$  nodes, and  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  is the set of edges. The topological structure of the graph can be described by an adjacency matrix  $A \in \{0, 1\}^{n \times n}$ , where  $A_{ij} = 1$  if there exists an edge between nodes  $v_i$  and  $v_j$ , and  $A_{ij} = 0$  otherwise. Each node  $v_i \in \mathcal{V}$  is associated with a  $d$ -dimensional feature vector. The features of all nodes are collectively represented by a node feature matrix  $X \in \mathbb{R}^{n \times d}$ . Consequently, a graph can be concisely represented as the tuple  $G = (A, X)$ .

**Unsupervised Graph-level OOD Detection.** In this paper, we focus on the task of unsupervised graph-level out-of-distribution detection. In this setting, we are given a training set of unlabeled graphs,  $\mathcal{D}_{\text{train}}^{\text{in}} = \{G_1, G_2, \dots, G_n\}$ , which are assumed to be sampled from an in-distribution, denoted as  $\mathbb{P}_{\text{in}}$ . The goal is to learn a scoring function,  $f(\cdot)$ , using only this ID training set. During the testing phase, this scoring function is used to assign an OOD score,  $s = f(G)$ , to any given graph  $G$ . A higher score indicates a greater likelihood that the graph originates from an unknown out-of-distribution,  $\mathbb{P}_{\text{out}}$ , where  $\mathbb{P}_{\text{in}} \neq \mathbb{P}_{\text{out}}$ . The model’s performance is evaluated on a test set composed of unseen ID samples  $\mathcal{D}_{\text{test}}^{\text{in}}$  and OOD samples  $\mathcal{D}_{\text{test}}^{\text{out}}$ , where  $\mathcal{D}_{\text{train}}^{\text{in}} \cap \mathcal{D}_{\text{test}}^{\text{in}} = \emptyset$ .

**Reinforcement Learning.** Reinforcement Learning (Sutton, Barto et al. 1998) provides a mathematical framework for an agent to learn an optimal policy  $\pi$  by interacting with an environment, which is formally modeled as a Markov Decision Process (MDP). An MDP is defined by a state space  $\mathcal{S}$ , an action space  $\mathcal{A}$ , a transition function  $P(s'|s, a)$ , a reward function  $R(s, a, s')$ , and a discount factor  $\gamma \in [0, 1)$ .

The agent’s goal is to learn a policy  $\pi(a|s)$  that maximizes the expected discounted sum of future rewards. The value of taking an action  $a$  in a state  $s$  is given by the action-value function (Q-function):

$$Q^\pi(s, a) = \mathbb{E}_\pi \left[ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a \right] \quad (1)$$

The objective of RL is to find the optimal Q-function  $Q^*(s, a) = \max_\pi Q^\pi(s, a)$ , which represents the maximum possible return. Once  $Q^*(s, a)$  is determined, the optimal policy  $\pi^*$  can be extracted by acting greedily at each state:

$$\pi^*(s) = \arg \max_{a \in \mathcal{A}} Q^*(s, a) \quad (2)$$

## Methodology

As illustrated in Figure 2, we propose a pseudo-outlier synthesis framework for unsupervised graph OOD detection. We first train a graph autoencoder with prototypical contrastive learning to cluster semantically similar graphs around distinct prototypes. Then, a reinforcement learning-based sampler explores low-density regions in the latent space—far from any prototype—to generate informative latent vectors, which are decoded into high-quality pseudo-outlier graphs. Finally, we train the OOD detection model on both the synthesized outliers and the original ID graphs to distinguish between the two distributions. Technical details are provided in the following sections.

## Prototypical Representation Learning for Graphs

To effectively synthesize outliers, our strategy first requires a structured latent space that accurately models the in-distribution data. To this end, we design a prototypical representation learning module to produce a latent space where ID graphs form compact, well-separated clusters. This process yields two critical components: (1) a graph encoder  $f_\theta$  that maps graphs into this structured space, consisting of a multi-layer GCN to learn node representations and a pooling layer to aggregate them into a graph-level embedding, and (2) a graph decoder  $g_\psi$  that generates graph data from a given latent vector. The overall architecture for this module consists of the GCN-MLP encoder-decoder pair and a set of  $K$  trainable prototypes  $\mathcal{C} = \{c_k\}_{k=1}^K$ , where the number of prototypes  $K$  is a pre-defined hyperparameter. We train the model by jointly optimizing a prototypical contrastive objective and a generative reconstruction loss.

**Prototypical Contrastive Objective.** This objective  $\mathcal{L}_{\text{PCO}}$  structures the latent space by pulling graph embeddings towards semantic prototypes and separating the prototypes themselves. By contrasting two augmented views of each graph  $G_i$  with embeddings  $z'_i$  and  $z''_i$ , this creates the low-density regions required for our sampling strategy. The objective is composed of three loss terms: debiased contrastive loss, prototypical consistency loss, and inter-prototype separation loss.

**Debiased Contrastive Loss  $\mathcal{L}_{\text{DC}}$**  mitigates sampling bias of conventional contrastive learning by using prototype information to identify and exclude potential false negatives.

$$\mathcal{L}_{\text{DC}} = - \sum_{i=1}^N \log \frac{\text{sim}(z'_i, z''_i)}{\text{sim}(z'_i, z''_i) + \sum_{z_j \in \mathcal{N}_i} \text{sim}(z'_i, z_j)} \quad (3)$$

where we define the exponentiated similarity score as  $\text{sim}(u, v) = \exp(u^T v / \tau)$ ,  $\tau$  is the temperature, the set of debiased negative samples for an embedding  $z'_i$  is  $\mathcal{N}_i = \{z_j \mid j \neq i, c(z_j) \neq c(z'_i)\}$ , herein  $c(z)$  denotes the nearest prototype for embedding  $z$ ,

**Prototypical Consistency Loss  $\mathcal{L}_{\text{PC}}$**  ensures clustering consistency between different augmented views of the same graph. It is defined as:

$$\mathcal{L}_{\text{PC}} = \frac{1}{2N} \sum_{i=1}^N [l(z'_i, z''_i) + l(z''_i, z'_i)] \quad (4)$$

where  $N$  is the batch size and the loss term  $l(z_a, z_b)$  is calculated by  $l(z_a, z_b) = - \sum_{k=1}^K p_k(z_b) \log p_k(z_a)$ . Here,  $p_k(z) = \frac{\exp(z^T c_k / \tau)}{\sum_{j=1}^K \exp(z^T c_j / \tau)}$  is the predicted assignment probability of embedding  $z$  to the  $k^{\text{th}}$  prototype.

**Inter-Prototype Separation Loss  $\mathcal{L}_{\text{IPS}}$**  is designed to push the prototypes away from each other. Its objective is to maximize the average squared Euclidean distance between all pairs of prototypes, thereby forcing them apart and creating well-defined, separated clusters.

$$\mathcal{L}_{\text{IPS}} = - \frac{1}{K(K-1)} \sum_{i=1}^K \sum_{j=1, j \neq i}^K \|c_i - c_j\|_2^2 \quad (5)$$

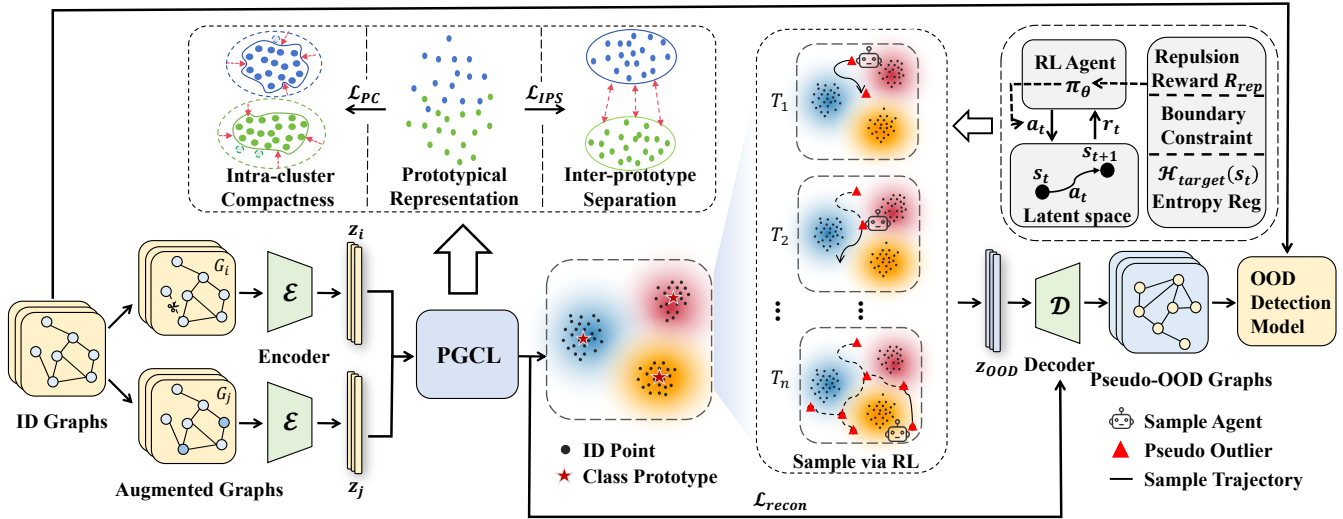


Figure 2: An overview of our PGOS framework based on pseudo-outlier synthesis.

These three components are combined into the prototypical contrastive objective as  $\mathcal{L}_{PCO} = \mathcal{L}_{DC} + \mathcal{L}_{PC} + \mathcal{L}_{IPS}$ .

**Generative Reconstruction Loss.** To ensure the embeddings are informative and to train the decoder  $g_{\psi}$ , we employ a generative reconstruction loss  $\mathcal{L}_{recon}$ . The decoder aims to reconstruct both the graph’s adjacency matrix  $\hat{A}$  and its node features  $\hat{X}$  from the latent node embeddings. The structure  $\hat{A}$  is decoded via an inner product of embeddings, while an MLP decoder decodes the features  $\hat{X}$ . The reconstruction loss is defined as:

$$\mathcal{L}_{recon} = \sum_{i=1}^N \left( \|X_i - \hat{X}_i\|_2^2 + \lambda \cdot \mathcal{L}_{struct}(A_i, \hat{A}_i) \right) \quad (6)$$

where  $\lambda$  is a balanced hyperparameter and  $\mathcal{L}_{struct}$  is the element-wise Cross-Entropy loss for the adjacency matrices.

The final training objective consists of the prototypical contrastive objective and generative reconstruction losses:

$$\mathcal{L}_{total} = \mathcal{L}_{PCO} + \gamma \mathcal{L}_{recon} \quad (7)$$

where  $\gamma$  is a balancing hyperparameter. Minimizing the total objective shapes a structured latent space suitable for sampling and a decoder for generating pseudo-OOD graphs.

### Policy-Guided Outlier Synthesis

Building upon a structured embedding space, we propose a reinforcement learning-based method that replaces previous non-adaptive sampling heuristics with an adaptive policy for exploratory outlier synthesis. Our approach introduces a novel guidance system for the RL agent, featuring a repulsion reward, hard boundary constraints, and dynamic spatial entropy regularization.

**MDP Formulation for Outlier Synthesis.** We model the latent space  $\mathcal{Z} \subset \mathbb{R}^D$  as an environment for the agent. The

agent’s goal is to learn an optimal policy  $\pi$  for navigation and exploration. The agent’s state  $s_t$  is its current coordinate within the latent space. At each timestep, the agent takes an action  $a_t$  which is a continuous displacement vector of the same dimension as the state. The subsequent state is determined by the transition  $s_{t+1} = s_t + a_t$ . The immediate reward  $r_t$  is determined by the next state  $s_{t+1}$  and is denoted by  $r_t = R(s_{t+1})$ . A high-quality OOD state should satisfy two key conditions: it must lie in the regions between the in-distribution clusters, and its position should not deviate excessively from the global boundary of these clusters.

**Reward Function Design.** To guide the agent’s exploration, our reward function is designed with a single and clear objective: to compel the agent to move into the low-density voids between in-distribution clusters. To achieve this, we introduce a repulsion reward  $R_{rep}$  which penalizes the agent for entering the dense regions of the ID clusters. For each cluster  $C_i$ , we define its centroid  $\mu_i$  and effective radius  $r_i$  as  $\mu_i = \frac{1}{|C_i|} \sum_{s \in C_i} s$  and  $r_i = \max_{s \in C_i} \|s - \mu_i\|_2$ . We apply a strong penalty when the agent’s distance  $d_i = \|s - \mu_i\|_2$  falls below a safety margin  $\delta_i$ , which is defined as a multiple of the cluster’s specific radius  $r_i$ . As a result, the agent learns to focus its exploration on the spaces between clusters rather than the intra-cluster areas. The reward is formally expressed as:

$$R_{rep}(s) = \sum_{i=1}^K \begin{cases} - \left(1 - \frac{d_i - r_i}{\delta_i}\right)^2 & \text{if } d_i < r_i + \delta_i \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

**Boundary Constraint.** To ensure the agent’s exploration remains focused on areas relevant to the in-distribution data, we confine the state space  $\mathcal{S}$  within a global boundary. This boundary is defined as a hypersphere centered at the global centroid of all ID embeddings,  $\mu_g$ , with a radius  $R_{max}$ . These parameters are pre-calculated from the training set  $\mathcal{D}_{train}^{in}$  as

$\mu_g = \frac{1}{|\mathcal{D}_{\text{train}}^{\text{in}}|} \sum_{s \in \mathcal{D}_{\text{train}}^{\text{in}}} s$  and  $R_{\text{max}} = \max_{s \in \mathcal{D}_{\text{train}}^{\text{in}}} \|s - \mu_g\|_2$ . Instead of using a reward penalty that can be inefficient for the agent to learn, we enforce a hard boundary constraint directly on the state transition dynamics. If an action  $a_t$  causes the next state  $s_{t+1} = s_t + a_t$  to lie outside this boundary, the agent’s position is deterministically projected back onto the surface of the hypersphere.

$$s_{t+1} = \mu_g + R_{\text{max}} \frac{s_{t+1} - \mu_g}{\|s_{t+1} - \mu_g\|_2} \quad (9)$$

This approach is more efficient and simplifies the agent’s learning task.

**Spatially-Aware Entropy Regularization.** To better explore the informative boundary regions of ID clusters, we innovate on the entropy regularization of Soft Actor-Critic (SAC) (Haarnoja et al. 2018). Instead of using a manually designed entropy coefficient, we leverage SAC’s automatic temperature tuning capability, guiding it with a dynamic target entropy. The core idea is to encourage maximal exploration near the boundaries of ID clusters. We first compute a characteristic length scale from data: the average radius of all ID clusters,  $\bar{r} = \frac{1}{K} \sum_{i=1}^K r_i$ . The target entropy  $\mathcal{H}_{\text{target}}(s_t)$  for the agent’s policy is then dynamically set based on the agent’s distance to the nearest cluster,  $d_{\text{min}}(s_t)$ , peaking when the agent is at the average boundary distance:

$$\mathcal{H}_{\text{target}}(s_t) = \mathcal{H}_{\text{max}} \cdot \exp\left(-\frac{(d_{\text{min}}(s_t) - \bar{r})^2}{2\bar{r}^2}\right) \quad (10)$$

where  $\mathcal{H}_{\text{max}}$  is the maximum target entropy, often set to a default value based on the action space dimension. This self-tuning mechanism automatically focuses the agent’s exploration on the most informative boundary regions.

**Learning the Exploration Policy.** We employ the SAC algorithm to learn the exploration policy  $\pi_\phi$ . The architecture follows a standard actor-critic framework composed of a stochastic actor network  $\pi_\phi$  to select actions and a pair of critic networks ( $Q_{\theta_1}, Q_{\theta_2}$ ) to evaluate them. Both networks are trained concurrently using mini-batches of transitions sampled from a replay buffer. The actor’s objective is to maximize both the expected return and the policy’s entropy, where the entropy term encourages exploration. This is achieved by minimizing the following loss function:

$$\mathcal{L}_\pi(\phi) = \mathbb{E}_{a_t \sim \pi_\phi} [\alpha \log \pi_\phi(a_t | s_t) - Q_{\text{min}}(s_t, a_t)] \quad (11)$$

where  $a_t$  is an action sampled from the policy  $\pi_\phi(\cdot | s_t)$ ,  $Q_{\text{min}}(s_t, a_t)$  is the minimum estimated action-value from the pair of critic networks, i.e.,  $Q_{\text{min}}(s_t, a_t) = \min_{i=1,2} Q_{\theta_i}(s_t, a_t)$  and  $\alpha$  is the entropy temperature parameter. This parameter is automatically tuned towards a target  $\mathcal{H}_{\text{target}}(s_t)$  by minimizing its own objective  $J(\alpha)$ . With a learning rate  $\lambda_\alpha$ , the update rule for  $\alpha$  is:

$$\alpha \leftarrow \alpha - \lambda_\alpha \mathbb{E}_{a_t \sim \pi_\phi} [-\log \pi_\phi(a_t | s_t) - \mathcal{H}_{\text{target}}(s_t)] \quad (12)$$

The critic networks are trained concurrently by minimizing the standard soft Bellman error.

Once the policy converges, we generate pseudo-OOD samples by executing the learned policy over multiple episodes to collect a set of latent representations. Each episode starts from the midpoint of two randomly selected prototype centroids. The pre-trained decoder then synthesizes these representations into final pseudo-OOD graphs until their total number matches that of the ID training set.

## Outlier-Regularized OOD Detection

We instantiate our OOD detection model using GOOD-D (Liu et al. 2023a) and integrate the pseudo-outlier into the training process following HGOE (Junwei et al. 2024). Specifically, we jointly optimize a standard OOD detection loss on in-distribution data and a boundary-aware regularization term on pseudo-outlier samples. The overall training objective is formulated as:

$$\mathcal{L} = \sum_{G \in \mathcal{D}_{\text{train}}^{\text{in}}} \left[ \mathcal{L}_{\text{ID}}(h(G)) + \beta \sum_{G' \in \mathcal{D}_{\text{OOD}}} \mathcal{L}_{\text{reg}}(s_{G'}) \right] \quad (13)$$

where  $\mathcal{L}_{\text{ID}}$  denotes the training loss from GOOD-D,  $\mathcal{D}_{\text{OOD}}$  is pseudo-OOD graphs sampling by our method,  $s_{G'} = \text{sigmoid}(h(G'))$  is the OOD confidence score of a pseudo-outlier  $G'$ , and  $\mathcal{L}_{\text{reg}}$  is the boundary-aware loss that penalizes uninformative outliers near or inside the in-distribution region and  $\beta$  controls the regularization strength.

## Experiment

### Experimental Setup

**Datasets.** To ensure fair and consistent evaluation, we adopt the benchmark protocol introduced by (Liu et al. 2023a), which covers tasks of graph OOD detection and graph anomaly detection. Besides, 15 datasets from the Tox21 challenge and the TU benchmark (Morris et al. 2020) are used for graph anomaly detection. Samples belonging to the minority class or the true anomalous class are treated as anomalies, whereas all others are considered normal. For all datasets, we adopt the same splitting strategy as defined in the UB-GOLD benchmark (Wang et al. 2025).

**Baselines.** We compare the baselines across four main categories, including 4 graph kernel with detector methods (Li et al. 2016; Neumann et al. 2016; Liu, Ting, and Zhou 2008; Amer, Goldstein, and Abdennadher 2013), 4 SSL with detector methods (Sun et al. 2020; You et al. 2020; Sehswag, Chiang, and Mittal 2021), 5 GNN-based graph-level anomaly detection methods like OCGTL (Qiu et al. 2022), SIGNET (Liu et al. 2023b), OCGIN (Zhao and Akoglu 2023), CVTGAD (Li et al. 2023), GLocalKD (Ma et al. 2023) and 2 GNN-based graph-level out-of-distribution detection methods (Liu et al. 2023a; Wang et al. 2024).

**Evaluation & Implementation.** We evaluate OOD detection performance using the AUC metric. We report the mean and standard deviation across 5 independent runs. Complete implementation details, hyperparameter settings, and additional experimental results are provided in the appendix.

ID dataset	BZR	PTC-MR	AIDS	ENZYMES	IMDB-M	Tox21	FreeSolv	BBBP	ClinTox	Esol	Avg.
OOD dataset	COX2	MUTAG	DHFR	PROTEIN	IMDB-B	SIDER	ToxCast	BACE	LIPO	MUV	Rank
PK-SVM	43.1±7.9	50.2±6.7	52.1±2.6	49.9±3.2	49.0±3.0	51.0±2.0	49.3±2.9	53.0±2.3	51.9±3.0	51.7±1.8	14.5
PK-iF	52.8±2.2	52.0±4.0	50.8±1.8	52.6±2.4	51.9±3.0	50.7±1.1	51.9±1.7	53.1±1.8	51.7±1.6	51.2±4.1	13.7
WL-SVM	50.2±5.2	52.4±7.8	51.4±3.3	52.9±3.2	52.9±2.6	50.7±1.9	50.4±3.8	52.9±2.0	50.8±3.7	52.0±2.2	13.7
WL-iF	50.0±3.0	52.8±1.9	50.8±0.9	51.4±3.0	52.6±3.5	51.3±0.6	53.0±3.2	50.6±0.7	51.4±2.5	51.6±1.3	14.0
IG-iF	62.7±8.6	50.2±6.4	91.0±1.3	57.3±2.4	60.7±2.0	57.6±0.9	59.6±2.1	55.7±2.1	46.8±2.5	55.8±4.2	11.9
IG-MD	85.8±6.3	52.6±8.2	70.8±11.3	53.2±4.2	80.9±0.5	58.6±1.2	59.1±5.7	72.3±4.3	46.9±4.7	75.6±2.5	9.9
GCL-iF	58.9±5.4	52.4±2.7	91.3±2.1	60.8±3.1	58.0±2.5	58.0±1.4	56.7±1.5	57.9±2.0	48.8±1.4	60.3±3.7	11.2
GCL-MD	84.8±5.4	70.9±2.0	92.0±1.0	51.9±4.4	81.2±1.6	60.7±1.3	59.7±4.3	74.3±1.4	53.3±2.6	76.8±2.7	8.5
OCGIN	83.1±0.1	74.0±0.6	95.9±0.0	63.5±0.1	80.8±0.0	68.1±0.2	68.3±0.2	77.2±0.0	62.2±0.0	87.3±0.1	5.7
OCGTL	80.9±0.2	75.3±0.6	<b>99.3±0.0</b>	67.6±0.0	67.4±0.7	60.9±0.0	64.6±0.0	76.3±0.0	54.8±1.1	84.9±0.0	6.3
SIGNET	86.9±0.3	<u>81.9±0.6</u>	<u>97.3±0.0</u>	62.1±0.1	<u>83.0±0.1</u>	65.6±0.1	<b>75.4±0.1</b>	<b>91.4±0.0</b>	<u>73.0±0.1</u>	87.2±0.0	3.4
CVTGAD	<b>96.9±0.0</b>	80.3±0.3	<u>99.0±0.0</u>	65.4±0.1	<u>82.5±0.0</u>	<u>68.1±0.1</u>	73.0±0.2	87.1±0.0	67.8±0.0	<b>92.7±0.0</b>	2.6
GLocalKD	80.7±0.2	75.0±0.6	94.0±0.0	58.9±0.1	81.4±0.1	56.7±0.1	70.7±0.3	68.0±0.4	56.7±0.1	88.0±0.3	7.0
GOOD-D	75.8±6.0	70.6±3.5	93.7±1.2	57.2±2.0	78.2±4.3	66.3±1.0	64.8±3.3	73.2±1.3	55.7±3.8	86.8±2.4	8.0
GOODAT	<u>96.5±0.0</u>	81.2±0.2	98.6±0.0	65.1±0.0	81.0±0.1	66.8±0.0	68.5±0.3	83.1±0.1	68.5±0.0	89.6±0.0	3.7
PGOS	94.8±1.1	<b>84.1±1.7</b>	96.8±2.3	<b>68.9±1.6</b>	<b>85.4±1.4</b>	<b>74.2±1.1</b>	74.9±0.3	<u>87.4±1.1</u>	<b>75.2±1.1</b>	<u>92.5±0.4</u>	<b>1.9</b>

Table 1: OOD detection performance in AUC (% , mean ± std), with the best and second-best results in **bold** and underlined.

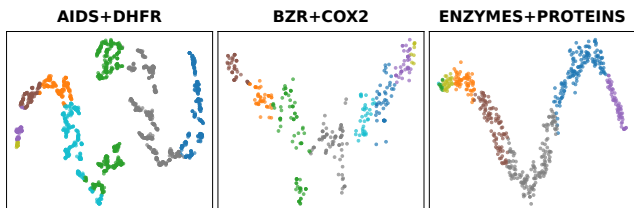


Figure 3: T-SNE visualizations of graph embeddings learned by PGCL on three different datasets. Each point denotes a graph, and colors represent the  $K = 8$  distinct clusters.

## Experimental Results and Analysis

To evaluate the effectiveness of our proposed method, we conduct comprehensive experiments including: (1) performance evaluation on OOD detection and anomaly detection tasks, (2) ablation study, (3) visualization analysis, and (4) sensitivity analysis of the number of prototypes.

**Performance Evaluation on OOD Detection.** As shown in Table 1, we compare our proposed PGOS method with 15 competitive baselines across 10 representative OOD detection benchmarks. PGOS achieves the best average rank of 1.9, demonstrating strong overall performance and generalization across diverse distributional shifts. The performance gains are particularly obvious on several datasets. For instance, PGOS surpasses the second-best method by 2.2% AUC on PTC-MR/MUTAG. On other benchmarks such as IMDB-M/IMDB-B and Tox21/SIDER, it also achieves significant improvements of 2.4% and 6.1% over the runner-up, respectively. Furthermore, even on challenging datasets like ENZYMES/PROTEIN where many methods struggle, PGOS achieves superior performance. These results consistently validate the effectiveness and robustness of our policy-guided outlier synthesis framework.

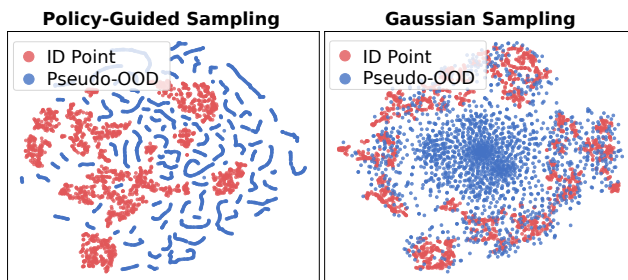


Figure 4: T-SNE visualization of sampled points generated by the RL-based and Gaussian sampling strategies on the BBBP+BACE OOD dataset.

**Performance Evaluation on Anomaly Detection.** To evaluate the effectiveness of our proposed method in the anomaly detection setting, we conduct comprehensive experiments on 15 graph-level anomaly detection benchmark datasets. As shown in Table 2, our method achieves new state-of-the-art performance on 7 out of 15 datasets. Even on datasets where PGOS is not the leading method, its performance remains highly competitive. Furthermore, PGOS demonstrates its superior performance most clearly on challenging datasets where other baselines struggle. For example, it achieves state-of-the-art performance on both HSE and COX2, outperforming the runner-up methods by significant margins of 5.9% and 4.0% AUC, respectively. These results confirm that our adaptive synthesis policy is a powerful and versatile solution for identifying both anomalous and out-of-distribution graphs.

**Ablation Study.** As presented in Table 3, we analyze the crucial role of each component within the PGOS framework. The results demonstrate that the policy-guided sampling module is the most critical element; its removal (PGOS-

Method	WL-SVM	IG-iF	GCL-iF	OCCIN	OCGTL	SIGNET	CVTGAD	GLocalKD	GOOD-D	GOODAT	PGOS
PROTEINS	52.9±3.8	55.8±2.0	59.1±2.5	76.3±0.0	74.2±0.0	72.6±0.1	74.4±0.0	71.4±0.5	72.4±0.0	77.9±2.4	<b>81.3±0.3</b>
ENZYMES	57.2±2.6	52.4±4.5	52.1±4.0	63.6±2.2	66.0±0.1	63.2±0.3	<b>69.4±0.7</b>	66.0±0.4	61.5±0.7	59.1±2.5	67.1±0.1
AIDS	49.2±3.4	72.8±2.4	78.0±2.7	<b>99.8±0.0</b>	<b>99.6±0.0</b>	92.2±0.3	98.6±0.0	97.3±0.0	95.9±0.1	96.0±1.0	96.9±0.1
DHFR	52.9±3.0	51.4±3.0	50.3±2.8	59.8±0.2	59.4±0.1	<u>72.7±0.6</u>	64.2±0.1	59.6±0.1	64.5±0.0	62.2±2.5	<b>73.2±0.2</b>
BZR	52.3±6.2	61.2±2.5	61.2±2.8	68.9±0.4	63.5±0.6	<u>79.3±1.3</u>	<b>80.0±0.5</b>	63.9±1.3	65.7±0.6	59.0±5.3	77.1±2.4
COX2	49.2±2.9	55.7±2.9	51.0±1.7	57.4±0.5	56.6±0.3	<u>70.6±0.2</u>	67.6±0.4	49.2±0.6	61.6±0.7	59.3±7.6	<b>74.6±2.3</b>
DD	46.2±1.1	57.8±2.8	53.8±2.0	<u>80.5±0.0</u>	80.4±0.0	74.1±0.1	74.2±0.0	78.3±1.4	78.5±0.0	78.2±2.2	<b>81.1±0.4</b>
NCI1	52.9±1.4	52.4±1.9	50.7±1.7	52.2±0.1	<b>79.0±0.0</b>	69.1±0.0	75.0±0.0	61.1±0.1	60.9±0.1	45.6±1.0	<u>76.1±0.1</u>
IMDB-B	52.7±1.8	64.9±3.1	54.9±3.5	61.2±0.2	63.1±0.0	<u>70.3±0.1</u>	<b>71.5±0.2</b>	50.6±0.2	67.6±0.2	65.5±4.3	67.7±2.4
REDDIT-B	47.4±2.9	67.5±3.0	75.0±1.6	<b>93.3±0.0</b>	<u>89.7±0.0</u>	85.8±0.1	87.7±0.0	79.5±0.1	88.5±0.0	80.3±0.8	86.4±0.7
COLLAB	62.0±1.8	45.0±2.8	49.3±2.8	62.2±0.1	51.4±0.0	<u>71.2±0.1</u>	65.2±0.0	48.0±0.0	63.4±0.0	45.3±0.9	<b>74.5±1.2</b>
HSE	64.9±1.1	51.5±3.0	50.9±1.8	72.3±0.1	59.2±0.0	67.2±0.0	67.0±0.3	58.8±0.0	<u>72.7±0.1</u>	63.4±1.1	<b>78.6±0.0</b>
MMP	58.3±2.0	56.9±1.8	54.3±1.6	69.0±0.0	60.8±0.5	<b>75.1±0.1</b>	69.9±0.0	60.8±0.4	71.6±0.0	69.4±0.4	<u>71.7±0.1</u>
p53	59.8±2.8	52.1±1.6	55.0±2.1	<u>69.3±0.1</u>	66.4±0.1	66.4±0.1	67.9±0.0	63.7±0.0	68.7±0.0	63.2±0.1	<b>69.8±0.1</b>
PPAR	58.9±1.8	53.4±2.9	51.8±1.8	<u>65.0±0.0</u>	64.7±0.0	<u>70.4±0.2</u>	66.1±0.0	65.7±0.0	<b>70.8±0.0</b>	66.8±2.5	68.0±0.1
Avg. Rank	9.6	9.8	10.2	4.5	5.6	4.1	<u>3.5</u>	7.4	4.2	6.7	<b>2.1</b>

Table 2: Anomaly detection performance in AUC (% , mean ± std), with the best and second-best results in bold and underlined.

Model Variants	BZR	PTC-MR	IMDB-M	Tox21
	COX2	MUTAG	IMDB-B	SIDER
PGOS- $\mathcal{L}_{IPS}$	93.3±1.1	82.2±2.2	83.7±1.4	72.9±1.7
PGOS-PGCL	93.4±2.7	80.9±3.7	81.6±1.3	71.7±2.4
PGOS- $\mathcal{H}_{target}(s_t)$	93.1±1.3	82.8±2.1	82.0±1.3	70.6±1.4
PGOS-RL	77.5±1.6	74.5±1.3	76.8±2.6	64.9±2.3
PGOS-Full	<b>94.8±1.1</b>	<b>84.1±1.7</b>	<b>85.4±1.4</b>	<b>74.2±1.1</b>

Table 3: Performance comparison of PGOS and its ablated variants, reported as AUC scores (% , mean ± std).

RL) leads to a significant performance degradation across all datasets, causing the AUC score to drop by an average of 11.2% and by as much as 17.3% on the BZR/COX2 dataset. This underscores the fundamental importance of an adaptive exploration policy. Furthermore, the other components also prove important for achieving optimal performance. Removing the inter-prototype separation loss (PGOS- $\mathcal{L}_{IPS}$ ) and the spatially-aware entropy regularization (PGOS- $\mathcal{H}_{target}(s_t)$ ) also results in noticeable performance drops, confirming their respective roles in structuring the latent space and efficiently guiding exploration. The superior performance of the full PGOS model over all ablated variants confirms that all components contribute synergistically to the framework’s overall effectiveness.

**Visualization.** To provide an intuitive understanding of the effectiveness of our PGCL module and policy-guided sampling strategy, we visualize both the learned embeddings and the sampled points on several representative datasets using T-SNE. As illustrated in Figure 3, PGCL produces compact and well-separated clusters, effectively structuring the latent space across diverse dataset distributions and facilitating pseudo-outlier synthesis. Figure 4 further shows that the policy-guided sampler generates pseudo-OOD samples clearly separated from ID clusters, while the Gaussian sampler adds isotropic noise around the midpoint of two ran-

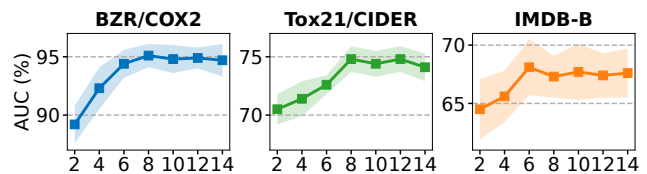


Figure 5: Sensitivity analysis of the number of prototypes  $K$  on OOD detection performance across different datasets.

domly selected cluster centers, yielding less distinguishable samples.

**Sensitivity Analysis of the Number of Prototypes.** Figure 5 illustrates the sensitivity of prototype count  $K$  on 2 OOD dataset pairs and 1 anomaly detection dataset. A small  $K$  performs poorly as too few clusters generate ineffective pseudo-OOD samples for regularization. Conversely, a sufficient  $K$  improves the representation of the ID data, which helps generate higher-quality outliers and enhances detection capability. This analysis confirms the importance of selecting an adequate number of prototypes to achieve optimal model performance.

## Conclusion

We introduce PGOS, a framework for unsupervised graph OOD detection that learns an adaptive policy to replace static sampling. Our method first structures the latent space using Prototypical Graph Contrastive Learning, then deploys a policy-guided agent to explore low-density regions and generate informative pseudo-outliers. These pseudo-outliers are then used to train a robust OOD detection model. Experimental results demonstrate the effectiveness of our approach, with PGOS achieving strong performance and state-of-the-art results on the majority of OOD and anomaly detection benchmarks. Future work includes exploring more advanced reward mechanisms and extending our paradigm to other data modalities and security-related tasks.

## Acknowledgements

This work is supported in part by the NSFC under grant 62202164, the Shandong Provincial Natural Science Foundation under grant ZR2025MS1038, and the Young Scholars Program of Shandong University. Philip S. Yu is supported in part by the NSF under grants III-2106758 and POSE-2346158.

## References

- Amer, M.; Goldstein, M.; and Abdennadher, S. 2013. Enhancing one-class support vector machines for unsupervised anomaly detection. In *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description*, 8–15.
- Ding, Z.; Shi, J.; Shen, S.; Shang, X.; Cao, J.; Wang, Z.; and Gong, Z. 2024. Sgood: Substructure-enhanced graph-level out-of-distribution detection. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 467–476.
- Du, X.; Sun, Y.; Zhu, J.; and Li, Y. 2023. Dream the impossible: Outlier imagination with diffusion models. *Advances in Neural Information Processing Systems*, 36: 60878–60901.
- Du, X.; Wang, Z.; Cai, M.; and Li, Y. 2022. VOS: Learning What You Don’t Know by Virtual Outlier Synthesis. In *International Conference on Learning Representations*.
- Gu, J.; Qiao, Z.; and Li, Z. 2025. SpectralGap: Graph-Level Out-of-Distribution Detection via Laplacian Eigenvalue Gaps. *arXiv preprint arXiv:2505.15177*.
- Guo, Y.; Yang, C.; Chen, Y.; Liu, J.; Shi, C.; and Du, J. 2023. A data-centric framework to endow graph neural networks with out-of-distribution detection ability. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 638–648.
- Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, 1861–1870. Pmlr.
- Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.
- Hendrycks, D.; Mazeika, M.; and Dietterich, T. 2019. Deep Anomaly Detection with Outlier Exposure. *Proceedings of the International Conference on Learning Representations*.
- Hofmann, C.; Schmid, S.; Lehner, B.; Klotz, D.; and Hochreiter, S. 2024. Energy-based hopfield boosting for out-of-distribution detection. *Advances in Neural Information Processing Systems*, 37: 131859–131919.
- Hou, Y.; Zhu, H.; Liu, R.; Su, Y.; Xia, J.; Wu, J.; and Xu, K. 2025. Structural Entropy Guided Unsupervised Graph Out-Of-Distribution Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17258–17266.
- Jiang, W.; Cheng, H.; Chen, M.; Wang, C.; and Wei, H. 2024. DOS: Diverse Outlier Sampling for Out-of-Distribution Detection. In *The Twelfth International Conference on Learning Representations*.
- Junwei, H.; Xu, Q.; Jiang, Y.; Wang, Z.; Sun, Y.; and Huang, Q. 2024. HGOE: Hybrid External and Internal Graph Outlier Exposure for Graph Out-of-Distribution Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 1544–1553.
- Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations*.
- Kumar, N.; Šegvić, S.; Eslami, A.; and Gumhold, S. 2023. Normalizing flow based feature synthesis for outlier-aware object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5156–5165.
- Lee, O.-J.; et al. 2025. Pre-Training Graph Neural Networks on Molecules by Using Subgraph-Conditioned Graph Information Bottleneck. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17204–17213.
- Li, A.; and Pan, Y. 2016. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory*, 62: 3290–3339.
- Li, H.; and Zhang, T. 2025. Outlier Synthesis via Hamiltonian Monte Carlo for Out-of-Distribution Detection. In *The Thirteenth International Conference on Learning Representations*.
- Li, J.; Xing, Q.; Wang, Q.; and Chang, Y. 2023. Cvtgad: Simplified transformer with cross-view attention for unsupervised graph-level anomaly detection. In *Joint European conference on machine learning and knowledge discovery in databases*, 185–200. Springer.
- Li, J.; Zhou, P.; Xiong, C.; and Hoi, S. C. 2021. Prototypical Contrastive Learning of Unsupervised Representations. In *International Conference on Learning Representations*.
- Li, P.; Yu, X.; Peng, H.; Xian, Y.; Wang, L.; Sun, L.; Zhang, J.; and Yu, P. S. 2025. Relational Prompt-Based Pre-Trained Language Models for Social Event Detection. *ACM Trans. Inf. Syst.*, 43(1): 12:1–12:43.
- Li, W.; Saidi, H.; Sanchez, H.; Schäf, M.; and Schweitzer, P. 2016. Detecting similar programs via the Weisfeiler-Leman graph kernel. In *International conference on software reuse*, 315–330. Springer.
- Lin, S.; Liu, C.; Zhou, P.; Hu, Z.-Y.; Wang, S.; Zhao, R.; Zheng, Y.; Lin, L.; Xing, E.; and Liang, X. 2022. Prototypical graph contrastive learning. *IEEE transactions on neural networks and learning systems*, 35(2): 2747–2758.
- Lin, X.; Cao, Y.; Sun, N.; Zou, L.; Zhou, C.; Zhang, P.; Zhang, S.; Zhang, G.; and Wu, J. 2025. Conformal graph-level out-of-distribution detection with adaptive data augmentation. In *Proceedings of the ACM on Web Conference 2025*, 4755–4765.
- Liu, F. T.; Ting, K. M.; and Zhou, Z.-H. 2008. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, 413–422. IEEE.
- Liu, Y.; Ding, K.; Liu, H.; and Pan, S. 2023a. Good-d: On unsupervised graph out-of-distribution detection. In *Proceedings of the sixteenth ACM international conference on web search and data mining*, 339–347.

- Liu, Z.; Zhang, W.; Zhang, X.; and Tang, J. 2023b. SIGNET: Self-supervised One-Class Graph-Level Anomaly Detection via Contrastive Learning. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*.
- Ma, X.; Wu, J.; Yang, J.; and Sheng, Q. Z. 2023. Towards graph-level anomaly detection via deep evolutionary mapping. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, 1631–1642.
- Morris, C.; Kriege, N. M.; Bause, F.; Kersting, K.; Mutzel, P.; and Neumann, M. 2020. TUDataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*.
- Neumann, M.; Garnett, R.; Bauckhage, C.; and Kersting, K. 2016. Propagation kernels for graphs. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, 4307–4313.
- Qiu, Z.; Liu, Y.; Cheng, P.; Ma, Y.; Ma, H.; and Yang, Q. 2022. OCGTL: One-class graph neural networks for temporal link anomaly detection. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, 875–883.
- Sehwag, V.; Chiang, M.; and Mittal, P. 2021. SSD: A Unified Framework for Self-Supervised Outlier Detection. In *International Conference on Learning Representations*.
- Shen, X.; Wang, Y.; Zhou, K.; Pan, S.; and Wang, X. 2024. Optimizing ood detection in molecular graphs: A novel approach with diffusion models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2640–2650.
- Song, Y.; Wei, Y.; Lu, Y.; Sun, Q.; Shao, M.; Wang, L.-e.; Hu, C.; Li, X.; and Fu, X. 2025. Mitigating Message Imbalance in Fraud Detection with Dual-View Graph Representation Learning. In Kwok, J., ed., *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, 3281–3289. International Joint Conferences on Artificial Intelligence Organization. Main Track.
- Sun, F.-Y.; Hoffmann, J.; Verma, V.; and Tang, J. 2020. InfoGraph: Unsupervised and interpretable graph-level representation learning via mutual information maximization. In *International Conference on Learning Representations (ICLR)*.
- Sun, L.; Du, Y.; Gao, S.; Ye, J.; Wang, F.; Ren, F.; Liang, M.; Wang, Y.; and Wang, S. 2023a. GroupAligner: A Deep Reinforcement Learning with Domain Adaptation for Social Group Alignment. *ACM Trans. Web*, 17(3): 17:1–17:30.
- Sun, L.; Huang, Z.; Wu, H.; Ye, J.; Peng, H.; Yu, Z.; and Yu, P. S. 2023b. DeepRicci: Self-supervised Graph Structure-Feature Co-Refinement for Alleviating Over-squashing. In *Proceedings of ICDM*, 558–567. IEEE.
- Sun, L.; Huang, Z.; Zhou, S.; Wan, Q.; Peng, H.; and Yu, P. S. 2025a. RiemannGFM: Learning a Graph Foundation Model from Riemannian Geometry. In *Proceedings of the ACM Web Conference*, 1154–1165. ACM.
- Sun, L.; Zhang, Z.; Wang, F.; Ji, P.; Wen, J.; Su, S.; and Yu, P. S. 2023c. Aligning Dynamic Social Networks: An Optimization Over Dynamic Graph Autoencoder. *IEEE Trans. Knowl. Data Eng.*, 35(6): 5597–5611.
- Sun, L.; Zhou, S.; Fang, B.; Zhang, H.; Ye, J.; Ye, Y.; and Yu, P. S. 2025b. Trace: Structural Riemannian Bridge Matching for Transferable Source Localization in Information Propagation. In *Proceedings of the 34th IJCAI*, 3308–3316.
- Sutton, R. S.; Barto, A. G.; et al. 1998. *Reinforcement learning: An introduction*. MIT press Cambridge.
- Tao, L.; Du, X.; Zhu, J.; and Li, Y. 2023. Non-parametric Outlier Synthesis. In *The Eleventh International Conference on Learning Representations*.
- Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.
- Wang, L.; He, D.; Zhang, H.; Liu, Y.; Wang, W.; Pan, S.; Jin, D.; and Chua, T.-S. 2024. GOODAT: towards test-time graph out-of-distribution detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15537–15545.
- Wang, Y.; Liu, Y.; Shen, X.; Li, C.; Miao, R.; Ding, K.; Wang, Y.; Pan, S.; and Wang, X. 2025. Unifying Unsupervised Graph-Level Anomaly Detection and Out-of-Distribution Detection: A Benchmark. In *The Thirteenth International Conference on Learning Representations*.
- Yang, L.; Chen, X.; Zhuo, J.; Jin, D.; Wang, C.; Cao, X.; Wang, Z.; and Guo, Y. 2025. Disentangled Graph Spectral Domain Adaptation. In *Proceedings of the 42nd International Conference on Machine Learning*.
- You, Y.; Chen, T.; Wang, Z.; Shen, Y.; et al. 2020. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 5812–5823.
- Zhang, G.; Yang, Z.; Wu, J.; Jiao, P.; and Yang, J. 2025. Enhancing Graph Neural Networks for Out-of-Distribution Graph Detection. *IEEE Transactions on Neural Networks and Learning Systems*.
- Zhao, L.; and Akoglu, L. 2023. On using classification datasets to evaluate graph outlier detection: Peculiar observations and new insights. *Big Data*, 11(3): 151–180.
- Zheng, H.; Wang, Q.; Fang, Z.; Xia, X.; Liu, F.; Liu, T.; and Han, B. 2023. Out-of-distribution detection learning with unreliable out-of-distribution sources. *Advances in neural information processing systems*, 36: 72110–72123.
- Zhu, J.; Geng, Y.; Yao, J.; Liu, T.; Niu, G.; Sugiyama, M.; and Han, B. 2023. Diversified outlier exposure for out-of-distribution detection via informative extrapolation. *Advances in neural information processing systems*, 36: 22702–22734.
- Zhuo, J.; Liu, Y.; Lu, Y.; Ma, Z.; Fu, K.; Wang, C.; Guo, Y.; Wang, Z.; Cao, X.; and Yang, L. 2025a. DUALFormer: Dual Graph Transformer. In *Proceedings of the 13th International Conference on Learning Representations*.
- Zhuo, J.; Ma, Z.; Lu, Y.; Liu, Y.; Fu, K.; Jin, D.; Wang, C.; Wu, W.; Wang, Z.; Cao, X.; and Yang, L. 2025b. A Closer Look at Graph Transformers: Cross-Aggregation and Beyond. In *Advances in the 39th Annual Conference on Neural Information Processing Systems*.