Graph Out-of-Distribution Detection Goes Neighborhood Shaping

Tianyi Bao [1] **Qitian Wu** [1] **Zetian Jiang** [1] **Yiting Chen** [1] **Jiawei Sun** [1] **Junchi Yan** [1]

Abstract

Despite the rich line of research works on out-of-distribution (OOD) detection on images, the literature on OOD detection for interdependent data, e.g., graphs, is still relatively limited. To fill this gap, we introduce *TopoOOD* as a principled approach that accommodates graph topology and neighborhood context for detecting OOD node instances on graphs. Meanwhile, we enrich the experiment settings by splitting in-distribution (ID) and OOD data based on distinct topological distributions, which presents new benchmarks for a more comprehensive analysis of graph-based OOD detection. The latter is designed to thoroughly assess the performance of these discriminators under distribution shifts involving structural information, providing a rigorous evaluation of methods in the emerging area of OOD detection on graphs. Our experimental results show the competitiveness of the proposed model across multiple datasets, as evidenced by up to a 15% increase in the AUROC and a 50% decrease in the FPR compared to existing state-of-the-art methods.

1. Introduction

Recognizing out-of-distribution (OOD) input perturbations to improve the robustness of machine learning models is crucial for real-world applications, particularly in safety-critical and high-stakes domains. Currently, the susceptibility to OOD samples remains a pivotal concern in developing reliable AI systems. For example, for large language models, plenty of work has been attempted to evaluate model robustness under the threats of confusing synonyms as well as typographical errors in prompts (Zhu et al., 2023), adversarial contents (Nie et al., 2019; Zhuo et al., 2023), and noisy inputs (Wang et al., 2023), demonstrating the lack of adequate robustness to uncertain outliers.

A similar challenge is also critical yet relatively underexplored when it comes to graph-structured data. Graph Neural Networks (GNNs) typically operate under the assumption that training and testing datasets share the same distribution, which does not mirror real-world scenarios where distribution shifts of various types widely exist (Koh et al., 2021; Wu et al., 2022; Zhu et al., 2021; Yang et al., 2022). More importantly, this assumption causes the classifiers to provide over-confidence predictions for OOD samples and hampers the reliability of GNN models. Simultaneously, compared to other types of data, the difficulty of OOD detection for graphs at the node level is further exacerbated by the interdependence within the neighborhood.

The current methods for node-level OOD detection either model uncertainty predominantly on the predicted categorical distribution but sideline the node feature embeddings (Hasanzadeh et al., 2020; Rong et al., 2020; Stadler et al., 2021; Zhao et al., 2020) or over-rely on the features of individual nodes, concurrently operating under the assumption that connected nodes predominantly tend to be sampled from similar distributions (Wu et al., 2023). Yet, while IDs and OODs intermingle in the neighborhood, GNNSafe fails to provide a discriminative energy score, as illustrated via the toy example given in Fig. 1. Such unsatisfactory performance highlights that, unlike vision data, node OOD detection requires the appropriate assessment of each node's neighborhood information.

In this paper, we start from the neighborhood topology, combining it with interdependent node features to develop a robust OOD detector, *TopoOOD*, for graph structural data. Inspired by the well-established Dirichlet energy (Zhou & Schölkopf, 2005), we adopt a comprehensive outlook, introducing a generalized node-wise Dirichlet energy as the confidence score. We further propose new experimental benchmarks by partitioning ID and OOD instances based on different topological distributions to enrich the limited dataset collection. **In summary, the contributions are:**

1) **An Innovative Detection Metric:** We design the node-wise k-hop Dirichlet energy (raw kHDE) to measure the level of turbulence in a node's neighborhood and quantify it as the confidence score. We demonstrate the superiority

[1]School of Artificial Intelligence & Department of Computer Science and Engineering & MoE Lab of AI, Shanghai Jiao Tong University, Shanghai, China. Correspondence to: Junchi Yan <yanjunchi@sjtu.edu.cn>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

of raw kHDE and provide theoretical justification for its effectiveness in the discrimination of IDs and OODs.

2) **Principled Detector Design:** In response to the energy convergence issue identified with raw kHDE in large-scale graphs, we introduce the **G**eneralized node-wise **D**irichlet **E**nergy (GDE), which improves the raw kHDE by incorporating a weighting mechanism for each node within each k-hop subgraph, based on their structural proximity to the central node. Significantly, TopoOOD, our proposed detector, employs GDE as a robust core component and can be seamlessly integrated with various GNN backbones.

3) **A New Evaluation Setting:** We craft a topology-aware setting for splitting ID and OOD instances based on structural information including triangleness, squareness, and cliqueness of graphs, contributing to enrich the benchmark settings of graph-structural data's OOD detection.

4) **Practical Efficacy:** Comprehensive experiments across numerous benchmarks, including ours, validate that our simple yet effective method achieves up to an approximately **15%** enhancement in AUROC and a **50%** reduction in FPR over SOTA models.

2. Preliminaries

2.1. Graph Notation and Node Classification

For a graph $\mathcal{G} = (V, E)$, we consider $V = \{i | 1 \leq i \leq N\}$ denotes the node set and $E = \{e_{ij}\}$ denotes the edge set where e_{ij} represents an edge from i to j. The adjacency matrix of $\mathcal{G}$ is $A = a_{ij} \in \mathbb{R}^{N \times N}$. We define the out-degree of each node $d_i = \sum_{j \in V} a_{ij}$. Let $\tilde{A} := A + I_n$ and $\tilde{D} := D + I_n$ be the adjacency and degree matrix of the graph augmented with self-loops. The augmented normalized Laplacian is then given by $\tilde{\Delta} := I_n - \tilde{P}$, where $\tilde{P} := \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$ is an augmented normalized adjacency matrix for the neighborhood aggregation in GNNs.

Each instance i has an input feature vector $\mathbf{x}_i \in \mathbb{R}^M$ and a label $y_i \in \{1, ..., C\}$ where M denotes the dimension of the input feature and C is the class number. The N instances are partially labeled and we define $\mathcal{I}_{lb}$ as the labeled node set and $\mathcal{I}_{unlb}$ as the unlabeled node set, i.e., $\mathcal{I} = \mathcal{I}_{lb} \cup \mathcal{I}_{unlb}$. The goal of standard (semi-)supervised learning on graphs is to train a node-level classifier f with $\hat{Y} = f(X, A)$, where $X = [\mathbf{x}_i]_{i \in \mathcal{I}}$ and $\hat{Y} = [\hat{y}_i]_{i \in \mathcal{I}}$, that predicts the labels for in-distribution instances in $\mathcal{I}_{unlb}$.

2.2. OOD Nodes Detection on Graphs

Graph Neural Networks (GNNs) (Scarselli et al., 2009) use message passing to aggregate features from neighboring nodes to update the central node's representations. The GNN models can be denoted as h with parameters θ, such that $h_\theta(\mathbf{x}_i, \mathcal{G}_{\mathbf{x}_i}) = \mathbf{x}_i^L$, where $\mathcal{G}_{\mathbf{x}_i} = (\mathbf{x}_{j \in \mathcal{N}_i}, A_i)$ represents the ego-graph centered at $\mathbf{x}_i$, $\mathcal{N}_i$ signifies the set of nodes within a designated distance from node i on the graph, and A_i is the corresponding adjacency matrix.

To enable the model's awareness of OOD instances, in line with the popular treatment in literature, such as (Sun et al., 2022; Wu et al., 2023), we integrate a detection mechanism for uncertainty alongside the primary classification model h to accurately classify ID instances while also recognizing OOD inputs as "unknown". Generally, the goal is to formulate an optimal decision function, F associated with model h that can evaluate any provided input $\mathbf{x}_i$:

$$F(\mathbf{x}_i, \mathcal{G}_{\mathbf{x}_i}; h) = \begin{cases} 1, & \text{if } i \text{ is an ID instance,} \\ 0, & \text{if } i \text{ is an OOD instance,} \end{cases} \tag{1}$$

Provided this framework, node OOD detection should depend on both individual node feature embeddings and their neighborhood context.

2.3. Dirichlet Energy

Dirichlet energy is designed to measure the embedding smoothness with the weighted node pair distance at the graph level as an effective training guide for GNNs (Zhou & Schölkopf, 2005). Formally, given the node embedding matrix $X^l = \left[\mathbf{x}_1^l, \cdots, \mathbf{x}_N^l\right]^\top \in \mathbb{R}^{N \times M}$ learned from GNNs at the l-th layer, the Dirichlet energy is

$$\operatorname{tr}\left(X^{l^\top} \tilde{\Delta} X^l\right) = \frac{1}{2} \sum a_{ij} \left\| \frac{\mathbf{x}_i^l}{\sqrt{1+d_i}} - \frac{\mathbf{x}_j^l}{\sqrt{1+d_j}} \right\|_2^2, \tag{2}$$

where $\operatorname{tr}(\cdot)$ denotes the matrix trace, $a_{ij} \in A$, and d_i represents the node degree as the i-th diagonal element of the matrix. Given that the aggregation step commonly postulates nodes with analogous properties are likely to cluster densely after gradient optimization (Stadler et al., 2021), eventually, ID instances will closely knit within the embedding space whereas the OOD instances will be more dispersed. This rationally lends to the potential application of Dirichlet energy in the area of graph OOD detection.

3. Methodology

To apply the classic Dirichlet energy for OOD node detection, we require a refined transformation to the node level. In this section, we first propose several assumptions to formulate the scenarios. We then propose the raw k-hop Dirichlet energy and theoretically prove its ability to differentiate between ID and OOD interdependent data. Subsequently, we elucidate the problem of convergence which is the primary limitation of raw kHDE. This then motivates our refinement into the generalized node-wise Dirichlet energy.

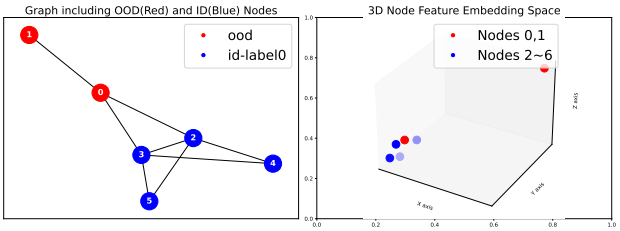


Figure 1. A toy example to visualize the limitation of GNNSafe++ (Wu et al., 2023). This example depicts a simple six-node graph (**Left**), with nodes 0 and 1 as OOD and 2-5 as IDs in class 1. Despite initial differences in logits (**Right**), node 0's final score, influenced by neighbor propagation in GNNSafe++, aligns closely with ID nodes, leading to potential misidentification as ID by GNNSafe++.

3.1. Assumptions

We first put forth a set of assumptions related to the properties of OOD detection in graph structural data. For succinctness in exposition, we use $\mathbf{x}_i$ to denote the logits of a node i, and ν_{id} (resp. ν_{ood}) to denote the ID (resp. OOD) instances in $\mathcal{G}$. We assume the logits of ID nodes $\mathbf{x}_{id}$ are in a probability distribution $P_{id}(\mathbf{x})$, i.e. $\mathbf{x}_{id} \sim P_{id}(\mathbf{x})$, and $\mathbf{x}_{ood} \sim P_{ood}(\mathbf{x})$. Lastly, without further specification, $\mathbb{E}$ denotes $\mathbb{E}_{\nu \in \mathcal{G}}$.

1) **IDs' Density in Neighborhoods:** Let p denote the expected number of ID instances within $\mathcal{N}_k(\nu_{id})$, and q signify the expected number of ID instances within $\mathcal{N}_k(\nu_{ood})$. We posit that IDs are more densely populated in the ID-centered subgraphs on average, i.e., $p > q$.

2) **Uniformity in Subgraph Sizes:** The sizes of ID-centered and OOD-centered k-hop subgraphs are postulated to maintain a similar scale on average. Formally:

$$\mathbb{E}[|\mathcal{N}_k(\nu_{id})|] = \mathbb{E}[|\mathcal{N}_k(\nu_{ood})|].$$

3) **Uniformity in Logit Scale:** Extracted from well-supervised Graph Neural Networks, the logits of ID and OOD samples are expected to exhibit uniform scale on average. Mathematically, this can be expressed as:

$$\mathbb{E}_{\mathbf{x}_i \sim P_{id}}[\|\mathbf{x}_i\|_2^2] = \mathbb{E}_{\mathbf{x}_i \sim P_{ood}}[\|\mathbf{x}_i\|_2^2].$$

4) **Class-based Clustering vs. Dispersal:** In the feature embedding space, ID logits are conjectured to cluster by class, while OOD logits disperse after passing through well-trained GNNs. Hence, we have

$$\mathbb{E}_{\mathbf{x}_1, \mathbf{x}_2 \sim P_{id}(\mathbf{x})}[\mathbf{x}_1^T \mathbf{x}_2] > \mathbb{E}_{\mathbf{x}_3, \mathbf{x}_4 \sim P_{ood}(\mathbf{x})}[\mathbf{x}_3^T \mathbf{x}_4].$$

Remarks. In the context of multi-graph scenarios, where IDs and OODs originate from distinct graph domains, we assume that the graph structure of $\mathcal{G}_{\text{id}}$ bears a structural resemblance to that of $\mathcal{G}_{\text{ood}}$. Consequently, all assumptions can still be preserved. Further, it is essential to emphasize that the formalization of these assumptions is primarily for the clarity and comprehensibility of proofs. Importantly, the implementation of TopoOOD **DOES NOT** technically rely on any of these assumptions.

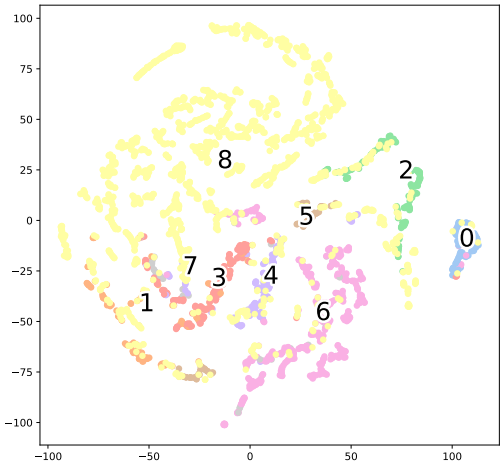


Figure 2. Logits of nodes from the Amazon-Photo dataset. ID and OOD instances are split based on the Triangleness. Label 8 (yellow) represents the OOD nodes and labels 0 to 7 (in different colors) represent ID instances from 8 classes. After the training phase, OOD data spread randomly over the embedding space, whereas ID instances are clustered based on their labels.

3.2. Modeling the Level of Disorganization in Neighborhood

After passing through a well-supervised GNN encoder, ID instances tend to cluster based on classes in the embedding space whereas OOD instances tend to disperse randomly over the entire space (visualized in Fig. 2). To capture such dissimilarity, for each node i, we define the Dirichlet energy of its k-hop subgraph, $\mathcal{N}_k(i)$ as the node's *raw k-hop Dirichlet energy (raw kHDE)*, to model the embedding smoothness of this i-centered neighborhood. Raw kHDE is denoted by $E_r(\mathcal{N}_k(\nu))$, and the formal definition is as follows:

$$E_r(\mathcal{N}_k(\nu)) = \frac{1}{2} \sum_{i,j \in \mathcal{N}_k(\nu)} a_{ij} \cdot \left\| \frac{\mathbf{x}_i^L}{\sqrt{1+d_i}} - \frac{\mathbf{x}_j^L}{\sqrt{1+d_j}} \right\|_2^2, \quad (3)$$

where i and j are two different nodes in $\mathcal{N}_k(\nu)$ and $a_{ij} \in A$. $\mathbf{x}_i^L$ denotes the logits of node i, and we call ν the *central node* of $\mathcal{N}_k(\nu)$. Based on the definition, the expectation of a node ν's raw kHDE for an arbitrary k, $\mathbb{E}[E_r(\mathcal{N}_k(\nu))] =$

$$\mathbb{E}[\left(\mathbf{x}_i^L\right)^\top \mathbf{x}_i^L] \cdot \mathbb{E}[|\mathcal{N}_k(\nu)|] - \mathbb{E}[\left(\mathbf{x}_i^L\right)^\top \mathbf{x}_j^L] \cdot \mathbb{E}[|\mathcal{N}_k(\nu)|] \cdot t, \quad (4)$$

where $i, j \in \mathcal{N}_k(\nu)$, and t is a subgraph-structure related coefficient (details in Appendix A).

Theorem 3.1 (Expectation Gap of kHDE Between ID and OOD Instances)**.** *For interdependent data in graphs, the*

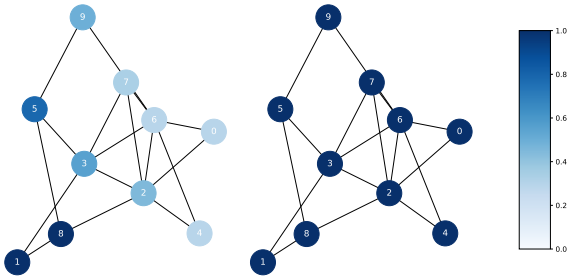


Figure 3. The visualization of weights for each node in calculating raw kHDE (right) and GDE (left). The central node is node 8. Deeper colors represent larger weights assigned.

expectation of ID instances' raw kHDE is smaller than that of OOD instances, i.e.,

$$\mathbb{E}[E_r(\mathcal{N}_k(\nu_{id}))] < \mathbb{E}[E_r(\mathcal{N}_k(\nu_{ood}))].$$

The proof is provided in Appendix A. This theorem aligns with the intuition that OOD instances with haphazard neighborhoods are likely to manifest elevated raw kHDE values compared to IDs. Such an expectation gap naturally empowers raw kHDE to undertake OOD detection. The rationale behind the mechanism of raw kHDE is to expand the estimation of node uncertainty to a subgraph domain, and such expansion essentially reinforces the confidence in node OOD detection compared to the score mechanism focusing on individual nodes. However, as k increases, the neighborhoods, $\mathcal{N}_k(\nu_{\text{id}})$ and $\mathcal{N}_k(\nu_{\text{ood}})$, start to exhibit strong resemblance, particularly in the single-graph scenario, i.e., $\nu_{\text{id}}, \nu_{\text{ood}} \in \mathcal{G}$ due to overlapping.

Theorem 3.2 (Energy Convergence)**.** *For an arbitrary node $\nu \in \mathcal{G}$, as k increases, $E_r(\mathcal{N}_k(\nu))$ will converge to the energy of the largest connected subgraph $\mathcal{G}_\nu$ containing ν, namely $E_r(\mathcal{N}_k(\nu)) = E_r(\mathcal{G}_\nu)$; if $\mathcal{G}$ is connected, $E_r(\mathcal{N}_k(\nu))$ will converge to the classic Dirichlet energy of $\mathcal{G}$, i.e. $E_r(\mathcal{N}_k(\nu)) = E_r(\mathcal{G})$.*

Proof is shown in Appendix A. This theorem indicates that the continuous expansion of ν-centered subgraphs tremendously diminishes the discriminative power of raw kHDE. This reveals a fundamental limitation: *raw kHDE assigns equal importance to all nodes in the subgraph, irrespective of their structural proximity to the central node*.

3.3. Node-Wise Generalized Dirichlet Energy

To address this issue, we introduce a straightforward but potent weighting scheme to apply weights to $\omega \in \mathcal{N}_k(\nu)$ step by step, based on i's structural proximity to the central node ν. Such k-hop Dirichlet energy with central-node-specific weights is termed as *generalized Dirichlet energy (GDE), denoted by* $E(\mathcal{N}_k(\nu))$. Namely, at the first step, we define $E(\mathcal{N}_1(\nu))$ as the node ν's raw 1-hop Dirichlet energy; for each step $k > 1$, we have

$$E(\mathcal{N}_k(\nu)) = \alpha E(\mathcal{N}_{k-1}(\nu)) + \frac{1-\alpha}{d_\nu} \sum_\omega a_{\nu\omega} E(\mathcal{N}_{k-1}(\omega)), \tag{5}$$

where node ω is a neighbor of node ν. Evidently, after k steps, information from all nodes in $\mathcal{N}_k(\nu)$ is aggregated, with nodes farther from the central node assigned a smaller weight (visualized in Fig. 3).

GDE of each node ν can be reformulated as:

$$E(\mathcal{N}_k(\nu)) = \frac{1}{2} \sum_{i,j \in \mathcal{N}_k(\nu)} a_{ij} \cdot b_i \cdot \left\| \frac{\mathbf{x}_i^L}{\sqrt{1+d_i}} - \frac{\mathbf{x}_j^L}{\sqrt{1+d_j}} \right\|_2^2, \tag{6}$$

where b_i denotes the weight of each node i in $\mathcal{N}_k(\nu)$.

Theorem 3.3 (Preserved Capability of Discrimination)**.** *Given the same assumption of Theorem 1, the expectation of ID nodes' GDE is smaller than that of OOD nodes, i.e.,* $\mathbb{E}[E(\mathcal{N}_k(\nu_{id}))] < \mathbb{E}[E(\mathcal{N}_k(\nu_{ood}))]$

Proof is in Appendix A. The theorem asserts that integrating weights through our aggregation scheme does not compromise the discrimination between ID and OOD samples. Hence, opting for GDE over raw kHDE in $\mathcal{N}_k(\nu)$ is prudent for two reasons. First, even two nodes with identical k-hop subgraphs now yield disparate energy scores, since the aggregation weights of neighbors are contingent on the central nodes. Such distinction solves the problem of convergence. Second, adding aggregation through such a propagation scheme will not interfere with the capability of discriminating the ID and OOD instances compared to raw kHDE.

3.4. Dirichlet Energy induced OOD Detector

Inspired by the Dirichlet energy, *TopoOOD* combine logits extracted from a GNN encoder and neighborhood topology structure to measure GDE of each node as the confidence score for detection, lower for IDs whereas higher for OODs. Concurrently, in line with the popular treatment in literature (Hendrycks et al., 2018; Wu et al., 2023),

In line with previous literature (Hendrycks et al., 2018; Wu et al., 2023), we investigate a scenario that more closely mirrors real-world conditions by incorporating a training dataset with outlier exposure, wherein OOD instances are also present within the training sets. This is achieved by integrating a regularization loss that operates under the assumption of OOD exposure during training, thereby instructing the network on improved data representation methodologies. Further elaboration on this approach is provided in Sec. 4.3. Overall, the decision function for OOD detection is formulated as:

$$F(\mathbf{x}_\nu, \mathcal{N}_k(\nu); h) = \mathbf{1}\left\{-E(\mathcal{N}_k(\nu)) \geq \tau\right\},$$

where τ is the threshold and the return of 1 indicates the identification for ID.

Remarks Despite the implications of its nomenclature, Dirichlet energy provides a mechanism to measure the weighted node pair distance within a graph or, in our scenarios, a subgraph. Hence, TopoOOD pioneers the exploration of distance-based approaches for OOD detection concerning graph-structured data in this nascent domain.

4. Experiments

In this section, we first introduce a novel metric, leveraging graph topological data to generate synthetic distribution shifts. Subsequently, we present an extensive experimental assessment of *TopoOOD*, examining the impact of critical hyper-parameters on our detector's efficacy.

4.1. A Topology-Aware Evaluation Metric for More Challenging Distribution Shift

Limitation of Current Metrics. Studies largely rely on *real distribution shifts* in datasets to assess the robustness and efficacy of OOD detectors for graph data (Stadler et al., 2021; Wu et al., 2023). Typically, these scenarios encompass *domain-based distribution shifts* in multi-graph datasets, where nodes are separated based on the graphs they originate from, and *feature-based distribution shifts* in single-graph datasets, where nodes are divided according to their temporal attributes, for instance. Though important, the availability and diversity of real shifts in graph datasets can be limited. Metrics have been introduced to generate synthetic distribution shifts to fully examine the OOD detectors. Some opt for *data-splitting* in the original graph datasets based on individual node classes (label leave-out) (Stadler et al., 2021), while others turn to *data-generation*, creating new OOD graphs through random node feature interpolation or structural modification (Wu et al., 2023). Such methods oversimplify the task by transferring it to the existing domain- or feature-based problem.

OOD Data Splitting from Topological Distribution. In this work, we propose that nodes within a graph can be segregated based on their structural information and term these settings as *topological distribution shifts*. We delineate three settings generated via such topological split as examples for illustration, which collectively cover a wide range of graph properties—from small-scale interactions (Triangleness) to medium-scale structures (Squareness) and large-scale cohesiveness (Cliqueness). Such broad coverage not only ensures that experimental settings capture various aspects of graph topology to provide a holistic evaluation of OOD detectors but also demonstrates relevance to phenomena commonly observed in real-world networks (See Appendix for more details).

i) Triangleness. We follow the definition of the clustering coefficient (Saramäki et al., 2007) and define a parameter called *triangleness*, which measures the density of the 1.5-degree egocentric graph for each node. Namely, when a node is deeply embedded within tight-knit groups in the graph and connects densely with its neighbors, the triangleness would be high. We here define the nodes with higher triangleness as having higher importance and being in-distribution instances:

$$tr_i = \frac{2T(i)}{d_i(d_i-1)}, \tag{7}$$

where $T(i)$ is the number of triangles through node i.

ii) Squareness. Beyond small groups in graphs, a node's inclination towards longer-range connections within the graph's small-world structure is also important. Utilizing the C_4 coefficient (Zhang et al., 2008), we introduce the *squareness* to measure the fraction of feasible squares around node i, essentially assessing the likelihood that a pair of i's neighbors have a distinct shared neighbor. Nodes with higher squareness usually have access to different small groups or clusters and are defined as ID instances. See the definition in Appendix B.

iii) Cliqueness. We here use *cliqueness* of node i to denote the number of maximal cliques in graph $\mathcal{G}$ which contain the node i. Since nodes involved in many maximal cliques usually belong to various fully connected groups in the graph, we assign higher importance to them and define them as ID instances.

These settings are crafted to highlight the complexity inherent in graph structures, particularly emphasizing the topological long-tail phenomenon where a minority of nodes, characterized by significant structural importance including triangleness, squareness, and cliqueness dominate the interaction dynamics. This focus on structural significance as the basis for splitting introduces a novel perspective on distribution shifts, emphasizing the disparities in node connectivity and interaction patterns as critical factors. Through this approach, we aim to tackle the challenges presented by structural long-tail distributions in detecting and analyzing topological distribution shifts, thereby offering a refined framework for understanding and addressing the complexities of graph-based data analysis in the context of out-of-distribution detection. Additional topological distribution shift settings are further provided in the Appendix B.3.

4.2. Datasets and Splits

We ground our experiments on six prominent real-world datasets, frequently employed in node classification benchmarks: **Twitch-Explicit** (Rozemberczki & Sarkar, 2021), **ogbn-Arxiv** (Hu et al., 2020), **Amazon-Photo** (McAuley et al., 2015), **Coauthor-CS** (Sinha et al., 2015), **Coauthor-Physics** (Shchur et al., 2018), and **Cora** (Sen et al., 2008). The detailed information is introduced below.

Table 1. Out-of-distribution detection by AUROC (↑) / AUPR (↑) / FPR95 (↓) on Twitch and Arxiv dataset based on the standard for data splitting mentioned in the previous section. The in-distribution testing accuracy is reported for calibration. For each column, the highest value is in red and the second highest value is in blue. Detailed results on each OOD dataset (i.e., sub-graph or year) are presented in Appendix D.4. GPN reports out-of-memory issue on Arxiv with a 24GB GPU.

Model	Twitch				Arxiv			
	AUROC (↑)	**AUPR(↑)**	**FPR (↓)**	**ID ACC(↑)**	**AUROC(↑)**	**AUPR(↑)**	**FPR(↓)**	**ID ACC(↑)**
MSP (2016)	33.59	49.14	97.45	68.72	63.91	75.85	90.59	53.78
Mahalanobis (2018b)	55.68	66.42	90.13	70.51	56.92	69.63	94.24	51.59
OE (2018)	55.72	70.18	95.07	70.73	69.80	80.15	85.16	52.39
ODIN (2020)	58.16	72.12	93.96	70.79	55.07	68.85	100.0	51.39
KNN (2022)	64.43	71.18	88.24	72.13	43.16	61.12	98.23	53.77
GKDE (2020)	46.48	62.11	95.62	67.44	58.32	72.62	93.84	50.76
GPN (2021)	51.73	66.36	95.51	68.09	-	-	-	-
GNNSafe++ (2023)	95.36	97.12	33.57	70.18	74.77	83.21	77.43	53.50
TopoOOD	99.72	99.99	0.00	71.19	87.11	87.61	32.10	52.80

Twitch. We use the multi-graph dataset Twitch to test our model's capability of handling the cross-domain distribution shift, with subgraph DE as ID data, EN as the OOD exposure for training, and other subgraphs as OOD data for testing.

Arxiv. We use this large-scale single-graph dataset to examine TopoOOD on temporal distribution shift with the papers published before 2015 as in-distribution data, those published in 2015 and 2016 as OOD exposure during training, and those published after 2017 for OOD testing.

Amazon-Photo, Coauthor-CS, Coauthor-Physics. We create synthetically topological distribution shifts according to the parameters defined in Sec. 4.1. The data-splitting details are in Appendix B.

Cora. We include a *label leave-out* setting, distinguishing certain classes as IDs and others as OODs, as previous work (Wu et al., 2023; Stadler et al., 2021). This approach is applied to Cora in addition to the three datasets.

4.3. Setup

Our implementation is based on Ubuntu 16.04, Cuda 11.0, Pytorch 1.13.0, and Pytorch Geometric 2.3.1. Most of the experiments run with an NVIDIA 2080Ti with 11GB memory, except for cases where the model requires larger GPU memory, for which we use an NVIDIA 3090 with 24GB memory for experiments.

Implementation details. We set the number of propagation steps k according to the size of the graph datasets, namely, k equal to 5 or 10 for most settings, and the propagation coefficient $\alpha = 0.5$. For fair comparison, the GCN model with layer depth 2 and hidden size 64 is used as the backbone encoder for all the OOD discriminators.

Loss design. We generally assume that detectors will be exposed to OOD data during their training phase in all scenarios, and such an OOD exposure (Hendrycks et al., 2018) allows to impose constraints on the disparity between GDE for ID and OOD nodes for supervision. We introduce a regularization loss, $\mathcal{L}_{reg}$, which narrows the GDE of ID nodes down to a range while pushing that of OOD ones further apart. The final objective is:

$$\mathcal{L} = \mathcal{L}_{sup} + \lambda \mathcal{L}_{reg}, \tag{8}$$

where $\mathcal{L}_{sup}$ denotes the classic negative log-likelihood (NLL) loss utilized for node classification, $\mathcal{L}_{reg}$ denotes the regularization loss for OOD detection, and λ represents a trading weight. There is much design flexibility for the regularization term $\mathcal{L}_{reg}$, and here we have

$$\begin{aligned}\mathcal{L}_{reg} = &\frac{1}{|\mathcal{I}_s|}\sum_{\nu\in\mathcal{I}_s}(\text{ReLU}(E(\mathcal{N}_k(\nu) - e_{in}))^2 \\ &+ \frac{1}{|\mathcal{I}_o|}\sum_{\nu\in\mathcal{I}_o}(\text{ReLU}(e_{out} - (E(\mathcal{N}_k(\nu))))^2.\end{aligned} \tag{9}$$

The instances we have in the training phase are $\mathcal{I} = \mathcal{I}_s \cup \mathcal{I}_o$, where $\mathcal{I}_s$ represents the set of in-distribution training data, and $\mathcal{I}_o$ represents some unwanted OOD training instances from a distinct distribution which were exposed to the model during the training phase. We set the values of e_{in} and e_{out} through grid search.

Evaluation metrics. We report the performance based on the metrics: (1) the area under the receiver operating characteristic curve (**AUROC**); (2) the area under the Precision-Recall curve (**AUPR**); (3) the false positive rate (**FPR95**) of OOD samples when the true positive rate of ID samples is at 95% and (4) ID classification accuracy (**ID ACC**).

Competitors. We mainly contrast our detector with eight baselines from two categories. The initial category encompasses those traditionally aligned with the OOD detection task for images, and we substitute the conventional CNN backbone with a GCN encoder, including MSP (Hendrycks & Gimpel, 2016), ODIN (Liang et al., 2020), Mahalanobis (Lee et al., 2018b), OE (Hendrycks et al., 2018), and KNN (Sun et al., 2022). The subsequent category encapsulates OOD detectors designed for interdependent data, including the Graph Posterior Network,

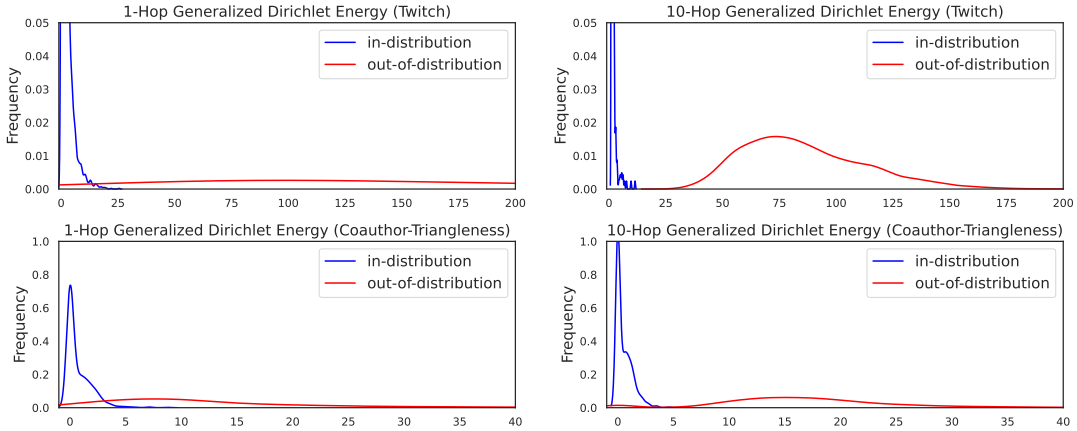


Figure 4. Distribution of 1-hop GDE of ID and OOD instances from Twitch (top right) and Coauthor-triangleness (bottom left), and the distribution of 10-hop GDE of ID and OOD instances from Twitch (top right) and Coauthor (bottom right).

termed GPN (Stadler et al., 2021), the Graph-based Kernel Dirichlet GCN method, termed GKDE (Zhao et al., 2020), and the established free-energy grounded baseline GNNSafe++ (Wu et al., 2023).

4.4. Experimental Results

Excellent Performance Compared to Competitors. In Tab. 1, we present the experimental results of TopoOOD on the Twitch and Arxiv datasets, juxtaposed against all the baselines discussed earlier. It is observed that TopoOOD consistently surpasses all competitors. Remarkably, even against the highly performing GNNSafe++ on Twitch, our model still exhibits superior performance, diminishing the average FPR95 by **33.57%** and elevating the average AUROC by **4.36%**. Moreover, TopoOOD considerably excels on Arxiv, improving the average AUROC by **16.58%** and decreasing the average FPR95 by **51.1%**. With these advancements, our detector consistently maintains classification accuracy for IDs. Collectively, these results reinforce the strong power of TopoOOD within real distribution shifts.

Concurrently, in Tab. 2, we can observe that TopoOOD also constantly performs outstandingly on synthetic topological distribution shifts compared to the aforementioned competitors. Particularly, in detecting OOD nodes split based on triangleness, TopoOOD exhibits a pronounced advantage, ameliorating the average AUROC on three datasets respectively by **20.56%**, **40.9%**, and **31.63%** over GNNSafe++, and **4.37%**, **9.67%**, and **1.1%** over the runner-ups. In contexts beyond triangleness, TopoOOD either parallels the best-performing competitor while greatly outperforming the others or significantly surpasses all baseline models. The full experimental results as well as the results for additional topological distribution shift settings are displayed in Appendix B. Overall, the results experimentally validate our insight: utilizing the neighborhood disorganization level to make OOD detection can provide a more accurate awareness of OOD nodes in all scenarios.

Table 2. Out-of-distribution detection performance measured by AUROC (↑) on datasets Amazon-Photo, Coauthor-CS, and Coauthor-Physics with three OOD types (**T**riangleness, **S**quareness, **C**liqueness). For each column, the highest value is in red and the second highest is in blue. Other results for AUPR, FPR95, and in-distribution accuracy are deferred to Appendix B.

Model	Amazon-Photo			Coauthor-CS			Coauthor-Physics		
	T	S	C	T	S	C	T	S	C
MSP (2016)	80.29	87.40	87.63	71.31	78.33	64.44	56.97	65.74	83.98
Mahalanobis (2018b)	82.21	75.28	29.72	86.06	79.41	60.29	87.50	87.06	41.87
OE (2018)	86.66	91.58	87.37	73.42	75.87	69.96	79.74	80.23	92.49
ODIN (2020)	19.33	49.43	11.86	49.23	49.25	49.55	49.95	32.20	41.32
KNN (2022)	48.49	70.59	72.56	61.17	63.75	51.92	40.35	47.06	72.07
GKDE (2020)	74.94	66.23	63.91	70.07	74.60	57.12	52.33	58.86	79.39
GPN (2021)	62.98	61.75	52.64	57.47	89.09	65.49	39.95	55.00	74.24
GNNSafe++ (2023)	70.47	80.41	95.87	54.83	70.91	93.36	56.97	76.41	95.49
TopoOOD	91.03	89.05	93.57	95.73	97.94	99.96	88.60	96.09	100.0

Additionally, Tabs 11 and 12 illustrate the superior efficacy of TopoOOD in the label leave-out scenario. On both the Coauthor-Physics and Amazon-Photo datasets, TopoOOD surpasses all baseline models across the three evaluation metrics. For the Coauthor-CS and Cora datasets, our approach achieves a reduction in FPR by **8.51%** and **13.2%** respectively, compared to the nearest competing model.

Backbone-Agnostic and Outstanding Performance. In Fig. 5, we evaluate the performance of TopoOOD against GNNSafe++ using various GNN encoders as backbones, including GCN, GAT (Veličković et al., 2018), JKNet (Xu et al., 2018), and MixHop (Abu-El-Haija et al., 2019). It turns out that TopoOOD consistently outperforms GNNSafe++ across all backbone types in diverse settings, underscoring its robust performance. These results experimentally highlight the model-agnostic nature of TopoOOD and affirm the efficacy of using GDE over free energy solely based on individual node features for the OOD discrimination over graph data.

Effectiveness of Propagation. Next, we experimentally

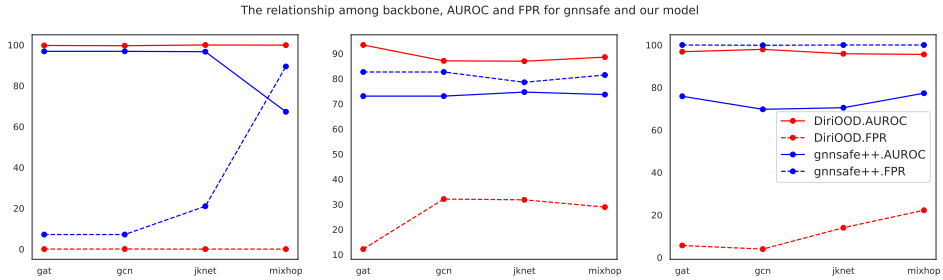


Figure 5. Study of the relationship between backbone types and model performance. We plot the AUROC and FPR of our approach and GNNSafe++ for Twitch, Arxiv, and Coauthor-CS-Squareness with four types of common GNN encoders. Red (dotted) lines present the AUROC (FPR) of TopoOOD whereas yellow ones represent that of GNNSafe++.

investigate the efficacy of our propagation scheme. In Fig. 4, we contrast the distribution of 1-hop GDE (without propagation) with the k-hop GDE (with k-step propagation) in multiple settings. Our findings reveal that, firstly, even at its base, the GDE for individual nodes possesses the inherent ability to differentiate between ID and OOD instances, experimentally supporting the idea that OOD instances are likely to deviate more from their neighbors in feature embedding space compared to IDs. Secondly, proper propagation with k in a reasonable range can minimize the overlap between the GDE distributions of IDs and OODs and provide a stronger capability of discrimination. Fig. 6 in Appendix B further supports such findings by displaying variation in AUROC of TopoOOD for propagation steps ranging from 0 to 20. Enhancement brought by the propagation plateaus beyond a threshold (e.g., $t = 8$ for Twitch), indicating beyond a certain subgraph size, including additional node information does not improve performance.

Effectiveness of Regularization Loss. We further investigate our detection-regularized training's efficacy by varying parameters e_{in}, e_{out}, and λ. As shown in Fig. 8, increasing λ within reasonable bounds bolsters OOD awareness without compromising node classification. Given a fixed λ, enlarging the gap between e_{in} and e_{out} can promote the performance of TopoOOD to a certain extent.

5. Related Works

We discuss the related works in the OOD area based on neural networks, which are mainly concerned with vision. Until very recently, OOD methods emerged in the graph domain.

OOD detection on Euclidean data. Extensive research has been conducted in recent literature to detect out-of-distribution samples. Recent works (Hendrycks & Gimpel, 2016; Hendrycks et al., 2018; Liang et al., 2020; Hsu et al., 2020) propose leveraging the softmax probability score to detect OOD samples, while other works (Lee et al., 2018b;a; Ren et al., 2019) incorporate generative methods to model the underlying data distribution and hence distinguish the samples from different distributions. Moreover, previous works (Malinin & Gales, 2018; 2019; Charpentier et al., 2020) employ Dirichlet-based uncertainty models to estimate the uncertainty and show outstanding performance in OOD detection compared to softmax-based confidence. Note that the Dirichlet-based uncertainty models fit the Dirichlet distribution, which is a different concept from the Dirichlet energy that we use. In particular, energy-based methods (Liu et al., 2020) propose the use of energy values instead of softmax scores to mitigate overconfident predictions for OOD samples. These methods mainly concentrate on scenarios where samples are independently generated (*e.g.* images), neglecting the common scenario where data samples exhibit inter-dependencies in graphs.

OOD detection on graphs. OOD detection on graph data can be broadly divided into two categories: graph-level and node-level detection. At the graph level, previous works (Li et al., 2022; Bazhenov et al., 2022) address OOD detection for graph classification where instances are independent graphs without inter-dependencies. To address OOD detection at the node level, previous works (Zhao et al., 2020; Stadler et al., 2021) propose Bayesian GNN models capable of detecting OOD nodes within graphs, while recent research GNNSafe++ (Wu et al., 2023) exploits the energy values extracted directly from predicted logits of standard GNNs to identify OOD nodes in graphs and adds an energy propagation based on a strong assumption that neighbors are likely from one distribution. Unlike graph-level OOD detection, the inter-dependencies between nodes (*i.e.* the topological context of the node) play a significant role in characterizing the node itself. However, the energy leveraged in (Wu et al., 2023) is solely based on the prediction on the node, neglecting the topological context of the node. We introduce a node-level OOD detection method that incorporates both node feature embeddings and the topological context.

6. Discussion

We would like to emphasize the potential **novelty** of this work: 1) **Dirichlet Energy Adaptation:** We have devised a node-wise Dirichlet energy model, a significant advancement over the classical graph-level Dirichlet energy, to address the OOD node detection task. This adaptation outperforms previous models by capturing the disorganization levels of neighborhoods and integrating feature embeddings with graph topology, without the restrictive assumption of distribution homogeneity. 2) **Neighborhood Disorganization:** We are the first to suggest that a node's in-distribution (ID) or out-of-distribution (OOD) identity can be detected by the disorganization level of its neighborhood—a shift from the traditional node feature level analysis. 3) **First Exploration of Distance-based Methods:** We innovate in the graph domain by adopting the essence of distance-based methods, typically used for image data OOD detection (measuring the relative difference between entities to determine their confidence scores (Sun et al., 2022)), thus pioneering their use in graph-structured data for node OOD detection. 4) **Experimental Frameworks Expansion:** Our research enhances the experimental landscape for node OOD detection by first introducing topology distribution shift scenarios. This expansion is particularly relevant to phenomena commonly observed in real-world networks and enables us to fully assess the capabilities of detectors, as detailed in the following sections.

Overall, the research motivation is that we aim to bolster the robustness of GNNs, enabling them to accurately validate predictions within their learned domain or cautiously signal inputs originating from out-of-distribution. This effort seeks to expand the operational domain and reliability of GNNs, promoting their wider application.

7. Conclusion

We have shown the intrinsic efficacy of making OOD node detection in graphs by using neighborhood disorganization information. We designed an OOD detector, TopoOOD, that leverages the generalized node-wise Dirichlet energy with a simple propagation scheme. Given the nature of the Dirichlet energy, TopoOOD can be viewed as pioneering a distance-based detection method for graph data. We further introduce novel experimental settings where OODs are split via our topology-related metrics. Experiments show that TopoOOD displays model-agnosticism, ease of use, and superior performance on both real and our challenging synthetic distribution shifts.

Impact Statement

Out-of-distribution (OOD) detection at the node level plays a vital role in a wide array of real-world applications, ensuring the reliability and safety of trustworthy AI systems handling graph-structural data. Specifically, our proposed node detector potentially benefits high-stake or safety-related domains including healthcare and medical diagnosis (Kukar, 2003), and autonomous driving (Dai & Van Gool, 2018). In all these applications, node OOD detection reduces the system's susceptibility to unknown or unexpected samples in test sets and improves the models' knowledge about what they do not know, thereby improving decision-making and system robustness.

References

Abu-El-Haija, S., Perozzi, B., Kapoor, A., Alipourfard, N., Lerman, K., Harutyunyan, H., Steeg, G. V., and Galstyan, A. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing, 2019.

Bazhenov, G., Ivanov, S., Panov, M., Zaytsev, A., and Burnaev, E. Towards OOD detection in graph classification from uncertainty estimation perspective. *CoRR*, abs/2206.10691, 2022.

Bron, C. and Kerbosch, J. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 1973.

Charpentier, B., Zügner, D., and Günnemann, S. Posterior network: Uncertainty estimation without OOD samples via density-based pseudo-counts. In *NeurIPS*, 2020.

Dai, D. and Van Gool, L. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *ITSC*, 2018.

Freeman, L. C. Centrality in social networks conceptual clarification. *Social Networks*, 1 (3):215–239, 1978. ISSN 0378-8733. doi: https://doi.org/10.1016/0378-8733(78)90021-7. URL https://www.sciencedirect.com/science/article/pii/0378873378900217.

Hasanzadeh, A., Hajiramezanali, E., Boluki, S., Zhou, M., Duffield, N., Narayanan, K., and Qian, X. Bayesian graph neural networks with adaptive connection sampling. In *ICML*, 2020.

Hendrycks, D. and Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Hendrycks, D., Mazeika, M., and Dietterich, T. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

Hsu, Y., Shen, Y., Jin, H., and Kira, Z. Generalized ODIN: detecting out-of-distribution image without learning from out-of-distribution data. In *CVPR*, 2020.

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. *NeurIPS*, 2020.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., Lee, T., David, E., Stavness, I., Guo, W., Earnshaw, B., Haque, I., Beery, S. M., Leskovec, J., Kundaje, A., Pierson, E., Levine, S., Finn, C., and Liang, P. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, pp. 5637–5664, 2021.

Kukar, M. Transductive reliability estimation for medical diagnosis. *Artificial intelligence in medicine*, 2003.

Langley, P. Crafting papers on machine learning. In *ICML*, 2000.

Lee, K., Lee, H., Lee, K., and Shin, J. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *ICLR*, 2018a.

Lee, K., Lee, K., Lee, H., and Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. 2018b.

Li, Z., Wu, Q., Nie, F., and Yan, J. Graphde: A generative framework for debiased learning and out-of-distribution detection on graphs. In *NeurIPS*, 2022.

Liang, S., Li, Y., and Srikant, R. Enhancing the reliability of out-of-distribution image detection in neural networks, 2020.

Liu, W., Wang, X., Owens, J., and Li, Y. Energy-based out-of-distribution detection. *NeurIPS*, 2020.

Malinin, A. and Gales, M. J. F. Predictive uncertainty estimation via prior networks. In *NeurIPS*, 2018.

Malinin, A. and Gales, M. J. F. Reverse KL-divergence training of prior networks: Improved uncertainty and adversarial robustness. In *NeurIPS*, 2019.

McAuley, J., Targett, C., Shi, Q., and Van Den Hengel, A. Image-based recommendations on styles and substitutes. In *SIGIR*, 2015.

Metcalf, L. and Casey, W. *Cybersecurity and Applied Mathematics*. Syngress Publishing, 1st edition, 2016. ISBN 0128044527.

Nie, Y., Williams, A., Dinan, E., Bansal, M., Weston, J., and Kiela, D. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*, 2019.

Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., DePristo, M. A., Dillon, J. V., and Lakshminarayanan, B. Likelihood ratios for out-of-distribution detection. In *NeurIPS*, 2019.

Rong, Y., Huang, W., Xu, T., and Huang, J. Dropedge: Towards deep graph convolutional networks on node classification. In *ICLR*, 2020.

Rozemberczki, B. and Sarkar, R. Twitch gamers: a dataset for evaluating proximity preserving and structural role-based node embeddings. *arXiv preprint arXiv:2101.03091*, 2021.

Saramäki, J., Kivelä, M., Onnela, J.-P., Kaski, K., and Kertész, J. Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E*, 2007.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. The graph neural network model. *IEEE Transactions on Neural Networks*, 2009.

Sen, P., Namata, G., Bilgic, M., Getoor, L., Galligher, B., and Eliassi-Rad, T. Collective classification in network data. *AI Magazine*, 29(3):93, Sep. 2008. doi: 10.1609/aimag.v29i3.2157. URL https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2157.

Shchur, O., Mumme, M., Bojchevski, A., and Günnemann, S. Pitfalls of graph neural network evaluation. *Relational Representation Learning Workshop, NeurIPS*, 2018.

Sinha, A., Shen, Z., Song, Y., Ma, H., Eide, D., Hsu, B.-J., and Wang, K. An overview of microsoft academic service (mas) and applications. In *WWW*, 2015.

Stadler, M., Charpentier, B., Geisler, S., Zügner, D., and Günnemann, S. Graph posterior network: Bayesian predictive uncertainty for node classification. In *NeurIPS*, 2021.

Sun, Y., Ming, Y., Zhu, X., and Li, Y. Out-of-distribution detection with deep nearest neighbors. In *ICML*, 2022.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. Graph attention networks. In *ICLR*, 2018.

Wang, J., Hu, X., Hou, W., Chen, H., Zheng, R., Wang, Y., Yang, L., Huang, H., Ye, W., Geng, X., et al. On the robustness of chatgpt: An adversarial and out-of-distribution perspective. *arXiv preprint arXiv:2302.12095*, 2023.

Wu, Q., Zhang, H., Yan, J., and Wipf, D. Handling distribution shifts on graphs: An invariance perspective, 2022.

Wu, Q., Chen, Y., Yang, C., and Yan, J. Energy-based out-of-distribution detection for graph neural networks. In *ICLR*, 2023.

Xu, K., Li, C., Tian, Y., Sonobe, T., Kawarabayashi, K.-i., and Jegelka, S. Representation learning on graphs with jumping knowledge networks. In *ICML*, 2018.

Yang, N., Zeng, K., Wu, Q., Jia, X., and Yan, J. Learning substructure invariance for out-of-distribution molecular representations. In *Advances in Neural Information Processing Systems*, 2022.

Zhang, P., Wang, J., Li, X., Li, M., Di, Z., and Fan, Y. Clustering coefficient and community structure of bipartite networks. *Physica A: Statistical Mechanics and its Applications*, 2008.

Zhao, X., Chen, F., Hu, S., and Cho, J.-H. Uncertainty aware semi-supervised learning on graph data. 2020.

Zhou, D. and Schölkopf, B. Regularization on discrete spaces. In *Pattern Recognition*, 2005.

Zhu, K., Wang, J., Zhou, J., Wang, Z., Chen, H., Wang, Y., Yang, L., Ye, W., Gong, N. Z., Zhang, Y., et al. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*, 2023.

Zhu, Q., Ponomareva, N., Han, J., and Perozzi, B. Shift-robust gnns: Overcoming the limitations of localized graph training data. *NeurIPS*, pp. 27965–27977, 2021.

Zhuo, T. Y., Li, Z., Huang, Y., Shiri, F., Wang, W., Haffari, G., and Li, Y.-F. On robustness of prompt-based semantic parsing with large pre-trained language model: An empirical study on codex. *arXiv preprint arXiv:2301.12868*, 2023.

A. Theoretical Proof

In this section, we delve into the theoretical foundations supporting the three theorems presented in Section 3. Given the under-developed state of OOD detection pertaining to graph-structured data, we commence by introducing a set of assumptions as Stadler et al. did in the work of GPN. Following this, we provide rigorous proofs underscoring the theoretical efficacy of our methodologies. We shall follow the notations and assumptions provided in Section 3.

A.1. Expectation of Raw kHDE

Initially, we elucidate the rationale behind the expected value of a node ν's raw k-hop Dirichlet energy, denoted as $\mathbb{E}[E_r(\mathcal{N}_k(\nu))]$. According to the definition of raw kHDE (given in Eq. 3), $\mathbb{E}[E_r(\mathcal{N}_k(\nu))]$

$$
\begin{aligned}
&= \frac{1}{2}\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)} a_{ij}\left\|\frac{\mathbf{x}_i}{\sqrt{1+d_i}}-\frac{\mathbf{x}_j}{\sqrt{1+d_j}}\right\|_2^2\right] \\
&= \frac{1}{2}\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)} a_{ij}\left(\frac{\mathbf{x}_i^\top\mathbf{x}_i}{1+d_i}+\frac{\mathbf{x}_j^\top\mathbf{x}_j}{1+d_j}-2\frac{\mathbf{x}_i^\top}{\sqrt{1+d_i}}\frac{\mathbf{x}_j}{\sqrt{1+d_j}}\right)\right] \\
&= \mathbb{E}\left[\sum_{i\in\mathcal{N}_k(\nu)} \mathbf{x}_i^\top\mathbf{x}_i\right] - \mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)} a_{ij}\frac{\mathbf{x}_i^\top}{\sqrt{1+d_i}}\frac{\mathbf{x}_j}{\sqrt{1+d_j}}\right] \\
&= \mathbb{E}[\mathbf{x}_i^\top\mathbf{x}_i]\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|] \\
&\quad - \mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)} \mathbf{x}_i^\top\mathbf{x}_j\right]\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{\sqrt{(1+d_i)(1+d_j)}}\right].
\end{aligned}
\tag{10}
$$

As we can see,

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{\sqrt{(1+d_i)(1+d_j)}}\right] \\
&\quad\le \mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{2}\left(\frac{1}{1+d_i}+\frac{1}{1+d_j}\right)\right] \\
&\quad= \mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{1+d_i}\right] \\
&\quad= \mathbb{E}\left[|\mathcal{N}_k(\nu)|\right],
\end{aligned}
\tag{11}
$$

and,

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{\sqrt{(1+d_i)(1+d_j)}}\right] \\
&\ge \mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{2a_{ij}}{(1+d_i)+(1+d_j)}\right] \\
&\ge \mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\right] \\
&= \mathbb{E}\left[|\mathcal{N}_k(\nu)|\cdot\frac{1+\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[d_i]}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\right],
\end{aligned}
\tag{12}
$$

For easy writing, we set t_ν for each $\mathcal{N}_k(\nu)$, such that

$$
\frac{1+\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[d_i]}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)} \le t_\nu \le 1.
$$

Collectively, we have $\mathbb{E}[E_r(\mathcal{N}_k(\nu))] =$

$$
\mathbb{E}[\mathbf{x}_i^\top\mathbf{x}_i]\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|] - \mathbb{E}[\mathbf{x}_i^\top\mathbf{x}_j]\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot t, \tag{13}
$$

where $t = \mathbb{E}[t_\nu]$, and, $i, j \in \mathcal{N}_k(\nu)$. Note that Eq. 13 is identical to Eq. 4 in Sec. 3.

A.2. Proof of Theorem 1

Lemma 1: Consider two arbitrary vectors $\mathbf{u}$ and $\mathbf{v}$ in space V. For a given vector $\mathbf{k}$ from a subspace U of V, the expected angle between $\mathbf{u}$ and $\mathbf{v}$ is identical to the expected angle between $\mathbf{k}$ and $\mathbf{v}$, namely:

$$
\mathbb{E}_{\mathbf{u},\mathbf{v}\sim V}[\theta(\mathbf{u},\mathbf{v})] = \mathbb{E}_{\mathbf{v}\sim V,\mathbf{k}\sim U}[\theta(\mathbf{k},\mathbf{v})].
$$

Theorem 1: The expectation of ID instances' raw kHDE is smaller than that of OOD instances, i.e.,

$$
\mathbb{E}[E_r(\mathcal{N}_k(\nu_{id}))] < \mathbb{E}[E_r(\mathcal{N}_k(\nu_{ood}))]
$$

.

Proof. In order to facilitate the comparison between $\mathbb{E}[E_r(\mathcal{N}_k(\nu_{id}))]$ and $\mathbb{E}[E_r(\mathcal{N}_k(\nu_{ood}))]$, we introduce a set of notations pertaining to the dot product of logits.

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{x}_i\sim P_{\text{id}}(\mathbf{x})}[\mathbf{x}_i^\top\mathbf{x}_i] = \mathbb{E}^{\text{id}}_{\mathbf{x}_i,\mathbf{x}_i}, \\
&\mathbb{E}_{\mathbf{x}_i\sim P_{\text{ood}}(\mathbf{x})}[\mathbf{x}_i^\top\mathbf{x}_i] = \mathbb{E}^{\text{ood}}_{\mathbf{x}_i,\mathbf{x}_i}, \\
&\mathbb{E}_{\mathbf{x}_i,\mathbf{x}_j\sim P_{\text{id}}(\mathbf{x})}[\mathbf{x}_i^\top\mathbf{x}_j] = \mathbb{E}^{\text{id}}_{\mathbf{x}_i,\mathbf{x}_j}, \\
&\mathbb{E}_{\mathbf{x}_i,\mathbf{x}_j\sim P_{\text{ood}}(\mathbf{x})}[\mathbf{x}_i^\top\mathbf{x}_j] = \mathbb{E}^{\text{ood}}_{\mathbf{x}_i,\mathbf{x}_j}, \\
&\mathbb{E}_{\mathbf{x}_i\sim P_{\text{id}}(\mathbf{x}),\mathbf{x}_j\sim P_{\text{ood}}(\mathbf{x})}[\mathbf{x}_i^\top\mathbf{x}_j] = \mathbb{E}^{\text{io}}_{\mathbf{x}_i,\mathbf{x}_j}.
\end{aligned}
\tag{14}
$$

With these notations in place, the expressions $\mathbb{E}[E_r(\mathcal{N}_k(\nu_{id}))]$ and $\mathbb{E}[E_r(\mathcal{N}_k(\nu_{ood}))]$ can be reformulated as follows:

$$\begin{aligned}\mathbb{E}[E_r(\mathcal{N}_k(\nu_{id}))] = &(p \cdot \mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_i} + (1-p) \cdot \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_i}) \cdot \mathbb{E}[|\mathcal{N}_k(\nu_{id})|] \\ &+ (p^2 \cdot \mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_j} + 2p(1-p) \cdot \mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j} \\ &+ (1-p)^2 \cdot \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_j}) \cdot \mathbb{E}[|\mathcal{N}_k(\nu_{id})|] \cdot \mathbb{E}[t_{\nu_{id}}],\end{aligned} \tag{15}$$

$$\begin{aligned}\mathbb{E}[E_r(\mathcal{N}_k(\nu_{ood}))] = &(q \cdot \mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_i} + (1-q) \cdot \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_i}) \cdot \mathbb{E}[|\mathcal{N}_k(\nu_{ood})|] \\ &+ (q^2 \cdot \mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_j} + 2q(1-q) \cdot \mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j} \\ &+ (1-q)^2 \cdot \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_j}) \cdot \mathbb{E}[|\mathcal{N}_k(\nu_{ood})|] \cdot \mathbb{E}[t_{\nu_{ood}}].\end{aligned} \tag{16}$$

In single-graph scenario, as k increases, $\mathcal{N}_k(\nu_{id})$ shares strong resemblance with $\mathcal{N}_k(\nu_{ood})$ due to overlapping; in multi-graph scenario, $\mathcal{N}_k(\nu_{id})$ and $\mathcal{N}_k(\nu_{ood})$ have similar graph structure. Thus, we have

$$\mathbb{E}[t_{\nu_{id}}] = \mathbb{E}[t_{\nu_{ood}}] = t.$$

Given Assumption 3, we have

$$\mathbb{E}^{id}_{\mathbf{x}_i\mathbf{x}_i} = \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_i}$$

.

Given Assumption 2, we have

$$\mathbb{E}[|\mathcal{N}_k(\nu_{id})|] = \mathbb{E}[|\mathcal{N}_k(\nu_{ood})|] = \mathbb{E}[|\mathcal{N}_k(\nu)|]$$

.

Hence,

$$\begin{aligned}&\mathbb{E}[E_r(\mathcal{N}_k(\nu_{id}))] - \mathbb{E}[E_r(\mathcal{N}_k(\nu_{ood}))] \\ &= -t \cdot \mathbb{E}[|\mathcal{N}_k(\nu)|] \cdot ((p^2 - q^2) \cdot \mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_j} \\ &\quad + (2p(1-p) - 2q(1-q))\mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j} \\ &\quad + ((1-p)^2 - (1-q)^2) \cdot \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_j}) \\ &= -t \cdot \mathbb{E}[|\mathcal{N}_k(\nu)|] \cdot (p-q) \cdot ((p+q)(\mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_j} - \mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j}) \\ &\quad + (2-p-q) \cdot (\mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j} - \mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_j}))\end{aligned} \tag{17}$$

Given Assumption 4, we have $\mathbb{E}^{id}_{x_i,x_j} > \mathbb{E}^{io}_{x_i,x_j}$.

Given the Lemma 1, we have $\mathbb{E}^{io}_{x_i,x_j} = \mathbb{E}^{ood}_{x_i,x_j}$.

Thus Eq. 17 < 0 □

A.3. Proof of Theorem 2

Theorem 2: For an arbitrary node ν in a graph $\mathcal{G}$, as k increases, $E_r(\mathcal{N}_k(\nu))$ will converge to the energy of the largest connected subgraph $\mathcal{G}_\nu$ containing ν, namely $E_r(\mathcal{N}_k(\nu)) = E_r(\mathcal{G}_\nu)$; if $\mathcal{G}$ is connected, $E_r(\mathcal{N}_k(\nu))$ will converge to the classic Dirichlet energy of $\mathcal{G}$, i.e. $E_r(\mathcal{N}_k(\nu)) = E_r(\mathcal{G})$.

Proof. This theorem is intuitively evident. To ensure rigorousness, we present a proof by contradiction.

Case 1: when $\mathcal{G}$ is a connected graph. We assume $E_r(\mathcal{G}) \neq E_r(\mathcal{N}_k(\nu))$ when k is sufficiently large.

If $E_r(\mathcal{G}) > E_r(\mathcal{N}_k(\nu))$, there is a node ν' such that $\nu' \in \mathcal{G}$ but $\nu' \notin \mathcal{N}_k(\nu)$, contradictory to k is sufficiently large.

If $E_r(\mathcal{G}) < E_r(\mathcal{N}_k(\nu))$, there is a node ν' such that $\nu' \notin \mathcal{G}$ but $\nu' \in \mathcal{N}_k(\nu)$, contradictory to $\mathcal{N}_k(\nu) \subseteq \mathcal{G}$.

Case 2: when $\mathcal{G}$ is not a connected graph. We assume $E_r(\mathcal{N}_k(\nu)) \neq E_r(\mathcal{G}_\nu)$ when k is sufficiently large.

If $E_r(\mathcal{G}_\nu) > E_r(\mathcal{N}_k(\nu))$, there is a node ν' such that $\nu' \in \mathcal{G}_\nu$ but $\nu' \notin \mathcal{N}_k(\nu)$, contradictory to k is sufficiently large.

If $E_r(\mathcal{G}_\nu) < E_r(\mathcal{N}_k(\nu))$, there is a node ν' such that $\nu' \notin \mathcal{G}_\nu$ but $\nu' \in \mathcal{N}_k(\nu)$, contradictory to $\mathcal{N}_k(\nu) \subseteq \mathcal{G}_\nu$. □

A.4. Proof of Theorem 3

Proof. First, the expected GDE of a node ν is $\mathbb{E}[E(\mathcal{N}_k(\nu))]$

$$\begin{aligned}&= \frac{1}{2} \cdot \mathbb{E}[\sum_{i,j \in \mathcal{N}_k(\nu)} a_{ij} b_i || \frac{\mathbf{x}_i}{\sqrt{1+d_i}} - \frac{\mathbf{x}_j}{\sqrt{1+d_j}} ||^2_2] \\ &= \mathbb{E}[\sum_{i \in \mathcal{N}_k(\nu)} b_i] \cdot \mathbb{E}[\mathbf{x}_i^\top \mathbf{x}_i] - \\ &\quad \mathbb{E}[\sum_{i,j \in \mathcal{N}(\nu)} \mathbf{x}_i^\top \mathbf{x}_j] \cdot \mathbb{E}[\sum_{i,j \in \mathcal{N}(\nu)} \frac{a_{ij} b_i}{\sqrt{(1+d_i)(1+d_j)}}]\end{aligned} \tag{18}$$

Similar to the raw kHDE, we have

$$\begin{aligned}&\mathbb{E}\left[\sum_{i,j \in \mathcal{N}_k(\nu)} \frac{a_{ij} b_i}{\sqrt{(1+d_i)(1+d_j)}}\right] \\ &\leq \mathbb{E}\left[\sum_{i,j \in \mathcal{N}_k(\nu)} \frac{a_{ij} b_i}{2} \cdot \left(\frac{1}{1+d_i} + \frac{1}{1+d_j}\right)\right] \\ &= \frac{1}{2}\left(\mathbb{E}\left[\sum_{i,j \in \mathcal{N}_k(\nu)} \frac{a_{ij} b_i}{1+d_i}\right] + \mathbb{E}\left[\sum_{i,j \in \mathcal{N}_k(\nu)} \frac{a_{ij} b_i}{1+d_j}\right]\right)\end{aligned} \tag{19}$$

As

$$\begin{aligned}&\mathbb{E}\left[\sum_{i,j \in \mathcal{N}_k(\nu)} \frac{a_{ij} b_i}{1+d_i}\right] \\ &= \mathbb{E}\left[\sum_{i \in \mathcal{N}_k(\nu)} b_i\right] \\ &= \mathbb{E}[|\mathcal{N}_k(\nu)|] \cdot \mathbb{E}\left[\mathbb{E}_{i \in \mathcal{N}_k(\nu)}[b_i]\right],\end{aligned} \tag{20}$$

and,

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}b_i}{1+d_j}\right]\\
&=\mathbb{E}\left[\sum_{i\in\mathcal{N}_k(\nu)}\left(b_i\cdot\sum_{j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{1+d_j}\right)\right]\\
&=\mathbb{E}\left[(\sum_{i\in\mathcal{N}_k(\nu)}\sum_{j\in\mathcal{N}_k(\nu)}\frac{a_{ij}}{1+d_j})\cdot\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[b_i]\right]\\
&=\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot\mathbb{E}\left[\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[b_i]\right].
\end{aligned}\tag{21}$$

We have Eq. 19

$$=\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot\mathbb{E}\left[\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[b_i]\right].\tag{22}$$

For the lower bound, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}b_i}{\sqrt{(1+d_i)(1+d_j)}}\right]\\
&\geq\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{2a_{ij}b_i}{(1+d_i)+(1+d_j)}\right]\\
&\geq\mathbb{E}\left[\sum_{i,j\in\mathcal{N}_k(\nu)}\frac{a_{ij}b_i}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\right]\\
&=\mathbb{E}\left[\frac{1}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\sum_{i\in\mathcal{N}_k(\nu)}b_i\cdot(1+d_i)\right]\\
&\geq\mathbb{E}\left[\frac{1}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\sum_{i\in\mathcal{N}_k(\nu)}b_i\cdot(1+\min_{i\in\mathcal{N}_k(\nu)}(d_i))\right]\\
&=\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot\mathbb{E}\left[\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[b_i]\right]\cdot\mathbb{E}\left[\frac{1+\min_{i\in\mathcal{N}_k(\nu)}(d_i)}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\right].
\end{aligned}\tag{23}$$

For easy writing, we use $\mathbb{E}[b_i]$ to denote $\mathbb{E}_{\nu\in\mathcal{G}}\mathbb{E}_{i\in\mathcal{N}_k(\nu)}[b_i]$. Accordingly, similar to Eq. 13, we have $\mathbb{E}[E(\mathcal{N}_k(\nu))]=$

$$\mathbb{E}[\mathbf{x}_i^\top\mathbf{x}_i]\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot\mathbb{E}[b_i]-\mathbb{E}[\mathbf{x}_i^\top\mathbf{x}_j]\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot\mathbb{E}[b_i]\cdot t,\tag{24}$$

where $t=\mathbb{E}[t_\nu]$ and

$$\frac{1+\min_{i\in\mathcal{N}_k(\nu)}(d_i)}{1+\max_{i\in\mathcal{N}_k(\nu)}(d_i)}\leq t_\nu\leq 1.\tag{25}$$

As weights for nodes in $\mathcal{N}_k(\nu_{id})$ and $\mathcal{N}_k(\nu_{ood})$ are assigned via the identical propagation scheme, $\mathbb{E}[\sum_{i\in\mathcal{N}_k(\nu)}b_i]=\mathbb{E}[\sum_{i\in\mathcal{N}_k(\nu)}b_i]=b$.

Hence,

$$\begin{aligned}
&\mathbb{E}[E(\mathcal{N}_k(\nu_{id}))]-\mathbb{E}[E(\mathcal{N}_k(\nu_{ood}))]\\
&=-t\cdot b\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot((p^2-q^2)\cdot\mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_j}\\
&\quad+(2p(1-p)-2q(1-q))\mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j}\\
&\quad+((1-p)^2-(1-q)^2)\cdot\mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_j})\\
&=-t\cdot b\cdot\mathbb{E}[|\mathcal{N}_k(\nu)|]\cdot(p-q)\cdot((p+q)(\mathbb{E}^{id}_{\mathbf{x}_i,\mathbf{x}_j}-\mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j})\\
&\quad+(2-p-q)\cdot(\mathbb{E}^{io}_{\mathbf{x}_i,\mathbf{x}_j}-\mathbb{E}^{ood}_{\mathbf{x}_i,\mathbf{x}_j}))
\end{aligned}\tag{26}$$

The subsequent steps align with the proof of Theorem 1. Consequently, based on the preceding derivations and established results, we can deduce that

$$\mathbb{E}[E(\mathcal{N}_k(\nu_{id}))]<\mathbb{E}[E(\mathcal{N}_k(\nu_{ood}))].\tag{27}$$

□

B. Experimental Details

B.1. Data Splitting

In Sec. 4.1, we provided the formula to calculate triangleness. For a more comprehensive understanding, we now present the formulas for squareness and cliqueness.

Squareness. We apply the definition of C_4 (Zhang et al., 2008) and define the squareness of a node i as follows,

$$sq_i=\frac{\sum_{m=1}^{d}\sum_{n=m+1}^{d}q_i(m,n)}{\sum_{m=1}^{d}\sum_{n=m+1}^{d}[B_i(m,n)+q_i(m,n)]},\tag{28}$$

where $B_i(m,n)=$

$$(d_m-(1+q_i(m,n)+a_{mn}))+(d_n-(1+q_i(m,n)+a_{mn})).\tag{29}$$

$q_i(m,n)$ represents the number of common neighbors of m and n other than i, and a_{mn} represents the adjacency between neighbors m and n.

Cliqueness. We define the cliqueness of each node i to be the number of maximal cliques in the graph $\mathcal{G}$ that pass through the node i. Specifically, we utilize the classic algorithm to find the maximal cliques MC in a graph (Bron & Kerbosch, 1973), and

$$cq_i=MC(i),\tag{30}$$

where $MC(i)$ represents the number of maximal cliques through node i.

B.2. Thresholds for Topological-Related Metrics

In our experiments, we defined specific thresholds, detailed in Tab. 3, to split nodes into ID instances, OOD instances for training (denoted by OOD_tr), and OOD instances for

testing (denoted by OOD_te) based on values of corresponding metrics. For clear illustration, we use triangleness and the Amazon-Photo dataset as an example:

Nodes for which $tr(i) > 0.47$ are categorized as IDs. Nodes fulfilling $0.28 < tr(i) \leq 0.47$ are designated as OODs for training. Nodes where $tr(i) \leq 0.28$ are designated as OODs for testing.

Remark. These thresholds in Tab. 3 were chosen to ensure the ratio of ID/OOD_tr/OOD_te in each dataset to be approximately 1:1:1.

B.3. Additional Topology Distribution Shifts Settings

We further provide three additional experiment settings, where the other three structural attributes are used for IDs/OODs splitting.

Average Neighbor Degree: The average neighborhood degree of a node i is

$$\text{AND}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} D_j$$

where $\mathcal{N}(i)$ are the 1-hop neighbors of node i and $\mathbf{D}_j$ is the degree of node j which belongs to $\mathcal{N}(i)$. A high AND for a node indicates that, on average, the node's immediate neighbors are highly connected within the network. This suggests that the node is part of a dense part of the graph.

Closeness Centrality: Closeness centrality (Freeman, 1978) of a node i is the reciprocal of the average shortest path distance to i over all $n-1$ reachable nodes.

$$\text{CC}(i) = \frac{n-1}{\sum_{j=1}^{n-1} d(j,i)}$$

where $d(j,i)$ is the shortest-path distance between j and i, and $n-1$ is the number of nodes reachable from i. The closeness is defined so that if a node is close to every other node, then the value is larger than if the vertex is not close to everything else (Metcalf & Casey, 2016).

Core Number: Lastly, we adopt the core number of each node for data-splitting. Mathematically, a k-core of a graph is a maximal connected subgraph in which all vertices have a degree of at least k. The core number of a node is then defined as the highest integer k for which that node is a part of the corresponding k-core. A high core number for a node typically implies that it is embedded within a highly interconnected community.

The thresholds to split IDs, OOD_tr, and OOD_te are provided in Tab. 4.

C. Datasets and Preprocessing

All datasets employed in our study are publicly accessible, serving as standard benchmarks for graph learning model evaluations. We used different sources for data loading, specifically relying on the OGB package for `ogbn-Arxiv` and the Pytorch Geometric package for the others.

- **Amazon-Photo** (McAuley et al., 2015):
  - **Description**: An item co-purchasing network on Amazon. Nodes symbolize products and edges represent that the two products are frequently purchased together. Node labels represent the categories of products.
  - **Specifications**: 7,650 nodes, 238,162 edges, 745 features, and 8 classes.
  - **Data Preparation**: We split IDs, OODs for training, and OODs for testing according to three metrics based on Tab. 3. For ID data, we employed the conventional random splits method (1:1:8 for training/validation/testing) as suggested by Kipf & Welling.
- **Coauthor-CS** (Sinha et al., 2015):
  - **Description**: A coauthor network for the computer science domain. Nodes represent authors and are connected by an edge if the two coauthored a paper. Node features represent paper keywords. The objective is to predict authors' fields of study using paper keyword features.
  - **Specifications**: 18,333 nodes, 163,788 edges, 6,805 features, and 15 classes.
  - **Data Preparation**: Similar to Amazon-Photo, we employed three methods via different thresholds to split OOD data. The ID data adheres to the 1:1:8 random splits convention.
- **Coauthor-Physics** (Shchur et al., 2018):
  - **Description**: A coauthor network for the physics domain. Nodes represent authors and edges indicate co-authorship. Features represent paper keywords and the class denotes authors' fields of study.
  - **Specifications**: 34,493 nodes, 247,962 edges, 8,415 features, and 5 classes.
  - **Data Preparation**: Similar to Coauthor-CS, we employed three metrics to split OOD data via different thresholds. The ID data adheres to the 1:1:8 random splits convention.
- **Twitch-Explicit** (Rozemberczki & Sarkar, 2021):
  - **Description**: A multi-graph dataset of Twitch gaming networks by region. Nodes denote Twitch

Table 3. Thresholds to split ID, OOD for training, and OOD for testing. Thresholds were chosen to ensure the ratio of ID/OOD_tr/OOD_te in each dataset to be approximately 1:1:1.

Dataset	Triangleness		Squareness		Cliqueness	
	ID_Bar	OOD_Bar	ID_Bar	OOD_Bar	ID_Bar	OOD_Bar
Coauthor-CS	0.39	0.18	0.06	0.02	5	3
Amazon-Photo	0.47	0.28	0.14	0.08	52	9
Coauthor-Physics	0.40	0.20	0.07	0.04	8	3

players, and edges show user follow relationships. Node features are embeddings of games played by the users.

  - **Specifications**: Node count varies from 1,912 to 9,498 per subgraph. Edge numbers range from 31,299 to 153,138, with a consistent feature dimension of 2,545. Nodes belong to 2 classes.
  - **Data Preparation**: Subgraph DE is chosen for in-distribution data with the familiar 1:1:8 split. Subgraphs ENGB, ES, FR, and RU are used as OOD data.

- **Ogbn-Arxiv** (Hu et al., 2020):
  - **Description**: A relatively large-scale citation network capturing data from 1960 to 2020. Nodes are papers and labels represent their subject areas. Edges represent citation relationships. Node features are 128-dimensional vectors extracted from titles and abstracts. 40 classes in total.
  - **Data Preparation**: As the way for data-splitting in (Hu et al., 2020) is disabled since we use temporal information for splitting the OOD instances, we adopted a 1:1:8 random split for the in-distribution segment in line with (Wu et al., 2023).

- **Cora** (Sen et al., 2008):
  - **Description**: A citation network where nodes represent scientific publications and edges denote citation links. The goal is to classify each paper into one of several predefined topics.
  - **Specifications**: 2,708 nodes, 5,429 edges, 1,433 features, and spans 7 distinct classes.
  - **Data Preparation**: This dataset is only used in leave-out settings. The in-distribution data splits for training, validation, and testing follow the semi-supervised setting of Kipf & Welling (2017), using the provided standard splits.

Table 4. Thresholds to split ID, OOD for training, and OOD for testing for additional topology distribution shift settings.

	Core (Coauthor-Physics)	AND (Amazon-Photo)	CC(Coauthor-CS)
ID_Bar	6	60	0.17
OOD_bar	4	30	0.14

D. Implementation Details

D.1. Encoder Architectures.

As highlighted in Section 4.1, we predominantly utilize the GCN as our backbone encoder. However, for a more comprehensive insight, we've also incorporated other models, including GAT, JKNet, and MixHop. Below, we outline the specific configurations for each encoder used in our research:

- **GCN** (Kipf & Welling, 2017):
  - Layers: Two `GCNConv` layers
  - Dimensions: Hidden size 64
  - Activation: ReLU
  - With self-loop and batch normalization
- **GAT** (Veličković et al., 2018):
  - Layers: Two `GATConv` layers
  - Dimensions: Hidden size 64
  - Activation: ELU
  - Head numbers configured as [2, 1]
  - With batch normalization
- **MixHop** (Abu-El-Haija et al., 2019):
  - Layers: Two `MixHop` layers with hop number 2
  - Dimensions: Hidden size 64
  - Activation: ReLU
  - With batch normalization
- **JKNet** (Xu et al., 2018):
  - Layers: Two `GCNConv` layers
  - Dimensions: Hidden size 64
  - Activation: ReLU
  - With self-loop, batch normalization, and using max-pooling in the jumping knowledge module

D.2. Evaluation Metrics

In our experiment, we assess OOD data detection performance utilizing three established metrics from the literature.

The **AUROC** (Area Under the Receiver Operating Characteristic curve) measures the area beneath the ROC curve and quantifies a model's ability to distinguish between positive

and negative classes. The ROC curve is plotted by juxtaposing the true positive rate against the false positive rate across various thresholds ranging from 0 to 1. In OOD detection tasks, the AUROC denotes the likelihood that, given a randomly selected ID-OOD pair, the ID sample will have a higher estimation score than the OOD one.

The **AUPR** (Area Under the Precision-Recall curve) is another critical metric, especially illustrative when dealing with class imbalances. It is defined as the area under the Precision-Recall curve. A model that makes perfect predictions has an AUPR of 1.

The **FPR95** is also a prevalent metric which indicates the False Positive Rate when the true positive rate (TPR) is fixed at 95%. In the OOD detection task, this metric captures the likelihood of mistakenly categorizing an out-of-distribution sample as in-distribution, given a 95% TPR.

D.3. Hyper-parameter Configurations.

We detail the default hyper-parameters utilized across all scenarios, as delineated in Tab. 5. These hyper-parameters were determined through a systematic grid search. Notably, for all scenarios, the propagation coefficient, denoted by α, was consistently set to 0.5, and the number of training epochs was set to 200.

D.4. Additional Experiments Results

We further provide additional experimental outcomes to complement our results in Sec. 4.3.

Real Distribution Shifts. For a more comprehensive analysis, we provide detailed results of TopoOOD in each OOD dataset for real distribution shifts. This includes different subgraphs for Twitch and distinct years for Arxiv, as illustrated in Tab. 6 and 7. The average values for AUROC, AUPR, and FPR of TopoOOD on Arxiv should be 87.11, 87.61, and 32.10, respectively. As a result, TopoOOD improves the average AUROC by 12.34%, and reduces the average FPR by 45.33% in Arxiv dataset compared to other GNNSafe++, significantly surpassing GNNSafe++ and all other baseline methods (displayed in Tab. 7).

Topological Distribution Shifts. First, for Triangleness, Squareness, and Cliqueness, we present the results of Amazon-Photo, Coauthor-CS, and Coauthor-Physics based on performance metrics of AUROC/AUPR/FPR95 alongside the in-distribution testing accuracy. These details can be gleaned from Tabs 8, 9 and 10. They serve as a supplement to the insights presented in Tab. 2.

Additionally, we further provided experimental results for three additional topology distribution shift settings, Average Neighbor Degree, Core Number, and Closeness Centrality. In Table 13

Label Distribution Shifts Detailed Experimental results for the label leave-out setting as been provided in Tabs 12 and 11.

Ablation Studies Hyper-parameter evaluations are incorporated in Figs. 6 and 8. For k, e_{in}, e_{out}, and λ, we discuss their influence on TopoOOD in the main text.

Additionally, we investigate the relationship between AUROC and the learning rate, lr by conducting experiments on the Arxiv dataset as well as three synthetic settings of the Coauthor-CS dataset. Specifically, we varied the learning rate, lr, across a set of values: $\{0.001, 0.003, 0.01, 0.03, 0.1\}$. The outcomes, as depicted in Fig. 7, suggest that the performance of TopoOOD remains robust across a reasonable range of learning rates.

Table 5. Hyper-parameters of TopoOOD in our experiments. As mentioned in the maintext, k denotes the propagation steps; e_{in} and e_{in} denote the constraints for regularization loss; λ denotes the regularization weight in the final objective function; lr is the learning rate.

Dataset	k	e_{in}	e_{out}	λ	lr
Twitch	10	0	5	1	0.01
Arxiv	20	0	20	0.003	0.003
Amazon-Triangleness	17	0	1	5	0.003
Amazon-Squareness	5	0	3	1	0.01
Amazon-Cliqueness	10	0	1	5	0.003
Coauthor-Triangleness	10	0	1	5	0.01
Coauthor-Squareness	10	0	1	5	0.01
Coauthor-Cliqueness	10	0	5	5	0.003
Coauthor-Physics-Triangleness	5	0	1	10	0.01
Coauthor-Physics-Squareness	50	0	3	5	0.01
Coauthor-Physics-Cliqueness	50	0	3	5	0.01

Table 6. OOD detection performance by $\mathrm{AUROC}(\uparrow)/\mathrm{AUPR}(\uparrow)/\mathrm{FPR95}(\downarrow)$ on OOD sub-graphs ES, FR and RU of Twitch dataset.

Model	Twitch-ES			Twitch-FR			Twitch-RU		
	AUROC	AUPR	FPR95	AUROC	AUPR	FPR95	AUROC	AUPR	FPR95
MSP	37.72	53.08	98.09	21.82	38.27	99.25	41.23	56.06	95.01
ODIN	83.83	80.43	33.28	59.82	64.63	92.57	58.67	72.58	93.98
Mahalanobis	45.66	58.82	95.48	40.40	46.69	95.54	55.68	66.42	90.13
GKDE	48.70	61.05	95.37	49.19	52.94	95.04	46.48	62.11	95.62
GPN	53.00	64.24	95.05	51.25	55.37	93.92	50.89	65.14	99.93
OE	55.97	69.49	94.94	45.66	54.03	95.48	55.72	70.18	95.07
KNN	80.95	86.03	73.26	56.75	60.46	96.46	55.58	67.06	95.01
GNNSafe++	94.54	97.17	44.06	93.45	95.44	51.06	98.10	98.74	5.59
TopoOOD	99.73	99.89	0.00	99.21	99.38	0.14	99.97	99.99	0.00

Table 7. Out-of-distribution detection performance measured by $\mathrm{AUROC}(\uparrow)/\mathrm{AUPR}(\uparrow)/\mathrm{FPR95}(\downarrow)$ on OOD datasets of papers published in 2018, 2019 and 2020, respectively, on Arxiv. GPN reports the out-of-memory issue with a 24GB GPU.

Model	Arxiv-2018			Arxiv-2019			Arxiv-2020		
	AUROC	AUPR	FPR95	AUROC	AUPR	FPR95	AUROC	AUPR	FPR95
MSP	61.66	70.63	91.67	63.07	66.00	90.82	67.00	90.92	89.28
ODIN	53.49	63.06	100.0	53.95	56.07	100.0	55.78	87.41	100.0
Mahalanobis	57.08	65.09	93.69	56.76	57.85	94.01	56.92	85.95	95.01
GKDE	56.29	66.78	94.31	57.87	62.34	93.97	60.79	88.74	93.31
GPN	-	-	-	-	-	-	-	-	-
OE	67.72	75.74	86.67	69.33	72.15	85.52	72.35	92.57	83.28
GNNSafe++	70.4	78.62	81.47	72.16	75.43	79.33	81.75	95.57	71.50
KNN	43.76	55.27	98.09	44.27	48.40	97.84	41.44	79.69	98.77
TopoOOD	80.95	79.28	41.30	89.14	86.09	28.71	91.25	97.47	26.30

Table 8. Evaluation results for different OOD detection baselines on Amazon-Photo datasets.

Model	Amazon-Photo-Triangleness				Amazon-Photo-Squareness				Amazon-Photo-Cliqueness			
	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC
OE	86.66	82.35	52.03	75.15	91.58	88.94	46.08	84.91	87.37	84.53	43.73	87.14
GNNSafe++	70.47	79.48	99.73	66.40	80.41	85.31	98.94	88.80	95.87	91.23	9.09	92.69
MSP	80.29	70.64	67.02	84.60	87.40	82.27	50.69	91.60	87.63	82.26	50.61	92.74
KNN	48.49	47.37	98.44	86.80	70.59	71.32	89.87	92.12	72.56	67.29	82.70	92.99
GPN	62.98	59.64	94.52	87.69	61.75	54.30	88.53	92.60	52.64	50.48	97.25	48.71
Mahalanobis	82.21	70.79	43.74	84.28	75.28	63.29	60.62	89.70	29.72	36.69	99.80	92.30
ODIN	19.33	30.42	96.99	82.55	49.43	45.29	100.00	91.31	11.86	28.66	99.51	92.55
GKDE	74.94	64.55	69.52	74.17	66.23	56.92	85.49	42.57	63.91	58.67	89.85	49.00
TopoOOD	91.03	84.93	19.68	82.87	89.05	80.83	52.15	84.34	93.57	79.63	6.43	91.82

Table 9. Evaluation results for different OOD detection baselines on Coauthor-CS datasets.

Model	Coauthor-CS-Triangleness				Coauthor-CS-Squareness				Coauthor-CS-Cliqueness			
	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC
OE	73.42	68.79	79.07	78.10	75.87	73.17	76.34	84.13	69.96	70.09	84.19	89.18
GNNSafe++	54.83	63.60	100.00	83.54	70.91	75.70	99.89	83.79	93.36	89.57	21.64	88.36
MSP	71.31	64.23	84.97	85.95	78.33	74.26	75.04	91.04	64.44	64.59	86.30	90.69
KNN	61.17	55.51	93.38	89.48	63.75	58.96	94.38	92.05	51.92	49.66	91.57	90.75
GPN	57.47	53.00	96.09	84.51	89.09	90.75	65.12	89.26	65.49	58.35	78.94	76.59
Mahalanobis	86.06	82.74	52.56	82.89	79.41	77.39	70.16	90.53	60.29	63.07	95.65	58.23
ODIN	49.23	44.34	100.00	84.55	49.25	46.52	100.00	90.73	49.55	49.13	100.00	90.14
GKDE	70.07	59.19	80.86	86.27	74.60	68.22	76.04	90.63	57.12	57.22	92.19	88.12
TopoOOD	95.73	97.11	27.56	88.29	97.94	98.42	3.93	92.17	99.96	99.89	0.10	86.72

Table 10. Evaluation results for different OOD detection baselines on Coauthor-Physics datasets.

Model	Coauthor-Physics-Triangleness				Coauthor-Physics-Squareness				Coauthor-Physics-Cliqueness			
	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC
MSP	56.97	55.96	90.61	93.58	65.74	55.17	88.30	94.97	83.98	88.47	77.72	96.32
KNN	40.35	48.05	97.14	93.74	47.06	42.98	97.15	95.01	72.07	78.21	89.39	96.28
GPN	39.95	51.32	100.00	93.93	55.00	51.99	97.18	94.06	74.24	72.20	61.02	95.35
ODIN	49.95	54.84	100.00	93.57	32.20	50.63	100.00	96.33	41.32	40.08	100.00	94.75
GKDE	52.33	51.61	87.85	94.32	58.86	47.38	87.72	95.53	79.39	84.36	75.58	95.90
Mahalanobis	87.50	86.87	42.39	92.10	87.06	82.02	44.31	88.67	41.87	55.35	99.16	95.30
OE	79.74	85.27	79.27	75.50	80.23	80.90	79.07	84.26	92.49	93.79	33.62	95.62
GNNSafe++	56.97	72.57	100.00	92.30	76.41	81.63	100.00	88.46	95.49	93.11	14.12	96.05
TopoOOD	88.60	93.45	88.15	92.28	96.09	97.75	0.00	88.31	100.00	100.00	0.00	95.74

Table 11. Evaluation results for different OOD detection baselines measured by AUROC(↑) / AUPR(↑) / FPR95(↓) on Cora (Sen et al., 2008) and Amazon-Photo with the OOD type **label leave-out**.

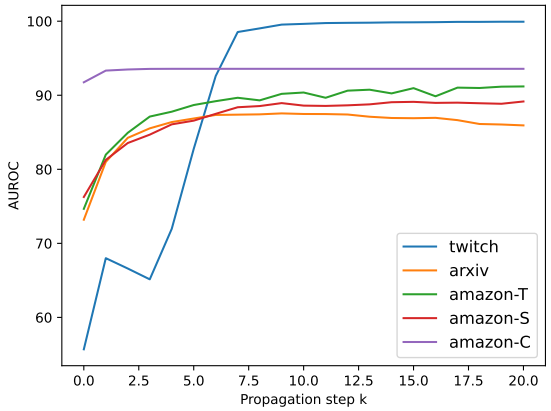
Model	Cora				Amazon			
	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC
MSP	91.36	78.03	34.99	88.92	93.97	91.32	26.65	95.76
ODIN	49.80	24.27	100.0	88.92	65.97	57.80	90.23	96.08
KNN	94.68	93.47	19.87	88.91	83.88	87.23	68.84	94.07
Mahalanobis	67.62	42.31	90.77	88.92	73.25	66.89	74.30	95.76
GKDE	57.23	27.50	88.95	89.87	65.58	65.20	96.87	89.37
GPN	90.34	77.40	37.42	91.46	92.72	90.34	37.16	90.07
OE	89.47	77.01	46.55	87.97	95.39	92.53	17.72	95.72
GNNSAFE++	92.75	82.64	34.08	91.46	97.51	97.07	6.18	95.84
TopoOOD	93.50	93.09	6.67	89.25	100.00	100.00	0.00	94.45

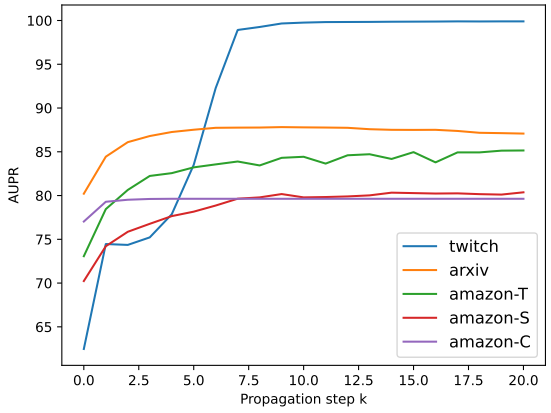
Table 12. Evaluation results for different OOD detection baselines measured by AUROC(↑) / AUPR(↑) / FPR95(↓) on Coauthor-CS and Coauthor-Physics with the OOD type **label leave-out**.

Model	Coauthor-CS				Coauthor-Physics			
	AUROC	AUPR	FPR95	ID ACC	AUROC	AUPR	FPR95	ID ACC
MSP	94.88	97.99	23.81	95.18	93.06	98.78	38.22	98.00
ODIN	51.44	74.79	100.0	95.15	50.56	86.63	100.00	98.02
Mahalanobis	85.36	93.61	45.41	95.19	39.64	86.87	100.00	74.89
GKDE	61.15	81.39	94.60	89.05	83.24	96.48	60.78	98.00
GPN	93.24	97.55	34.78	91.68	80.68	95.78	67.12	97.80
OE	96.04	98.50	18.17	95.10	92.58	98.66	37.57	98.00
KNN	86.05	90.58	48.73	93.25	85.13	97.04	59.62	98.02
GNNSAFE++	97.89	99.24	9.43	95.24	97.24	99.44	11.57	98.04
TopoOOD	99.32	98.94	0.92	92.86	99.92	99.99	0.14	97.69

	AND (Amazon-Photo)				CC (Coauthor-CS)				Core Number (Coauthor-Physics)			
	AUROC↑	AUPR↑	FPR↓	ID ACC	AUROC↑	AUPR↑	FPR↓	ID ACC	AUROC↑	AUPR↑	FPR↓	ID ACC
TopoOOD	87.56	85.12	12.88	84.01	97.35	99.79	4.91	89.92	99.99	99.99	0.06	96.21
GNNSafe++	93.96	97.91	74.46	86.60	95.51	99.71	15.13	91.03	94.83	95.94	18.52	96.14
GPN	56.56	74.11	91.16	81.21	50.46	94.84	85.48	74.83	64.87	73.69	91.87	83.29
ODIN	32.11	57.95	98.73	86.17	38.58	93.00	95.71	91.66	12.21	43.89	99.98	96.31
GKDE	58.04	72.55	94.46	64.25	43.14	94.66	82.21	90.47	76.70	83.46	69.93	96.21
Mahalanobis	38.28	64.00	98.05	85.94	25.27	92.71	100.00	91.64	38.55	57.23	98.74	96.31
KNN	71.61	86.42	95.58	89.17	68.15	97.71	89.57	91.92	72.31	81.54	95.61	96.34
MSP	70.07	81.15	75.88	88.44	58.45	96.58	78.73	91.74	87.31	91.91	59.00	96.37
OE	85.55	90.42	42.25	84.24	90.77	99.51	37.01	89.56	95.26	96.93	22.82	95.49

Table 13. Evaluation results for different OOD detection baselines measured by AUROC(↑) / AUPR(↑) / FPR95(↓) on three additional topology distribution shifts.





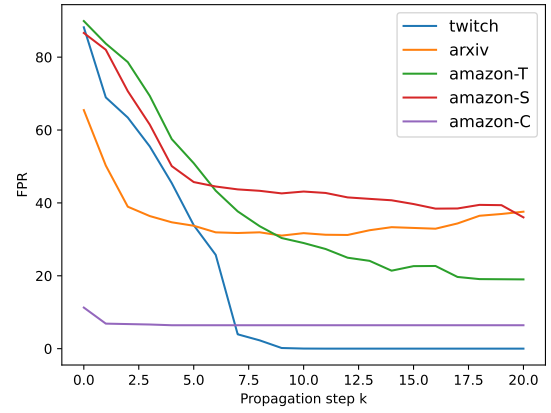


Figure 6. Hyper-parameter study for propagation steps k. The performance of TopoOOD on Twitch, Arxiv, and the three OOD types of Amazon-Photo are plotted as the propagation steps increase.

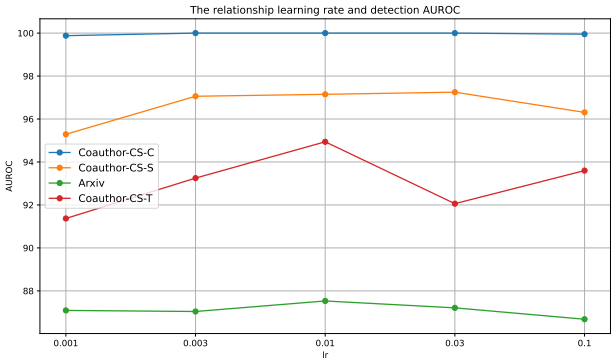
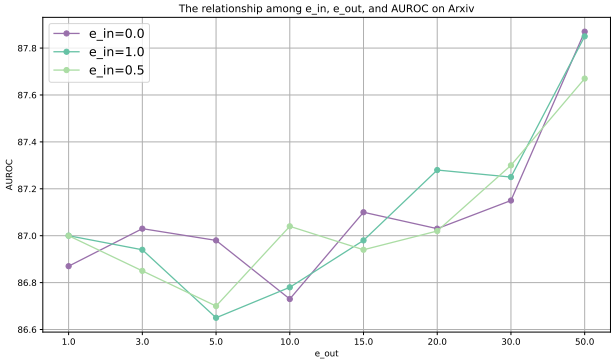


Figure 7. Hyper-parameter study for the model learning rate lr. The results for the Arxiv and the three OOD types of Coauthor-CS are plotted as the learning rate equal to $0.001, 0.003, 0.01, 0.03, 0.1$.



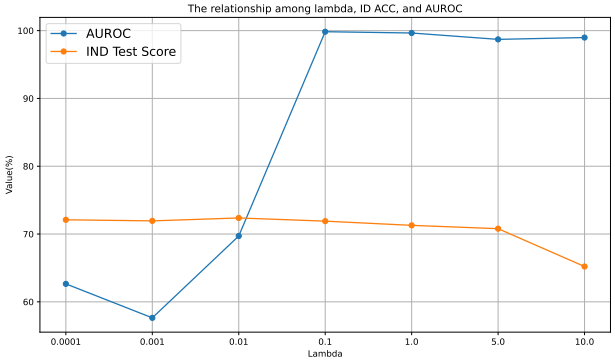


Figure 8. Impact of margin hyper-parameters e_{in}, e_{out} (the upper one) and regularization weight λ (the lower one). We use the Twitch dataset to display the influence of λ, and the Arxiv dataset to show how e_{in} and e_{out} influence performance.