GGADBENCH: A COMPREHENSIVE BENCHMARK FOR GENERALIST GRAPH ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generalist Graph Anomaly Detection (GGAD) aims to detect anomalies across different graphs with a unified model and has developed rapidly in recent years. However, existing methods are evaluated on different datasets and under different transfer settings and protocols, making their performance difficult to compare reliably. We present GGADBench, the first comprehensive benchmark for GGAD, covering **30** methods and **22** graphs with both organic and injected anomalies under a unified evaluation protocol. GGADBench evaluates methods in terms of overall effectiveness, cross-graph transferability, anomaly retrieval, and efficiency. We further introduce anomaly residual polarity, a dataset-level measure that characterizes how anomalies differ from their local neighborhoods. Our evaluation reveals three key findings: (i) prompt-tuned graph foundation models are competitive with specialized GGAD methods on organic targets, whereas strong AUROC does not necessarily translate into effective anomaly retrieval; (ii) cross-graph performance is driven more by the target graph than by the source anomaly type or source quantity; and (iii) residual-based rankings can reverse on targets with negative ARP. These findings highlight the importance of transferable feature construction and target-adaptive anomaly modeling for future GGAD. The code is available at https://anonymous.4open.science/r/GGADBench-B8FD.

1 INTRODUCTION

Graphs are widely used to model complex systems, such as social networks Fan et al. (2019), financial transactions Weber et al. (2019a), knowledge engineering Li et al. (2025) and e-commerce platforms Ying et al. (2018). Graph Anomaly Detection (GAD) aims to identify abnormal nodes that exhibit unusual attributes or connections, and supports many real-world applications, including financial crime detection Weber et al. (2019a), malicious account detection Dou et al. (2020b), and spam review detection Rayana & Akoglu (2015a). Many recent GAD methods build on graph neural networks and use both node attributes and graph structure to detect anomalies Ding et al. (2019); Qiao & Pang (2023). However, these methods are typically designed for a single graph, with a separate detector trained for each dataset Liu et al. (2024); Niu et al. (2024), and do not explicitly target generalization across different graphs.

Generalist Graph Anomaly Detection (GGAD) seeks a *one-for-all* detector that learns general anomaly knowledge across graphs and can be applied to unseen graphs. As shown in Figure 1, GGAD has grown rapidly since 2024: early studies such as ARC Liu et al. (2024) and UNPrompt Niu et al. (2024) were followed by AnomalyGFM Qiao et al. (2025), IA-GGAD Zhang et al. (2025), and ZeroGAD Zheng et al. (2025), with a much larger number of methods appearing in 2026. However, these methods are developed and evaluated under different settings, making fair comparison difficult and leaving the overall progress of GGAD unclear.

A unified and comprehensive benchmark is therefore needed, yet none exists for GGAD, which obscures a clear understanding of progress in this field. ❶ **Dataset preparation.** Existing studies use different graph collections, preprocessing steps, and anomaly construction settings, making their results difficult to compare directly. ❷ **Experimental protocol.** The choice and number of source graphs, source–target combinations, and target-side access vary widely across studies, leading to inconsistent transfer settings. ❸ **Performance evaluation.** Existing studies lack a consistent and comprehensive evaluation standard. Most focus on detection effectiveness, typically measured by

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

AUROC, while anomaly retrieval, cross-graph transferability, and computational efficiency are often overlooked.

In this work, we establish GGADBench, the first comprehensive benchmark for generalist graph anomaly detection. Our main contributions are summarized as follows:

- **First comprehensive GGAD benchmark.** GGADBench provides a unified evaluation of 30 methods, including 15 GGAD methods, 10 conventional GAD baselines, and 5 graph foundation models, across 22 graphs, including 14 with organic anomalies and 8 with injected anomalies. All methods are evaluated under consistent data processing and transfer protocols.
- **Multi-faceted evaluation and analysis.** Beyond conventional AUROC-based comparison, GGADBench examines anomaly retrieval, how cross-graph performance varies with source and target choices, and computational efficiency on large graphs. We further introduce *Anomaly Residual Polarity* (ARP), a dataset-level measure that characterizes whether anomalies tend to deviate from or blend into their local neighborhoods.
- **Key findings.** Our unified evaluation reveals that: ***(i)*** On organic targets, the two strongest GGAD methods both avoid fixed PCA/SVD-based feature alignment, suggesting that cross-graph feature semantics may deserve more attention than dimensional compatibility alone; ***(ii)*** Prompt-tuned graph foundation models are competitive with GGAD methods on organic targets; ***(iii)*** High AUROC does not always translate into effective anomaly retrieval on organic targets; ***(iv)*** Cross-graph performance is driven more by the target graph than by the source anomaly type or source quantity; ***(v)*** Residual-based rankings can reverse on negative-ARP targets, showing that a fixed anomaly criterion does not transfer consistently across graphs. These findings highlight the importance of transferable feature representation and target-adaptive anomaly modeling for future GGAD.

2 PRELIMINARIES

Attributed Graph. An attributed graph is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, where $\mathcal{V} = \{v_1, \ldots, v_n\}$ is the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set, and $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the node-attribute matrix. The graph structure is represented by an adjacency matrix $\mathbf{A} \in \{0,1\}^{n \times n}$, where $\mathbf{A}_{ij} = 1$ iff $(v_i, v_j) \in \mathcal{E}$.

Generalist Graph Anomaly Detection. Let $\mathcal{S} = \{\mathcal{D}^{(1)}, \ldots, \mathcal{D}^{(M)}\}$ denote a collection of source datasets, where $\mathcal{D}^{(m)} = (\mathcal{G}^{(m)}, \mathbf{y}^{(m)})$ and $\mathbf{y}^{(m)} \in \{0,1\}^{n_m}$ denotes the node-level anomaly labels. Let $\mathcal{D}^{(t)} = (\mathcal{G}^{(t)}, \mathbf{y}^{(t)})$ be an unseen target dataset such that $\mathcal{D}^{(t)} \notin \mathcal{S}$. The source and target graphs may differ in their sizes, structures, and attribute spaces. GGAD aims to learn a shared anomaly detector f_θ from the source datasets and generalize it to the unseen target graph. Given $\mathcal{G}^{(t)}$, the detector produces an anomaly-score vector $\mathbf{s}^{(t)} = f_\theta(\mathcal{G}^{(t)}) \in \mathbb{R}^{n_t}$, where a larger $s_i^{(t)}$ indicates that node $v_i^{(t)}$ is more likely to be anomalous. The target labels $\mathbf{y}^{(t)}$ are excluded from source-side training and are used for evaluation, subject to the target-side access allowed by the specific zero-shot or few-shot protocol.

Anomaly Residual Polarity. Let $\mathbf{x}_i = \mathbf{X}_{i,:}$ denote the attribute vector of node v_i. Its neighborhood residual is defined as

$$r_i = \left\| \mathbf{x}_i - \frac{1}{|\mathcal{N}(i)|} \sum_{v_j \in \mathcal{N}(i)} \mathbf{x}_j \right\|_2,$$

where $\mathcal{N}(i)$ denotes the set of neighbors of v_i. The residual r_i measures the attribute deviation of v_i from its neighborhood. Let $\mathcal{V}_a = \{v_i : y_i = 1\}$ and $\mathcal{V}_n = \{v_i : y_i = 0\}$ denote the anomalous and normal node sets, respectively. The *Anomaly Residual Polarity* (ARP) of graph $\mathcal{G}$ is defined as

$$\mathrm{ARP}(\mathcal{G}) = \Pr(r_a > r_n) - \Pr(r_a < r_n), \qquad v_a \sim \mathrm{Unif}(\mathcal{V}_a), \quad v_n \sim \mathrm{Unif}(\mathcal{V}_n),$$

where r_a and r_n are the neighborhood residuals of v_a and v_n,respectively. By definition, $\mathrm{ARP}(\mathcal{G}) \in [-1, 1]$. A positive ARP indicates that anomalous nodes tend to have larger neighborhood residuals than normal nodes, whereas a negative ARP indicates the opposite; $\mathrm{ARP}(\mathcal{G}) = 0$ indicates no consistent ordering between the two groups.

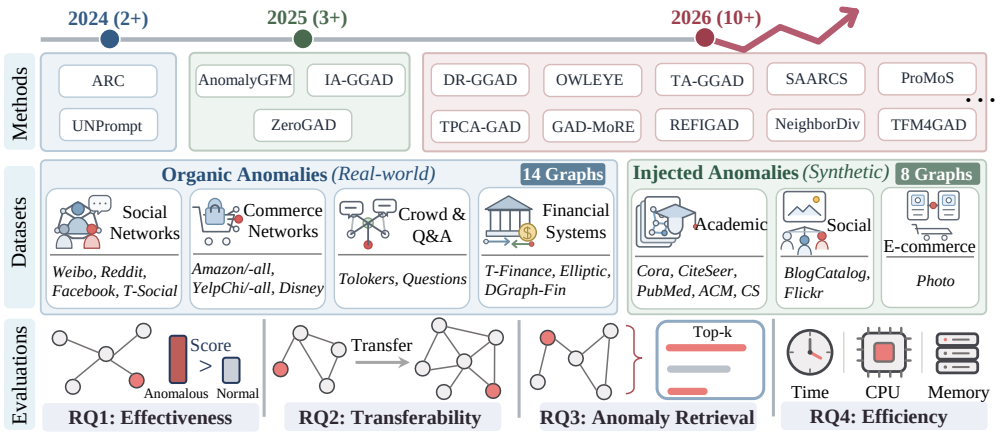

Figure 1: Overview of GGADBench, including the rapid development of GGAD methods, benchmark graphs with organic and injected anomalies, and four evaluation dimensions.

3 The Setup of GGADBench

Figure 1 provides an overview of GGADBench, including the benchmark models, datasets, and evaluation dimensions. In this section, we describe the benchmark models (Section 3.1), datasets (Section 3.2), and research questions (Section 3.3).

3.1 Benchmark Models

GGADBench evaluates ***30*** methods from three categories: conventional GAD baselines, generalist GAD methods, and Graph Foundation Models (GFMs). More details are provided in Appendix A.

Conventional Baselines. Conventional GAD methods are typically developed and trained for individual graphs rather than for cross-graph generalization. We include 10 representative baselines for comparison. Specifically, DOMINANT Ding et al. (2019), CoLA Liu et al. (2021b), and TAM Qiao & Pang (2023) are unsupervised GAD methods, while BWGNN Tang et al. (2022) and GHRN Gao et al. (2023b) are supervised GAD methods. We also include MLP, GCN Kipf & Welling (2017), and GAT Veličković et al. (2018) as standard neural baselines, together with RF-Graph and XGB-Graph Tang et al. (2023), two tree-based baselines with neighborhood aggregation.

Generalist GAD Methods. Generalist GAD moves beyond the one-model-for-one-dataset paradigm by using a shared detector across different graphs. We include 15 representative GGAD methods: ARC Liu et al. (2024), UNPrompt Niu et al. (2024), AnomalyGFM Qiao et al. (2025), ZeroGAD Zheng et al. (2025), IA-GGAD Zhang et al. (2025), DR-GGAD Fu et al. (2026), OWLEYE Zheng et al. (2026), TA-GGAD Zhang et al. (2026), ProMoS Xu et al. (2026), SAARCS Liu et al. (2026d), TPCA-GAD Yao et al. (2026), GAD-MoRE Zhao et al. (2026), NeighborDiv Wei et al. (2026), REFIGAD Liu et al. (2026b), and TFM4GAD Liu et al. (2026c).

Graph Foundation Models. Graph Foundation Models (GFMs) aim to learn transferable graph representations for different graphs and downstream tasks. We include 5 representative GFMs: GraphPrompt Liu et al. (2023), SAMGPT Yu et al. (2025), MDGFM Wang et al. (2025a), BRIDGE Yuan et al. (2025), and GILT Ma et al. (2025). All GFMs are evaluated under the same cross-graph protocol as the GGAD methods.

3.2 Benchmark Datasets

GGADBench includes two types of graph anomalies: *organic anomalies* from real-world datasets and *injected anomalies* generated synthetically. In total, we evaluate ***22*** attributed graphs, including 14 organic-anomaly graphs and 8 injected-anomaly graphs. More details are provided in Appendix B.

Organic Anomaly Datasets. We include 14 graphs with naturally occurring anomaly labels from real-world applications: Weibo Zhao et al. (2020); Liu et al. (2022b), Reddit Kumar et al. (2019); Liu et al. (2022b), Facebook Xu et al. (2022), Amazon and Amazon-all McAuley & Leskovec (2013); Dou et al. (2020a), YelpChi and YelpChi-all Rayana & Akoglu (2015b); Dou et al. (2020a), Tolokers and Questions Platonov et al. (2023), T-Finance Tang et al. (2022), Elliptic Weber et al. (2019b), DGraph-Fin Huang et al. (2022), T-Social Tang et al. (2022), and Disney Sánchez et al. (2013). These graphs cover social networks, e-commerce, crowdsourcing, and financial systems, with anomaly labels defined by their original applications.

Injected Anomaly Datasets. We further include 8 graphs with synthetically injected anomalies: Cora, Citeseer, and PubMed Sen et al. (2008), ACM Tang et al. (2008), BlogCatalog and Flickr Ding et al. (2019); Tang & Liu (2009), and CS and Photo Shchur et al. (2018). Following standard GAD settings Ding et al. (2019); Liu et al. (2022b), structural and/or attribute anomalies are injected into the original graphs to construct anomalous nodes.

3.3 Research Questions

We systematically design the GGADBench to comprehensively evaluate the existing GGAD algorithms and inspire future research. In particular, we aim to investigate the following research questions.

RQ1: How much progress has been made by the existing GGAD algorithms?

Motivation and Experiment Design. Existing GGAD methods are developed under inconsistent training sources, target splits, and label budgets, making performance comparisons unfair. Given the fair data and evaluation environment of GGADBench, **RQ1** aims to understand the strengths and weaknesses of GGAD methods and whether purpose-built designs genuinely outperform classic baselines and graph foundation models. To achieve this, all methods are trained on a fixed set of real-world source graphs spanning distinct domains (Facebook for social networks, T-Finance for financial systems, and Amazon for e-commerce) and evaluated on each unseen real-world target graph, where every method shares identical graph reading, preprocessing, splitting, and scoring, and each source graph is excluded when it serves as the target. We report AUROC averaged over 11 random seeds (0 to 10).

RQ2: Does performance depend more on the source or on the target?

Motivation and Experiment Design. Cross-graph performance may vary with both the source graphs used for learning and the unseen target graph being evaluated. To examine their respective effects, we vary the source anomaly type (*organic* vs. *injected*), the number of source graphs (*single* vs. *multi-source*), and evaluate each setting on both organic and injected unseen targets. This results in four source–target combinations: organic→organic, organic→injected, injected→organic, and injected→injected. For organic sources, we use Facebook in the single-source setting and {Facebook, T-Finance, Amazon} in the multi-source setting; for injected sources, we use Cora and {Cora, Citeseer, ACM, BlogCatalog}, respectively. We report the mean AUROC over 11 random seeds and compare the performance variation caused by changing the source configuration with that across target graphs.

RQ3: How precisely do GGAD methods retrieve anomalies?

Motivation and Experiment Design. AUROC measures the overall ranking performance but is less sensitive to whether anomalies are ranked near the top. This distinction is important in GAD, where anomalies usually constitute only a small fraction of the nodes. Therefore, **RQ3** further evaluates anomaly retrieval using AUPRC, which jointly reflects precision and recall for the anomaly class. We follow the same multi-source setting as RQ1 and report the mean AUPRC over 11 random seeds on each unseen organic target.

RQ4: How efficient are these GGAD methods in terms of time and space?

Motivation and Experiment Design. Existing GGAD methods generally introduce various complex modules to capture transferable anomaly patterns, a process that naturally incurs extra computational and spatial complexity compared to vanilla GNNs. However, the algorithm efficiency has been largely overlooked, where **RQ4** is proposed to understand the trade-off between efficiency and

Table 1: AUROC (%) on real-world target graphs under the fixed multi-source setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	92.1 ± 1.4	53.9 ± 1.8	27.7 ± 7.7	37.3 ± 1.2	46.3 ± 0.2	55.6 ± 0.7	56.1 ± 2.0	46.8 ± 6.2	38.4 ± 2.9	66.9 ± 8.4	41.0 ± 4.6	51.1	22
CoLA	32.1 ± 11.8	56.9 ± 2.8	39.9 ± 13.1	43.4 ± 2.5	46.0 ± 2.3	58.3 ± 3.0	52.1 ± 1.1	56.9 ± 2.6	42.5 ± 3.6	57.8 ± 6.2	58.1 ± 18.7	49.5	28
TAM	33.5 ± 13.4	54.6 ± 0.5	76.2 ± 0.4	50.7 ± 0.8	43.5 ± 2.2	50.0 ± 2.2	50.1 ± 0.9	53.5 ± 1.1	**65.8** ± 0.7	73.1 ± 0.8	48.8 ± 3.4	54.5	16
BWGNN	39.6 ± 25.4	50.6 ± 8.0	51.6 ± 17.2	50.8 ± 4.6	49.0 ± 3.9	51.9 ± 8.2	49.8 ± 9.1	52.1 ± 9.4	51.1 ± 6.5	48.0 ± 11.0	51.0 ± 8.2	49.6	27
GHRN	56.6 ± 27.9	46.5 ± 4.8	74.3 ± 16.0	50.2 ± 5.8	49.6 ± 5.3	49.4 ± 6.3	48.4 ± 4.9	49.0 ± 12.0	47.9 ± 6.2	52.9 ± 14.9	53.1 ± 6.8	52.5	18
MLP	53.1 ± 10.3	42.1 ± 1.5	88.9 ± 3.8	44.9 ± 2.6	54.4 ± 2.3	50.0 ± 4.7	49.4 ± 4.6	59.7 ± 9.0	47.0 ± 4.0	44.7 ± 6.6	33.6 ± 6.8	51.6	20
GCN	43.2 ± 26.5	46.2 ± 8.0	64.5 ± 20.6	52.5 ± 3.7	50.9 ± 2.4	40.8 ± 8.8	46.9 ± 9.6	53.0 ± 7.3	46.6 ± 3.7	51.2 ± 5.4	52.1 ± 6.6	49.8	26
GAT	48.9 ± 24.9	54.1 ± 3.9	44.4 ± 10.9	52.5 ± 6.3	48.8 ± 1.5	49.4 ± 9.9	50.4 ± 4.9	44.1 ± 12.4	51.6 ± 3.5	53.0 ± 13.3	57.0 ± 7.4	50.4	23
RF-Graph	72.9 ± 10.8	43.0 ± 1.3	**96.7** ± 0.1	54.7 ± 0.3	57.3 ± 0.4	48.8 ± 1.9	49.7 ± 0.6	45.2 ± 2.2	47.1 ± 0.7	36.9 ± 3.1	49.9 ± 5.4	54.7	15
XGB-Graph	78.9 ± 10.1	50.9 ± 1.8	96.1 ± 0.2	44.8 ± 1.0	53.4 ± 0.5	34.2 ± 0.9	45.6 ± 3.6	68.0 ± 3.1	57.2 ± 1.2	50.3 ± 4.1	45.3 ± 3.0	56.8	12
Generalist GAD Methods													
ARC	88.8 ± 0.7	59.2 ± 0.9	83.8 ± 2.0	54.5 ± 0.5	**61.2** ± 1.7	51.6 ± 2.8	59.9 ± 1.0	26.6 ± 5.1	47.8 ± 3.8	48.3 ± 3.6	47.8 ± 6.8	57.2	9
UNPrompt	89.9 ± 1.8	52.8 ± 1.9	92.0 ± 0.8	53.7 ± 2.1	55.9 ± 1.3	47.9 ± 2.2	49.0 ± 0.9	34.1 ± 1.9	57.5 ± 3.0	39.6 ± 2.4	56.4 ± 5.8	57.1	10
AnomalyGFM	59.0 ± 6.5	53.1 ± 3.2	82.7 ± 6.0	46.6 ± 0.3	60.7 ± 0.9	56.6 ± 4.4	54.2 ± 0.4	58.1 ± 0.6	53.4 ± 1.7	49.1 ± 2.1	72.5 ± 2.0	58.7	7
ZeroGAD	25.1 ± 1.7	50.0 ± 1.8	80.1 ± 2.3	OOM	OOM	38.3 ± 0.5	OOM	OOM	OOM	OOM	45.9 ± 1.8	47.9	29
IA-GGAD	91.7 ± 0.4	54.8 ± 2.2	82.2 ± 1.8	53.3 ± 1.1	47.9 ± 1.7	49.9 ± 2.8	58.5 ± 0.6	24.4 ± 1.1	48.4 ± 1.6	51.3 ± 2.9	41.9 ± 4.0	54.9	14
DR-GGAD	53.6 ± 1.1	58.2 ± 0.5	52.7 ± 3.9	42.5 ± 1.3	55.3 ± 1.5	44.8 ± 0.5	53.0 ± 0.3	42.0 ± 0.7	50.0 ± 1.2	59.3 ± 2.0	40.0 ± 0.3	50.1	24
OWLEYE	84.9 ± 0.4	57.2 ± 0.6	36.9 ± 3.9	53.8 ± 0.6	58.8 ± 0.8	50.7 ± 3.4	51.1 ± 1.1	13.3 ± 0.7	41.8 ± 1.3	49.8 ± 8.6	49.6 ± 1.0	49.8	25
TA-GGAD	91.2 ± 0.2	55.6 ± 2.7	80.5 ± 3.8	54.8 ± 1.8	59.1 ± 2.6	50.8 ± 3.8	58.9 ± 0.4	24.6 ± 0.9	54.4 ± 4.0	51.4 ± 1.1	44.1 ± 1.5	56.8	11
ProMoS	92.2 ± 0.1	55.2 ± 2.4	59.2 ± 2.0	53.4 ± 2.1	58.9 ± 3.4	38.9 ± 1.7	49.6 ± 0.4	13.4 ± 0.2	41.3 ± 0.2	47.1 ± 0.1	53.2 ± 0.1	51.1	21
SAARCS	85.9 ± 2.7	54.1 ± 2.8	60.7 ± 3.8	42.0 ± 0.5	58.4 ± 1.7	52.7 ± 3.4	54.2 ± 1.0	40.1 ± 4.5	55.4 ± 6.0	60.4 ± 6.7	48.8 ± 5.4	55.7	13
TPCA-GAD	81.4 ± 5.5	49.4 ± 0.5	59.0 ± 8.0	70.0 ± 1.0	51.1 ± 0.2	55.1 ± 0.3	45.6 ± 0.2	40.0 ± 1.8	51.6 ± 1.1	35.7 ± 2.9	55.7 ± 4.9	54.1	17
GAD-MoRE	88.2 ± 0.4	57.8 ± 1.4	21.1 ± 0.1	49.8 ± 0.5	56.8 ± 0.9	45.6 ± 0.6	54.2 ± 0.3	31.2 ± 0.4	50.3 ± 0.8	60.0 ± 0.2	55.1 ± 0.4	51.8	19
NeighborDiv	79.8 ± 0.7	51.2 ± 0.0	68.8 ± 0.7	58.1 ± 0.0	50.7 ± 0.2	52.9 ± 0.0	45.5 ± 0.0	56.6 ± 0.0	52.1 ± 0.0	56.4 ± 0.0	72.5 ± 0.0	58.6	8
REFIGAD	85.5 ± 5.9	52.9 ± 4.4	70.8 ± 2.7	73.9 ± 5.8	54.7 ± 4.3	53.2 ± 3.1	56.1 ± 4.0	77.9 ± 4.2	54.1 ± 3.6	83.1 ± 6.4	**85.6** ± 8.0	**68.0**	**2**
TFM4GAD	**97.5** ± 1.6	55.5 ± 4.6	92.5 ± 3.9	**74.0** ± 7.1	61.0 ± 3.2	**67.2** ± 8.3	**62.9** ± 2.8	**85.0** ± 2.9	59.9 ± 6.9	**93.0** ± 4.7	–	**74.8**	**1**
Graph Foundation Models													
GraphPrompt	90.8 ± 6.3	54.1 ± 5.4	73.2 ± 5.9	67.9 ± 2.9	50.9 ± 1.4	52.8 ± 3.8	55.2 ± 7.0	51.7 ± 11.1	52.1 ± 4.0	76.1 ± 9.2	57.1 ± 15.7	62.0	6
SAMGPT	96.2 ± 5.8	**59.5** ± 3.9	79.3 ± 4.4	66.8 ± 5.7	51.7 ± 2.3	61.6 ± 6.0	54.2 ± 2.9	74.6 ± 6.7	59.3 ± 9.3	85.9 ± 5.3	47.6 ± 25.7	**67.0**	**3**
MDGFM	91.0 ± 3.7	55.3 ± 2.3	89.8 ± 2.8	57.7 ± 4.7	55.5 ± 5.1	63.0 ± 6.8	56.7 ± 3.6	69.2 ± 6.6	59.0 ± 8.9	73.5 ± 5.4	41.2 ± 22.1	64.7	5
BRIDGE	96.2 ± 2.7	57.5 ± 4.7	82.4 ± 1.1	62.7 ± 5.4	51.0 ± 1.1	62.4 ± 7.6	56.9 ± 3.8	68.7 ± 7.5	59.1 ± 8.0	74.7 ± 2.9	40.7 ± 23.2	64.7	4
GILT	40.7 ± 22.8	47.5 ± 6.7	47.6 ± 17.9	44.5 ± 5.3	48.4 ± 5.2	46.8 ± 6.6	48.2 ± 4.9	55.5 ± 12.3	46.6 ± 8.8	45.9 ± 12.4	44.6 ± 23.7	46.9	30

scalability. To achieve this, we evaluate the algorithm efficiency by reporting the training time and peak GPU and CPU memory consumption on consistent configurations.

4 Experimental Results

In this section, we compare GGAD methods, conventional baselines, and graph foundation models under the unified GGADBench protocol, organized around the four research questions (RQ1–RQ4) posed in Section 3.3. Detailed experiment settings and additional results on more metrics and backbones can be found in Appendix C, Appendix D.

4.1 Effectiveness Evaluation for GGAD Methods (RQ1)

Using Facebook, T-Finance, and Amazon as source graphs, we evaluate all 30 methods on 11 unseen real-world targets. Table 1 reports the AUROC results.

❶ A shared feature dimension may not imply a shared feature space. PCA/SVD makes heterogeneous features dimensionally compatible, but the latent axes are estimated independently and may not preserve consistent semantics across datasets. Notably, the two strongest methods avoid relying on fixed low-dimensional alignment, suggesting that feature representation may be an important factor in cross-graph performance. **❷ Residual-based methods can rank anomalies in the wrong direction.** On negative-ARP targets, many residual- and reconstruction-based methods achieve AUROC below 50, indicating that nodes with larger neighborhood deviations are more often normal than anomalous. Methods based on other anomaly signals do not show this reversal as consistently. **❸ Prompt-tuned GFMs are competitive with specialized GGAD methods.** Among the five GFMs evaluated, SAMGPT, MDGFM, and BRIDGE rank third to fifth overall. In contrast, the tuning-free GILT achieves a lower performance under the same benchmark. This indicates that current GFM competitiveness in GGAD is closely related to target-side adaptation. **❹ Conventional baselines remain strong in specific targets.** Despite being designed for single-graph learning, RF-

Graph and TAM achieve the best results on Amazon-all and DGraph-Fin, respectively. This shows that strong cross-graph generalization does not guaranty the best detection performance on every target graph.

> **RQ1 Insight** **(i) Feature alignment should preserve transferable information, not only match dimensionality.** Projecting heterogeneous features into the same dimension does not ensure that they remain comparable across graphs. The strongest methods instead preserve transferable information representations with consistent cross-graph meaning. **(ii) Cross-graph detection requires target-adaptive anomaly modeling.** Different anomaly signals and method families perform differently across target graphs, and the best-performing method varies by target. This suggests that GGAD should adapt its anomaly criterion to the target graph rather than rely on a fixed detection rule across all graphs.

4.2 Transferability Evaluation for GGAD Methods (RQ2)

In this section, we investigate how cross-graph performance changes with the source and the target. We vary the source anomaly type and source quantity, and evaluate the resulting models on both organic and injected unseen target graphs.

Table 2: AUROC (%) on injected targets. Models are trained with multiple organic-anomaly graphs. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	84.2 ± 3.7	75.2 ± 0.5	83.8 ± 3.5	63.1 ± 0.8	76.6
CoLA	68.1 ± 3.2	51.6 ± 4.9	49.4 ± 2.8	54.2 ± 4.2	55.8
TAM	60.5 ± 1.4	54.0 ± 1.9	70.1 ± 2.9	58.2 ± 0.8	60.7
BWGNN	47.5 ± 14.4	49.8 ± 11.6	49.7 ± 14.1	49.5 ± 7.4	49.1
GHRN	50.9 ± 12.5	57.0 ± 11.7	54.6 ± 12.1	51.9 ± 7.5	53.6
MLP	41.4 ± 6.0	35.1 ± 9.3	43.3 ± 7.7	49.7 ± 10.4	42.4
GCN	50.3 ± 6.4	50.7 ± 10.2	52.5 ± 6.2	50.1 ± 5.8	50.9
GAT	48.6 ± 9.6	49.3 ± 5.7	49.7 ± 7.2	48.5 ± 2.4	49.0
RF-Graph	50.2 ± 0.9	52.8 ± 6.9	56.3 ± 1.0	54.4 ± 1.0	53.4
XGB-Graph	50.0 ± 2.5	47.5 ± 2.9	56.2 ± 2.8	58.0 ± 3.5	52.9
ARC	85.4 ± 0.3	73.7 ± 1.8	82.9 ± 0.3	75.3 ± 0.9	79.3
UNPrompt	69.0 ± 1.5	68.8 ± 0.6	69.6 ± 1.6	62.6 ± 2.6	67.5
AnomalyGFM	49.9 ± 2.7	53.6 ± 1.8	57.2 ± 3.1	55.8 ± 2.8	54.1
ZeroGAD	OOM	50.7 ± 0.8	OOM	62.4 ± 2.3	–
IA-GGAD	88.6 ± 0.6	68.0 ± 1.5	93.0 ± 0.4	72.7 ± 1.6	**80.6**

Method	PubMed	Flickr	CS	Photo	Avg.
DR-GGAD	**95.4** ± 0.4	73.7 ± 0.3	**97.9** ± 0.3	67.1 ± 1.8	**83.5**
OWLEYE	77.2 ± 0.4	**78.2** ± 4.6	81.2 ± 0.2	69.3 ± 1.8	76.5
TA-GGAD	87.5 ± 0.5	76.4 ± 1.3	88.7 ± 1.0	77.3 ± 2.7	**82.5**
ProMoS	81.9 ± 0.4	69.4 ± 0.9	76.3 ± 0.8	70.1 ± 1.1	74.4
SAARCS	84.2 ± 0.3	76.0 ± 0.7	82.8 ± 0.9	**78.3** ± 2.3	80.3
TPCA-GAD	44.3 ± 2.4	41.8 ± 0.8	60.1 ± 1.9	47.1 ± 0.5	48.3
GAD-MoRE	79.9 ± 0.5	73.9 ± 0.5	83.5 ± 0.3	75.6 ± 0.2	78.2
NeighborDiv	72.0 ± 0.0	44.8 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	67.3
REFIGAD	92.4 ± 1.8	68.6 ± 3.5	91.0 ± 3.4	65.7 ± 5.7	79.4
TFM4GAD	74.6 ± 3.7	55.4 ± 4.9	75.6 ± 3.1	76.1 ± 2.8	70.4
GraphPrompt	70.7 ± 1.9	55.7 ± 6.1	61.9 ± 9.1	51.0 ± 4.8	59.9
SAMGPT	74.9 ± 7.0	65.2 ± 12.2	69.4 ± 7.2	56.7 ± 5.2	66.5
MDGFM	55.2 ± 8.0	65.0 ± 7.6	61.3 ± 9.4	62.5 ± 4.5	61.0
BRIDGE	57.1 ± 7.2	56.2 ± 13.3	57.9 ± 3.3	53.6 ± 3.4	56.2
GILT	46.4 ± 5.0	47.4 ± 11.3	47.7 ± 8.8	53.4 ± 7.4	48.7

4.2.1 Transferability from Organic to Injected Anomalies

Using {Facebook, T-Finance, Amazon} as organic source graphs, we evaluate all methods on four unseen injected targets: {PubMed, Flickr, CS, Photo}. Table 2 reports the AUROC results.

❶ **The best-performing methods differ between organic and injected targets.** The relative performance of methods changes noticeably between the two target types. For example, TFM4GAD ranks first on organic targets, but its advantage becomes much weaker on injected targets, where several GGAD (TAGGAD,IAGGAD) achieve higher AUROC. This indicates that method effectiveness depends strongly on the target anomaly type. ❷ **The same injected anomalies can have very different difficulty across graphs.** Despite using the same injection procedure, PubMed and CS are generally much easier to detect than Flickr and Photo. The best-performing method also changes across targets. ❸ **Conventional methods can be highly competitive on injected targets.** DOMINANT achieves an average AUROC of 76.6, outperforming many GGAD methods and prompt-tuned GFMs. This shows that strong performance on injected anomalies is not limited to methods designed for cross-graph generalization.

4.2.2 Transferability from Injected to Injected Anomalies

Using {Cora, Citeseer, ACM, BlogCatalog} as injected source graphs, we evaluate all methods on the four unseen injected targets: {PubMed, Flickr, CS, Photo}. Table 3 reports the AUROC results and their changes relative to the organic-to-injected setting.

Table 3: AUROC (%) on injected targets. Models are trained with multiple injected-anomaly graphs. Δ denotes the change in average AUROC relative to organic-anomaly source training.

Method	PubMed	Flickr	CS	Photo	Δ
DOMINANT	83.5 ± 0.4	76.3 ± 0.1	86.0 ± 0.2	63.9 ± 0.0	+0.9
CoLA	67.7 ± 2.9	52.9 ± 2.9	49.6 ± 3.4	61.8 ± 6.4	+2.2
TAM	63.5 ± 2.7	59.4 ± 2.1	69.3 ± 2.8	64.1 ± 5.8	+3.3
BWGNN	56.1 ± 14.5	51.4 ± 12.7	57.1 ± 13.6	52.7 ± 7.3	+5.2
GHRN	61.0 ± 9.7	68.1 ± 9.6	67.2 ± 8.3	54.4 ± 7.0	+9.1
MLP	64.0 ± 1.9	73.2 ± 0.5	73.1 ± 1.4	55.4 ± 4.7	+24.1
GCN	56.7 ± 7.6	59.0 ± 6.8	57.4 ± 8.6	54.7 ± 7.5	+6.0
GAT	70.9 ± 5.7	42.8 ± 3.6	64.2 ± 4.7	55.1 ± 3.7	+9.2
RF-Graph	88.8 ± 0.6	70.2 ± 1.3	89.1 ± 0.7	74.9 ± 1.1	+27.3
XGB-Graph	68.5 ± 1.4	65.7 ± 1.2	79.1 ± 0.6	64.5 ± 1.2	+16.5
ARC	85.4 ± 0.2	73.6 ± 1.8	82.7 ± 0.5	75.3 ± 1.0	-0.1
UNPrompt	67.4 ± 1.2	70.0 ± 0.4	69.8 ± 0.8	55.0 ± 4.5	-2.0
AnomalyGFM	52.7 ± 3.7	53.8 ± 2.6	54.6 ± 3.4	51.8 ± 3.1	-0.9
ZeroGAD	OOM	46.9 ± 0.3	OOM	74.0 ± 2.3	–
IA-GGAD	88.1 ± 0.6	70.0 ± 1.3	92.6 ± 0.4	78.1 ± 1.9	+1.6

Method	PubMed	Flickr	CS	Photo	Δ
DR-GGAD	**96.0** ± 0.3	73.4 ± 0.4	97.8 ± 0.3	67.6 ± 1.2	+0.1
OWLEYE	78.1 ± 0.1	74.9 ± 2.9	82.3 ± 0.2	75.2 ± 0.3	+1.1
TA-GGAD	87.7 ± 0.6	77.6 ± 1.8	88.7 ± 0.9	76.1 ± 3.4	+0.1
ProMoS	82.3 ± 0.5	69.2 ± 0.6	77.8 ± 0.5	70.8 ± 0.6	+0.6
SAARCS	84.2 ± 0.3	76.0 ± 0.7	82.8 ± 0.9	**78.3** ± 2.3	0.0
TPCA-GAD	73.7 ± 0.9	38.5 ± 6.6	73.9 ± 2.1	50.2 ± 2.3	+10.7
GAD-MoRE	74.4 ± 0.3	70.3 ± 0.5	78.5 ± 0.2	65.4 ± 2.6	-6.1
NeighborDiv	72.1 ± 0.0	44.8 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	–
REFIGAD	95.9 ± 1.2	**82.1** ± 4.3	**98.2** ± 0.7	63.0 ± 3.7	+5.4
TFM4GAD	74.6 ± 3.7	55.4 ± 4.9	75.6 ± 3.2	76.1 ± 2.8	–
GraphPrompt	58.7 ± 11.1	51.2 ± 2.8	54.1 ± 7.4	51.2 ± 4.8	-6.1
SAMGPT	74.7 ± 6.1	64.2 ± 13.4	69.0 ± 6.8	56.6 ± 5.1	-0.4
MDGFM	56.8 ± 7.4	66.5 ± 14.8	59.8 ± 8.0	59.3 ± 6.3	-0.4
BRIDGE	57.7 ± 4.6	58.4 ± 13.3	56.9 ± 3.3	53.5 ± 2.6	+0.4
GILT	52.8 ± 5.8	56.4 ± 8.1	52.3 ± 9.7	42.5 ± 12.0	+2.3

❶ Matching the source and target anomaly types brings limited gains to GGAD methods. Changing the source graphs from organic to injected anomalies produces only small performance changes for most GGAD methods. For example, ARC, DR-GGAD, TA-GGAD, and SAARCS change by no more than 0.1 AUROC on average. This shows that GGAD performance is generally insensitive to whether the source and target share the same anomaly type. **❷ Conventional baselines benefit much more from matched anomaly types.** In contrast to GGAD methods, conventional baselines improve substantially when both source and target graphs contain injected anomalies. Their average gain is about 10.4 AUROC, with RF-Graph, MLP, and XGB-Graph improving by 27.3, 24.1, and 16.5 points, respectively. This indicates that conventional models are more sensitive to the anomaly type of the source graphs. **❸ Prompt-tuned GFMs show little benefit from source–target anomaly-type matching.** The performance of SAMGPT, MDGFM, and BRIDGE changes by less than 0.5 AUROC, while GraphPrompt decreases by 6.1 points. Thus, using injected rather than organic source graphs does not consistently improve the evaluated GFMs.

4.2.3 Transferability under Single- and Multi-Source Training

Using Facebook and Cora as the organic and injected single sources, and {Facebook, T-Finance, Amazon} and {Cora, Citeseer, ACM, BlogCatalog} as their multi-source counterparts, we compare single- and multi-source performance on the same unseen targets.

(i) Multi-source training provides only limited gains. The overall improvement from single- to multi-source training is only +0.45 AUROC points on average. The gain is also small on both organic and injected targets, and performance can either increase or decrease across individual method–target pairs. *(ii)* Most method–target pairs are insensitive to source quantity. Overall, 68.2% of paired evaluations change by less than 3 AUROC points between single- and multi-source training. This shows that increasing the number of source graphs has little effect on most evaluations.

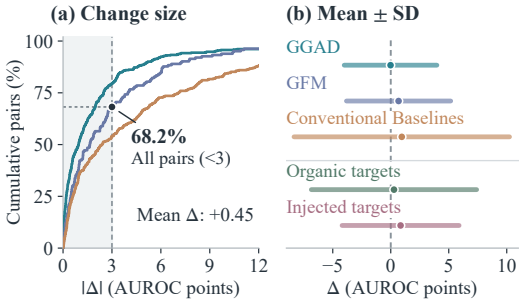

Figure 2: Single- versus multi-source training (Δ = multi − single). Matching colors link method families; bars show ±1 population SD across paired evaluations.

4.2.4 Transferability from Injected to Organic Anomalies

Using {Cora, CiteSeer, ACM, BlogCatalog} as injected source graphs, we evaluate all methods on the same 11 unseen organic targets used in RQ1. Table 12 in Appendix D.1 reports the complete AUROC results. **Strong performance on injected targets does not transfer directly to organic**

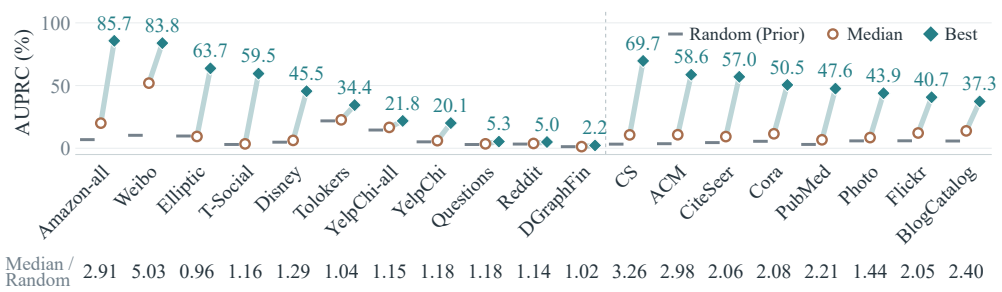

Figure 3: AUPRC (%) on unseen real-world targets, showing the random baseline, method median, and the best method with its gain over random (Δ, pp). Method colors denote Conv./GGAD/GFM.

targets. With the same injected-source training, REFIGAD, DR-GGAD, and TA-GGAD achieve the top three average AUROC scores on injected targets, at 84.8, 83.7, and 82.5, respectively. On organic targets, their average AUROC becomes 67.8, 49.4, and 54.5, while TFM4GAD, REFIGAD, and SAMGPT rank highest overall. This shows that the methods that perform best on injected anomalies are not necessarily the strongest on organic anomalies.

Robustness to Source Selection. To examine whether the above observations depend on a particular choice of source graphs, we repeat the evaluation with an alternative set of organic and injected sources. The main trends remain consistent under the new source split in Appendix D.4.

RQ2 Insight **(i) Changing or adding source graphs provides limited gains for generalist methods.** Matching the source and target anomaly types brings little improvement to GGAD and prompt-tuned GFMs, while multi-source training improves AUROC by only 0.45 points on average. **(ii) Target difficulty changes little across source types.** The target-wise mean AUROC under organic and injected source training is highly correlated ($r = 0.93$), indicating that easy and difficult organic targets remain largely the same after changing the source anomaly type.

4.3 Anomaly Retrieval Evaluation for GGAD Methods (RQ3)

Using Facebook, T-Finance, and Amazon as source graphs, we evaluate anomaly retrieval on the same 11 real-world targets with AUPRC. Figure 3 reports the results.

❶ **Ranking quality does not imply effective anomaly retrieval.** Despite the moderate AUROC achieved in RQ1, anomaly retrieval remains weak on most targets. On 9 of the 11 targets, the median AUPRC is only 0.98–$1.3\times$ the random baseline, including $0.98\times$ on Elliptic and $1.0\times$ on Tolokers and DGraph. ❷ **Anomaly retrieval is strongly method-selective.** Large gaps arise between the median and the best AUPRC on several targets. For example, the median and best AUPRC are 19.4 and 85.7 on Amazon-all, 9.5 and 63.7 on Elliptic, 3.4 and 59.5 on T-Social, and 6.3 and 45.5 on Disney, respectively. ❸ **Anomaly prevalence does not determine retrieval performance.** T-Social and Questions have the same 3.0% class prior, yet their best AUPRCs differ substantially, at 59.5 and 5.3, respectively. Conversely, Tolokers has the highest class prior (21.8%) while its median AUPRC remains close to the random baseline.

RQ3 Insight **Anomaly retrieval is under-evaluated.** AUROC alone can obscure poor anomaly retrieval, as moderate ranking quality may still correspond to near-random AUPRC on many targets. Future GGAD should place greater emphasis on AUPRC alongside AUROC to assess whether anomalies can be retrieved with meaningful precision.

4.4 Efficiency and Scalability Analysis (RQ4)

Using an RTX 3090, we measure peak GPU and CPU memory usage and wall-clock time for one training epoch under single- and multi-source settings, and for testing under the multi-source setting on DGraphFin and T-Social (Figure 4).

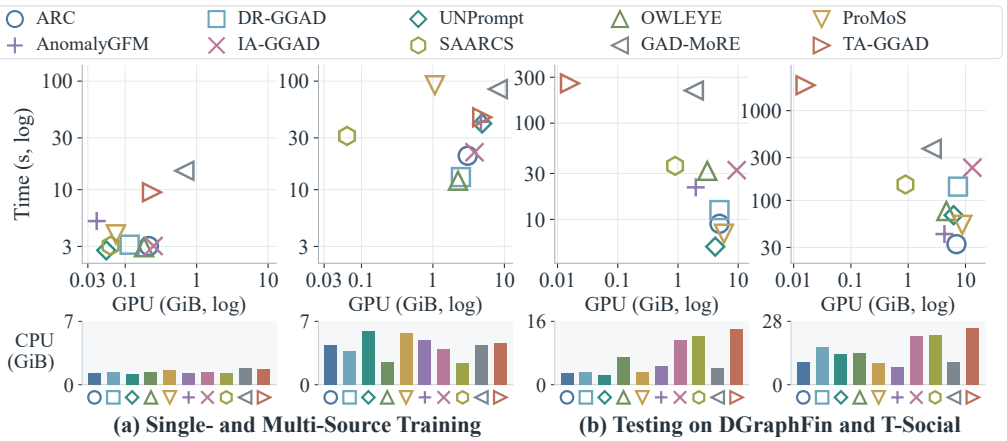

Figure 4: Peak GPU and CPU memory usage and wall-clock time for (a) single- and multi-source training and (b) testing the multi-source models on DGraphFin and T-Social.

❶ **Efficiency varies widely across methods.** Both training and testing costs differ greatly across methods, and the runtime gap becomes much larger on large target graphs. ❷ **Low GPU memory does not mean low overall cost.** Some methods use little GPU memory but require much longer runtime and much more CPU memory, showing clear trade-offs between different resources. ❸ **Large graphs create different bottlenecks.** As target graphs become larger, different methods are limited by different combinations of runtime, GPU memory, and CPU memory.

RQ4 Insight **(i) Large target graphs are the main efficiency challenge.** Efficiency gaps become much larger on large target graphs, showing that efficient target processing is important for practical GGAD. **(ii) Efficiency must be measured from multiple sides.** Low GPU memory can come with high runtime or CPU memory, so no single resource can fully reflect efficiency. Future GGAD should reduce the overall cost of processing target graphs rather than focusing on only one resource.

5 Conclusion and Future Directions

This paper presents GGADBench, the first comprehensive benchmark for Generalist Graph Anomaly Detection, integrating 30 methods across 22 graphs with organic and injected anomalies. We conduct a systematic evaluation of existing methods in terms of effectiveness, cross-graph transferability, anomaly retrieval, and efficiency under unified protocols. We further introduce *anomaly residual polarity* (ARP) to characterize the relationship between anomalies and their local neighborhoods. Drawing upon our empirical analysis and insights, we highlight several promising directions for future GGAD research:

❶ **Transferable Representation.** Cross-graph feature heterogeneity cannot be addressed by dimensional alignment alone. Future research should explore representations that preserve anomaly-relevant semantics across heterogeneous graphs rather than directly aligning their raw feature spaces.

❷ **Target Adaptation and Generalization.** Unseen target graphs can exhibit substantially different anomaly patterns, making a fixed detection criterion difficult to generalize. Future GGAD should develop target-adaptive mechanisms that can adjust anomaly modeling to new graphs with little or no target supervision.

❸ **Practicality and Scalability.** Practical GGAD requires not only accurate anomaly ranking but also effective retrieval and affordable computation on large graphs. Future research should jointly improve retrieval quality and reduce target-side time and memory costs for large-scale deployment.

Ethics Statement

GGADBench uses previously released graph datasets and does not collect new data or involve direct interaction with human participants. We use these datasets solely for research evaluation and do not attempt to identify or infer the identities of individuals represented in the data.

AI Use Statement

Generative AI tools were used solely for language polishing. All AI-assisted revisions were carefully reviewed and verified by the authors. Generative AI was not used to generate or fabricate experimental results, citations, or scientific claims. The authors take full responsibility for the accuracy and content of the final manuscript.

References

Ziwei Chai, Siqi You, Yang Yang, Shiliang Pu, Jiarong Xu, Haoyang Cai, and Weihao Jiang. Can abnormality be detected by graph neural networks? In *IJCAI*, pp. 1945–1951, 2022.

Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *SIAM*, pp. 594–602, 2019.

Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *CIKM*, pp. 315–324, 2020a.

Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *CIKM*, pp. 315–324, 2020b.

Jingcan Duan, Siwei Wang, Pei Zhang, En Zhu, Jingtao Hu, Hu Jin, Yue Liu, and Zhibin Dong. Graph anomaly detection via multi-scale contrastive learning networks with augmented view. In *AAAI*, volume 37, pp. 7459–7467, 2023.

Haoyi Fan, Fengbin Zhang, and Zuoyong Li. Anomalydae: Dual autoencoder for anomaly detection on attributed networks. In *ICASSP*, pp. 5685–5689, 2020.

Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. Graph neural networks for social recommendation. In *WWW*, pp. 417–426, 2019.

Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

Changlong Fu, Zhenli He, Xiong Zhang, Cheng Xie, Xin Jin, and Yun Yang. Dr-ggad: Dual residual centering for mitigating anomaly non-discriminativity in generalist graph anomaly detection. In *ICLR*, volume 2026, pp. 71971–71997, 2026.

Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. Alleviating structural distribution shift in graph anomaly detection. In *WSDM*, pp. 357–365, 2023a.

Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. Addressing heterophily in graph anomaly detection: A perspective of graph spectrum. In *WWW*, pp. 1528–1538, 2023b.

Fengrui Hua, Yiyan Qi, Zikai Wei, Yuxing Tian, Chengjin Xu, Xiaojun Wu, Jia Li, and Jian Guo. Bag: Benchmarking anomaly detection on dynamic graphs. In *AAAI*, volume 40, pp. 14892–14900, 2026.

Xuanwen Huang, Yang Yang, Yang Wang, Chunping Wang, Zhisheng Zhang, Jiarong Xu, Lei Chen, and Michalis Vazirgiannis. Dgraph: A large-scale financial dataset for graph anomaly detection. *NeurIPS*, 35:22765–22777, 2022.

Ming Jin, Yixin Liu, Yu Zheng, Lianhua Chi, Yuan-Fang Li, and Shirui Pan. Anemone: Graph anomaly detection with multi-scale contrastive learning. In *CIKM*, pp. 3122–3126, 2021.

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

Srijan Kumar, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *KDD*, pp. 1269–1278, 2019.

Linyu Li, Zhi Jin, Xuan Zhang, Haoran Duan, Jishu Wang, Zhengwei Tao, Haiyan Zhao, and Xiaofeng Zhu. Multi-view riemannian manifolds fusion enhancement for knowledge graph completion. *TKDE*, 37(5):2756–2770, 2025.

Fanzhen Liu, Xiaoxiao Ma, Jia Wu, Jian Yang, Shan Xue, Amin Beheshti, Chuan Zhou, Hao Peng, Quan Z Sheng, and Charu C Aggarwal. Dagad: Data augmentation for graph anomaly detection. In *ICDM*, pp. 259–268, 2022a.

Kay Liu, Yingtong Dou, Yue Zhao, Xueying Ding, Xiyang Hu, Ruitong Zhang, Kaize Ding, Canyu Chen, Hao Peng, Kai Shu, et al. Bond: Benchmarking unsupervised outlier node detection on static attributed graphs. *NeurIPS*, 35:27021–27035, 2022b.

Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Pick and choose: a gnn-based imbalanced learning approach for fraud detection. In *WWW*, pp. 3168–3177, 2021a.

Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *TNNLS*, 33(6):2378–2392, 2021b.

Yixin Liu, Shiyuan Li, Yu Zheng, Qingfeng Chen, Chengqi Zhang, and Shirui Pan. Arc: A generalist graph anomaly detector with in-context learning. *NeurIPS*, 37:50772–50804, 2024.

Yixin Liu, Shiyuan Li, Yu Zheng, Qingfeng Chen, Chengqi Zhang, Philip S Yu, and Shirui Pan. From few-shot to zero-shot: Towards generalist graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 2026a.

Yujing Liu, Yixin Liu, Yu Zheng, Alan Wee-Chung Liew, Xiaofeng Cao, and Shirui Pan. Rethinking feature alignment in generalist graph anomaly detection: A relational fingerprint-based approach. In *ICML*, 2026b.

Yunhui Liu, Tieke He, Yongchao Liu, Can Yi, Hong Jin, and Chuntao Hong. Tabular foundation models are strong graph anomaly detectors. In *WWW*, pp. 8785–8788, 2026c.

Zemin Liu, Xingtong Yu, Yuan Fang, and Xinming Zhang. Graphprompt: Unifying pre-training and downstream tasks for graph neural networks. In *WWW*, pp. 417–428, 2023.

Zhaowei Liu, Leilei Jiang, and Haitao Yang. Multi-dimensional adaptive mix-hop contextual learning framework for universal graph anomaly detection. In *AAAI*, volume 40, pp. 15412–15420, 2026d.

Zhiwei Liu, Yingtong Dou, Philip S Yu, Yutong Deng, and Hao Peng. Alleviating the inconsistency problem of applying graph neural network to fraud detection. In *SIGIR*, pp. 1569–1572, 2020.

Xuexiong Luo, Jia Wu, Amin Beheshti, Jian Yang, Xiankun Zhang, Yuan Wang, and Shan Xue. Comga: Community-aware attributed graph anomaly detection. In *WSDM*, pp. 657–665, 2022.

Weishuo Ma, Yanbo Wang, Xiyuan Wang, Lei Zou, and Muhan Zhang. Gilt: An llm-free, tuning-free graph foundational model for in-context learning. *arXiv preprint arXiv:2510.04567*, 2025.

Julian John McAuley and Jure Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *WWW*, pp. 897–908, 2013.

Chaoxi Niu, Hezhe Qiao, Changlu Chen, Ling Chen, and Guansong Pang. Zero-shot generalist graph anomaly detection with unified neighborhood prompts. In *IJCAI*, pp. 3226–3234, 2024.

Junjun Pan, Yixin Liu, Yizhen Zheng, and Shirui Pan. Prem: A simple yet effective approach for node-level graph anomaly detection. In *ICDM*, pp. 1253–1258, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of gnns under heterophily: Are we really making progress? In *ICLR*, 2023.

Tim Poštuvan, Claas Grohnfeldt, Michele Russo, and Giulio Lovisotto. Learning-based link anomaly detection in continuous-time dynamic graphs. *arXiv preprint arXiv:2405.18050*, 2024.

Hezhe Qiao and Guansong Pang. Truncated affinity maximization: One-class homophily modeling for graph anomaly detection. *NeurIPS*, 36:49490–49512, 2023.

Hezhe Qiao, Chaoxi Niu, Ling Chen, and Guansong Pang. Anomalygfm: Graph foundation model for zero/few-shot anomaly detection. In *KDD*, pp. 2326–2337, 2025.

Shebuti Rayana and Leman Akoglu. Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining*, pp. 985–994, 2015a.

Shebuti Rayana and Leman Akoglu. Collective opinion spam detection: Bridging review networks and metadata. In *KDD*, pp. 985–994, 2015b.

Amit Roy, Juan Shu, Jia Li, Carl Yang, Olivier Elshocht, Jeroen Smeets, and Pan Li. Gad-nr: Graph anomaly detection via neighborhood reconstruction. In *WSDM*, pp. 576–585, 2024.

Patricia Iglesias Sánchez, Emmanuel Müller, Fabian Laforet, Fabian Keller, and Klemens Böhm. Statistical selection of congruent subspaces for mining attributed graphs. In *ICDM*, pp. 647–656, 2013.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–106, 2008.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

Fengzhao Shi, Yanan Cao, Yanmin Shang, Yuchen Zhou, Chuan Zhou, and Jia Wu. H2-fdetector: A gnn-based fraud detector with homophilic and heterophilic connections. In *Proceedings of the ACM web conference 2022*, pp. 1486–1494, 2022.

Jianheng Tang, Jiajin Li, Ziqi Gao, and Jia Li. Rethinking graph neural networks for anomaly detection. In *ICML*, pp. 21076–21089, 2022.

Jianheng Tang, Fengrui Hua, Ziqi Gao, Peilin Zhao, and Jia Li. Gadbench: Revisiting and benchmarking supervised graph anomaly detection. *NeurIPS*, 36:29628–29653, 2023.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In *KDD*, pp. 990–998, 2008.

Lei Tang and Huan Liu. Relational learning via latent social dimensions. In *KDD*, pp. 817–826, 2009.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018.

Shuo Wang, Bokui Wang, Zhixiang Shen, Boyan Deng, and Zhao Kang. Multi-domain graph foundation models: Robust knowledge transfer via topology alignment. In *ICML*, 2025a.

Yili Wang, Yixin Liu, Xu Shen, Chenyu Li, Rui Miao, Kaize Ding, Ying Wang, Shirui Pan, and Xin Wang. Unifying unsupervised graph-level anomaly detection and out-of-distribution detection: A benchmark. In *ICLR*, volume 2025, pp. 91903–91929, 2025b.

Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I Weidele, Claudio Bellei, Tom Robinson, and Charles E Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv preprint arXiv:1908.02591*, 2019a.

Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I Weidele, Claudio Bellei, Tom Robinson, and Charles E Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv preprint arXiv:1908.02591*, 2019b.

Kaifeng Wei, Teng Liu, Liang Dong, Xiubo Liang, and Yuke Li. Neighbordiv: Training-free zero-shot generalist graph anomaly detection via neighbor diversity. *arXiv preprint arXiv:2605.20879*, 2026.

Yiming Xu, Zihan Chen, Zhen Peng, Song Wang, Bin Shi, Bo Dong, and Chao Shen. Generalist graph anomaly detection via prototype-based distillation. In *ICML*, 2026.

Zhiming Xu, Xiao Huang, Yue Zhao, Yushun Dong, and Jundong Li. Contrastive attributed network anomaly detection with data augmentation. In *PAKDD*, pp. 444–457, 2022.

Youjian Yao, Tailun Chen, Yiyao Yang, Qisen Yan, and Wen Chen. Tpca-gad: Topology preference-consistency aggregation for zero-shot graph anomaly detection. *Expert Systems with Applications*, pp. 133224, 2026.

Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *KDD*, pp. 974–983, 2018.

Xingtong Yu, Zechuan Gong, Chang Zhou, Yuan Fang, and Hui Zhang. Samgpt: Text-free graph foundation model for multi-domain pre-training and cross-domain adaptation. In *WWW*, pp. 1142–1153, 2025.

Haonan Yuan, Qingyun Sun, Junhua Shi, Xingcheng Fu, Bryan Hooi, Jianxin Li, and Philip S Yu. How much can transfer? bridge: Bounded multi-domain graph foundation model with generalization guarantees. In *ICML*, 2025.

Xiong Zhang, Zhenli He, Changlong Fu, and Cheng Xie. Ia-ggad: Zero-shot generalist graph anomaly detection via invariant and affinity learning. *NeurIPS*, 38:83728–83760, 2025.

Xiong Zhang, Hong Peng, Changlong Fu, Xin Jin, Yun Yang, and Cheng Xie. Ta-ggad: Testing-time adaptive graph model for generalist graph anomaly detection. *arXiv preprint arXiv:2603.09349*, 2026.

Tong Zhao, Chuchen Deng, Kaifeng Yu, Tianwen Jiang, Daheng Wang, and Meng Jiang. Error-bounded graph anomaly loss for gnns. In *CIKM*, pp. 1873–1882, 2020.

Xinyu Zhao, Qingyun Sun, Jiayi Luo, Xingcheng Fu, and Jianxin Li. Zero-shot generalizable graph anomaly detection with mixture of riemannian experts. In *ICDM*. IEEE, 2026.

Lecheng Zheng, Dongqi Fu, Zihao Li, and Jingrui He. Owleye: Zero-shot learner for cross-domain graph data anomaly detection. In *ICLR*, 2026.

Xiangping Zheng, Xuan Feng, Bo Wu, Bin Ren, Wei Li, Xiuxin Hao, Xun Liang, Bin Tang, and Zhiwen Yu. Breaking semantic barriers: A zero-shot generalized framework for graph anomaly detection. In *MM*, pp. 1423–1432, 2025.

Yu Zheng, Ming Jin, Yixin Liu, Lianhua Chi, Khoa T Phan, and Yi-Ping Phoebe Chen. Generative and contrastive self-supervised learning for graph anomaly detection. *TKDE*, 35(12):12220–12233, 2021.

Appendix

A Detailed Description of Models **15**

A.1 Conventional Baselines 15

A.2 Generalist Graph Anomaly Detection Methods 16

A.3 Graph Foundation Models 18

B Detailed Description of Datasets **18**

B.1 Organic Anomaly Datasets 18

B.2 Injected Anomaly Datasets 20

C Detailed Experimental Settings **21**

C.1 Unified Evaluation Protocol 21

C.2 Model Configuration and Hyperparameter Search 21

C.3 Feature Preprocessing for Conventional Baselines 21

C.4 Implementation and Reproducibility 22

D Additional Experimental Results **23**

D.1 Transferability Evaluation (RQ2) 23

D.2 Anomaly Retrieval Evaluation (RQ3) 24

D.3 Efficiency and Scalability Analysis (RQ4) 29

D.4 Robustness to Alternative Source Selection 32

E Additional Related Work **40**

E.1 Graph Anomaly Detection 40

E.2 Generalist Graph Anomaly Detection 41

E.3 Graph Anomaly Detection Benchmarks 41

Table 4: Categorization of all models used in our evaluation.

Category	Models
Conventional Baselines	MLP, GCN, GAT, DOMINANT, CoLA, TAM, BWGNN, GHRN, RF-Graph, XGB-Graph
Generalist GAD Methods	ARC, UNPrompt, AnomalyGFM, ZeroGAD, IA-GGAD, DR-GGAD, OWLEYE, TA-GGAD, SAARCS, TPCA-GAD, GAD-MoRE, NeighborDiv, ProMoS, REFIGAD, TFM4GAD
Graph Foundation Models	GraphPrompt, SAMGPT, MDGFM, BRIDGE, GILT

A Detailed Description of Models

We include ***30*** models in our benchmark, consisting of ***15*** specialized GGAD methods, ***10*** conventional baselines, and ***5*** graph foundation models. Below, we provide detailed descriptions of their original model designs and key mechanisms.

A.1 Conventional Baselines

We include 10 conventional baselines, covering generic neural networks, graph neural networks, unsupervised and supervised GAD methods, and tree ensembles with neighborhood aggregation. Their detailed descriptions are provided below.

- **MLP** is a feed-forward neural network composed of multiple fully connected layers. It independently transforms node attributes without using graph structural information and serves as a non-graph baseline for node anomaly classification.
- **GCN** Kipf & Welling (2017) performs layer-wise neighborhood aggregation based on a first-order approximation of spectral graph convolution. By repeatedly propagating and transforming neighboring node features, it learns representations that jointly encode node attributes and local graph structure.
- **GAT** Veličković et al. (2018) introduces self-attention into neighborhood aggregation. Instead of assigning identical aggregation weights to all neighboring nodes, GAT learns attention coefficients that determine the relative importance of different neighbors when updating a node representation.
- **DOMINANT** Ding et al. (2019) is an unsupervised attributed-network anomaly detection method based on graph autoencoding. It employs a GCN-based encoder to jointly capture graph structure and node attributes, followed by an attribute decoder and a structure decoder to reconstruct the node feature matrix and graph topology, respectively. The anomaly score of each node is computed as a weighted combination of its attribute reconstruction error and structural reconstruction error.
- **CoLA** Liu et al. (2021b) is a contrastive self-supervised method for unsupervised graph anomaly detection. It constructs "target node–local subgraph" instance pairs, where a target node and its own contextual subgraph form a positive pair, while the node is paired with another sampled subgraph to form a negative pair. A GNN encoder and discriminator are trained to measure the agreement between nodes and subgraphs, and the anomaly score is obtained from the contrastive agreement scores over multiple sampling rounds.
- **TAM** Qiao & Pang (2023) is an unsupervised GAD method motivated by the one-class homophily observation that normal nodes generally exhibit stronger affinity with their neighbors than anomalous nodes. It introduces local affinity maximization networks to learn a representation space in which local node affinity is directly optimized for anomaly detection.
- **BWGNN** Tang et al. (2022) is a spectral GNN designed for supervised graph anomaly detection. It is motivated by the "right-shift" phenomenon that graph anomalies shift spectral energy from low-frequency components toward higher frequencies. BWGNN constructs a family of Beta-wavelet filters that cover different graph frequency bands and uses their spatially and spectrally localized representations to capture anomaly-related high-frequency signals.
- **GHRN** Gao et al. (2023b) addresses the strong heterophily between anomalous and normal nodes in graph anomaly detection. It introduces a label- or prediction-aware high-frequency indicator

to estimate whether an edge is likely to connect nodes from different classes. Edges with high heterophily scores are pruned to construct a purified graph, on which a GNN is subsequently trained for anomaly classification.

- **RF-Graph** Tang et al. (2023) extends Random Forest to graph anomaly detection by incorporating parameter-free neighborhood aggregation. Starting from the original node attributes, it repeatedly aggregates neighboring features for multiple hops and concatenates the representations from different aggregation depths. The resulting structure-enhanced feature vectors are then fed into a Random Forest classifier to predict node anomaly scores.
- **XGB-Graph** Tang et al. (2023) follows the same graph feature construction scheme as RF-Graph, using parameter-free multi-hop neighborhood aggregation to incorporate graph structural information into node features. The concatenated original and aggregated features are subsequently processed by XGBoost, which learns an additive ensemble of decision trees for node anomaly classification.

A.2 Generalist Graph Anomaly Detection Methods

Recent years have witnessed a rapid expansion of GGAD research. We include ***15*** specialized GGAD methods proposed between 2024 and 2026, including two methods from 2024, three from 2025, and ten from 2026, and provide their detailed descriptions below.

- **ARC** Liu et al. (2024) proposes an in-context learning framework for GGAD. It consists of a smoothness-based feature alignment module, an ego-neighbor residual graph encoder, and a cross-attentive in-context anomaly scoring module. The feature alignment module projects heterogeneous node features into a common space and reorders feature dimensions according to spatial smoothness. The residual encoder captures multi-hop differences between an ego node and its neighborhood. Given a few labeled normal nodes as context, ARC reconstructs query-node embeddings through cross-attention and uses the distance between the original and reconstructed embeddings as the anomaly score.
- **UNPrompt** Niu et al. (2024) trains a one-for-all GAD model on a single graph dataset and directly applies it to other graph datasets without retraining or fine-tuning. It first performs coordinate-wise normalization-based node attribute unification to align the dimensionality and semantics of node attributes across graphs. It then learns unified neighborhood prompts through latent node attribute prediction. The predictability of latent node attributes from neighborhood information is directly used as the anomaly measure, with normal nodes expected to exhibit higher predictability than anomalous nodes.
- **AnomalyGFM** Qiao et al. (2025) is a GAD-oriented graph foundation model supporting both zero-shot inference and few-shot prompt tuning. It represents each node by its representation residual, defined as the deviation of the node representation from its neighboring representations. During pre-training, data-independent learnable normal and abnormal class prototypes are aligned with the residual features, thereby learning graph-agnostic discriminative class representations. For zero-shot inference, anomaly scores are determined from the similarities between node residuals and the normal and abnormal prototypes. When a few labeled normal nodes are available, AnomalyGFM further refines the normal prototype through one-class prompt tuning.
- **Zero-GAD** Zheng et al. (2025) is a zero-shot GGAD framework designed to reduce semantic dependence across heterogeneous graph domains. It projects graph signals into the spectral domain and normalizes their frequency-space energy distributions. Then a graph autoencoder reconstructs node representations and the discrepancy between the original and reconstructed representations is used for anomaly scoring.
- **IA-GGAD** Zhang et al. (2025) explicitly models two cross-domain challenges, namely Feature Space Shift (FSS) and Graph Structure Shift (GSS). To address FSS, it introduces anomaly-driven graph invariant learning to obtain domain-invariant node representations. To address GSS, it develops structure-insensitive affinity learning to capture cross-domain structural correspondences through affinity-based representations. The two modules are jointly optimized, after which the trained detector can be directly applied to unseen graphs without target-domain retraining or fine-tuning.
- **DR-GGAD** Fu et al. (2026) focuses on Anomaly non-Discriminativity, which describes the overlap between normal and anomalous representations after cross-domain transfer. It introduces a

multi-scale hyper residual Center that measures node-to-center residual distances and an affinity-residual module that models local residual directional consistency. The two complementary residual signals are fused to obtain the final anomaly score, while model parameters remain frozen on the target graph.

- **OWLEYE** Zheng et al. (2026) is a zero-shot GAD framework that learns transferable patterns of normal behavior from multiple graphs. It first uses a cross-domain feature alignment module to harmonize heterogeneous feature distributions while preserving domain-specific semantics. It then adopts multi-domain multi-pattern dictionary learning to encode shared structural and attribute patterns. Finally, a truncated attention-based reconstruction module performs anomaly detection on unseen graph data without requiring target labels.
- **TA-GGAD** Zhang et al. (2026) is a testing-time adaptive framework for zero-shot generalist graph anomaly detection. It first constructs complementary anomaly scores from high-order residual representations and low-order structural affinity, and uses anomaly-disassortativity indicators to obtain a domain-aware score. During inference, its Testing-time Adapter generates pseudo-labels from multiple anomaly scores and learns reliability weights to adaptively combine them for the target graph, enabling target-specific adjustment without retraining or labeled target-domain data.
- **SAARCS** Liu et al. (2026d) is a universal GAD framework composed of feature alignment, adaptive graph representation learning, and contextual anomaly prediction. It aligns heterogeneous feature dimensions using composite spatial smoothness and learns node representations with an adaptive-hop attention encoder. A small set of normal context nodes is then used to estimate the abnormality of query nodes through contextual reconstruction.
- **TPCA-GAD** Yao et al. (2026) is a topology-prioritized zero-shot GAD framework that uses structural consistency as the main cross-domain detection signal. It first constructs multi-scale structural fingerprints to encode domain-agnostic node roles from local and multi-hop topology. Its Topology Preference-Consistency Aggregation mechanism evaluates structural compatibility between connected nodes and reweights neighborhood messages accordingly. Gated residual connections are further used to preserve structural information and alleviate over-smoothing.
- **GAD-MoRE** Zhao et al. (2026) models the geometric heterogeneity of graph anomaly patterns through a Mixture of Riemannian Experts. Each expert operates in a Riemannian space with a different curvature, allowing anomaly patterns with different geometric characteristics to be modeled in different spaces. An anomaly-aware multi-curvature feature alignment module maps raw node features into the corresponding Riemannian representations, while a memory-based dynamic router assigns inputs to suitable experts according to their historical reconstruction performance.
- **ProMoS** Xu et al. (2026) is an unsupervised generalist GAD framework that models normality from unlabeled graph data. It distills normality priors from a frozen self-supervised GNN teacher into a Mixture-of-Students architecture containing a shared global branch and lightweight personalized branches. Prototype-guided soft-label distillation aligns teacher and student representations in a shared prototype space. At inference time, anomalies are detected on unseen graphs using distillation bias together with prototype geometric deviation.
- **NeighborDiv** Wei et al. (2026) is a training-free zero-shot GGAD framework based on the Neighbor-to-Neighbor Diversity Paradigm. Instead of measuring node-to-neighbor consistency, it characterizes the internal dispersion of a node's neighborhood. Specifically, neighbor diversity is quantified by the variance of pairwise feature similarities among neighboring nodes, and the deviation of this statistic from a graph-level reference is directly used for anomaly scoring. The method requires neither model training nor source-graph data.
- **ReFi-GAD** Liu et al. (2026b) rethinks the use of dimensionality-based feature alignment in GGAD and replaces heterogeneous raw node features with a semantics-aware relational fingerprint. The fingerprint encodes anomaly-indicative contextual and structural relations in a common relational feature space. Based on fingerprint, a transformer-based domain-shared encoder learns domain-invariant representations, followed by an SNR-guided refinement module that performs domain-specific adaptation for unseen graphs.
- **TFM4GAD** Liu et al. (2026c) adapts pre-trained Tabular Foundation Models (TFMs) to graph anomaly detection through graph-to-tabular flattening. It represents each node using an augmented feature vector containing raw node attributes, Laplacian embeddings, explicit structural characteristics such as degree and PageRank, and anomaly-sensitive neighborhood aggregation features. The resulting table is directly processed by a pre-trained TFM, which uses a small set

of labeled nodes as in-context examples to infer anomaly probabilities for the remaining nodes without graph-specific training or gradient updates.

A.3 Graph Foundation Models

We further include five graph foundation models (GFMs), covering graph prompt learning and text-free multi-domain graph pre-training.

- **GraphPrompt** Liu et al. (2023) connects graph pre-training and downstream tasks through a unified prompting framework. It pre-trains a GNN with self-supervised link prediction and keeps the encoder frozen during downstream adaptation. A learnable prompt is applied to the graph readout operation to extract task-relevant information for few-shot prediction.
- **SAMGPT** Yu et al. (2025) is a text-free graph foundation model designed for multi-domain pre-training and cross-domain adaptation. It introduces learnable structure tokens into the GNN aggregation layers during pre-training to capture domain-specific structural patterns across source graphs. For a target graph, holistic prompts transfer shared structural knowledge, while specific prompts combine source-domain structure tokens for target-specific adaptation.
- **MDGFM** Wang et al. (2025a) focuses on aligning graph topology across domains. Instead of directly using the original graph structure, it learns a refined graph through a Graph Structure Learning module and uses a balance token to control the contribution of node features and topology. The refined representations are pre-trained across multiple domains and then transferred to a target graph through lightweight prompts.
- **BRIDGE** Yuan et al. (2025) focuses on controlling which source-domain knowledge should be transferred to a target graph. It first aligns node features from different source domains during pre-training. For downstream adaptation, a lightweight Mixture-of-Experts module selects and combines useful source knowledge according to the target graph. It further introduces a spectral regularizer to constrain cross-domain transfer and reduce harmful transfer.
- **GILT** Ma et al. (2025) is an LLM-free and tuning-free graph foundation model for in-context learning. It first uses a graph-native tokenizer to combine structure-aware node representations with class prototypes, converting labeled support examples and unlabeled queries into a unified token format. For a new target graph, GILT uses an in-context Transformer to learn the task from the support examples and predict the queries directly, without updating any model parameters.

B Detailed Description of Datasets

We include ***22*** attributed graph datasets spanning citation, social, e-commerce, crowdsourcing, and financial domains. Among them, 14 datasets contain real-world anomaly labels, while the remaining 8 datasets are constructed with injected anomalies. The dataset statistics are summarized in Table 5; below, we provide detailed descriptions of each dataset.

B.1 Organic Anomaly Datasets

- **Weibo** Zhao et al. (2020); Liu et al. (2022b) is constructed from Tencent Weibo and originally contains user, post, and hashtag information. Following the preprocessing adopted by BOND Liu et al. (2022b), users constitute the target nodes and two users are connected if they use the same hashtag. User features combine location information and bag-of-words representations of post content. Temporal posting information is used to identify suspicious behavior: users exhibiting at least five suspicious posting events are labeled as suspicious and treated as anomalies.
- **Reddit** Kumar et al. (2019); Liu et al. (2022b) is constructed from one month of activity on the Reddit platform as a user–subreddit interaction graph. The benchmark retains the most active users and subreddits, and textual posts are represented using Linguistic Inquiry and Word Count (LIWC) features. Users banned by the platform are treated as anomalous users.
- **Facebook** Xu et al. (2022) is a social network in which nodes represent users and edges denote social relationships between them. Anomalous nodes are synthetically constructed by either connecting selected nodes into manually generated dense clusters or assigning node attributes that differ substantially from those of their contextual neighbors.

Table 5: Statistics of the benchmark datasets. Undirected edges are counted once after removing self-loops and merging duplicates; directed edge counts are reported for Elliptic and DGraph-Fin. Injected datasets are measured after anomaly injection.

Dataset	#Nodes	#Edges	#Feat.	#Anom.	Ratio (%)	Avg. deg.	#CC	Spars. (%)	Hom. (%)	ARP
Organic Anomalies										
Facebook	1,081	27,552	576	25	2.31	50.98	1	98.03	99.71	+0.4150
Reddit	10,984	78,516	64	366	3.33	14.30	3	0.00	94.70	+0.1874
Weibo	8,405	377,271	400	868	10.33	89.77	2	0.04	97.66	+0.6855
Amazon	10,224	175,608	25	693	6.78	34.35	331	40.96	80.75	+0.0099
Amazon-all	11,944	4,398,392	25	821	6.87	736.50	1	41.56	95.36	-0.3601
YelpChi	23,831	49,315	32	1,217	5.11	4.14	7,308	0.09	99.64	+0.1422
YelpChi-all	45,941	3,846,979	32	6,674	14.53	167.47	13	0.10	77.30	+0.1528
Tolokers	11,758	519,000	10	2,566	21.82	88.28	1	48.03	59.45	-0.2074
Questions	48,921	153,540	301	1,460	2.98	6.28	1	58.46	83.96	-0.1403
T-Finance	39,357	21,222,543	10	1,803	4.58	1078.46	4	0.00	97.08	-0.6560
Elliptic[†]	203,769	234,355	166	4,545	9.76	1.15	49	0.00	95.37	-0.2565
DGraph-Fin[‡]	3,700,550	4,300,999	17	15,509	1.27	1.16	1	5.75	98.68	-0.0056
T-Social	5,781,065	73,105,508	10	174,280	3.01	25.29	3,437	13.81	62.39	-0.1291
Disney	124	335	28	6	4.84	5.40	1	8.06	94.93	+0.1384
Injected Anomalies										
Cora	2,708	5,802	1,433	150	5.54	4.29	73	98.72	90.83	+0.2455
CiteSeer	3,327	5,077	3,703	150	4.51	3.05	413	99.14	92.04	+0.1621
PubMed	19,717	46,423	500	600	3.04	4.71	1	90.05	94.56	+0.3542
ACM	16,484	74,073	8,337	597	3.62	8.99	25	99.65	92.84	+0.4374
BlogCatalog	5,196	172,779	8,189	300	5.77	66.50	1	99.03	88.12	+0.4809
Flickr	7,575	241,301	12,047	450	5.94	63.71	1	99.75	88.47	+0.4591
CS	18,333	83,993	6,805	600	3.27	9.16	1	99.08	93.72	+0.4191
Photo	7,650	120,653	745	450	5.88	31.54	132	64.15	88.97	+0.1177

Definitions. Avg. deg.: mean number of unique neighbors (outgoing neighbors for directed graphs). #CC: number of connected components (weakly connected for directed graphs), including isolated nodes. Spars.: percentage of exactly zero entries in the stored raw attributes. Hom.: percentage of same-label edges among edges with two labeled endpoints; undirected edges are counted once. Homophily is not adjusted for class imbalance. ARP uses raw attributes and outgoing neighbors for directed graphs; isolated nodes have a zero neighborhood mean.

[†]For Elliptic, the anomaly ratio and ARP use 46,564 labeled transactions; neighborhood residuals use the full graph. Its labeled induced graph contains 36,624 directed edges (73,248 adjacency entries after bidirectional symmetrization).

[‡]For DGraph-Fin, the anomaly ratio and ARP use 1,225,601 labeled nodes rather than all 3,700,550 nodes. Unlabeled nodes remain in the graph for degree and neighborhood computations.

- **Amazon** McAuley & Leskovec (2013); Dou et al. (2020a) is constructed from reviews in the Musical Instruments category of Amazon. Nodes represent users and are associated with handcrafted behavioral features. Users with more than 80% helpful votes are labeled as benign, whereas users with fewer than 20% helpful votes are labeled as fraudulent. The original graph contains three relations: U-P-U connects users reviewing at least one common product, U-S-U connects users giving the same star rating within one week, and U-V-U connects users with high mutual review-text similarity.
- **Amazon-all** McAuley & Leskovec (2013); Dou et al. (2020a). Amazon-all is constructed from the same Amazon fraud dataset as Amazon. Unlike the U-P-U-only version, Amazon-all merges the U-P-U, U-S-U, and U-V-U relations into a single homogeneous graph while retaining the original node features and fraud labels.
- **YelpChi** Rayana & Akoglu (2015b); Dou et al. (2020a) contains hotel and restaurant reviews collected from Yelp. Nodes correspond to reviews and are represented by handcrafted review features. Its labels are derived from Yelp's filtering mechanism: filtered reviews are used as suspicious or fraudulent instances, whereas recommended reviews are regarded as legitimate. The graph contains R-U-R, R-S-R, and R-T-R relations, connecting reviews according to common users, ratings, products, and posting time.
- **YelpChi-all** Rayana & Akoglu (2015b); Dou et al. (2020a) uses the same review features and labels as YelpChi but merges all three relational graphs—R-U-R, R-S-R, and R-T-R—into a single homogeneous graph.
- **Disney** Sánchez et al. (2013) is a movie co-purchase network derived from Amazon. Nodes correspond to movies, edges represent co-purchase relationships, and node attributes include information such as prices, ratings, and numbers of reviews. Its anomaly labels were manually annotated by high-school students and determined through majority voting.

- **Tolokers** Platonov et al. (2023) is constructed from the Toloka crowdsourcing platform. Nodes correspond to workers, and two workers are connected if they have worked on the same task. Node features are based on worker profile information and task-performance statistics.
- **Questions** Platonov et al. (2023). Questions is constructed from the Yandex Q question-answering platform. Nodes correspond to users, and an edge connects two users if one answered a question posted by the other. Node features are obtained by averaging word embeddings from user descriptions, together with an additional indicator for users without descriptions.
- **T-Finance** Tang et al. (2022) is a real-world financial transaction network released for graph anomaly detection. Nodes are anonymized accounts with behavioral features describing information such as registration duration, login activities, and interaction frequency, while an edge indicates that two accounts have transaction records. Human experts annotate accounts associated with fraud, money laundering, online gambling, and related abnormal activities as anomalies.
- **T-Social** Tang et al. (2022) is a large-scale social network released together with T-Finance. It uses the same type of account features and anomaly annotations as T-Finance, while its graph structure is defined by social relations: two accounts are connected if their friendship has been maintained for more than three months.
- **Elliptic** Weber et al. (2019b) is a temporal Bitcoin transaction graph in which nodes represent transactions and directed edges represent Bitcoin flows between transactions. The original dataset contains local transaction features and aggregated neighborhood features, and transactions are categorized as licit, illicit, or unknown. Following standard GAD preprocessing, the anomaly-detection task is defined on labeled transactions, with illicit transactions treated as anomalies and licit transactions as normal nodes.
- **DGraph-Fin** Huang et al. (2022) is a real-world dynamic financial graph released by Finvolution Group. Nodes represent user accounts, and an edge is created when one account lists another as an emergency contact. For users with borrowing records, accounts with overdue histories are labeled anomalous and those without overdue histories are labeled normal; the original dataset additionally contains a large number of background users without borrowing records.

B.2 Injected Anomaly Datasets

The following datasets were originally introduced for graph representation learning or node classification and do not contain native anomaly labels. Following established GAD protocols, anomaly labels are constructed by injecting structural and contextual anomalies Ding et al. (2019).

- **Cora** Sen et al. (2008) is a citation network in which nodes represent scientific publications and edges correspond to citation relationships. Each paper is represented by a sparse bag-of-words vector derived from its textual content.
- **CiteSeer** Sen et al. (2008) is a citation network of scientific publications. Nodes correspond to papers, edges represent citation links, and node attributes are bag-of-words representations of publication content.
- **PubMed** Sen et al. (2008) is a citation network of biomedical publications related to diabetes. Nodes correspond to papers and edges represent citation relationships. Node features are TF/IDF weighted word representations derived from publication content.
- **ACM** Tang et al. (2008) is an academic citation network derived from the ArnetMiner academic network. Nodes correspond to research papers, edges represent citation relationships, and node attributes encode textual information associated with the papers.
- **BlogCatalog** Ding et al. (2019); Tang & Liu (2009) is a social network of bloggers. Nodes correspond to users and edges represent social relationships between bloggers. Its high-dimensional node attributes are constructed from blog-related metadata and textual information, including tags and content associated with users.
- **Flickr** Ding et al. (2019); Tang & Liu (2009) is a social network collected from the Flickr photo-sharing platform. Nodes represent users and edges describe social relationships between users. Node attributes are constructed from tags associated with user-generated photo content.
- **CS** Shchur et al. (2018) is a co-authorship network derived from the Microsoft Academic Graph. Nodes represent authors and an edge connects two authors who have co-authored a paper. Node features are constructed from keywords of the authors' publications.

- **Photo** Shchur et al. (2018) is a product co-purchase network. Nodes correspond to products and edges connect products that are frequently purchased together. Node attributes are sparse bag-of-words representations extracted from product reviews.

Anomaly Injection. For Cora, CiteSeer, PubMed, ACM, BlogCatalog, Flickr, CS, and Photo, we follow the anomaly-injection protocol commonly adopted in graph anomaly detection Ding et al. (2019); Liu et al. (2021b). Structural anomalies are generated by randomly sampling nodes and connecting them to form small dense cliques. Contextual anomalies are generated by selecting a target node, randomly sampling a set of candidate nodes, and replacing the target node's attributes with those of the candidate having the largest feature-space distance from the target.

C Detailed Experimental Settings

This section provides additional details of the unified evaluation protocol, hyperparameter selection, feature processing, and computation resource.

C.1 Unified Evaluation Protocol

We evaluate all methods under the same cross-graph protocol. For every source–target configuration, the target graph is excluded from source-domain training or pre-training. ❶ Supervised methods are trained with labels from the designated source graphs, whereas unsupervised methods do not access source anomaly labels. ❷ GGAD methods further follow their original zero-shot or few-shot protocols; in particular, zero-shot methods are directly applied to unseen target graphs without target-domain training. ❸ Training-free and source-independent methods are evaluated directly according to their original inference procedures.

All methods output continuous anomaly scores for target nodes. We evaluate these scores using AUROC and AUPRC, with anomalous nodes treated as the positive class. Since both metrics operate directly on continuous scores, no decision threshold is introduced. Unless otherwise specified, all main results are reported as mean $\pm$ standard deviation over 11 random seeds (0–10).

C.2 Model Configuration and Hyperparameter Search

For specialized GGAD methods with publicly available official implementations, we directly use the configurations reported in their papers or official repositories without additional tuning. TPCA-GAD and SAARCS are the only exceptions because no official code is publicly available; we reimplement and tune them using the same protocol as the conventional baselines and GFMs.

For all tuned methods, we use Optuna TPE with 100 trials per method. Each candidate is evaluated with seed 0 on Reddit, Tolokers, Weibo, and Questions, and the configuration with the best average performance over these four graphs is selected. AUROC and AUPRC lead to very similar selections in our experiments. The selected configuration is then fixed for all 11 seeds in the main evaluation, without target-specific retuning.

For neural models, unless otherwise specified, the shared learning-rate candidates are $\mathcal{R} = \{10^{-6}, 10^{-5}, 5\times10^{-5}, 10^{-4}, 5\times10^{-4}, 10^{-3}, 5\times10^{-3}, 10^{-2}, 10^{-1}\}$, and the weight-decay candidates are $\mathcal{W} = \{0, 10^{-6}, 10^{-5}, 5\times10^{-5}, 10^{-4}, 5\times10^{-4}, 10^{-3}, 5\times10^{-3}, 10^{-2}\}$. Tables 6 and 7 summarize the method-specific search spaces.

C.3 Feature Preprocessing for Conventional Baselines

The conventional baselines were originally designed for learning and evaluation on a single graph, where the input feature dimension remains unchanged. In our cross-graph setting, however, source and target graphs can have substantially different feature dimensions. We therefore introduce a common feature preprocessing step only for these baselines, so that the same trained model can be applied to unseen graphs.

Specifically, we first use truncated SVD to project the node features of each graph into a common dimension, considering output dimensions of 8 and 64. After dimensionality reduction, we compare four simple ways of processing the projected features: (1) using the SVD features directly;

Table 6: Hyperparameter search spaces for conventional baselines. h, L, p, and E denote hidden dimension, number of layers, dropout rate, and training epochs, respectively.

Method	Search space
MLP	$h \in \{32, 64, 128\}$; $L \in \{1, 2, 3, 4\}$; $p \in \{0, 0.3, 0.5\}$; $E \in \{100, 200, 300\}$.
GCN	$h \in \{32, 64, 128\}$; $L \in \{1, 2, 3\}$; $p \in \{0, 0.3, 0.5\}$; $E \in \{100, 200, 300\}$.
GAT	$h \in \{8, 16, 32\}$; $L \in \{1, 2, 3\}$; heads $\in \{1, 2, 4, 8\}$; $p \in \{0, 0.3, 0.6\}$; $E \in \{100, 200, 300\}$.
RF-Graph	trees $\in \{25, 50, 100, 150, 200\}$; aggregation depth $\in \{1, 2, 3, 4\}$; aggregation $\in \{\text{sum}, \text{mean}, \text{max}\}$; sample fraction $\in \{0.5, 0.75, 1\}$.
XGB-Graph	trees $\in \{25, 50, 100, 150, 200\}$; learning rate $\in \{0.05, 0.1, 0.2, 0.3, 0.5\}$; regularization $\in \{0, 1, 10\}$; sample fraction $\in \{0.5, 0.75, 1\}$; aggregation depth $\in \{1, 2, 3, 4\}$.
CoLA	$h \in \{32, 64, 128\}$; batch size $\in \{128, 256, 300\}$; subgraph size $\in \{4, 8\}$; negative ratio $\in \{1, 2\}$; $E \in \{100, 200, 400\}$.
DOMINANT	$h \in \{16, 32, 64, 128\}$; $p \in \{0, 0.1, 0.3, 0.5\}$; reconstruction weight $\in \{0.2, 0.4, 0.6, 0.8\}$; $E \in \{100, 200, 300\}$.
TAM	$h \in \{32, 64, 128\}$; truncations $\in \{1, 4, 7\}$; ensemble size $\in \{1, 3\}$; regularization weight $\in \{0, 0.5, 1\}$; $E \in \{100, 300, 500\}$.
BWGNN	$h \in \{32, 64, 128\}$; polynomial order $\in \{2, 3, 4, 5\}$; $E \in \{100, 200, 300\}$.
GHRN	$h \in \{32, 64, 128\}$; polynomial order $\in \{2, 3, 5\}$; edge-deletion ratio $\in \{0, 0.015, 0.03, 0.1\}$; $E_{\text{pre}} \in \{50, 100, 200\}$; $E \in \{100, 200, 300\}$.

Table 7: Hyperparameter search spaces for GGAD methods without official implementations and graph foundation models. h, L, and E denote hidden dimension, number of layers, and training epochs, respectively.

Method	Search space
GGAD methods without official implementations	
TPCA-GAD	attribute width $\in \{8, 16, 32\}$; $h \in \{64, 128, 256\}$; $L \in \{1, 2, 3\}$; $\tau \in \{0.25, 0.5, 1\}$; consistency weight $\in \{0, 0.1, 0.5, 1\}$; score weight $\in \{0, 0.25, 0.5, 0.75, 1\}$; $E \in \{50, 100, 200\}$.
SAARCS	feature width $\in \{32, 64, 128\}$; $h \in \{128, 256, 512, 1024\}$; hops $\in \{1, 2, 3, 4, 5\}$; cosine margin $\in \{0.3, 0.5, 0.7\}$; spatial weight $\in \{0.25, 0.5, 0.75\}$; $E \in \{50, 100, 200\}$.
Graph foundation models	
GraphPrompt	$h \in \{64, 128\}$; $L \in \{2, 3\}$; $p \in \{0.3, 0.5\}$; queries/class $\in \{64, 128\}$; $E \in \{50, 100, 200\}$.
GILT	$h \in \{96, 192\}$; $L_{\text{GCN}} \in \{2, 4, 6\}$; $L_{\text{Transformer}} \in \{8, 12\}$; attention heads $\in \{4, 6, 8\}$; FFN expansion factor $\in \{2, 4\}$; features/group $\in \{2, 4\}$; Transformer and GCN dropout $\in \{0, 0.1\}$; GCN normalization and residual switches $\in \{\text{on}, \text{off}\}$; feature positional embedding $\in \{\text{subspace}, \text{learned}\}$; queries/class $\in \{64, 128\}$; gradient clipping $\in \{0.5, 1, 2\}$; $E \in \{25, 50, 100\}$.
SAMGPT	$h \in \{64, 128\}$; $L \in \{2, 3\}$; prompt combination $\in \{\text{add}, \text{multiply}\}$; $\alpha, \beta \in \{0.5, 1, 2\}$; $E_{\text{pre}} \in \{100, 200\}$; $E_{\text{adapt}} \in \{100, 200, 400\}$.
MDGFM	$h \in \{64, 128, 256\}$; $L \in \{2, 3\}$; kNN size $\in \{5, 15\}$; graph-mixing weight $\in \{0.25, 0.5, 0.75\}$; alignment temperature $\in \{0.1, 0.2\}$; prototype temperature $\in \{0.5, 1\}$; $E_{\text{pre}} \in \{30, 60, 100\}$; $E_{\text{adapt}} \in \{100, 200\}$.
BRIDGE	$h \in \{64, 128, 256\}$; $L \in \{2, 3\}$; variance weight $\in \{10^2, 10^4, 10^6\}$; entropy weight $\in \{0.05, 0.2\}$; regularization weight $\in \{0.1, 1, 10\}$; negatives $\in \{25, 50\}$; $E_{\text{pre}}, E_{\text{adapt}} \in \{100, 200\}$.

(2) subtracting the feature-wise mean from each dimension; (3) standardizing each dimension by subtracting its mean and dividing by its standard deviation; and (4) following ARC, reordering the projected feature dimensions according to their graph smoothness. The last operation changes only the ordering of feature dimensions and does not modify the graph structure or node set.

We select the best feature preprocessing based on the mean AUROC over the four tuning graphs and use it throughout the main evaluation. No additional feature preprocessing search is performed for the specialized GGAD methods or GFMs, for which we retain their original feature handling procedures.

C.4 Implementation and Reproducibility

We implement the benchmark in Python using PyTorch Paszke et al. (2019) and PyTorch Geometric Fey & Lenssen (2019). For methods with publicly available official implementations, we build the benchmark upon their released code while preserving the original model architectures, training objectives, and inference procedures. TPCA-GAD and SAARCS are reimplemented according to their papers because no official implementations are publicly available. All methods are integrated into a unified pipeline for data loading, source–target construction, repeated evaluation, and metric computation.

All experiments are conducted on a Linux server equipped with an Intel Xeon Platinum 8173M CPU, approximately 252 GB of RAM, and NVIDIA GeForce RTX 3090 GPUs with 24 GB memory.

Table 8: AUROC (%) on unseen organic targets under the single-source organic-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	**98.9** ± 0.1	56.5 ± 0.3	22.1 ± 0.0	43.2 ± 2.8	46.0 ± 0.3	54.4 ± 0.7	52.4 ± 1.2	53.6 ± 1.1	63.2 ± 0.6	72.7 ± 1.8	35.7 ± 0.5	54.4	12
CoLA	53.3 ± 10.9	54.0 ± 2.8	54.9 ± 13.5	48.7 ± 2.9	49.3 ± 3.2	52.8 ± 4.7	48.9 ± 2.6	49.8 ± 3.8	50.7 ± 5.8	54.4 ± 7.7	50.9 ± 17.1	51.6	21
TAM	60.6 ± 13.1	55.2 ± 0.4	75.9 ± 0.6	51.4 ± 0.7	41.6 ± 0.9	50.8 ± 0.1	49.5 ± 0.3	53.7 ± 1.9	**65.9** ± 1.1	68.9 ± 1.4	49.7 ± 0.3	56.7	9
BWGNN	39.9 ± 25.9	50.5 ± 7.8	51.3 ± 17.7	50.8 ± 4.6	48.9 ± 3.8	51.9 ± 8.0	49.6 ± 8.8	51.8 ± 10.1	51.2 ± 6.5	47.6 ± 11.1	51.3 ± 7.8	49.5	29
GHRN	55.6 ± 30.3	47.0 ± 5.0	66.0 ± 19.1	50.8 ± 5.8	48.0 ± 5.8	49.2 ± 6.4	48.2 ± 5.3	49.5 ± 12.8	47.7 ± 6.4	54.5 ± 15.1	54.4 ± 7.5	51.9	20
MLP	57.4 ± 15.7	51.5 ± 4.0	65.3 ± 20.5	49.7 ± 4.1	49.9 ± 3.8	57.2 ± 7.0	51.6 ± 4.0	42.0 ± 15.4	52.2 ± 6.2	48.4 ± 7.7	48.8 ± 10.4	52.2	18
GCN	43.6 ± 28.7	50.9 ± 8.1	62.5 ± 20.7	52.9 ± 3.8	51.1 ± 2.9	44.7 ± 8.4	48.7 ± 9.9	52.5 ± 6.2	47.8 ± 3.8	51.6 ± 5.7	50.0 ± 2.5	50.6	24
GAT	41.8 ± 21.1	56.4 ± 4.5	48.3 ± 12.9	55.8 ± 6.6	48.7 ± 1.4	49.4 ± 9.8	50.2 ± 5.5	49.7 ± 14.2	51.2 ± 3.2	52.7 ± 15.7	54.0 ± 6.5	50.7	23
RF-Graph	23.9 ± 5.8	55.4 ± 0.8	50.5 ± 11.8	54.5 ± 2.2	49.1 ± 1.1	58.7 ± 2.9	52.6 ± 1.6	49.8 ± 1.4	50.2 ± 1.2	49.6 ± 4.9	65.8 ± 8.9	50.9	22
XGB-Graph	34.2 ± 4.1	50.0 ± 0.0	60.5 ± 3.5	54.7 ± 3.0	50.1 ± 0.4	56.8 ± 3.8	50.4 ± 1.2	50.2 ± 1.2	48.2 ± 1.4	34.5 ± 3.5	64.8 ± 3.3	50.4	25
Generalist GAD Methods													
ARC	89.1 ± 0.8	**59.0** ± 1.1	64.0 ± 3.2	49.0 ± 0.5	58.9 ± 1.7	50.6 ± 4.5	57.4 ± 0.4	25.5 ± 3.2	47.9 ± 3.8	49.2 ± 5.3	43.5 ± 7.8	54.0	15
UNPrompt	90.0 ± 1.0	53.4 ± 0.5	79.9 ± 2.2	58.3 ± 1.1	56.2 ± 1.5	46.7 ± 1.4	49.1 ± 0.8	18.9 ± 0.7	39.7 ± 1.3	36.5 ± 1.2	62.0 ± 2.0	53.7	16
AnomalyGFM	63.8 ± 4.9	55.0 ± 4.0	82.7 ± 5.9	46.9 ± 0.4	58.7 ± 1.5	54.9 ± 7.6	53.6 ± 1.4	56.7 ± 1.4	47.2 ± 5.5	49.2 ± 2.5	63.1 ± 10.1	57.4	8
ZeroGAD	30.3 ± 4.3	50.6 ± 3.0	86.0 ± 1.7	OOM	OOM	43.3 ± 1.2	OOM	OOM	OOM	OOM	60.5 ± 3.7	54.1	13
IA-GGAD	91.7 ± 0.4	56.8 ± 1.7	76.8 ± 2.4	50.6 ± 1.6	45.0 ± 1.3	50.5 ± 3.7	57.0 ± 0.4	23.9 ± 1.1	48.6 ± 1.7	50.9 ± 3.1	42.6 ± 3.4	54.0	14
DR-GGAD	53.7 ± 1.0	57.8 ± 0.5	50.3 ± 2.3	41.4 ± 1.3	54.3 ± 0.8	43.5 ± 0.6	52.9 ± 0.3	42.3 ± 0.6	50.0 ± 0.7	59.7 ± 2.3	40.0 ± 0.5	49.6	28
OWLEYE	89.5 ± 0.3	57.4 ± 0.6	32.0 ± 1.8	48.6 ± 1.2	55.5 ± 0.6	50.3 ± 5.9	48.4 ± 1.8	22.6 ± 7.6	43.4 ± 1.4	50.3 ± 1.7	49.5 ± 1.5	49.8	26
TA-GGAD	91.5 ± 0.3	54.1 ± 2.8	65.6 ± 2.9	51.1 ± 1.9	56.2 ± 4.1	52.1 ± 5.1	57.8 ± 0.3	24.5 ± 1.0	55.4 ± 4.5	51.4 ± 1.2	44.2 ± 1.3	54.9	11
ProMoS	92.8 ± 0.1	53.0 ± 3.1	53.0 ± 3.8	50.1 ± 2.9	54.2 ± 3.8	38.9 ± 2.8	48.8 ± 0.6	14.1 ± 0.3	41.0 ± 0.4	47.1 ± 0.3	53.1 ± 0.1	49.7	27
SAARCS	85.7 ± 2.8	54.1 ± 2.8	60.7 ± 3.8	42.1 ± 0.5	58.4 ± 1.7	52.7 ± 3.5	54.2 ± 1.0	41.7 ± 5.2	55.4 ± 6.0	60.6 ± 6.6	49.7 ± 4.4	55.9	10
TPCA-GAD	68.8 ± 9.7	48.4 ± 1.3	65.5 ± 1.0	66.6 ± 2.6	51.3 ± 0.7	55.7 ± 1.6	45.3 ± 1.4	41.4 ± 4.9	51.5 ± 2.5	32.3 ± 3.8	55.4 ± 1.9	52.9	17
GAD-MoRE	87.8 ± 0.4	55.8 ± 1.3	21.5 ± 0.1	57.0 ± 0.3	51.2 ± 0.4	48.0 ± 1.2	55.5 ± 0.3	31.1 ± 0.4	50.3 ± 0.5	60.0 ± 0.3	55.0 ± 0.3	52.1	19
NeighborDiv	79.9 ± 0.7	51.2 ± 0.0	68.8 ± 0.7	58.2 ± 0.0	50.7 ± 0.3	52.9 ± 0.0	45.5 ± 0.0	56.6 ± 0.0	52.1 ± 0.0	56.4 ± 0.0	72.5 ± 0.0	58.6	7
REFIGAD	82.4 ± 6.4	53.9 ± 5.9	67.5 ± 1.6	71.1 ± 10.6	53.3 ± 2.8	54.1 ± 3.1	55.0 ± 6.8	74.5 ± 2.7	54.0 ± 6.2	86.6 ± 2.5	**78.6** ± 6.8	**66.5**	**3**
TFM4GAD	97.5 ± 1.6	55.5 ± 4.6	**92.5** ± 3.9	**74.0** ± 7.1	**61.0** ± 3.2	**67.2** ± 8.3	**62.9** ± 2.8	**85.0** ± 2.9	59.9 ± 6.9	**93.0** ± 4.7	–	**74.8**	**1**
Graph Foundation Models													
GraphPrompt	87.2 ± 6.5	51.8 ± 3.9	73.1 ± 2.6	68.4 ± 1.6	51.0 ± 1.6	48.7 ± 5.9	49.7 ± 5.3	50.7 ± 8.9	51.2 ± 4.7	81.0 ± 3.8	44.0 ± 25.9	59.7	6
SAMGPT	96.4 ± 5.4	58.7 ± 4.2	79.1 ± 4.3	66.8 ± 6.1	51.7 ± 2.5	61.1 ± 6.0	54.6 ± 3.6	75.7 ± 5.8	59.1 ± 9.7	86.8 ± 5.3	47.3 ± 27.0	**67.0**	**2**
MDGFM	88.7 ± 4.6	54.3 ± 4.5	88.5 ± 4.5	58.6 ± 4.9	53.0 ± 3.4	65.7 ± 4.0	57.2 ± 5.1	75.3 ± 7.7	59.5 ± 7.9	70.4 ± 4.4	49.7 ± 25.7	65.5	4
BRIDGE	96.2 ± 3.5	57.9 ± 3.7	81.6 ± 1.0	63.1 ± 5.2	50.5 ± 0.8	64.1 ± 4.1	56.6 ± 3.6	71.4 ± 7.7	59.4 ± 8.6	76.5 ± 4.9	43.4 ± 23.7	65.5	5
GILT	33.0 ± 17.8	54.7 ± 3.1	31.8 ± 14.7	51.8 ± 10.2	47.0 ± 4.6	52.8 ± 6.4	50.1 ± 3.4	46.4 ± 9.4	50.9 ± 9.2	48.4 ± 12.4	64.9 ± 16.5	48.3	30

The server runs Ubuntu 22.04.4 LTS. Our main software environment consists of Python 3.10.14, PyTorch 2.2.2 with CUDA 12.1, PyTorch Geometric 2.6.1, DGL 2.4.0, scikit-learn 1.6.0, Optuna 4.9.0, and XGBoost 3.2.0.

We release the benchmark code, data-processing scripts, complete hyperparameter configurations, and experiment scripts to facilitate reproducibility.

D Additional Experimental Results

This section provides additional results for RQ2–RQ4. It includes complete AUROC and AUPRC results, detailed efficiency measurements, and robustness experiments using alternative source graphs. Unless otherwise specified, all performance results are reported as mean ± standard deviation over 11 random seeds.

D.1 Additional Results for Transferability (RQ2)

We report complete AUROC results for the source configurations considered in RQ2.

D.1.1 Single-Source Results

Using Facebook and Cora as the organic and injected single sources, respectively, Tables 8–11 report all four source–target anomaly-type combinations.

D.1.2 Multi-Source Results

Using {Facebook, T-Finance, Amazon} as organic sources and {Cora, CiteSeer, ACM, BlogCatalog} as injected sources, Tables 12 report the remaining multi-source transfer settings.

Table 9: AUROC (%) on unseen organic targets under the single-source injected-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	**98.8** ± 0.1	56.4 ± 0.1	22.3 ± 0.1	42.7 ± 3.0	46.1 ± 0.3	55.3 ± 0.2	56.9 ± 0.3	53.0 ± 0.7	63.0 ± 0.8	70.4 ± 1.8	34.9 ± 0.5	54.5	10
CoLA	50.2 ± 11.5	54.3 ± 3.1	53.6 ± 13.5	48.3 ± 2.9	49.4 ± 3.3	53.4 ± 4.8	48.8 ± 2.4	51.1 ± 3.6	49.9 ± 5.7	55.0 ± 7.3	52.2 ± 17.1	51.5	18
TAM	41.4 ± 3.0	54.5 ± 0.3	75.4 ± 0.5	50.3 ± 1.1	42.1 ± 0.7	51.0 ± 0.1	51.1 ± 1.0	53.4 ± 0.4	63.8 ± 1.9	57.6 ± 7.9	46.3 ± 2.9	53.4	14
BWGNN	41.6 ± 25.5	50.5 ± 7.8	50.1 ± 17.4	50.6 ± 4.6	48.5 ± 3.8	52.3 ± 8.2	51.3 ± 8.4	51.2 ± 10.2	51.4 ± 6.4	49.5 ± 11.0	51.1 ± 8.3	49.8	24
GHRN	63.5 ± 27.3	46.6 ± 5.0	62.6 ± 18.0	50.3 ± 6.1	47.6 ± 5.9	50.8 ± 7.5	49.7 ± 5.7	46.1 ± 12.2	47.7 ± 6.1	59.4 ± 14.3	53.6 ± 7.7	52.5	16
MLP	66.1 ± 15.8	48.9 ± 4.6	75.2 ± 14.7	45.9 ± 4.2	51.0 ± 3.6	57.4 ± 7.7	48.8 ± 3.5	42.9 ± 14.7	51.2 ± 4.9	47.8 ± 7.8	43.8 ± 9.4	52.6	15
GCN	48.2 ± 26.8	46.5 ± 8.0	59.3 ± 21.9	51.6 ± 3.8	50.4 ± 3.4	47.6 ± 8.1	48.1 ± 10.1	52.0 ± 5.9	47.6 ± 3.8	51.7 ± 5.6	50.5 ± 1.5	50.3	23
GAT	48.4 ± 23.5	53.0 ± 5.0	46.3 ± 13.4	44.8 ± 6.5	49.5 ± 1.7	54.6 ± 8.0	49.3 ± 5.6	52.8 ± 13.3	52.5 ± 3.6	57.7 ± 15.5	48.5 ± 9.0	50.7	21
RF-Graph	4.7 ± 0.2	50.0 ± 1.8	77.1 ± 1.3	46.9 ± 0.8	46.5 ± 0.2	56.3 ± 1.4	49.9 ± 0.3	44.7 ± 0.5	47.1 ± 0.3	77.5 ± 0.5	14.8 ± 2.5	46.9	30
XGB-Graph	45.1 ± 12.4	50.3 ± 2.5	68.8 ± 4.9	33.9 ± 0.6	47.1 ± 0.2	46.7 ± 1.5	42.5 ± 1.1	45.2 ± 1.6	46.6 ± 0.4	57.0 ± 5.2	44.7 ± 9.7	48.0	29
Generalist GAD Methods													
ARC	89.4 ± 0.4	58.7 ± 1.0	63.8 ± 2.7	48.6 ± 0.5	58.7 ± 1.8	50.6 ± 4.6	57.4 ± 0.4	25.4 ± 3.1	47.9 ± 3.8	49.6 ± 5.7	48.0 ± 8.3	54.4	12
UNPrompt	45.6 ± 1.8	55.6 ± 1.5	58.0 ± 10.5	47.4 ± 5.7	51.4 ± 2.0	48.2 ± 4.3	45.2 ± 1.6	46.0 ± 4.3	**64.3** ± 0.9	50.3 ± 9.3	49.0 ± 5.8	51.0	19
AnomalyGFM	31.1 ± 10.0	50.0 ± 3.7	49.1 ± 6.6	46.8 ± 0.6	55.7 ± 2.5	46.9 ± 10.3	52.8 ± 2.3	59.7 ± 1.1	41.1 ± 8.3	46.5 ± 2.7	63.8 ± 4.6	49.4	27
ZeroGAD	27.1 ± 1.4	51.7 ± 1.7	80.4 ± 2.4	OOM	OOM	44.3 ± 1.1	OOM	OOM	OOM	OOM	50.3 ± 3.1	50.8	20
IA-GGAD	91.8 ± 0.4	56.7 ± 1.7	75.2 ± 2.6	50.6 ± 0.8	45.2 ± 1.2	51.0 ± 3.5	56.9 ± 0.5	24.0 ± 1.1	48.5 ± 1.7	50.7 ± 3.1	42.6 ± 3.5	53.9	13
DR-GGAD	53.6 ± 1.2	58.0 ± 0.5	50.5 ± 2.3	41.7 ± 1.4	54.4 ± 0.9	43.7 ± 0.6	52.9 ± 0.3	42.3 ± 0.6	49.9 ± 0.7	60.0 ± 2.2	40.0 ± 0.5	49.7	25
OWLEYE	90.1 ± 1.0	57.1 ± 1.0	31.9 ± 1.7	47.9 ± 1.3	55.7 ± 0.7	47.8 ± 5.8	47.6 ± 0.4	21.9 ± 4.4	42.7 ± 1.4	51.4 ± 3.8	49.1 ± 1.5	49.4	28
TA-GGAD	91.5 ± 0.3	55.1 ± 2.0	62.3 ± 4.0	50.9 ± 1.5	56.1 ± 2.9	51.7 ± 5.1	57.7 ± 0.3	24.6 ± 1.0	54.7 ± 4.4	51.4 ± 1.3	44.6 ± 1.4	54.6	9
ProMoS	92.9 ± 0.2	55.3 ± 1.8	55.3 ± 5.4	47.0 ± 2.5	55.6 ± 3.2	35.8 ± 1.4	48.5 ± 0.7	14.0 ± 0.2	41.1 ± 0.4	47.3 ± 0.2	53.1 ± 0.1	49.6	26
SAARCS	85.8 ± 2.6	54.1 ± 2.8	60.8 ± 3.8	42.1 ± 0.5	58.4 ± 1.7	52.6 ± 3.5	54.2 ± 1.0	43.1 ± 5.1	55.4 ± 6.0	60.6 ± 6.6	51.0 ± 6.4	56.2	8
TPCA-GAD	88.6 ± 3.8	49.4 ± 2.5	66.0 ± 2.0	67.8 ± 2.5	51.5 ± 0.5	57.7 ± 3.5	44.7 ± 2.3	37.4 ± 4.2	49.5 ± 1.2	36.4 ± 7.7	50.6 ± 2.2	54.5	11
GAD-MoRE	88.1 ± 0.4	56.0 ± 0.8	21.9 ± 0.1	55.7 ± 0.6	50.7 ± 0.4	39.9 ± 0.2	55.8 ± 0.3	33.1 ± 0.5	51.0 ± 0.3	59.9 ± 0.4	56.0 ± 0.3	51.6	17
NeighborDiv	79.9 ± 0.7	51.2 ± 0.0	68.8 ± 0.7	58.2 ± 0.0	50.7 ± 0.3	52.9 ± 0.0	45.5 ± 0.0	56.6 ± 0.0	52.1 ± 0.0	56.4 ± 0.0	72.5 ± 0.0	58.6	7
REFIGAD	84.6 ± 3.0	54.0 ± 4.6	65.7 ± 2.2	69.3 ± 6.9	54.7 ± 3.9	54.1 ± 3.5	57.7 ± 5.8	72.2 ± 6.6	54.1 ± 3.3	87.7 ± 1.5	**80.3** ± 5.9	**66.7**	**3**
TFM4GAD	97.5 ± 1.6	55.5 ± 4.6	**92.5** ± 3.9	**74.0** ± 7.1	**61.0** ± 3.2	**67.2** ± 8.3	**62.9** ± 2.8	**85.0** ± 2.9	59.9 ± 6.9	**93.0** ± 4.7	–	**74.8**	**1**
Graph Foundation Models													
GraphPrompt	83.3 ± 9.9	51.1 ± 6.2	66.4 ± 7.9	62.1 ± 9.7	50.1 ± 1.3	54.1 ± 9.1	50.2 ± 5.9	52.6 ± 8.5	49.4 ± 4.0	76.9 ± 5.9	70.1 ± 18.3	60.6	6
SAMGPT	96.3 ± 5.4	**58.8** ± 4.2	78.7 ± 4.7	66.9 ± 6.0	51.7 ± 2.5	61.6 ± 6.0	54.5 ± 3.6	75.7 ± 5.6	59.1 ± 9.8	87.0 ± 5.5	47.3 ± 26.9	**67.1**	**2**
MDGFM	87.5 ± 4.5	53.1 ± 3.1	88.4 ± 4.6	56.2 ± 4.8	52.6 ± 4.3	63.0 ± 6.9	58.4 ± 4.9	72.4 ± 7.5	57.3 ± 9.0	68.9 ± 5.8	44.7 ± 26.7	63.9	5
BRIDGE	94.7 ± 2.6	56.5 ± 5.5	81.8 ± 0.8	60.6 ± 4.9	50.6 ± 0.8	64.2 ± 4.7	56.3 ± 3.8	70.7 ± 7.5	59.3 ± 8.3	78.5 ± 3.9	42.9 ± 28.6	65.1	4
GILT	53.6 ± 21.0	51.2 ± 6.3	50.0 ± 23.6	46.0 ± 7.0	49.3 ± 5.3	51.8 ± 8.4	49.8 ± 5.2	51.1 ± 11.8	47.5 ± 6.3	49.6 ± 11.2	54.3 ± 20.3	50.4	22

Table 10: AUROC (%) on unseen injected targets under the single-source organic-to-injected setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	88.9 ± 0.6	74.6 ± 0.8	88.3 ± 0.3	64.1 ± 0.2	79.0
CoLA	52.8 ± 5.1	50.5 ± 5.0	51.8 ± 4.6	49.4 ± 9.3	51.1
TAM	60.2 ± 2.5	48.9 ± 3.5	71.2 ± 1.6	56.9 ± 2.5	59.3
BWGNN	48.5 ± 13.8	50.0 ± 11.6	50.5 ± 13.4	49.8 ± 7.5	49.7
GHRN	51.5 ± 13.0	58.0 ± 11.9	54.5 ± 12.7	51.5 ± 7.8	53.9
MLP	49.7 ± 9.6	46.5 ± 15.0	55.2 ± 12.6	54.9 ± 10.8	51.6
GCN	46.0 ± 7.0	41.2 ± 4.9	44.9 ± 5.7	47.1 ± 7.1	44.8
GAT	47.8 ± 8.8	51.2 ± 4.8	48.8 ± 7.0	47.8 ± 2.7	48.9
RF-Graph	48.6 ± 2.6	51.3 ± 4.2	48.4 ± 1.8	48.1 ± 2.0	49.1
XGB-Graph	49.6 ± 0.9	50.0 ± 0.1	50.1 ± 0.1	49.9 ± 0.4	49.9
ARC	85.4 ± 0.2	73.7 ± 1.5	83.0 ± 0.4	75.6 ± 1.0	79.4
UNPrompt	70.5 ± 0.9	69.5 ± 0.3	70.7 ± 0.6	64.8 ± 0.9	68.9
AnomalyGFM	47.8 ± 1.8	51.4 ± 1.6	54.5 ± 2.4	52.5 ± 2.4	51.6
ZeroGAD	OOM	49.6 ± 0.7	OOM	74.9 ± 1.2	–
IA-GGAD	89.5 ± 0.5	69.1 ± 1.2	93.5 ± 0.4	77.3 ± 2.0	**82.3**

Method	PubMed	Flickr	CS	Photo	Avg.
DR-GGAD	**96.0** ± 0.3	73.7 ± 0.4	**98.0** ± 0.2	68.4 ± 1.2	**84.0**
OWLEYE	78.3 ± 0.2	**79.4** ± 4.8	83.1 ± 2.2	75.2 ± 0.4	79.0
TA-GGAD	87.8 ± 0.6	76.5 ± 0.8	89.0 ± 0.9	76.0 ± 3.1	**82.3**
ProMoS	81.0 ± 0.7	69.2 ± 1.0	76.4 ± 0.7	68.2 ± 2.2	73.7
SAARCS	84.2 ± 0.3	76.0 ± 0.7	82.8 ± 0.9	**78.3** ± 2.3	80.3
TPCA-GAD	57.3 ± 6.9	44.3 ± 1.8	68.9 ± 2.0	47.5 ± 2.0	54.5
GAD-MoRE	82.4 ± 0.3	75.0 ± 0.3	83.0 ± 0.2	78.2 ± 0.3	79.7
NeighborDiv	72.1 ± 0.0	44.8 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	67.3
REFIGAD	88.8 ± 5.7	74.4 ± 6.9	90.8 ± 3.9	67.4 ± 3.7	80.3
TFM4GAD	74.6 ± 3.7	55.4 ± 4.9	75.6 ± 3.2	76.1 ± 2.8	70.4
GraphPrompt	62.6 ± 7.7	52.2 ± 5.0	54.6 ± 5.7	52.3 ± 7.0	55.4
SAMGPT	75.0 ± 5.9	65.1 ± 12.3	68.8 ± 7.3	57.8 ± 6.1	66.7
MDGFM	54.7 ± 6.0	64.6 ± 9.1	58.7 ± 7.7	62.9 ± 4.5	60.2
BRIDGE	57.8 ± 5.9	57.9 ± 14.0	57.0 ± 5.4	53.5 ± 2.3	56.5
GILT	50.4 ± 7.6	47.4 ± 13.9	46.5 ± 9.5	52.1 ± 10.0	49.1

D.2 Additional Results for Anomaly Retrieval (RQ3)

We further report the complete AUPRC results under the same source configurations. These tables complement the retrieval analysis in the main paper and allow direct comparison across source anomaly types and source quantities.

D.2.1 Single-Source Results

Tables 13–16 report AUPRC under all four single-source source–target combinations.

Table 11: AUROC (%) on unseen injected targets under the single-source injected-to-injected setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	87.9 ± 0.4	75.2 ± 0.1	87.6 ± 0.1	63.9 ± 0.2	78.6
CoLA	55.1 ± 4.9	51.0 ± 5.1	51.8 ± 4.6	51.1 ± 9.1	52.3
TAM	68.2 ± 2.0	50.2 ± 2.2	75.1 ± 2.2	67.4 ± 2.3	65.2
BWGNN	52.5 ± 14.3	50.1 ± 12.3	53.7 ± 13.7	51.1 ± 7.4	51.9
GHRN	54.8 ± 12.6	60.0 ± 12.0	56.6 ± 12.3	53.4 ± 8.0	56.2
MLP	55.7 ± 8.5	49.6 ± 12.8	61.8 ± 11.1	57.9 ± 6.7	56.3
GCN	51.4 ± 8.9	48.7 ± 9.3	51.8 ± 8.0	51.6 ± 7.9	50.9
GAT	59.1 ± 7.1	47.8 ± 6.3	55.8 ± 6.8	52.1 ± 3.3	53.7
RF-Graph	81.8 ± 0.9	40.6 ± 2.7	79.3 ± 1.1	67.8 ± 1.4	67.4
XGB-Graph	60.2 ± 5.2	62.3 ± 4.4	61.6 ± 3.4	61.7 ± 2.5	61.5
ARC	85.4 ± 0.2	73.7 ± 1.5	83.1 ± 0.4	75.5 ± 1.0	79.4
UNPrompt	80.5 ± 2.8	69.2 ± 0.5	72.8 ± 2.1	68.4 ± 5.1	72.7
AnomalyGFM	49.9 ± 2.6	51.8 ± 2.1	53.1 ± 2.6	49.6 ± 4.0	51.1
ZeroGAD	OOM	46.2 ± 0.4	OOM	75.1 ± 1.2	–
IA-GGAD	89.4 ± 0.5	69.8 ± 1.4	93.5 ± 0.4	77.5 ± 2.1	**82.5**
DR-GGAD	**96.0** ± 0.3	73.6 ± 0.4	**98.0** ± 0.2	67.8 ± 1.4	**83.8**
OWLEYE	78.4 ± 0.5	**76.7** ± 4.5	82.5 ± 0.5	75.4 ± 0.3	78.3
TA-GGAD	87.9 ± 0.6	76.2 ± 0.9	89.2 ± 0.9	75.5 ± 3.4	82.2
ProMoS	81.1 ± 0.7	67.4 ± 0.9	74.6 ± 1.1	67.4 ± 2.0	72.6
SAARCS	84.2 ± 0.3	76.0 ± 0.7	82.8 ± 0.9	**78.3** ± 2.3	80.3
TPCA-GAD	67.7 ± 8.1	49.0 ± 5.0	70.4 ± 2.0	48.2 ± 4.5	58.8
GAD-MoRE	76.3 ± 0.4	69.7 ± 0.2	80.4 ± 0.2	67.3 ± 0.7	73.4
NeighborDiv	72.1 ± 0.0	44.8 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	67.3
REFIGAD	95.7 ± 1.7	76.2 ± 4.6	97.7 ± 1.1	64.9 ± 4.3	**83.6**
TFM4GAD	74.6 ± 3.7	55.4 ± 4.9	75.6 ± 3.2	76.1 ± 2.8	70.4
GraphPrompt	57.1 ± 10.5	50.0 ± 2.8	58.0 ± 4.5	50.0 ± 4.4	53.8
SAMGPT	75.0 ± 5.7	65.5 ± 11.5	68.8 ± 7.1	57.8 ± 6.2	66.8
MDGFM	55.5 ± 4.4	69.5 ± 5.3	58.4 ± 8.4	60.5 ± 5.4	61.0
BRIDGE	57.6 ± 7.0	57.9 ± 14.4	55.6 ± 3.3	53.8 ± 3.2	56.2
GILT	52.7 ± 6.7	49.8 ± 10.3	52.1 ± 8.1	52.2 ± 10.1	51.7

Table 12: AUROC (%) on unseen organic targets under the multi-source injected-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	**98.8** ± 0.1	54.9 ± 0.1	22.2 ± 0.0	54.1 ± 3.6	45.7 ± 0.3	54.4 ± 0.3	52.3 ± 0.2	53.3 ± 0.9	**63.8** ± 0.7	71.5 ± 1.9	34.9 ± 0.3	55.1	9
CoLA	36.2 ± 8.6	55.1 ± 2.3	43.8 ± 12.2	46.0 ± 2.5	50.1 ± 3.2	55.4 ± 4.9	52.2 ± 1.1	58.8 ± 3.3	45.0 ± 4.1	59.5 ± 4.5	57.6 ± 20.8	50.9	21
TAM	58.7 ± 7.2	54.6 ± 1.0	74.7 ± 2.5	50.5 ± 1.3	42.1 ± 1.4	50.6 ± 0.9	51.2 ± 1.8	52.9 ± 1.2	63.7 ± 2.5	57.4 ± 9.3	47.6 ± 2.3	54.9	10
BWGNN	43.8 ± 25.4	50.8 ± 7.7	49.2 ± 16.9	50.6 ± 4.6	48.2 ± 3.8	52.4 ± 8.1	52.7 ± 8.1	50.2 ± 10.2	51.0 ± 6.4	50.2 ± 11.0	51.3 ± 7.8	50.0	24
GHRN	71.2 ± 20.7	45.6 ± 4.9	67.1 ± 14.3	50.7 ± 5.9	47.6 ± 5.7	48.9 ± 7.5	50.1 ± 5.4	37.6 ± 9.6	44.1 ± 5.2	58.7 ± 13.0	53.9 ± 6.2	52.3	15
MLP	74.5 ± 2.1	43.6 ± 1.9	73.5 ± 6.4	48.2 ± 3.4	48.3 ± 2.3	44.6 ± 3.5	48.8 ± 1.1	21.7 ± 4.9	39.1 ± 2.6	47.0 ± 4.4	55.5 ± 4.8	49.5	27
GCN	50.6 ± 27.3	45.6 ± 6.9	55.3 ± 22.6	51.0 ± 3.7	49.8 ± 3.3	47.8 ± 8.8	47.4 ± 10.2	52.4 ± 6.1	46.8 ± 3.8	51.7 ± 5.8	50.5 ± 1.7	49.9	25
GAT	42.5 ± 23.8	52.6 ± 4.5	50.4 ± 16.7	36.7 ± 4.6	48.5 ± 1.0	52.6 ± 7.3	48.6 ± 3.7	60.6 ± 10.4	48.8 ± 3.7	70.7 ± 9.2	50.0 ± 11.4	51.1	18
RF-Graph	69.6 ± 7.5	51.5 ± 1.3	81.8 ± 1.1	37.5 ± 0.6	49.0 ± 0.3	46.5 ± 0.7	50.8 ± 0.7	34.9 ± 0.8	47.0 ± 0.5	69.4 ± 2.0	23.3 ± 2.7	51.0	19
XGB-Graph	83.2 ± 9.2	49.4 ± 1.0	45.8 ± 10.3	33.0 ± 0.6	47.3 ± 0.4	52.5 ± 1.9	45.5 ± 0.6	48.3 ± 0.5	47.2 ± 0.2	58.8 ± 4.8	48.5 ± 4.4	50.8	22
Generalist GAD Methods													
ARC	89.4 ± 0.6	58.3 ± 1.0	59.5 ± 3.5	47.5 ± 0.6	57.5 ± 1.9	51.1 ± 5.2	57.4 ± 0.6	25.6 ± 3.1	47.9 ± 3.7	49.3 ± 4.9	48.4 ± 8.2	53.8	13
UNPrompt	55.1 ± 3.9	53.1 ± 1.6	44.9 ± 11.6	54.4 ± 3.6	49.6 ± 2.6	46.0 ± 4.2	41.9 ± 1.1	35.4 ± 4.5	45.6 ± 5.5	37.3 ± 3.2	52.1 ± 4.0	46.8	30
AnomalyGFM	27.8 ± 8.3	48.0 ± 1.7	47.0 ± 11.8	46.6 ± 0.5	57.9 ± 2.0	45.9 ± 5.5	54.4 ± 0.9	56.7 ± 1.8	44.7 ± 8.3	49.8 ± 3.1	66.1 ± 1.6	49.5	26
ZeroGAD	24.3 ± 1.3	53.5 ± 1.6	79.6 ± 3.7	OOM	OOM	43.9 ± 1.0	OOM	OOM	OOM	OOM	53.1 ± 2.0	50.9	20
IA-GGAD	91.7 ± 0.4	55.2 ± 2.1	77.7 ± 2.2	50.3 ± 1.1	45.7 ± 1.6	50.6 ± 3.8	57.2 ± 0.3	24.0 ± 1.2	48.7 ± 1.6	50.7 ± 3.1	42.6 ± 3.4	54.0	12
DR-GGAD	53.4 ± 1.1	57.8 ± 0.5	48.5 ± 1.9	41.2 ± 1.3	53.5 ± 0.9	43.6 ± 0.6	52.8 ± 0.3	42.3 ± 0.6	49.9 ± 0.7	60.2 ± 2.2	40.0 ± 0.4	49.4	28
OWLEYE	88.1 ± 0.4	58.7 ± 1.0	30.3 ± 0.9	47.6 ± 1.3	57.1 ± 0.7	45.9 ± 4.2	48.4 ± 2.3	19.4 ± 0.7	42.2 ± 1.0	51.9 ± 3.2	50.1 ± 0.6	49.1	29
TA-GGAD	91.5 ± 0.3	51.7 ± 1.4	63.3 ± 3.6	52.3 ± 1.4	55.8 ± 3.2	52.0 ± 4.8	57.9 ± 0.3	24.6 ± 1.0	55.0 ± 4.3	51.3 ± 1.3	44.2 ± 1.2	54.5	11
ProMoS	92.2 ± 0.1	52.2 ± 1.6	54.5 ± 4.4	52.5 ± 1.9	58.0 ± 4.7	39.8 ± 1.5	50.0 ± 0.3	13.7 ± 0.1	41.4 ± 0.3	47.3 ± 0.1	53.2 ± 0.1	50.4	23
SAARCS	86.3 ± 2.2	54.1 ± 2.8	60.7 ± 3.8	42.1 ± 0.5	58.4 ± 1.7	52.8 ± 3.4	54.2 ± 1.0	42.2 ± 5.0	55.4 ± 6.0	60.5 ± 6.2	50.1 ± 5.9	56.1	8
TPCA-GAD	72.2 ± 12.2	51.0 ± 1.8	66.6 ± 2.9	68.4 ± 3.9	51.9 ± 0.3	59.9 ± 1.1	41.6 ± 2.9	33.6 ± 3.0	51.6 ± 0.8	33.9 ± 6.8	53.2 ± 3.0	53.1	14
GAD-MoRE	89.8 ± 0.7	54.3 ± 0.5	23.3 ± 0.8	57.3 ± 0.3	51.4 ± 0.3	44.2 ± 1.3	56.8 ± 0.9	27.6 ± 0.7	48.2 ± 0.7	59.4 ± 0.5	55.0 ± 0.5	51.6	17
NeighborDiv	79.9 ± 0.7	51.2 ± 0.0	68.8 ± 0.7	58.2 ± 0.0	50.7 ± 0.3	52.9 ± 0.0	45.5 ± 0.0	56.6 ± 0.0	52.1 ± 0.0	56.4 ± 0.0	72.5 ± 0.0	58.6	7
REFIGAD	89.6 ± 2.8	55.1 ± 4.8	66.7 ± 2.5	71.9 ± 6.2	53.4 ± 3.4	54.6 ± 2.8	57.8 ± 5.3	73.4 ± 3.5	53.6 ± 3.5	88.0 ± 2.7	**82.2** ± 7.1	**67.8**	**2**
TFM4GAD	97.5 ± 1.6	55.5 ± 4.6	**92.5** ± 3.9	**74.0** ± 7.1	**61.0** ± 3.2	**67.2** ± 8.3	**62.9** ± 2.8	**85.0** ± 2.9	59.9 ± 6.9	**93.0** ± 4.7	–	**74.8**	**1**
Graph Foundation Models													
GraphPrompt	86.8 ± 8.0	53.0 ± 4.9	72.2 ± 4.6	68.2 ± 3.1	50.0 ± 2.2	52.4 ± 5.1	54.0 ± 6.1	60.4 ± 13.6	50.9 ± 3.1	87.8 ± 2.2	56.5 ± 18.4	62.9	6
SAMGPT	96.3 ± 5.5	**59.1** ± 4.5	79.2 ± 4.4	65.9 ± 5.8	52.0 ± 2.4	61.6 ± 5.8	53.8 ± 3.5	75.4 ± 6.7	59.5 ± 9.2	82.6 ± 7.0	46.8 ± 26.3	**66.6**	**3**
MDGFM	93.6 ± 2.0	53.6 ± 4.8	91.3 ± 2.3	58.9 ± 4.0	55.1 ± 5.6	64.7 ± 4.2	57.8 ± 3.3	68.4 ± 10.5	60.0 ± 2.6	70.2 ± 7.9	38.8 ± 24.6	64.8	4
BRIDGE	96.5 ± 1.5	55.6 ± 3.1	81.1 ± 1.0	62.3 ± 4.7	51.0 ± 1.8	61.1 ± 6.4	55.6 ± 3.1	66.9 ± 9.4	59.8 ± 4.7	75.8 ± 3.4	44.1 ± 29.4	64.5	5
GILT	66.9 ± 23.3	48.0 ± 6.1	52.2 ± 16.8	52.4 ± 9.5	49.0 ± 4.2	49.2 ± 5.5	47.9 ± 4.9	45.4 ± 14.1	50.4 ± 7.8	46.6 ± 8.1	64.7 ± 17.7	52.1	16

Table 13: AUPRC (%) on unseen organic targets under the single-source organic-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	**94.2** ± 0.7	3.8 ± 0.0	4.1 ± 0.0	4.9 ± 0.5	13.5 ± 0.2	24.7 ± 0.6	3.6 ± 0.2	9.9 ± 0.2	2.1 ± 0.1	12.5 ± 2.1	4.4 ± 0.1	16.2	7
CoLA	38.6 ± 11.5	3.7 ± 0.3	14.5 ± 8.0	6.3 ± 0.6	14.9 ± 1.5	22.7 ± 2.6	2.8 ± 0.2	8.8 ± 0.6	1.3 ± 0.2	3.1 ± 0.5	18.8 ± 11.7	12.3	19
TAM	23.0 ± 11.1	4.1 ± 0.1	18.1 ± 0.4	5.5 ± 0.2	12.3 ± 0.4	23.1 ± 0.1	3.5 ± 0.0	10.6 ± 0.4	**2.2** ± 0.1	4.7 ± 0.4	8.3 ± 0.5	10.5	24
BWGNN	33.5 ± 24.1	3.7 ± 1.0	7.7 ± 2.6	6.0 ± 1.3	14.4 ± 1.5	25.3 ± 5.5	3.8 ± 1.3	9.8 ± 2.3	1.3 ± 0.3	3.1 ± 0.9	5.5 ± 1.2	10.4	25
GHRN	43.7 ± 24.4	3.1 ± 0.5	21.3 ± 11.8	5.9 ± 1.3	14.2 ± 2.3	22.5 ± 3.8	3.4 ± 0.8	9.5 ± 3.0	1.2 ± 0.2	4.5 ± 2.4	10.3 ± 3.2	12.7	18
MLP	38.4 ± 14.5	3.8 ± 0.7	20.6 ± 12.6	5.3 ± 0.8	15.0 ± 1.5	26.7 ± 4.2	3.7 ± 0.9	9.6 ± 6.9	1.3 ± 0.2	2.9 ± 0.6	12.6 ± 7.3	12.7	17
GCN	41.4 ± 24.0	4.0 ± 1.3	16.4 ± 8.4	7.4 ± 1.3	15.7 ± 1.4	20.5 ± 5.6	3.3 ± 1.1	9.4 ± 1.0	1.2 ± 0.2	3.3 ± 0.9	5.2 ± 1.4	11.6	22
GAT	21.5 ± 12.4	4.0 ± 0.6	8.6 ± 4.1	8.0 ± 1.6	14.3 ± 0.5	21.8 ± 4.5	3.0 ± 0.5	9.8 ± 3.3	1.3 ± 0.1	3.5 ± 1.4	17.6 ± 9.1	10.3	26
RF-Graph	8.5 ± 1.5	4.1 ± 0.1	8.3 ± 2.4	5.5 ± 0.3	14.2 ± 0.3	27.9 ± 1.4	3.1 ± 0.1	10.2 ± 0.3	1.2 ± 0.0	2.9 ± 0.3	15.7 ± 4.5	9.2	29
XGB-Graph	17.6 ± 3.5	3.3 ± 0.0	11.4 ± 1.5	5.7 ± 0.3	14.6 ± 0.1	25.1 ± 1.7	3.0 ± 0.1	10.8 ± 0.4	1.2 ± 0.0	2.5 ± 0.2	15.3 ± 4.2	10.1	27
Generalist GAD Methods													
ARC	63.7 ± 1.3	4.3 ± 0.2	10.6 ± 2.2	5.2 ± 0.1	20.1 ± 1.4	22.8 ± 1.5	3.9 ± 0.1	6.1 ± 0.3	1.2 ± 0.1	3.0 ± 0.5	6.0 ± 1.5	13.4	15
UNPrompt	66.7 ± 1.9	3.5 ± 0.1	23.9 ± 3.6	6.9 ± 0.5	18.1 ± 1.0	21.0 ± 0.7	4.3 ± 0.2	5.6 ± 0.0	1.0 ± 0.0	2.2 ± 0.0	10.6 ± 2.3	14.9	9
AnomalyGFM	22.6 ± 6.4	3.5 ± 0.3	47.5 ± 16.4	5.5 ± 0.1	18.3 ± 1.1	26.7 ± 4.2	3.3 ± 0.2	11.0 ± 0.8	1.1 ± 0.1	2.7 ± 0.1	9.9 ± 2.9	13.8	13
ZeroGAD	7.4 ± 0.6	3.6 ± 0.4	36.8 ± 9.9	OOM	OOM	18.3 ± 0.5	OOM	OOM	OOM	OOM	17.9 ± 4.5	16.8	5
IA-GGAD	72.8 ± 0.3	4.2 ± 0.3	25.1 ± 3.9	5.6 ± 0.2	12.6 ± 0.3	22.7 ± 1.0	3.9 ± 0.1	6.0 ± 0.1	1.2 ± 0.1	2.9 ± 0.2	5.2 ± 0.6	14.7	10
DR-GGAD	36.5 ± 0.9	**4.9** ± 0.4	6.4 ± 0.3	4.5 ± 0.2	17.3 ± 0.7	18.6 ± 0.4	3.2 ± 0.1	7.7 ± 0.1	1.2 ± 0.0	3.9 ± 0.2	5.1 ± 0.1	9.9	28
OWLEYE	62.8 ± 0.7	4.1 ± 0.2	4.7 ± 0.1	5.1 ± 0.2	16.6 ± 0.4	22.3 ± 2.3	3.4 ± 0.2	6.0 ± 0.8	1.0 ± 0.0	2.9 ± 0.1	5.6 ± 0.2	12.2	20
TA-GGAD	72.8 ± 0.3	3.6 ± 0.3	11.2 ± 2.8	5.2 ± 0.3	17.9 ± 2.8	23.2 ± 1.5	3.9 ± 0.0	6.0 ± 0.1	1.6 ± 0.1	2.9 ± 0.1	5.6 ± 0.1	14.0	12
ProMoS	73.9 ± 0.3	3.9 ± 0.4	9.8 ± 1.9	5.4 ± 0.5	17.9 ± 2.5	17.6 ± 0.9	3.2 ± 0.2	5.4 ± 0.0	1.0 ± 0.0	2.6 ± 0.0	6.5 ± 0.0	13.4	14
SAARCS	59.1 ± 5.6	3.6 ± 0.2	9.2 ± 0.7	4.1 ± 0.0	19.1 ± 0.9	24.2 ± 1.6	3.7 ± 0.1	8.2 ± 0.8	1.5 ± 0.2	4.8 ± 1.2	6.5 ± 1.0	13.1	16
TPCA-GAD	31.6 ± 14.7	3.2 ± 0.2	12.1 ± 0.5	8.8 ± 0.8	15.7 ± 0.4	25.7 ± 1.1	2.5 ± 0.1	7.8 ± 0.8	1.3 ± 0.1	2.0 ± 0.1	7.9 ± 2.1	10.8	23
GAD-MoRE	59.2 ± 0.5	3.8 ± 0.1	4.1 ± 0.0	6.5 ± 0.1	14.5 ± 0.3	22.5 ± 0.5	3.5 ± 0.1	6.5 ± 0.0	1.3 ± 0.0	4.4 ± 0.1	6.2 ± 0.1	12.0	21
NeighborDiv	55.8 ± 0.7	3.6 ± 0.0	17.7 ± 0.3	6.9 ± 0.0	15.2 ± 0.1	21.7 ± 0.0	2.8 ± 0.0	11.3 ± 0.0	1.5 ± 0.0	3.7 ± 0.0	16.2 ± 0.0	14.2	11
REFIGAD	39.8 ± 5.9	3.9 ± 0.6	13.8 ± 3.1	10.8 ± 4.5	16.8 ± 1.4	24.2 ± 1.5	3.9 ± 1.0	25.2 ± 6.3	1.6 ± 0.4	12.9 ± 3.5	**30.9** ± 11.8	16.7	6
TFM4GAD	83.2 ± 6.6	4.2 ± 0.7	**70.7** ± 13.5	**20.1** ± 8.1	**21.5** ± 2.6	**34.4** ± 7.2	**5.3** ± 1.5	**63.7** ± 6.0	1.8 ± 0.5	**59.5** ± 13.3	–	**36.4**	**1**
Graph Foundation Models													
GraphPrompt	64.7 ± 8.7	3.6 ± 0.5	19.4 ± 2.0	9.3 ± 0.9	15.0 ± 0.9	21.6 ± 2.8	3.0 ± 0.5	11.0 ± 2.7	1.4 ± 0.2	15.8 ± 8.9	4.3 ± 5.1	15.4	8
SAMGPT	81.6 ± 19.8	4.9 ± 0.8	27.2 ± 7.4	11.0 ± 2.8	15.6 ± 1.2	30.0 ± 3.1	4.1 ± 0.8	28.6 ± 6.9	2.0 ± 0.6	48.4 ± 7.3	3.6 ± 3.2	**23.3**	**2**
MDGFM	61.9 ± 12.0	4.0 ± 0.7	65.6 ± 16.6	7.6 ± 1.4	16.0 ± 1.4	31.7 ± 2.8	4.5 ± 1.2	28.1 ± 11.4	1.9 ± 0.5	7.6 ± 2.5	6.2 ± 7.8	**21.4**	**3**
BRIDGE	82.4 ± 7.5	4.3 ± 0.7	28.9 ± 1.8	9.6 ± 1.7	14.8 ± 0.3	30.8 ± 2.3	4.1 ± 1.0	23.3 ± 7.3	2.1 ± 0.5	10.1 ± 5.7	3.1 ± 3.2	19.4	4
GILT	20.8 ± 13.5	3.8 ± 0.4	5.6 ± 1.7	6.4 ± 2.4	13.6 ± 1.4	24.2 ± 3.4	3.0 ± 0.2	8.6 ± 1.6	1.4 ± 0.4	3.0 ± 0.8	6.7 ± 6.8	8.8	30

Table 14: AUPRC (%) on unseen injected targets under the single-source organic-to-injected setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	13.3 ± 1.1	24.9 ± 0.5	16.4 ± 0.6	8.0 ± 0.0	15.7
CoLA	3.6 ± 0.6	10.9 ± 2.9	7.4 ± 2.3	6.1 ± 1.9	7.0
TAM	6.0 ± 0.5	15.6 ± 1.1	9.3 ± 1.0	6.8 ± 0.8	9.4
BWGNN	4.0 ± 1.8	12.7 ± 6.4	5.7 ± 2.0	7.7 ± 3.4	7.5
GHRN	5.5 ± 2.0	21.0 ± 6.1	12.8 ± 6.6	7.9 ± 3.8	11.8
MLP	5.8 ± 3.6	18.0 ± 11.0	11.6 ± 5.1	8.5 ± 3.7	11.0
GCN	3.1 ± 0.8	7.6 ± 2.7	3.1 ± 0.6	5.3 ± 0.9	4.8
GAT	3.2 ± 0.8	7.0 ± 1.2	3.9 ± 0.9	5.2 ± 0.3	4.8
RF-Graph	3.1 ± 0.2	7.0 ± 0.7	3.3 ± 0.1	6.5 ± 0.2	5.0
XGB-Graph	3.2 ± 0.1	5.9 ± 0.0	3.3 ± 0.0	5.9 ± 0.0	4.6
ARC	33.5 ± 0.3	38.7 ± 0.6	36.5 ± 0.1	33.4 ± 1.1	35.5
UNPrompt	10.3 ± 0.5	27.2 ± 0.5	17.3 ± 0.4	14.0 ± 2.3	17.2
AnomalyGFM	3.0 ± 0.1	6.5 ± 0.3	3.9 ± 0.2	6.5 ± 0.3	5.0
ZeroGAD	OOM	6.7 ± 0.4	OOM	27.3 ± 4.0	–
IA-GGAD	30.5 ± 1.2	36.5 ± 0.6	37.8 ± 1.3	16.6 ± 2.6	30.3

Method	PubMed	Flickr	CS	Photo	Avg.
DR-GGAD	**51.9** ± 2.6	38.1 ± 0.4	**70.0** ± 1.8	16.5 ± 1.3	**44.1**
OWLEYE	30.2 ± 0.3	39.5 ± 0.7	39.6 ± 0.3	41.9 ± 1.9	**37.8**
TA-GGAD	35.7 ± 0.5	39.4 ± 0.3	39.0 ± 0.5	31.9 ± 1.5	36.5
ProMoS	30.0 ± 0.5	36.1 ± 0.3	33.7 ± 0.3	27.1 ± 7.7	31.8
SAARCS	35.5 ± 0.3	**40.7** ± 0.2	38.5 ± 0.1	**43.9** ± 1.2	**39.6**
TPCA-GAD	3.7 ± 0.8	5.1 ± 0.2	7.5 ± 1.3	5.3 ± 0.2	5.4
GAD-MoRE	22.5 ± 1.4	32.1 ± 0.1	35.4 ± 0.7	36.4 ± 1.1	31.6
NeighborDiv	10.9 ± 0.0	5.0 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	10.8
REFIGAD	23.2 ± 8.3	11.8 ± 3.6	22.8 ± 7.9	14.0 ± 5.0	17.9
TFM4GAD	17.2 ± 9.3	7.6 ± 1.6	10.7 ± 2.7	31.2 ± 10.9	16.7
GraphPrompt	4.8 ± 1.4	6.9 ± 1.4	3.7 ± 0.8	6.8 ± 1.6	5.5
SAMGPT	12.9 ± 4.2	9.9 ± 2.6	7.0 ± 2.0	9.1 ± 3.4	9.7
MDGFM	4.1 ± 1.2	19.3 ± 6.5	6.7 ± 4.2	13.1 ± 7.1	10.8
BRIDGE	4.0 ± 0.8	7.9 ± 1.8	3.9 ± 0.7	6.5 ± 0.5	5.6
GILT	3.5 ± 0.9	14.8 ± 9.5	4.9 ± 3.2	6.7 ± 2.7	7.5

Table 15: AUPRC (%) on unseen organic targets under the single-source injected-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	**92.7** ± 0.9	3.8 ± 0.0	4.1 ± 0.0	4.3 ± 0.4	13.5 ± 0.1	25.3 ± 0.2	4.4 ± 0.1	9.9 ± 0.2	2.2 ± 0.2	10.9 ± 1.0	4.3 ± 0.1	15.9	6
CoLA	35.4 ± 10.5	3.7 ± 0.4	13.3 ± 7.1	6.2 ± 0.6	15.0 ± 1.5	23.1 ± 2.7	2.8 ± 0.2	9.0 ± 0.6	1.3 ± 0.2	3.1 ± 0.5	18.7 ± 10.9	12.0	19
TAM	9.0 ± 1.0	3.9 ± 0.0	17.5 ± 0.4	5.3 ± 0.1	12.2 ± 0.3	23.1 ± 0.1	3.6 ± 0.1	10.6 ± 0.1	2.1 ± 0.2	3.4 ± 0.8	6.7 ± 1.5	8.9	28
BWGNN	34.7 ± 24.3	3.8 ± 1.1	7.4 ± 2.5	6.0 ± 1.3	14.3 ± 1.5	25.6 ± 5.5	4.1 ± 1.3	9.6 ± 2.2	1.3 ± 0.3	3.2 ± 1.0	5.5 ± 1.4	10.5	23
GHRN	49.5 ± 24.0	3.0 ± 0.5	17.1 ± 9.4	5.8 ± 1.3	14.1 ± 2.3	23.7 ± 4.6	3.6 ± 0.9	8.8 ± 2.6	1.2 ± 0.2	5.8 ± 3.2	10.0 ± 3.3	13.0	17
MLP	45.7 ± 15.6	3.5 ± 0.5	23.9 ± 13.8	4.8 ± 0.8	15.6 ± 1.6	26.6 ± 3.8	3.5 ± 1.0	10.0 ± 8.4	1.3 ± 0.2	2.8 ± 0.5	8.4 ± 4.8	13.3	14
GCN	45.2 ± 22.8	3.5 ± 1.2	15.4 ± 9.1	7.0 ± 1.2	15.5 ± 1.6	22.0 ± 5.4	3.2 ± 1.1	9.3 ± 1.0	1.2 ± 0.2	3.3 ± 0.9	5.2 ± 1.0	11.9	20
GAT	29.5 ± 20.0	3.8 ± 0.7	8.2 ± 3.8	5.3 ± 1.1	14.7 ± 1.1	24.2 ± 4.0	2.9 ± 0.4	10.3 ± 3.1	1.3 ± 0.1	4.2 ± 1.9	10.0 ± 3.7	10.4	24
RF-Graph	5.6 ± 0.1	3.7 ± 0.1	19.2 ± 1.4	4.5 ± 0.1	13.4 ± 0.1	26.6 ± 0.8	2.9 ± 0.0	11.1 ± 0.9	1.2 ± 0.0	8.0 ± 0.2	3.3 ± 0.1	9.0	26
XGB-Graph	15.8 ± 7.9	3.4 ± 0.2	14.8 ± 3.1	3.5 ± 0.0	13.7 ± 0.2	21.0 ± 0.6	2.4 ± 0.1	9.1 ± 0.8	1.1 ± 0.0	4.2 ± 0.9	6.2 ± 2.3	8.6	29
Generalist GAD Methods													
ARC	63.9 ± 1.2	4.3 ± 0.2	10.3 ± 1.5	5.1 ± 0.1	20.0 ± 1.5	22.9 ± 1.5	3.9 ± 0.1	6.1 ± 0.2	1.2 ± 0.1	3.0 ± 0.5	6.6 ± 1.5	13.4	13
UNPrompt	15.1 ± 2.4	3.8 ± 0.3	14.3 ± 7.1	5.0 ± 1.0	15.0 ± 0.8	21.2 ± 1.8	2.8 ± 0.1	8.7 ± 0.8	**2.6** ± 0.1	3.3 ± 1.3	7.0 ± 2.3	9.0	27
AnomalyGFM	7.9 ± 1.8	3.1 ± 0.3	6.6 ± 1.3	5.4 ± 0.1	16.6 ± 1.1	20.9 ± 4.8	3.2 ± 0.2	12.9 ± 0.8	1.0 ± 0.2	2.7 ± 0.1	9.6 ± 1.2	8.2	30
ZeroGAD	7.1 ± 0.4	3.9 ± 0.4	46.9 ± 9.1	OOM	OOM	18.8 ± 0.3	OOM	OOM	OOM	OOM	6.8 ± 1.2	16.7	5
IA-GGAD	72.9 ± 0.3	4.2 ± 0.4	23.2 ± 3.2	5.6 ± 0.2	12.7 ± 0.3	22.8 ± 0.9	3.8 ± 0.1	6.0 ± 0.1	1.2 ± 0.1	2.9 ± 0.2	5.2 ± 0.6	14.6	8
DR-GGAD	36.2 ± 1.0	**4.9** ± 0.3	6.4 ± 0.4	4.6 ± 0.2	17.3 ± 0.7	18.7 ± 0.4	3.2 ± 0.1	7.7 ± 0.1	1.2 ± 0.0	4.0 ± 0.2	5.1 ± 0.1	9.9	25
OWLEYE	64.3 ± 1.8	3.9 ± 0.2	4.7 ± 0.1	5.0 ± 0.2	16.9 ± 0.6	21.4 ± 2.1	3.3 ± 0.1	5.9 ± 0.4	1.0 ± 0.0	3.0 ± 0.3	5.6 ± 0.2	12.3	18
TA-GGAD	72.8 ± 0.3	3.8 ± 0.3	9.7 ± 1.1	5.2 ± 0.3	17.8 ± 2.0	23.0 ± 1.5	3.9 ± 0.1	6.0 ± 0.1	1.6 ± 0.1	2.9 ± 0.1	5.6 ± 0.2	13.8	10
ProMoS	74.0 ± 0.3	4.2 ± 0.5	11.5 ± 2.9	4.9 ± 0.3	18.5 ± 2.5	16.5 ± 0.6	3.2 ± 0.1	5.4 ± 0.0	1.0 ± 0.0	2.6 ± 0.0	6.5 ± 0.0	13.5	12
SAARCS	59.2 ± 5.4	3.6 ± 0.2	9.2 ± 0.7	4.1 ± 0.0	19.1 ± 0.9	24.2 ± 1.6	3.7 ± 0.1	8.2 ± 0.7	1.5 ± 0.2	4.8 ± 1.2	7.0 ± 1.9	13.1	16
TPCA-GAD	57.0 ± 8.3	3.3 ± 0.2	12.8 ± 1.0	8.7 ± 0.9	15.9 ± 0.1	28.0 ± 3.0	2.6 ± 0.2	7.2 ± 0.5	1.2 ± 0.0	2.2 ± 0.3	6.4 ± 0.6	13.2	15
GAD-MoRE	60.4 ± 0.6	3.6 ± 0.1	4.1 ± 0.0	5.9 ± 0.1	14.2 ± 0.2	18.9 ± 0.1	3.4 ± 0.0	6.7 ± 0.0	1.3 ± 0.0	4.3 ± 0.1	6.6 ± 0.1	11.8	21
NeighborDiv	55.8 ± 0.7	3.6 ± 0.0	17.7 ± 0.3	6.9 ± 0.0	15.2 ± 0.1	21.7 ± 0.0	2.8 ± 0.0	11.3 ± 0.0	1.5 ± 0.0	3.7 ± 0.0	16.2 ± 0.0	14.2	9
REFIGAD	35.2 ± 6.0	4.3 ± 0.7	14.5 ± 3.7	8.8 ± 2.9	17.4 ± 1.7	23.3 ± 2.0	4.5 ± 0.8	20.7 ± 8.5	1.4 ± 0.1	14.8 ± 5.6	**26.2** ± 11.5	15.5	7
TFM4GAD	83.2 ± 6.6	4.2 ± 0.7	**70.7** ± 13.5	**20.1** ± 8.1	**21.5** ± 2.6	**34.4** ± 7.2	**5.3** ± 1.5	**63.7** ± 6.0	1.8 ± 0.5	**59.5** ± 13.3	–	**36.4**	**1**
Graph Foundation Models													
GraphPrompt	49.4 ± 16.5	3.5 ± 0.7	15.9 ± 4.6	8.1 ± 1.7	14.5 ± 0.5	24.8 ± 4.5	3.1 ± 0.5	13.1 ± 5.5	1.3 ± 0.1	8.5 ± 3.0	7.7 ± 7.1	13.6	11
SAMGPT	81.1 ± 19.8	4.9 ± 0.8	26.9 ± 7.9	11.0 ± 2.8	15.4 ± 1.2	30.2 ± 3.1	4.0 ± 0.8	28.6 ± 6.8	2.0 ± 0.6	48.7 ± 7.8	3.6 ± 3.2	**23.3**	**2**
MDGFM	55.8 ± 11.8	3.8 ± 0.7	65.7 ± 16.0	6.8 ± 1.0	15.5 ± 1.6	30.1 ± 4.5	4.8 ± 1.6	27.1 ± 13.0	1.7 ± 0.5	7.1 ± 2.8	5.3 ± 5.7	**20.3**	**3**
BRIDGE	80.8 ± 8.5	4.2 ± 0.8	29.8 ± 1.5	9.0 ± 2.3	14.9 ± 0.5	30.6 ± 3.2	4.1 ± 1.2	21.7 ± 6.9	2.1 ± 0.5	12.8 ± 5.1	3.4 ± 3.3	19.4	4
GILT	39.5 ± 17.6	3.4 ± 0.5	11.5 ± 7.8	5.2 ± 1.5	14.8 ± 2.1	23.1 ± 3.9	3.0 ± 0.5	10.0 ± 3.5	1.3 ± 0.3	2.9 ± 0.7	5.2 ± 6.7	10.9	22

Table 16: AUPRC (%) on unseen injected targets under the single-source injected-to-injected setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	12.7 ± 0.3	24.5 ± 0.1	15.3 ± 0.2	8.0 ± 0.0	15.1
CoLA	3.9 ± 0.7	10.9 ± 2.8	7.3 ± 2.0	6.3 ± 2.0	7.1
TAM	7.2 ± 1.8	14.4 ± 1.7	10.2 ± 1.1	11.9 ± 1.9	10.9
BWGNN	4.6 ± 2.2	12.4 ± 6.3	6.5 ± 2.3	8.0 ± 3.6	7.9
GHRN	5.6 ± 2.0	19.6 ± 6.7	11.9 ± 6.2	7.8 ± 3.3	11.2
MLP	6.9 ± 3.2	20.7 ± 9.5	14.0 ± 5.0	9.0 ± 3.5	12.6
GCN	3.5 ± 1.1	8.0 ± 2.8	3.7 ± 0.9	6.0 ± 1.1	5.3
GAT	4.7 ± 1.6	6.7 ± 0.8	4.7 ± 0.9	5.8 ± 0.5	5.5
RF-Graph	32.8 ± 0.9	5.3 ± 0.9	21.6 ± 0.8	22.7 ± 1.1	20.6
XGB-Graph	4.4 ± 0.8	23.0 ± 2.4	8.2 ± 1.5	9.8 ± 2.4	11.3
ARC	33.5 ± 0.3	38.6 ± 0.6	36.6 ± 0.1	33.3 ± 1.1	35.5
UNPrompt	13.6 ± 4.5	21.6 ± 2.2	10.0 ± 2.4	14.8 ± 5.8	15.0
AnomalyGFM	3.3 ± 0.2	6.6 ± 0.3	4.2 ± 0.4	6.0 ± 0.4	5.0
ZeroGAD	OOM	5.6 ± 0.1	OOM	21.3 ± 1.5	–
IA-GGAD	30.1 ± 1.1	36.7 ± 0.6	37.5 ± 1.2	16.5 ± 2.6	30.2

Method	PubMed	Flickr	CS	Photo	Avg.
DR-GGAD	**51.9** ± 2.7	38.2 ± 0.3	**70.0** ± 1.7	15.6 ± 1.4	**43.9**
OWLEYE	29.8 ± 0.3	39.6 ± 0.6	39.6 ± 0.3	42.6 ± 1.8	**37.9**
TA-GGAD	35.7 ± 0.5	39.3 ± 0.3	39.2 ± 0.5	31.6 ± 2.1	36.4
ProMoS	30.6 ± 0.5	35.9 ± 0.2	33.9 ± 0.3	27.5 ± 9.1	32.0
SAARCS	35.5 ± 0.3	**40.6** ± 0.2	38.5 ± 0.1	**43.9** ± 1.2	**39.6**
TPCA-GAD	15.7 ± 11.6	6.2 ± 1.1	12.8 ± 6.6	5.4 ± 0.5	10.0
GAD-MoRE	15.1 ± 0.5	29.9 ± 0.2	28.0 ± 0.9	12.3 ± 0.5	21.3
NeighborDiv	10.9 ± 0.0	5.0 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	10.8
REFIGAD	40.0 ± 12.4	17.1 ± 5.3	63.2 ± 7.0	11.7 ± 2.1	33.0
TFM4GAD	17.2 ± 9.3	7.6 ± 1.6	10.7 ± 2.7	31.2 ± 10.9	16.7
GraphPrompt	4.1 ± 1.1	6.3 ± 0.6	4.3 ± 0.7	6.0 ± 0.8	5.2
SAMGPT	12.4 ± 3.6	9.9 ± 2.9	6.9 ± 1.9	9.0 ± 3.4	9.6
MDGFM	4.3 ± 0.9	20.9 ± 4.1	6.4 ± 3.9	11.4 ± 8.0	10.8
BRIDGE	4.2 ± 1.0	8.1 ± 2.1	4.2 ± 0.7	6.4 ± 0.8	5.7
GILT	3.7 ± 1.1	15.6 ± 7.1	5.5 ± 2.6	6.9 ± 3.2	7.9

Table 17: AUPRC (%) on unseen organic targets under the multi-source organic-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	80.0 ± 1.9	3.6 ± 0.1	4.4 ± 0.6	3.7 ± 0.1	13.6 ± 0.1	26.8 ± 0.5	4.3 ± 0.2	8.8 ± 1.1	0.9 ± 0.1	8.1 ± 3.2	5.0 ± 0.5	14.5	15
CoLA	18.2 ± 8.3	4.1 ± 0.6	8.6 ± 4.1	5.3 ± 0.4	13.4 ± 0.8	26.1 ± 2.2	3.1 ± 0.1	10.1 ± 0.6	1.1 ± 0.1	3.3 ± 0.5	17.8 ± 12.0	10.1	27
TAM	8.3 ± 2.3	4.1 ± 0.1	18.7 ± 0.6	5.4 ± 0.1	13.5 ± 1.2	22.5 ± 1.4	3.6 ± 0.2	10.5 ± 0.2	**2.3** ± 0.1	5.6 ± 0.2	7.8 ± 1.1	9.3	30
BWGNN	33.2 ± 23.7	3.8 ± 1.0	7.7 ± 2.6	6.0 ± 1.3	14.5 ± 1.6	25.2 ± 5.6	3.8 ± 1.4	9.7 ± 2.1	1.3 ± 0.3	3.1 ± 0.9	5.5 ± 1.2	10.3	26
GHRN	44.0 ± 22.5	3.0 ± 0.4	31.7 ± 15.6	5.7 ± 1.2	14.9 ± 2.2	22.5 ± 3.7	3.4 ± 0.8	9.3 ± 2.7	1.2 ± 0.2	4.1 ± 2.1	9.3 ± 2.2	13.5	18
MLP	34.2 ± 8.7	2.7 ± 0.1	58.8 ± 12.3	4.5 ± 0.3	17.4 ± 1.0	22.1 ± 2.3	3.5 ± 1.3	12.4 ± 5.1	1.1 ± 0.1	2.6 ± 0.4	6.6 ± 4.4	15.1	14
GCN	40.8 ± 23.3	3.3 ± 1.0	17.2 ± 8.0	7.3 ± 1.3	15.5 ± 1.2	19.0 ± 5.7	3.1 ± 1.0	9.5 ± 1.3	1.2 ± 0.2	3.2 ± 0.4	6.6 ± 3.3	11.5	23
GAT	29.7 ± 19.7	3.8 ± 0.5	8.1 ± 3.2	7.1 ± 1.4	14.3 ± 0.6	21.9 ± 4.9	3.0 ± 0.4	8.4 ± 1.8	1.3 ± 0.1	3.4 ± 1.2	21.2 ± 9.8	11.1	25
RF-Graph	48.2 ± 7.1	2.8 ± 0.1	**85.7** ± 0.4	5.5 ± 0.0	18.2 ± 0.2	22.0 ± 0.7	3.1 ± 0.1	8.3 ± 0.3	1.2 ± 0.0	2.3 ± 0.1	6.4 ± 0.8	18.5	7
XGB-Graph	44.7 ± 6.3	3.4 ± 0.1	82.5 ± 0.6	4.8 ± 0.1	16.0 ± 0.3	15.7 ± 0.2	2.6 ± 0.3	21.2 ± 1.8	1.8 ± 0.1	2.9 ± 0.2	5.5 ± 0.3	18.3	8
Generalist GAD Methods													
ARC	63.0 ± 1.6	4.2 ± 0.2	46.9 ± 6.8	6.4 ± 0.2	**21.8** ± 1.2	23.0 ± 0.8	4.5 ± 0.2	6.2 ± 0.5	1.2 ± 0.1	2.9 ± 0.3	6.3 ± 1.2	16.9	9
UNPrompt	55.4 ± 4.0	3.4 ± 0.2	80.3 ± 1.0	5.4 ± 0.3	17.9 ± 0.5	22.2 ± 0.9	3.1 ± 0.1	6.9 ± 0.2	1.5 ± 0.2	2.3 ± 0.1	8.4 ± 1.6	18.8	6
AnomalyGFM	30.1 ± 6.8	3.4 ± 0.2	53.9 ± 16.5	5.5 ± 0.1	19.4 ± 1.0	26.4 ± 2.4	3.3 ± 0.1	11.9 ± 0.3	1.3 ± 0.1	2.7 ± 0.1	12.7 ± 2.2	15.5	12
ZeroGAD	9.6 ± 1.8	3.3 ± 0.2	23.8 ± 5.3	OOM	OOM	16.8 ± 0.2	OOM	OOM	OOM	OOM	5.0 ± 0.3	11.7	21
IA-GGAD	72.7 ± 0.3	4.1 ± 0.4	31.3 ± 3.1	6.0 ± 0.3	13.4 ± 0.5	22.6 ± 0.9	4.3 ± 0.1	6.0 ± 0.1	1.2 ± 0.1	2.9 ± 0.2	5.1 ± 0.7	15.4	13
DR-GGAD	35.4 ± 1.3	**5.0** ± 0.3	7.0 ± 1.0	4.7 ± 0.3	17.9 ± 1.1	19.4 ± 0.3	3.2 ± 0.2	7.7 ± 0.1	1.2 ± 0.0	3.8 ± 0.2	5.1 ± 0.1	10.0	28
OWLEYE	51.9 ± 1.1	4.0 ± 0.2	5.0 ± 0.3	6.2 ± 0.2	18.7 ± 0.9	23.2 ± 1.1	3.6 ± 0.1	5.4 ± 0.0	1.0 ± 0.0	3.5 ± 2.8	5.7 ± 0.2	11.7	22
TA-GGAD	72.2 ± 0.3	3.8 ± 0.3	37.2 ± 11.0	6.1 ± 0.4	19.4 ± 2.0	22.7 ± 1.1	4.4 ± 0.1	6.0 ± 0.1	1.5 ± 0.1	2.9 ± 0.1	5.6 ± 0.2	16.5	10
ProMoS	72.7 ± 0.2	3.9 ± 0.3	13.8 ± 2.1	6.1 ± 0.5	20.2 ± 2.9	17.6 ± 0.7	3.3 ± 0.1	5.4 ± 0.0	1.0 ± 0.0	2.6 ± 0.0	6.5 ± 0.0	13.9	17
SAARCS	59.3 ± 5.5	3.6 ± 0.2	9.2 ± 0.7	4.1 ± 0.0	19.0 ± 0.9	24.2 ± 1.6	3.7 ± 0.1	7.9 ± 0.7	1.5 ± 0.2	4.7 ± 1.1	6.2 ± 1.0	13.0	19
TPCA-GAD	39.9 ± 12.2	3.3 ± 0.1	10.5 ± 3.2	8.9 ± 0.5	15.7 ± 0.1	24.7 ± 0.2	2.5 ± 0.0	7.6 ± 0.3	1.3 ± 0.1	2.1 ± 0.1	6.7 ± 0.8	11.2	24
GAD-MoRE	59.2 ± 0.7	4.0 ± 0.2	4.1 ± 0.0	5.5 ± 0.1	17.2 ± 0.7	21.5 ± 0.2	3.2 ± 0.0	6.5 ± 0.0	1.3 ± 0.0	4.3 ± 0.1	6.3 ± 0.1	12.1	20
NeighborDiv	55.8 ± 0.7	3.6 ± 0.0	17.7 ± 0.3	6.9 ± 0.0	15.2 ± 0.1	21.7 ± 0.0	2.8 ± 0.0	11.3 ± 0.0	1.5 ± 0.0	3.7 ± 0.0	16.2 ± 0.0	14.2	16
REFIGAD	45.3 ± 11.4	3.8 ± 0.6	20.0 ± 5.5	12.6 ± 2.9	18.0 ± 2.0	23.6 ± 2.0	4.0 ± 0.7	28.9 ± 7.2	1.4 ± 0.2	14.8 ± 10.8	**45.5** ± 9.3	19.8	4
TFM4GAD	83.2 ± 6.6	4.2 ± 0.7	70.7 ± 13.5	**20.1** ± 8.1	21.5 ± 2.6	**34.4** ± 7.2	**5.3** ± 1.5	**63.7** ± 6.0	1.8 ± 0.5	**59.5** ± 13.3	–	**36.4**	**1**
Graph Foundation Models													
GraphPrompt	68.3 ± 17.3	4.2 ± 0.6	18.7 ± 4.4	8.4 ± 1.1	15.0 ± 0.7	23.6 ± 1.7	3.9 ± 0.8	11.4 ± 3.9	1.4 ± 0.2	14.3 ± 9.0	3.1 ± 1.4	15.7	11
SAMGPT	81.3 ± 20.0	4.8 ± 0.7	27.4 ± 7.1	10.9 ± 2.9	15.4 ± 1.1	30.3 ± 2.9	4.0 ± 0.7	28.4 ± 6.9	2.0 ± 0.6	48.1 ± 6.9	3.0 ± 2.2	**23.2**	**2**
MDGFM	60.0 ± 15.2	4.1 ± 0.6	71.6 ± 7.1	6.9 ± 1.2	17.4 ± 2.3	30.0 ± 3.8	4.2 ± 1.3	21.3 ± 9.2	1.8 ± 0.5	11.5 ± 6.8	2.8 ± 1.8	**21.1**	**3**
BRIDGE	**83.8** ± 5.0	4.2 ± 0.7	30.6 ± 1.3	9.8 ± 2.2	14.9 ± 0.5	29.5 ± 4.9	3.8 ± 0.8	20.2 ± 5.2	2.0 ± 0.4	10.6 ± 6.2	3.5 ± 3.7	19.4	5
GILT	31.2 ± 14.9	3.1 ± 0.4	7.9 ± 4.8	4.6 ± 0.7	14.3 ± 1.5	20.4 ± 3.3	3.1 ± 0.6	11.4 ± 4.2	1.2 ± 0.4	2.8 ± 0.9	7.1 ± 13.9	9.7	29

D.2.2 Multi-Source Results

Tables 17–20 report AUPRC under all four multi-source source–target combinations.

Table 18: AUPRC (%) on unseen injected targets under the multi-source organic-to-injected setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	10.2 ± 2.9	20.2 ± 3.0	12.1 ± 3.2	7.9 ± 0.1	12.6
CoLA	6.7 ± 1.3	10.1 ± 2.5	5.4 ± 1.4	6.7 ± 1.1	7.2
TAM	4.0 ± 0.5	13.5 ± 1.0	10.2 ± 1.0	6.8 ± 0.3	8.6
BWGNN	4.1 ± 2.1	12.7 ± 6.4	5.8 ± 2.2	7.6 ± 3.4	7.5
GHRN	5.7 ± 2.0	21.2 ± 5.7	13.4 ± 6.6	8.1 ± 3.9	12.1
MLP	3.6 ± 2.5	9.6 ± 6.9	5.7 ± 2.1	6.0 ± 1.5	6.2
GCN	3.5 ± 0.8	8.5 ± 1.3	3.8 ± 0.5	5.8 ± 0.9	5.4
GAT	3.3 ± 1.1	6.9 ± 1.0	3.9 ± 0.7	5.3 ± 0.3	4.8
RF-Graph	3.6 ± 0.1	9.5 ± 2.4	4.1 ± 0.1	6.9 ± 0.3	6.0
XGB-Graph	3.0 ± 0.2	10.0 ± 2.7	4.2 ± 0.6	7.7 ± 1.6	6.2
ARC	32.2 ± 0.3	38.7 ± 0.6	36.4 ± 0.1	32.9 ± 2.2	35.1
UNPrompt	5.8 ± 0.4	26.6 ± 1.2	9.5 ± 0.7	8.9 ± 1.7	12.7
AnomalyGFM	3.2 ± 0.1	6.9 ± 0.4	4.0 ± 0.2	7.0 ± 0.4	5.3
ZeroGAD	OOM	6.3 ± 0.5	OOM	14.7 ± 0.7	–
IA-GGAD	29.3 ± 1.4	37.5 ± 0.5	35.9 ± 1.4	14.5 ± 2.4	29.3

Method	PubMed	Flickr	CS	Photo	Avg.
DR-GGAD	**47.6** ± 4.0	38.0 ± 0.6	**69.7** ± 3.0	15.2 ± 2.1	**42.6**
OWLEYE	25.5 ± 0.8	37.4 ± 0.5	38.2 ± 0.2	23.5 ± 7.0	31.1
TA-GGAD	35.0 ± 0.5	39.9 ± 0.3	38.6 ± 0.5	29.4 ± 4.1	**35.7**
ProMoS	30.2 ± 0.3	36.3 ± 0.2	33.5 ± 0.1	26.5 ± 3.9	31.6
SAARCS	35.5 ± 0.3	**40.7** ± 0.2	38.5 ± 0.1	**43.9** ± 1.2	**39.6**
TPCA-GAD	2.5 ± 0.1	4.9 ± 0.1	4.2 ± 0.3	5.2 ± 0.1	4.2
GAD-MoRE	16.0 ± 0.7	30.4 ± 0.3	33.1 ± 0.7	39.2 ± 1.2	29.7
NeighborDiv	10.9 ± 0.0	5.0 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	10.8
REFIGAD	32.6 ± 8.3	11.6 ± 1.6	26.7 ± 6.8	14.1 ± 5.6	21.3
TFM4GAD	17.2 ± 9.3	7.6 ± 1.6	10.7 ± 2.7	31.2 ± 10.9	16.7
GraphPrompt	8.1 ± 2.4	7.7 ± 1.5	6.1 ± 3.1	6.1 ± 0.9	7.0
SAMGPT	13.4 ± 4.3	9.9 ± 2.7	7.4 ± 2.8	8.4 ± 2.3	9.8
MDGFM	4.3 ± 1.3	19.6 ± 4.9	6.8 ± 3.9	12.6 ± 7.7	10.8
BRIDGE	4.2 ± 0.9	8.3 ± 1.6	4.1 ± 0.7	6.3 ± 0.6	5.7
GILT	3.1 ± 0.7	13.4 ± 6.7	4.3 ± 2.0	6.4 ± 1.7	6.8

Table 19: AUPRC (%) on unseen organic targets under the multi-source injected-to-organic setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Weibo	Reddit	Amz-all	YelpChi	Yelp-all	Toloker	Question	Elliptic	DGraph	T-Social	Disney	Avg.	Rank
Conventional Baselines													
DOMINANT	**92.6** ± 0.7	3.7 ± 0.0	4.1 ± 0.0	6.9 ± 1.3	13.2 ± 0.3	24.4 ± 0.2	3.6 ± 0.0	9.9 ± 0.2	**2.2** ± 0.1	10.6 ± 0.9	4.3 ± 0.1	15.9	8
CoLA	18.3 ± 8.2	3.9 ± 0.6	8.4 ± 3.1	5.7 ± 0.4	15.2 ± 1.5	24.8 ± 3.0	3.1 ± 0.1	10.6 ± 0.8	1.1 ± 0.1	3.4 ± 0.4	17.1 ± 10.4	10.1	27
TAM	24.7 ± 11.2	4.1 ± 0.1	17.0 ± 1.3	5.3 ± 0.2	12.3 ± 0.7	22.9 ± 0.5	3.7 ± 0.3	10.4 ± 0.2	2.1 ± 0.2	3.4 ± 0.8	6.9 ± 1.3	10.2	26
BWGNN	36.1 ± 24.3	3.8 ± 1.0	7.2 ± 2.4	6.0 ± 1.3	14.2 ± 1.5	25.7 ± 5.5	4.4 ± 1.3	9.4 ± 2.1	1.3 ± 0.3	3.3 ± 1.0	5.5 ± 1.6	10.6	24
GHRN	57.5 ± 17.5	3.0 ± 0.5	16.8 ± 8.0	5.8 ± 1.2	14.1 ± 2.2	22.6 ± 4.4	3.7 ± 0.8	7.3 ± 1.4	1.1 ± 0.1	4.9 ± 2.4	10.5 ± 4.1	13.4	13
MLP	60.7 ± 1.3	3.0 ± 0.3	14.4 ± 4.6	4.9 ± 0.5	14.0 ± 0.9	19.7 ± 1.1	4.0 ± 0.6	5.8 ± 0.3	1.0 ± 0.1	2.7 ± 0.2	12.0 ± 4.6	12.9	16
GCN	47.1 ± 24.1	3.2 ± 0.9	13.9 ± 9.4	6.9 ± 1.3	15.2 ± 1.5	22.4 ± 5.6	3.2 ± 1.1	9.4 ± 1.0	1.2 ± 0.2	3.3 ± 0.8	5.5 ± 2.2	11.9	18
GAT	26.3 ± 19.5	3.6 ± 0.6	10.1 ± 5.9	4.0 ± 0.6	14.4 ± 0.5	23.1 ± 3.3	2.8 ± 0.3	12.1 ± 2.6	1.2 ± 0.1	5.4 ± 1.4	12.7 ± 8.1	10.5	25
RF-Graph	18.1 ± 3.1	3.7 ± 0.2	46.0 ± 4.2	3.8 ± 0.0	14.5 ± 0.2	20.9 ± 0.4	2.9 ± 0.1	7.0 ± 0.1	1.1 ± 0.0	6.2 ± 0.4	3.7 ± 0.2	11.6	21
XGB-Graph	47.9 ± 7.5	3.3 ± 0.2	7.1 ± 2.3	3.4 ± 0.0	13.4 ± 0.1	23.2 ± 1.0	2.8 ± 0.1	9.9 ± 0.7	1.1 ± 0.0	3.8 ± 0.7	6.8 ± 1.3	11.1	23
Generalist GAD Methods													
ARC	63.9 ± 1.2	4.2 ± 0.2	8.7 ± 0.9	4.9 ± 0.2	18.0 ± 1.0	23.0 ± 1.7	3.9 ± 0.1	6.1 ± 0.3	1.2 ± 0.1	3.0 ± 0.4	7.4 ± 1.8	13.1	15
UNPrompt	22.3 ± 6.7	3.4 ± 0.2	6.3 ± 1.4	6.2 ± 1.1	14.4 ± 0.9	20.1 ± 1.8	2.4 ± 0.1	7.3 ± 0.8	1.1 ± 0.2	2.2 ± 0.1	9.2 ± 4.9	8.6	29
AnomalyGFM	7.6 ± 1.7	2.9 ± 0.1	8.4 ± 5.2	5.5 ± 0.1	17.9 ± 1.0	20.1 ± 2.4	3.3 ± 0.1	11.8 ± 0.8	1.1 ± 0.2	2.9 ± 0.2	9.9 ± 1.3	8.3	30
ZeroGAD	6.5 ± 0.2	4.2 ± 0.4	44.9 ± 15.1	OOM	OOM	18.9 ± 0.3	OOM	OOM	OOM	OOM	13.6 ± 8.9	17.6	6
IA-GGAD	72.9 ± 0.3	4.1 ± 0.3	26.3 ± 3.2	5.5 ± 0.3	12.7 ± 0.4	22.7 ± 1.0	3.9 ± 0.1	6.0 ± 0.1	1.2 ± 0.1	2.9 ± 0.2	5.2 ± 0.6	14.8	9
DR-GGAD	35.9 ± 1.0	**4.9** ± 0.1	6.1 ± 0.2	4.6 ± 0.2	16.5 ± 0.6	18.7 ± 0.4	3.2 ± 0.1	7.7 ± 0.1	1.1 ± 0.0	4.0 ± 0.2	5.1 ± 0.1	9.8	28
OWLEYE	59.7 ± 1.0	4.1 ± 0.2	4.6 ± 0.1	5.1 ± 0.2	18.2 ± 0.7	20.8 ± 1.5	3.5 ± 0.3	5.7 ± 0.0	1.0 ± 0.0	3.0 ± 0.4	5.7 ± 0.1	11.9	19
TA-GGAD	72.8 ± 0.3	3.4 ± 0.1	10.3 ± 2.9	5.5 ± 0.2	17.6 ± 2.2	23.2 ± 1.5	3.9 ± 0.1	6.0 ± 0.1	1.6 ± 0.1	2.9 ± 0.1	5.5 ± 0.1	13.9	11
ProMoS	72.5 ± 0.2	3.5 ± 0.1	10.9 ± 3.0	5.9 ± 0.5	20.0 ± 3.3	17.9 ± 0.7	3.3 ± 0.1	5.4 ± 0.0	1.0 ± 0.0	2.6 ± 0.0	6.5 ± 0.0	13.6	12
SAARCS	60.0 ± 4.3	3.6 ± 0.2	9.2 ± 0.7	4.1 ± 0.0	19.1 ± 0.9	24.5 ± 1.7	3.7 ± 0.1	8.1 ± 0.7	1.5 ± 0.2	4.8 ± 1.1	7.1 ± 2.5	13.2	14
TPCA-GAD	35.6 ± 11.7	3.7 ± 0.3	14.1 ± 1.7	9.4 ± 1.3	15.9 ± 0.2	28.4 ± 1.4	2.4 ± 0.2	6.7 ± 0.3	1.3 ± 0.1	2.1 ± 0.3	6.5 ± 0.5	11.5	22
GAD-MoRE	58.9 ± 1.5	3.6 ± 0.1	4.2 ± 0.0	6.5 ± 0.1	15.2 ± 0.2	20.2 ± 0.5	4.0 ± 0.3	6.2 ± 0.1	1.2 ± 0.0	4.2 ± 0.1	6.3 ± 0.2	11.9	20
NeighborDiv	55.8 ± 0.7	3.6 ± 0.0	17.7 ± 0.3	6.9 ± 0.0	15.2 ± 0.1	21.7 ± 0.0	2.8 ± 0.0	11.3 ± 0.0	1.5 ± 0.0	3.7 ± 0.0	16.2 ± 0.0	14.2	10
REFIGAD	56.9 ± 4.5	4.4 ± 0.8	13.4 ± 2.7	10.0 ± 2.9	16.2 ± 1.2	25.1 ± 2.7	4.4 ± 0.8	19.6 ± 3.9	1.3 ± 0.2	21.3 ± 9.1	**38.5** ± 11.7	19.2	4
TFM4GAD	83.2 ± 6.6	4.2 ± 0.7	70.7 ± 13.5	**20.1** ± 8.1	**21.5** ± 2.6	**34.4** ± 7.2	**5.3** ± 1.5	**63.7** ± 6.0	1.8 ± 0.5	**59.5** ± 13.3	–	**36.4**	**1**
Graph Foundation Models													
GraphPrompt	55.3 ± 18.5	3.7 ± 0.5	20.5 ± 5.0	9.0 ± 0.8	14.6 ± 1.0	23.5 ± 2.8	3.8 ± 0.8	16.8 ± 6.6	1.4 ± 0.2	23.4 ± 13.2	4.2 ± 2.7	16.0	7
SAMGPT	81.2 ± 19.9	4.8 ± 0.8	27.6 ± 7.4	10.3 ± 2.8	15.5 ± 1.2	29.9 ± 2.8	4.0 ± 0.8	28.6 ± 7.1	2.0 ± 0.6	43.9 ± 7.7	3.1 ± 2.5	**22.8**	**2**
MDGFM	72.7 ± 5.7	3.8 ± 0.7	**76.3** ± 2.1	7.6 ± 1.0	17.3 ± 3.2	31.7 ± 3.5	4.7 ± 1.2	21.5 ± 12.9	1.8 ± 0.3	9.8 ± 6.1	2.4 ± 1.0	**22.7**	**3**
BRIDGE	82.7 ± 8.4	4.0 ± 0.5	28.4 ± 1.9	9.9 ± 2.3	15.0 ± 1.0	28.8 ± 3.1	4.1 ± 1.0	18.8 ± 6.2	2.1 ± 0.4	8.6 ± 3.8	8.3 ± 14.6	19.2	5
GILT	52.8 ± 21.7	3.2 ± 0.7	9.7 ± 7.0	6.2 ± 2.3	14.5 ± 1.8	21.9 ± 2.3	3.1 ± 0.6	8.8 ± 2.3	1.4 ± 0.4	2.7 ± 0.4	15.1 ± 27.7	12.7	17

Table 20: AUPRC (%) on unseen injected targets under the multi-source injected-to-injected setting. marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational.

Method	PubMed	Flickr	CS	Photo	Avg.
DOMINANT	8.9 ± 0.3	27.6 ± 0.2	12.5 ± 0.3	8.0 ± 0.0	14.2
CoLA	6.6 ± 0.9	9.9 ± 2.0	4.9 ± 1.2	8.9 ± 2.9	7.6
TAM	5.6 ± 2.0	16.4 ± 1.8	8.0 ± 0.9	9.8 ± 3.5	9.9
BWGNN	5.2 ± 2.3	12.7 ± 6.0	7.5 ± 2.9	8.5 ± 3.9	8.5
GHRN	8.3 ± 2.6	27.6 ± 5.4	17.8 ± 6.3	8.1 ± 3.5	15.5
MLP	11.0 ± 1.1	37.3 ± 0.3	22.1 ± 1.9	7.2 ± 1.5	19.4
GCN	3.9 ± 1.0	8.7 ± 0.6	4.3 ± 1.2	6.5 ± 1.1	5.9
GAT	8.8 ± 2.8	6.2 ± 0.9	6.3 ± 1.1	6.3 ± 0.7	6.9
RF-Graph	38.8 ± 1.7	31.6 ± 0.6	33.5 ± 1.2	25.1 ± 1.4	32.3
XGB-Graph	6.8 ± 0.5	28.8 ± 1.3	20.0 ± 0.8	8.3 ± 0.5	16.0
ARC	33.5 ± 0.4	38.9 ± 0.7	35.8 ± 0.2	32.3 ± 1.3	35.1
UNPrompt	5.3 ± 0.7	24.5 ± 2.4	10.9 ± 0.8	6.8 ± 1.9	11.9
AnomalyGFM	3.6 ± 0.2	6.9 ± 0.4	4.1 ± 0.3	6.3 ± 0.4	5.2
ZeroGAD	OOM	5.6 ± 0.1	OOM	22.2 ± 2.0	–
IA-GGAD	29.2 ± 1.5	37.4 ± 0.5	34.9 ± 1.2	16.7 ± 2.4	29.6

Method	PubMed	Flickr	CS	Photo	Avg.
DR-GGAD	**52.4** ± 2.6	38.2 ± 0.3	**68.3** ± 2.2	15.7 ± 1.2	**43.6**
OWLEYE	29.9 ± 0.4	38.4 ± 0.3	39.2 ± 0.2	37.3 ± 3.3	36.2
TA-GGAD	35.7 ± 0.4	39.5 ± 0.5	38.8 ± 0.4	31.8 ± 1.9	**36.4**
ProMoS	29.9 ± 0.4	36.3 ± 0.1	33.5 ± 0.1	26.7 ± 3.8	31.6
SAARCS	35.5 ± 0.3	**40.7** ± 0.2	38.5 ± 0.1	**43.9** ± 1.2	**39.6**
TPCA-GAD	22.1 ± 6.7	4.9 ± 0.9	20.7 ± 4.5	5.5 ± 0.3	13.3
GAD-MoRE	15.2 ± 0.3	27.2 ± 0.3	29.1 ± 0.5	12.4 ± 1.6	21.0
NeighborDiv	10.9 ± 0.0	5.0 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	10.8
REFIGAD	36.4 ± 10.4	28.5 ± 5.5	56.1 ± 10.9	9.5 ± 1.6	32.6
TFM4GAD	17.2 ± 9.3	7.6 ± 1.6	10.7 ± 2.7	31.2 ± 10.9	16.7
GraphPrompt	5.7 ± 2.0	6.5 ± 0.7	4.4 ± 1.0	6.6 ± 1.3	5.8
SAMGPT	12.3 ± 3.9	9.9 ± 3.1	7.6 ± 3.6	8.4 ± 2.2	9.6
MDGFM	4.8 ± 2.1	19.9 ± 7.1	5.4 ± 2.7	10.7 ± 6.7	10.2
BRIDGE	4.2 ± 0.9	8.5 ± 1.8	4.2 ± 0.6	6.2 ± 0.5	5.8
GILT	3.7 ± 0.7	20.1 ± 7.0	6.5 ± 3.0	5.2 ± 2.1	8.9

D.3 Additional Results for Efficiency and Scalability Analysis (RQ4)

All methods are profiled on an RTX 3090 with four CPU threads under seed 0. The single-source setting uses Facebook, while the multi-source setting uses Facebook, T-Finance, and Amazon. We report target-side wall time, peak GPU allocation, and peak CPU RSS. The target phase includes data preparation, target-side adaptation when applicable, anomaly scoring, and evaluation. NeighborDiv requires no source-side training.

Table 21: Target-wise test time (s) on Weibo, Reddit, Amazon-all, YelpChi, YelpChi-all, and Tolokers. Each entry reports Single / Multi.

Method	Weibo	Reddit	Amazon-all	YelpChi	YelpChi-all	Tolokers
ARC	0.54 / 0.49	0.24 / 0.18	3.58 / 3.58	0.21 / 0.19	3.45 / 3.35	0.32 / 0.28
DR-GGAD	0.67 / 0.64	0.19 / 0.18	5.72 / 5.76	0.18 / 0.15	5.17 / 5.03	0.37 / 0.44
UNPrompt	0.93 / 0.62	0.35 / 0.17	5.28 / 5.10	0.25 / 0.13	5.06 / 4.53	0.39 / 0.30
OWLEYE	0.40 / 0.42	0.21 / 0.24	1.43 / 1.44	0.28 / 0.31	1.35 / 1.54	0.21 / 0.28
ProMoS	0.35 / 0.44	0.12 / 0.12	1.94 / 1.89	0.12 / 0.15	1.71 / 1.75	0.20 / 0.20
AnomalyGFM	0.82 / 0.86	0.36 / 0.19	9.22 / 5.70	0.27 / 0.27	5.40 / 5.85	0.72 / 0.32
IA-GGAD	0.92 / 0.85	0.28 / 0.25	7.55 / 7.44	0.24 / 0.25	6.67 / 6.58	0.49 / 0.46
SAARCS	0.84 / 1.07	0.40 / 0.43	5.54 / 5.49	0.41 / 0.41	5.46 / 5.68	0.55 / 0.53
GAD-MoRE	1.18 / 1.20	0.76 / 0.93	3.80 / 4.06	1.31 / 1.36	5.13 / 5.26	0.90 / 0.94
TA-GGAD	4.53 / 4.66	4.98 / 4.90	9.68 / 9.74	9.41 / 9.44	21.45 / 21.60	5.33 / 5.27
TPCA-GAD	0.65 / 1.54	0.28 / 0.30	19.03 / 11.34	0.26 / 1.19	10.67 / 13.52	0.29 / 0.35
REFIGAD	2.72 / 2.53	0.83 / 0.79	64.98 / 67.66	0.96 / 1.07	12.53 / 12.27	0.92 / 0.97
NeighborDiv	0.39 / 0.21	0.08 / 0.06	0.50 / 0.43	0.16 / 0.06	0.64 / 0.47	0.05 / 0.04

Table 22: Target-wise test time (s) on Questions, Elliptic, DGraphFin, T-Social, and Disney. Each entry reports Single / Multi.

Method	Questions	Elliptic	DGraphFin	T-Social	Disney
ARC	0.54 / 0.48	8.28 / 7.76	13.04 / 9.00	39.30 / 32.64	0.11 / 0.06
DR-GGAD	0.49 / 0.60	7.99 / 7.82	12.95 / 12.50	142.89 / 142.56	0.05 / 0.09
UNPrompt	1.11 / 0.53	10.28 / 7.17	9.40 / 5.21	63.28 / 68.56	0.11 / 0.04
OWLEYE	0.58 / 0.75	8.29 / 8.80	22.02 / 31.93	65.12 / 76.15	0.13 / 0.11
ProMoS	0.42 / 0.45	7.92 / 8.03	7.04 / 7.10	53.26 / 53.51	0.08 / 0.06
AnomalyGFM	1.19 / 0.82	11.51 / 8.25	20.97 / 21.52	59.41 / 42.30	0.08 / 0.05
IA-GGAD	0.60 / 0.73	8.28 / 8.42	33.94 / 32.48	278.02 / 231.69	0.09 / 0.09
SAARCS	0.93 / 1.05	9.61 / 9.71	43.07 / 36.00	163.56 / 150.50	0.16 / 0.15
GAD-MoRE	3.97 / 4.66	29.64 / 25.23	204.29 / 218.39	566.60 / 377.00	0.13 / 0.11
TA-GGAD	8.44 / 8.37	28.57 / 28.10	280.64 / 259.51	1903.90 / 1917.41	1.62 / 1.58
TPCA-GAD	2.00 / 0.79	51.46 / 9.48	61.69 / 24.77	282.14 / 208.88	0.13 / 0.13
REFIGAD	2.31 / 0.90	17.03 / 8.09	34.70 / 26.72	496.45 / 508.28	0.03 / 0.03
NeighborDiv	0.31 / 0.28	7.50 / 7.17	3.04 / 2.88	17.71 / 17.10	0.03 / 0.01

Table 23: Target-wise peak GPU memory (GiB) on the first six organic targets. Each entry reports Single / Multi.

Method	Weibo	Reddit	Amazon-all	YelpChi	YelpChi-all	Tolokers
ARC	0.32 / 0.32	0.40 / 0.40	0.43 / 0.43	0.80 / 0.80	1.49 / 1.49	0.42 / 0.42
DR-GGAD	0.49 / 0.49	0.33 / 0.34	0.83 / 0.83	0.69 / 0.69	1.24 / 1.24	0.51 / 0.51
UNPrompt	0.05 / 0.05	0.05 / 0.05	0.67 / 0.67	0.08 / 0.08	0.68 / 0.68	0.06 / 0.06
OWLEYE	0.31 / 0.48	0.31 / 0.48	0.31 / 0.47	0.33 / 0.49	0.36 / 0.52	0.32 / 0.48
ProMoS	0.78 / 0.79	0.20 / 0.21	0.52 / 0.53	0.17 / 0.18	0.57 / 0.58	1.07 / 1.08
AnomalyGFM	0.12 / 0.15	0.11 / 0.14	0.47 / 0.51	0.21 / 0.24	0.69 / 0.72	0.13 / 0.16
IA-GGAD	1.50 / 1.65	0.66 / 0.80	1.03 / 1.17	1.27 / 1.41	2.60 / 2.73	1.01 / 1.15
SAARCS	0.22 / 0.22	0.28 / 0.28	0.30 / 0.30	0.58 / 0.58	0.77 / 0.77	0.30 / 0.30
GAD-MoRE	0.47 / 0.47	0.46 / 0.46	0.46 / 0.46	0.53 / 0.53	0.53 / 0.53	0.47 / 0.47
TA-GGAD	1.37 / 1.40	0.68 / 0.71	0.80 / 0.83	1.38 / 1.41	2.59 / 2.62	0.82 / 0.85
TPCA-GAD	0.51 / 0.51	0.28 / 0.28	0.55 / 0.55	0.23 / 0.23	0.61 / 0.61	0.52 / 0.52
REFIGAD	0.90 / 0.90	1.38 / 1.38	1.78 / 1.78	6.38 / 6.38	2.05 / 2.05	1.57 / 1.57

Table 24: Target-wise peak GPU memory (GiB) on the remaining five organic targets. Each entry reports Single / Multi.

Method	Questions	Elliptic	DGraphFin	T-Social	Disney
ARC	1.58 / 1.58	2.36 / 2.36	4.90 / 4.90	6.96 / 6.96	0.06 / 0.06
DR-GGAD	0.96 / 0.96	1.37 / 1.37	4.88 / 4.88	7.40 / 7.40	0.03 / 0.03
UNPrompt	0.14 / 0.14	0.61 / 0.60	4.16 / 4.16	6.21 / 6.21	0.02 / 0.02
OWLEYE	0.36 / 0.53	0.33 / 0.50	3.07 / 3.07	4.64 / 4.64	0.10 / 0.15
ProMoS	0.44 / 0.45	0.79 / 0.81	5.80 / 5.81	8.77 / 8.79	0.02 / 0.03
AnomalyGFM	0.40 / 0.43	1.56 / 1.59	1.95 / 1.98	4.26 / 4.29	0.02 / 0.05
IA-GGAD	2.46 / 2.60	3.00 / 3.15	9.29 / 9.43	13.04 / 13.19	0.14 / 0.28
SAARCS	0.77 / 0.77	0.77 / 0.77	0.90 / 0.90	0.90 / 0.90	0.04 / 0.04
GAD-MoRE	0.54 / 0.54	0.60 / 0.60	1.84 / 1.84	2.58 / 2.58	0.08 / 0.08
TA-GGAD	2.76 / 2.79	1.08 / 1.11	0.02 / 0.02	0.02 / 0.02	0.07 / 0.11
TPCA-GAD	0.52 / 0.52	0.30 / 0.30	0.36 / 0.36	0.69 / 0.69	0.02 / 0.02
REFIGAD	1.02 / 1.02	1.55 / 1.55	1.99 / 1.99	1.95 / 1.95	0.24 / 0.24

Table 25: Target-wise peak CPU RSS (GiB) on the first six organic targets. Each entry reports Single / Multi.

Method	Weibo	Reddit	Amazon-all	YelpChi	YelpChi-all	Tolokers
ARC	1.36 / 1.39	1.33 / 1.35	2.10 / 2.11	1.33 / 1.32	2.06 / 2.09	1.36 / 1.37
DR-GGAD	1.56 / 1.57	1.48 / 1.50	2.23 / 2.29	1.53 / 1.56	2.22 / 2.21	1.50 / 1.53
UNPrompt	1.43 / 1.44	1.31 / 1.32	2.37 / 2.39	1.41 / 1.33	2.35 / 2.32	1.35 / 1.37
OWLEYE	1.59 / 1.61	1.56 / 1.58	1.90 / 1.87	1.59 / 1.61	1.82 / 1.79	1.56 / 1.59
ProMoS	2.06 / 2.07	1.68 / 1.69	1.86 / 2.00	1.70 / 1.65	1.97 / 1.94	1.64 / 1.70
AnomalyGFM	1.47 / 1.49	1.37 / 1.41	2.28 / 2.29	1.44 / 1.47	2.20 / 2.27	1.39 / 1.42
IA-GGAD	1.57 / 1.60	1.54 / 1.53	2.46 / 2.48	1.57 / 1.66	2.42 / 2.49	1.54 / 1.57
SAARCS	1.35 / 1.34	1.30 / 1.32	1.57 / 1.61	1.35 / 1.34	1.60 / 1.64	1.31 / 1.34
GAD-MoRE	1.96 / 2.03	1.89 / 1.90	2.28 / 2.18	1.94 / 1.90	2.25 / 2.22	1.90 / 1.92
TA-GGAD	1.80 / 1.84	1.74 / 1.79	2.51 / 2.53	1.81 / 1.77	2.46 / 2.51	1.74 / 1.79
TPCA-GAD	1.45 / 1.49	1.43 / 1.44	1.70 / 1.67	1.45 / 1.46	1.78 / 1.79	1.43 / 1.44
REFIGAD	1.63 / 1.64	1.63 / 1.59	2.36 / 2.47	1.63 / 1.59	2.07 / 2.06	1.63 / 1.59
NeighborDiv	0.88 / 0.62	0.56 / 0.56	0.78 / 0.65	0.59 / 0.60	0.74 / 0.72	0.56 / 0.56

Table 26: Target-wise peak CPU RSS (GiB) on the remaining five organic targets. Each entry reports Single / Multi.

Method	Questions	Elliptic	DGraphFin	T-Social	Disney
ARC	1.44 / 1.44	2.05 / 2.02	2.78 / 2.82	10.07 / 10.04	1.33 / 1.35
DR-GGAD	1.56 / 1.56	2.18 / 2.23	3.06 / 3.08	16.67 / 16.71	1.47 / 1.51
UNPrompt	1.45 / 1.41	2.26 / 2.27	2.58 / 2.35	13.55 / 13.60	1.27 / 1.29
OWLEYE	1.65 / 1.64	2.26 / 2.32	7.05 / 7.05	14.15 / 14.14	1.54 / 1.58
ProMoS	1.75 / 1.76	2.49 / 2.50	3.20 / 3.07	9.31 / 9.31	1.63 / 1.67
AnomalyGFM	1.48 / 1.51	2.39 / 2.47	4.72 / 4.70	7.48 / 7.55	1.35 / 1.38
IA-GGAD	1.54 / 1.53	2.42 / 2.48	11.06 / 11.15	21.49 / 21.54	1.54 / 1.52
SAARCS	1.44 / 1.42	1.89 / 1.89	12.21 / 12.26	21.76 / 21.79	1.28 / 1.28
GAD-MoRE	2.21 / 2.06	2.83 / 2.83	4.36 / 4.16	9.76 / 9.77	1.86 / 1.88
TA-GGAD	1.80 / 1.80	3.13 / 2.90	14.02 / 14.09	25.19 / 24.84	1.70 / 1.77
TPCA-GAD	1.55 / 1.54	2.31 / 2.33	7.32 / 7.29	12.42 / 12.53	1.42 / 1.42
REFIGAD	1.63 / 1.59	2.04 / 1.78	2.72 / 2.71	11.89 / 11.76	1.63 / 1.58
NeighborDiv	0.66 / 0.66	1.91 / 1.67	1.32 / 1.50	5.09 / 5.25	0.54 / 0.54

D.4 Robustness to Alternative Source Selection

To examine whether the benchmark conclusions depend on a particular choice of source graphs, we repeat the GGAD experiments with an alternative source split. For organic sources, we use Amazon in the single-source setting and {Amazon, Questions, Reddit} in the multi-source setting. For injected sources, we use CiteSeer in the single-source setting and {CiteSeer, CS, PubMed} in the multi-source setting. All other preprocessing, model configurations, target-side access, evaluation metrics, and random seeds follow the main benchmark protocol. We report 15 GGAD configurations.

D.4.1 AUROC Results

Alternative single-source results. Tables 27–30 report AUROC under the four alternative single-source source–target combinations.

Alternative multi-source results. Tables 31–34 report AUROC under the four alternative multi-source source–target combinations.

Table 27: AUROC (%) on unseen organic targets under the alternative single-source organic-to-organic setting (Amazon source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amz-all	Disney	Question	Reddit	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	65.1 ± 1.4	81.1 ± 1.9	49.1 ± 6.6	59.2 ± 0.7	**59.3** ± 1.1	50.9 ± 3.0	26.5 ± 4.9	89.0 ± 0.3	53.9 ± 0.3	60.9 ± 1.6	47.9 ± 3.8	25.3 ± 2.9	49.0 ± 6.2	55.2	5
UNPrompt	54.5 ± 1.2	**94.7** ± 0.5	58.7 ± 2.7	54.5 ± 0.5	51.5 ± 1.8	46.7 ± 1.6	17.1 ± 0.9	83.4 ± 1.7	58.2 ± 0.8	58.8 ± 1.1	43.7 ± 1.2	26.5 ± 1.0	39.2 ± 2.0	52.9	9
AnomalyGFM	84.8 ± 1.1	82.4 ± 0.9	68.1 ± 0.6	52.7 ± 0.6	49.3 ± 0.6	39.7 ± 0.5	50.0 ± 0.8	69.8 ± 0.8	46.3 ± 0.1	58.0 ± 0.6	39.9 ± 0.7	60.7 ± 0.5	55.0 ± 2.2	58.2	4
ZeroGAD	60.3 ± 0.9	78.3 ± 4.8	42.4 ± 0.8	OOM	50.9 ± 0.6	32.9 ± 0.4	OOM	35.5 ± 2.0	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	72.9 ± 1.8	80.8 ± 2.5	42.0 ± 4.1	58.8 ± 0.8	56.3 ± 1.7	47.9 ± 3.0	19.9 ± 2.7	91.5 ± 0.3	52.6 ± 1.4	48.4 ± 1.1	48.2 ± 1.6	23.9 ± 1.1	50.8 ± 2.9	53.4	8
DR-GGAD	82.7 ± 2.6	53.0 ± 2.2	39.8 ± 0.3	53.2 ± 0.3	58.0 ± 0.5	44.6 ± 0.7	13.3 ± 0.6	53.7 ± 0.9	43.5 ± 1.4	56.0 ± 1.1	49.8 ± 0.7	42.3 ± 0.6	60.0 ± 1.1	50.0	13
OWLEYE	68.7 ± 1.1	35.4 ± 2.5	51.7 ± 1.7	50.0 ± 0.5	56.8 ± 0.9	49.4 ± 4.0	24.3 ± 2.0	87.7 ± 0.4	55.5 ± 0.6	59.2 ± 0.7	42.4 ± 1.5	13.9 ± 0.5	50.7 ± 3.5	49.7	14
TA-GGAD	68.2 ± 1.0	76.5 ± 4.3	44.2 ± 1.4	58.9 ± 0.5	55.3 ± 2.8	50.2 ± 3.9	22.3 ± 4.0	91.4 ± 0.2	54.4 ± 1.7	59.7 ± 2.6	54.5 ± 4.3	24.4 ± 1.0	51.4 ± 1.2	54.7	6
ProMoS	67.5 ± 0.7	64.0 ± 2.1	53.0 ± 0.1	50.1 ± 0.5	54.9 ± 2.9	39.3 ± 1.1	20.1 ± 0.5	93.0 ± 0.1	52.8 ± 2.8	56.8 ± 3.0	41.6 ± 0.5	14.2 ± 0.4	47.1 ± 0.2	50.3	11
SAARCS	63.5 ± 2.7	60.8 ± 3.8	50.9 ± 7.2	54.2 ± 1.0	54.1 ± 2.8	52.6 ± 3.5	29.0 ± 5.7	85.6 ± 2.8	42.0 ± 0.5	58.4 ± 1.7	55.4 ± 6.0	43.2 ± 5.2	60.5 ± 6.5	54.6	7
TPCA-GAD	82.9 ± 4.9	54.9 ± 8.2	48.0 ± 8.3	45.9 ± 1.7	49.0 ± 1.7	50.4 ± 2.3	30.6 ± 7.2	86.6 ± 5.1	70.5 ± 0.8	51.3 ± 0.7	50.8 ± 0.9	27.9 ± 1.8	35.0 ± 1.9	52.6	10
GAD-MoRE	74.3 ± 0.9	23.6 ± 0.2	54.6 ± 0.4	54.8 ± 0.1	47.6 ± 2.5	41.1 ± 0.6	16.0 ± 0.2	90.9 ± 0.2	55.4 ± 0.5	51.6 ± 0.6	49.9 ± 0.6	31.8 ± 0.5	60.6 ± 0.6	50.2	12
NeighborDiv	61.2 ± 0.0	68.8 ± 0.7	72.5 ± 0.0	45.5 ± 0.0	51.2 ± 0.0	52.9 ± 0.0	75.2 ± 0.4	79.8 ± 0.7	58.1 ± 0.0	50.7 ± 0.2	52.1 ± 0.0	56.6 ± 0.0	56.4 ± 0.0	**60.1**	**3**
REFIGAD	86.6 ± 4.5	71.4 ± 3.0	**81.1** ± 8.7	56.4 ± 4.5	53.2 ± 3.9	54.0 ± 4.0	76.3 ± 7.9	86.4 ± 4.3	70.1 ± 5.7	55.1 ± 3.9	53.7 ± 4.0	75.1 ± 3.9	84.3 ± 4.2	**69.5**	**2**
TFM4GAD	**99.5** ± 0.7	92.5 ± 3.9	–	**62.9** ± 2.8	55.5 ± 4.6	**67.2** ± 8.3	**91.8** ± 2.5	**97.5** ± 1.6	**74.0** ± 7.1	**61.0** ± 3.2	**59.9** ± 6.9	**85.0** ± 2.9	**93.0** ± 4.7	**78.3**	**1**

Table 28: AUROC (%) on unseen injected targets under the alternative single-source organic-to-injected setting (Amazon source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	CiteSeer	ACM	BlogCat.	Flickr	PubMed	CS	Photo	Avg.	Rank
ARC	87.2 ± 1.0	91.2 ± 0.5	79.8 ± 0.2	74.3 ± 0.4	73.6 ± 1.8	85.5 ± 0.3	82.9 ± 0.3	75.7 ± 0.9	81.3	5
UNPrompt	57.6 ± 1.2	51.9 ± 1.3	67.2 ± 1.1	69.5 ± 0.5	69.6 ± 0.2	65.3 ± 1.4	68.4 ± 1.6	60.5 ± 2.1	63.7	12
AnomalyGFM	56.2 ± 1.3	51.1 ± 0.3	57.2 ± 1.6	51.3 ± 1.2	52.8 ± 0.3	55.9 ± 1.8	59.3 ± 1.3	55.1 ± 1.2	54.9	13
ZeroGAD	61.3 ± 0.3	51.9 ± 0.6	OOM	62.2 ± 0.5	48.3 ± 0.7	OOM	OOM	42.5 ± 0.4	–	–
IA-GGAD	88.2 ± 0.4	91.8 ± 0.4	85.5 ± 1.1	74.2 ± 0.7	67.8 ± 1.4	90.1 ± 0.4	93.9 ± 0.4	72.1 ± 1.9	**82.9**	**3**
DR-GGAD	**92.6** ± 0.6	**94.5** ± 0.5	**91.1** ± 0.5	74.6 ± 0.5	73.7 ± 0.4	**95.8** ± 0.3	**98.0** ± 0.2	67.5 ± 1.5	**86.0**	**1**
OWLEYE	79.5 ± 0.6	84.4 ± 0.4	78.5 ± 1.1	**76.7** ± 2.1	**76.6** ± 3.9	78.3 ± 0.7	82.2 ± 1.9	74.2 ± 1.3	78.8	7
TA-GGAD	87.3 ± 0.5	91.6 ± 0.3	82.9 ± 0.4	75.1 ± 0.3	76.4 ± 1.1	87.9 ± 0.5	88.7 ± 0.8	75.8 ± 8.3	**83.2**	**2**
ProMoS	83.1 ± 0.4	89.2 ± 0.4	74.8 ± 1.8	73.4 ± 0.9	70.4 ± 0.9	81.6 ± 1.1	77.6 ± 1.1	69.3 ± 2.9	77.4	9
SAARCS	86.2 ± 0.9	92.5 ± 0.4	79.2 ± 0.5	75.8 ± 0.2	76.0 ± 0.7	84.2 ± 0.3	82.8 ± 0.9	**78.2** ± 2.3	81.9	4
TPCA-GAD	43.6 ± 10.6	45.0 ± 10.2	58.7 ± 12.2	44.2 ± 1.0	41.3 ± 5.6	44.9 ± 11.9	53.6 ± 10.2	46.1 ± 2.0	47.2	14
GAD-MoRE	78.1 ± 0.9	84.7 ± 0.6	80.4 ± 0.1	72.7 ± 0.1	75.1 ± 0.2	79.6 ± 0.2	82.9 ± 0.1	74.8 ± 0.2	78.5	8
NeighborDiv	68.6 ± 0.0	71.5 ± 0.0	66.8 ± 0.0	46.7 ± 0.0	44.8 ± 0.0	72.0 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	65.4	11
REFIGAD	88.6 ± 2.1	90.7 ± 0.9	84.2 ± 3.8	63.1 ± 3.1	69.3 ± 3.3	92.7 ± 1.7	92.0 ± 2.2	67.6 ± 4.9	81.0	6
TFM4GAD	74.3 ± 2.7	80.2 ± 3.0	65.6 ± 9.8	57.1 ± 5.5	55.4 ± 4.9	74.6 ± 3.7	75.6 ± 3.1	76.1 ± 2.8	69.9	10

Table 29: AUROC (%) on unseen organic targets under the alternative single-source injected-to-organic setting (CiteSeer source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amazon	Amz-all	Disney	Question	Reddit	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	68.3 ± 1.0	63.8 ± 2.4	63.5 ± 2.8	46.1 ± 6.0	57.4 ± 0.4	**58.8** ± 1.2	50.6 ± 4.6	26.2 ± 4.9	89.3 ± 0.4	48.8 ± 0.5	58.8 ± 1.7	47.9 ± 3.7	25.4 ± 3.0	49.5 ± 5.5	53.9	6
UNPrompt	59.5 ± 9.3	56.3 ± 4.3	55.2 ± 5.1	35.1 ± 5.4	48.8 ± 1.6	55.8 ± 1.6	46.0 ± 1.8	32.8 ± 7.8	47.5 ± 4.1	37.5 ± 5.1	50.1 ± 1.2	**62.3** ± 0.4	51.6 ± 3.1	70.9 ± 6.9	50.7	10
AnomalyGFM	54.3 ± 9.5	56.7 ± 10.9	51.4 ± 13.4	41.7 ± 13.0	50.4 ± 3.2	45.9 ± 6.3	52.4 ± 10.0	56.3 ± 4.1	38.9 ± 18.5	47.8 ± 1.0	53.5 ± 3.6	53.8 ± 9.4	47.7 ± 5.7	46.9 ± 1.8	49.8	14
ZeroGAD	82.8 ± 1.1	46.8 ± 3.9	42.9 ± 7.6	56.2 ± 1.8	OOM	54.7 ± 1.0	43.6 ± 0.5	OOM	25.1 ± 2.1	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	74.6 ± 1.7	56.3 ± 3.8	77.3 ± 2.5	42.6 ± 3.6	57.1 ± 0.4	56.8 ± 1.9	51.3 ± 3.9	20.5 ± 3.8	91.7 ± 0.4	50.6 ± 1.2	45.2 ± 1.5	48.4 ± 1.8	23.9 ± 1.1	50.6 ± 3.0	53.4	7
DR-GGAD	83.0 ± 2.3	60.9 ± 1.7	49.8 ± 2.0	39.9 ± 0.3	52.9 ± 0.3	57.9 ± 0.5	43.8 ± 0.7	13.3 ± 0.6	53.6 ± 1.1	41.6 ± 1.4	53.7 ± 1.0	49.9 ± 0.7	42.3 ± 0.6	59.7 ± 1.1	50.2	11
OWLEYE	67.2 ± 1.0	62.5 ± 3.9	31.9 ± 1.9	49.5 ± 2.2	47.4 ± 0.5	57.7 ± 1.0	47.8 ± 5.5	21.8 ± 0.5	89.5 ± 0.8	48.2 ± 1.3	56.5 ± 0.7	45.7 ± 3.2	20.2 ± 0.5	54.0 ± 7.4	50.0	12
TA-GGAD	69.3 ± 1.0	61.7 ± 2.8	64.4 ± 2.6	44.1 ± 1.3	57.8 ± 0.3	54.7 ± 2.7	52.8 ± 5.2	22.9 ± 5.0	91.5 ± 0.2	50.9 ± 1.7	55.2 ± 3.4	55.3 ± 4.4	24.6 ± 1.0	51.4 ± 1.2	54.0	5
ProMoS	69.5 ± 0.5	63.2 ± 2.4	52.2 ± 5.4	53.2 ± 0.2	48.4 ± 0.7	55.9 ± 2.2	36.5 ± 1.3	20.1 ± 1.0	92.5 ± 0.1	46.4 ± 3.3	57.8 ± 2.5	41.3 ± 0.5	13.9 ± 0.1	47.4 ± 0.2	49.9	13
SAARCS	63.5 ± 2.7	60.9 ± 2.7	60.8 ± 3.8	49.9 ± 4.7	54.2 ± 1.0	54.1 ± 2.8	52.6 ± 3.5	29.0 ± 5.7	85.8 ± 2.6	42.1 ± 0.5	58.4 ± 1.7	55.4 ± 6.0	43.1 ± 5.1	60.6 ± 6.6	55.0	4
TPCA-GAD	57.0 ± 11.6	46.9 ± 1.1	66.2 ± 1.4	51.7 ± 2.7	46.0 ± 2.5	50.3 ± 1.6	58.4 ± 2.7	25.3 ± 9.4	76.6 ± 12.6	67.1 ± 1.9	51.4 ± 0.4	51.1 ± 1.8	35.3 ± 4.0	35.4 ± 5.5	51.3	9
GAD-MoRE	80.3 ± 0.5	55.0 ± 1.6	24.7 ± 0.3	54.6 ± 0.3	55.8 ± 0.1	56.5 ± 0.6	47.0 ± 0.9	15.5 ± 0.1	91.7 ± 0.1	56.8 ± 0.4	50.1 ± 0.3	47.8 ± 0.5	28.8 ± 0.3	60.5 ± 0.5	51.8	8
NeighborDiv	61.2 ± 0.0	52.0 ± 0.0	68.8 ± 0.7	72.5 ± 0.0	45.5 ± 0.0	51.2 ± 0.0	52.9 ± 0.0	75.2 ± 0.4	79.8 ± 0.7	58.1 ± 0.0	50.7 ± 0.2	52.1 ± 0.0	56.6 ± 0.0	56.4 ± 0.0	**59.5**	**3**
REFIGAD	92.7 ± 2.0	61.3 ± 7.0	67.6 ± 2.5	**76.4** ± 8.3	57.3 ± 5.4	53.3 ± 3.9	54.9 ± 3.5	77.0 ± 3.3	88.5 ± 3.1	68.4 ± 6.9	55.4 ± 4.2	54.1 ± 5.7	75.5 ± 3.0	88.5 ± 1.7	**69.4**	**2**
TFM4GAD	**99.5** ± 0.7	**93.3** ± 2.5	**92.5** ± 3.9	–	**62.9** ± 2.8	55.5 ± 4.6	**67.2** ± 8.3	**91.8** ± 2.5	**97.5** ± 1.6	**74.0** ± 7.1	**61.0** ± 3.2	59.9 ± 6.9	**85.0** ± 2.9	**93.0** ± 4.7	**79.4**	**1**

Table 30: AUROC (%) on unseen injected targets under the alternative single-source injected-to-injected setting (CiteSeer source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	ACM	BlogCat.	Flickr	PubMed	CS	Photo	Avg.	Rank
ARC	86.5 ± 0.6	79.8 ± 0.2	74.2 ± 0.6	73.6 ± 1.6	85.4 ± 0.2	83.0 ± 0.4	75.5 ± 1.0	79.7	6
UNPrompt	77.3 ± 1.5	70.7 ± 2.6	68.6 ± 1.0	68.2 ± 0.9	85.9 ± 1.5	80.6 ± 1.6	76.0 ± 4.5	75.3	9
AnomalyGFM	48.5 ± 4.8	52.3 ± 3.5	55.5 ± 2.9	50.5 ± 2.1	49.4 ± 3.9	51.8 ± 3.1	45.4 ± 4.2	50.5	14
ZeroGAD	74.9 ± 0.4	OOM	63.0 ± 0.5	49.8 ± 0.5	OOM	OOM	70.7 ± 1.0	–	–
IA-GGAD	87.4 ± 0.5	85.8 ± 1.2	74.7 ± 0.5	69.4 ± 1.2	89.5 ± 0.5	93.5 ± 0.4	78.2 ± 2.4	**82.6**	**3**
DR-GGAD	92.6 ± 0.6	**91.0** ± 0.5	74.6 ± 0.4	73.6 ± 0.4	**96.0** ± 0.3	**98.0** ± 0.2	68.4 ± 1.3	**84.9**	**1**
OWLEYE	80.2 ± 0.7	78.7 ± 1.2	74.6 ± 1.1	**76.9** ± 4.8	78.3 ± 0.5	82.6 ± 0.6	75.0 ± 0.4	78.1	8
TA-GGAD	86.6 ± 0.4	82.9 ± 0.5	75.1 ± 0.5	76.9 ± 1.1	87.8 ± 0.7	89.3 ± 0.8	74.9 ± 3.5	81.9	4
ProMoS	84.3 ± 0.2	70.1 ± 1.3	72.6 ± 0.8	67.9 ± 0.9	81.3 ± 0.7	75.4 ± 0.9	68.1 ± 3.1	74.2	10
SAARCS	86.2 ± 0.9	79.2 ± 0.5	**75.8** ± 0.2	76.0 ± 0.7	84.2 ± 0.3	82.8 ± 0.9	**78.2** ± 2.3	80.3	5
TPCA-GAD	69.0 ± 9.1	76.0 ± 0.7	43.5 ± 0.8	46.2 ± 6.3	71.0 ± 2.9	72.4 ± 1.8	52.1 ± 5.1	61.4	13
GAD-MoRE	84.3 ± 0.5	81.5 ± 0.1	72.8 ± 0.1	74.5 ± 0.1	83.3 ± 0.2	82.9 ± 0.1	78.0 ± 0.1	79.6	7
NeighborDiv	68.6 ± 0.0	66.8 ± 0.0	46.7 ± 0.0	44.8 ± 0.0	72.0 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	64.5	12
REFIGAD	**93.6** ± 2.8	89.3 ± 3.3	68.1 ± 3.4	73.3 ± 4.7	95.4 ± 1.8	96.6 ± 2.7	65.1 ± 4.7	**83.0**	**2**
TFM4GAD	74.3 ± 2.7	65.6 ± 9.8	57.1 ± 5.5	55.4 ± 4.9	74.6 ± 3.7	75.6 ± 3.1	76.1 ± 2.8	68.4	11

Table 31: AUROC (%) on unseen organic targets under the alternative multi-source organic-to-organic setting (Amazon, Questions, and Reddit sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amz-all	Disney	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	64.4 ± 1.6	86.9 ± 1.9	44.3 ± 8.6	51.0 ± 2.1	26.5 ± 5.0	89.1 ± 0.4	55.2 ± 0.6	61.3 ± 1.5	47.9 ± 3.8	25.3 ± 2.8	48.6 ± 5.6	54.6	6
UNPrompt	53.9 ± 2.8	**93.2** ± 1.3	54.4 ± 3.9	44.5 ± 2.0	17.4 ± 1.3	71.1 ± 2.5	60.1 ± 1.4	**63.2** ± 1.0	43.1 ± 1.7	33.6 ± 1.1	32.0 ± 1.4	51.5	10
AnomalyGFM	72.4 ± 1.3	62.5 ± 5.9	67.4 ± 2.2	47.5 ± 5.6	44.0 ± 2.0	74.5 ± 1.7	46.4 ± 0.1	58.0 ± 0.7	40.4 ± 0.6	59.0 ± 0.7	45.5 ± 3.2	56.1	4
ZeroGAD	85.3 ± 1.1	76.6 ± 5.0	46.5 ± 1.4	37.8 ± 0.7	OOM	21.3 ± 1.1	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	71.6 ± 1.7	84.5 ± 1.6	41.3 ± 4.5	50.2 ± 2.1	20.0 ± 2.8	91.5 ± 0.3	53.2 ± 0.9	48.1 ± 1.2	48.3 ± 1.6	23.9 ± 1.1	50.8 ± 3.0	53.0	9
DR-GGAD	81.2 ± 3.5	61.5 ± 2.5	39.6 ± 0.3	46.3 ± 0.7	13.4 ± 0.5	53.6 ± 1.1	46.3 ± 1.1	58.0 ± 1.1	49.8 ± 0.9	42.3 ± 0.6	59.8 ± 1.2	50.2	12
OWLEYE	65.9 ± 4.2	36.3 ± 5.7	49.1 ± 6.8	47.8 ± 0.7	26.6 ± 4.1	85.7 ± 1.5	55.2 ± 1.0	59.2 ± 0.6	40.8 ± 1.5	14.4 ± 3.4	52.0 ± 4.7	48.5	14
TA-GGAD	67.8 ± 0.9	84.5 ± 2.3	43.4 ± 1.5	51.1 ± 3.4	22.3 ± 4.0	91.3 ± 0.3	55.7 ± 1.5	59.7 ± 4.0	55.3 ± 4.2	24.4 ± 1.0	51.4 ± 1.2	55.2	5
ProMoS	68.3 ± 0.4	61.4 ± 2.1	53.0 ± 0.1	41.5 ± 2.2	20.6 ± 0.4	92.8 ± 0.1	54.6 ± 1.6	58.7 ± 2.8	41.6 ± 0.3	14.2 ± 0.4	47.3 ± 0.1	50.4	11
SAARCS	63.5 ± 2.7	60.7 ± 3.8	49.8 ± 5.4	52.7 ± 3.4	29.0 ± 5.7	86.1 ± 2.3	42.0 ± 0.5	58.4 ± 1.7	55.4 ± 6.0	42.4 ± 4.9	60.1 ± 6.2	54.6	7
TPCA-GAD	78.8 ± 3.9	57.8 ± 7.9	51.7 ± 2.7	52.0 ± 1.6	23.0 ± 3.4	91.8 ± 3.8	70.4 ± 0.9	52.2 ± 0.4	50.8 ± 0.8	26.3 ± 1.5	36.0 ± 3.0	53.7	8
GAD-MoRE	68.2 ± 1.2	25.5 ± 1.1	54.7 ± 0.4	43.1 ± 0.4	17.0 ± 0.2	90.8 ± 0.3	52.4 ± 0.5	50.4 ± 0.3	49.2 ± 0.5	30.1 ± 0.4	59.3 ± 0.4	49.2	13
NeighborDiv	61.2 ± 0.0	68.8 ± 0.7	72.5 ± 0.0	52.9 ± 0.0	75.2 ± 0.4	79.8 ± 0.7	58.1 ± 0.0	50.7 ± 0.2	52.1 ± 0.0	56.6 ± 0.0	56.4 ± 0.0	**62.2**	**3**
REFIGAD	91.2 ± 3.4	72.1 ± 3.1	**80.3** ± 6.7	54.7 ± 2.8	78.1 ± 7.3	87.9 ± 4.4	72.1 ± 3.9	56.1 ± 2.8	52.8 ± 4.2	72.8 ± 4.3	87.3 ± 3.5	**73.2**	**2**
TFM4GAD	**99.5** ± 0.7	92.5 ± 3.9	–	**67.2** ± 8.3	**91.8** ± 2.5	**97.5** ± 1.6	**74.0** ± 7.1	61.0 ± 3.2	**59.9** ± 6.9	**85.0** ± 2.9	**93.0** ± 4.7	**82.1**	**1**

Table 32: AUROC (%) on unseen injected targets under the alternative multi-source organic-to-injected setting (Amazon, Questions, and Reddit sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	CiteSeer	ACM	BlogCat.	Flickr	PubMed	CS	Photo	Avg.	Rank
ARC	87.2 ± 0.9	91.0 ± 0.4	79.8 ± 0.2	74.3 ± 0.4	73.6 ± 1.8	85.4 ± 0.3	82.9 ± 0.2	73.4 ± 1.4	80.9	6
UNPrompt	50.7 ± 0.6	43.4 ± 1.2	59.7 ± 2.4	68.3 ± 0.5	69.1 ± 0.2	61.3 ± 3.3	64.0 ± 2.5	55.3 ± 2.7	59.0	12
AnomalyGFM	57.6 ± 0.7	52.6 ± 0.5	58.7 ± 1.3	51.8 ± 0.9	54.1 ± 0.7	58.8 ± 0.9	60.4 ± 0.7	57.6 ± 1.2	56.5	13
ZeroGAD	70.1 ± 0.4	72.2 ± 0.8	OOM	46.9 ± 1.5	45.9 ± 0.4	OOM	OOM	72.5 ± 0.9	–	–
IA-GGAD	88.0 ± 0.4	91.5 ± 0.4	85.1 ± 1.2	74.5 ± 0.5	70.6 ± 1.5	89.4 ± 0.5	93.3 ± 0.4	71.1 ± 1.5	**82.9**	**2**
DR-GGAD	**92.4** ± 0.7	**94.2** ± 0.5	**90.9** ± 0.6	74.5 ± 0.6	73.7 ± 0.5	**95.4** ± 0.3	**98.1** ± 0.2	71.6 ± 1.6	**86.3**	**1**
OWLEYE	79.2 ± 1.7	83.5 ± 1.9	78.6 ± 1.4	75.0 ± 1.6	75.4 ± 4.6	77.1 ± 5.0	82.0 ± 1.8	71.0 ± 4.7	77.7	9
TA-GGAD	87.3 ± 0.5	91.4 ± 0.2	82.9 ± 0.4	75.1 ± 0.4	**76.8** ± 0.9	87.5 ± 0.4	88.5 ± 0.7	72.7 ± 4.0	**82.8**	**3**
ProMoS	83.9 ± 0.2	90.4 ± 0.2	77.8 ± 0.9	73.7 ± 0.4	70.5 ± 0.2	82.1 ± 0.3	78.9 ± 0.4	71.0 ± 0.8	78.5	7
SAARCS	86.2 ± 0.9	92.5 ± 0.4	79.2 ± 0.5	**75.8** ± 0.2	76.0 ± 0.7	84.2 ± 0.3	82.8 ± 0.9	**78.3** ± 2.3	81.9	5
TPCA-GAD	51.1 ± 4.7	51.0 ± 4.8	68.4 ± 4.5	40.5 ± 0.8	34.3 ± 2.3	62.8 ± 2.7	62.1 ± 4.8	41.8 ± 0.9	51.5	14
GAD-MoRE	77.4 ± 0.6	81.4 ± 0.7	80.0 ± 0.2	73.5 ± 0.1	73.5 ± 0.4	79.3 ± 0.4	82.3 ± 0.1	75.0 ± 0.8	77.8	8
NeighborDiv	68.6 ± 0.0	71.5 ± 0.0	66.8 ± 0.0	46.7 ± 0.0	44.8 ± 0.0	72.0 ± 0.0	88.3 ± 0.0	64.0 ± 0.0	65.4	11
REFIGAD	88.2 ± 4.6	92.9 ± 1.6	85.9 ± 2.0	63.9 ± 3.3	70.8 ± 4.7	93.4 ± 3.3	94.2 ± 1.6	66.6 ± 2.2	82.0	4
TFM4GAD	74.3 ± 2.7	80.2 ± 3.0	65.6 ± 9.8	57.1 ± 5.5	55.4 ± 4.9	74.6 ± 3.7	75.6 ± 3.1	76.1 ± 2.8	69.9	10

Table 33: AUROC (%) on unseen organic targets under the alternative multi-source injected-to-organic setting (CiteSeer, CS, and PubMed sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amazon	Amz-all	Disney	Question	Reddit	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	68.9 ± 1.1	58.3 ± 2.2	54.8 ± 5.0	48.9 ± 6.9	57.1 ± 0.6	**58.3** ± 1.2	50.6 ± 5.3	26.1 ± 4.8	89.4 ± 0.4	46.0 ± 0.8	57.3 ± 1.7	47.9 ± 3.7	25.3 ± 2.8	49.5 ± 5.3	52.7	7
UNPrompt	67.7 ± 3.8	52.7 ± 3.1	44.8 ± 6.2	55.8 ± 7.8	45.0 ± 1.8	53.7 ± 1.2	47.7 ± 4.0	33.7 ± 7.5	55.6 ± 3.0	53.2 ± 2.4	45.6 ± 1.5	43.9 ± 3.8	32.7 ± 2.6	44.2 ± 5.0	48.3	14
AnomalyGFM	64.4 ± 12.5	51.4 ± 14.0	49.2 ± 11.5	67.6 ± 1.9	52.8 ± 1.5	48.5 ± 3.2	42.5 ± 5.6	61.4 ± 4.2	33.3 ± 11.1	47.2 ± 0.7	55.8 ± 5.9	42.9 ± 8.4	57.0 ± 1.3	46.2 ± 2.1	51.4	8
ZeroGAD	82.8 ± 1.1	46.8 ± 3.9	42.9 ± 7.4	56.4 ± 1.9	OOM	54.7 ± 1.0	43.6 ± 0.5	OOM	25.1 ± 2.1	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	74.6 ± 1.6	55.6 ± 3.8	74.5 ± 1.8	42.9 ± 3.4	56.6 ± 0.6	55.0 ± 1.7	52.7 ± 3.7	20.1 ± 3.5	91.8 ± 0.4	50.1 ± 1.2	44.9 ± 1.3	48.6 ± 1.7	23.9 ± 1.1	50.6 ± 3.1	53.0	5
DR-GGAD	83.0 ± 2.2	60.7 ± 1.7	49.0 ± 2.0	40.0 ± 0.4	52.8 ± 0.3	57.8 ± 0.4	43.9 ± 0.7	13.3 ± 0.6	53.4 ± 1.2	41.3 ± 1.3	53.4 ± 1.1	49.8 ± 0.7	42.2 ± 0.6	59.8 ± 1.1	50.0	12
OWLEYE	65.6 ± 2.4	53.3 ± 3.3	30.3 ± 0.9	49.7 ± 1.2	46.8 ± 0.7	55.8 ± 1.2	47.4 ± 5.4	22.7 ± 1.8	89.3 ± 0.3	44.6 ± 1.1	54.9 ± 0.7	43.5 ± 1.7	23.4 ± 0.6	52.1 ± 5.7	48.5	13
TA-GGAD	69.6 ± 0.9	54.9 ± 2.0	60.3 ± 3.8	44.4 ± 1.4	57.5 ± 0.5	52.6 ± 2.0	53.9 ± 4.9	21.9 ± 4.3	91.5 ± 0.2	49.6 ± 1.8	53.2 ± 3.1	55.1 ± 4.4	24.5 ± 1.0	51.3 ± 1.3	52.9	6
ProMoS	69.3 ± 0.4	63.5 ± 2.7	53.6 ± 2.7	53.1 ± 0.1	49.9 ± 0.2	52.6 ± 2.2	39.3 ± 1.2	20.4 ± 0.5	92.2 ± 0.1	51.8 ± 2.3	58.3 ± 3.8	41.3 ± 0.2	13.7 ± 0.1	47.3 ± 0.2	50.4	11
SAARCS	63.5 ± 2.8	60.9 ± 2.8	60.7 ± 3.8	50.8 ± 5.3	54.1 ± 1.0	54.1 ± 2.8	52.7 ± 3.4	29.0 ± 5.7	86.2 ± 2.2	42.0 ± 0.5	58.4 ± 1.7	55.4 ± 6.0	42.9 ± 5.0	60.4 ± 6.2	55.1	4
TPCA-GAD	58.2 ± 13.5	46.7 ± 1.1	64.3 ± 2.1	53.2 ± 2.9	45.8 ± 2.4	50.6 ± 2.7	55.9 ± 2.9	21.3 ± 7.7	75.2 ± 12.1	68.5 ± 2.3	51.6 ± 0.4	51.1 ± 1.1	32.7 ± 3.6	37.3 ± 7.5	50.9	10
GAD-MoRE	54.7 ± 2.4	68.1 ± 4.1	24.4 ± 0.6	53.8 ± 0.5	55.0 ± 0.9	57.0 ± 0.4	57.3 ± 0.7	15.4 ± 0.4	90.9 ± 0.7	54.0 ± 1.0	52.5 ± 0.9	48.0 ± 1.0	31.1 ± 0.8	57.4 ± 0.5	51.4	9
NeighborDiv	61.2 ± 0.0	52.0 ± 0.0	68.8 ± 0.7	72.5 ± 0.0	45.5 ± 0.0	51.2 ± 0.0	52.9 ± 0.0	75.2 ± 0.4	79.8 ± 0.7	58.1 ± 0.0	50.7 ± 0.2	52.1 ± 0.0	56.6 ± 0.0	56.4 ± 0.0	**59.5**	**3**
REFIGAD	90.9 ± 2.8	61.3 ± 6.0	68.9 ± 2.9	**77.4** ± 7.4	57.2 ± 6.0	53.2 ± 4.7	54.5 ± 2.8	78.2 ± 6.0	87.7 ± 3.3	67.2 ± 6.9	55.2 ± 5.1	53.9 ± 4.8	73.6 ± 5.1	87.2 ± 1.3	**69.0**	**2**
TFM4GAD	**99.5** ± 0.7	**93.3** ± 2.5	**92.5** ± 3.9	–	**62.9** ± 2.8	55.5 ± 4.6	**67.2** ± 8.3	**91.8** ± 2.5	**97.5** ± 1.6	**74.0** ± 7.1	**61.0** ± 3.2	**59.9** ± 6.9	**85.0** ± 2.9	**93.0** ± 4.7	**79.4**	**1**

Table 34: AUROC (%) on unseen injected targets under the alternative multi-source injected-to-injected setting (CiteSeer, CS, and PubMed sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	ACM	BlogCat.	Flickr	Photo	Avg.	Rank
ARC	86.6 ± 0.8	79.8 ± 0.2	74.3 ± 0.6	73.6 ± 1.6	75.3 ± 0.9	77.9	6
UNPrompt	52.4 ± 2.9	71.4 ± 0.3	69.1 ± 0.5	70.2 ± 0.4	60.4 ± 3.3	64.7	11
AnomalyGFM	54.2 ± 3.4	59.1 ± 2.9	57.5 ± 2.4	53.3 ± 3.1	50.7 ± 3.7	55.0	14
ZeroGAD	74.9 ± 0.4	OOM	63.0 ± 0.6	49.8 ± 0.6	70.7 ± 1.0	–	–
IA-GGAD	87.0 ± 0.6	85.2 ± 1.3	74.8 ± 0.4	69.6 ± 1.5	**81.5** ± 1.3	**79.6**	**2**
DR-GGAD	92.6 ± 0.7	91.0 ± 0.5	74.7 ± 0.4	73.5 ± 0.4	68.3 ± 1.1	**80.0**	**1**
OWLEYE	80.1 ± 0.9	78.1 ± 0.2	74.9 ± 1.2	74.6 ± 2.9	74.8 ± 0.3	76.5	7
TA-GGAD	86.4 ± 0.4	83.3 ± 0.4	75.0 ± 0.3	**77.0** ± 1.2	74.0 ± 3.2	**79.2**	**3**
ProMoS	84.4 ± 0.2	75.7 ± 0.5	73.5 ± 0.7	70.1 ± 0.3	70.9 ± 0.5	74.9	8
SAARCS	86.2 ± 0.9	79.2 ± 0.5	**75.8** ± 0.2	76.0 ± 0.7	78.4 ± 2.3	79.1	4
TPCA-GAD	69.5 ± 8.0	75.0 ± 0.6	42.1 ± 0.8	45.9 ± 6.7	48.9 ± 5.6	56.3	13
GAD-MoRE	69.6 ± 0.5	77.1 ± 0.3	74.2 ± 0.3	75.9 ± 0.5	74.2 ± 1.1	74.2	9
NeighborDiv	68.6 ± 0.0	66.8 ± 0.0	46.7 ± 0.0	44.8 ± 0.0	64.0 ± 0.0	58.2	12
REFIGAD	**94.1** ± 2.2	**93.6** ± 1.7	67.8 ± 3.6	73.3 ± 5.0	62.9 ± 5.0	78.4	5
TFM4GAD	74.3 ± 2.7	65.6 ± 9.8	57.1 ± 5.5	55.4 ± 4.9	76.1 ± 2.8	65.7	10

D.4.2 AUPRC Results

Alternative single-source results. Tables 35–38 report AUPRC under the four alternative single-source source–target combinations.

Table 35: AUPRC (%) on unseen organic targets under the alternative single-source organic-to-organic setting (Amazon source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amz-all	Disney	Question	Reddit	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	5.2 ± 1.0	38.5 ± 6.5	6.8 ± 1.4	4.4 ± 0.1	4.3 ± 0.2	22.9 ± 1.0	2.9 ± 0.2	63.2 ± 1.2	6.3 ± 0.2	21.4 ± 1.1	1.2 ± 0.1	6.1 ± 0.2	3.0 ± 0.5	14.3	4
UNPrompt	2.5 ± 0.1	**84.1** ± 0.7	6.6 ± 0.6	3.4 ± 0.1	3.4 ± 0.2	21.4 ± 0.6	2.6 ± 0.0	52.8 ± 2.7	7.0 ± 0.2	19.3 ± 0.6	1.1 ± 0.0	6.2 ± 0.1	2.3 ± 0.1	**16.4**	**3**
AnomalyGFM	14.5 ± 5.0	49.1 ± 2.7	11.3 ± 1.8	3.0 ± 0.1	3.0 ± 0.0	17.7 ± 0.4	5.0 ± 0.4	28.2 ± 6.0	5.4 ± 0.0	17.5 ± 0.5	1.0 ± 0.0	14.1 ± 0.6	3.1 ± 0.1	13.3	8
ZeroGAD	3.4 ± 0.1	23.1 ± 9.7	4.6 ± 0.1	OOM	3.1 ± 0.1	15.3 ± 0.1	OOM	13.6 ± 0.3	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	7.1 ± 1.2	29.2 ± 3.7	5.1 ± 0.7	4.3 ± 0.1	4.0 ± 0.5	22.0 ± 0.8	2.6 ± 0.1	72.4 ± 0.3	5.9 ± 0.3	13.6 ± 0.3	1.2 ± 0.1	6.0 ± 0.1	2.9 ± 0.2	13.6	7
DR-GGAD	14.3 ± 3.9	6.9 ± 0.5	5.0 ± 0.1	3.4 ± 0.1	**5.0** ± 0.2	19.2 ± 0.4	2.5 ± 0.0	36.4 ± 0.9	4.9 ± 0.3	18.4 ± 0.9	1.1 ± 0.0	7.7 ± 0.1	3.9 ± 0.1	9.9	14
OWLEYE	7.6 ± 1.2	4.9 ± 0.2	6.0 ± 0.2	3.8 ± 0.1	3.8 ± 0.2	22.5 ± 1.6	2.8 ± 0.1	57.3 ± 0.8	6.7 ± 0.2	18.9 ± 0.8	1.0 ± 0.0	5.4 ± 0.0	3.3 ± 1.4	11.1	11
TA-GGAD	7.5 ± 1.3	26.1 ± 12.1	5.5 ± 0.1	4.3 ± 0.1	3.7 ± 0.3	22.7 ± 1.1	2.7 ± 0.1	72.4 ± 0.3	5.9 ± 0.4	19.7 ± 2.0	1.5 ± 0.1	6.0 ± 0.1	2.9 ± 0.1	13.9	6
ProMoS	7.0 ± 1.4	20.6 ± 2.9	6.5 ± 0.0	3.6 ± 0.1	3.9 ± 0.4	17.6 ± 0.6	2.6 ± 0.0	74.1 ± 0.3	6.0 ± 0.6	18.9 ± 2.2	1.0 ± 0.0	5.4 ± 0.0	2.6 ± 0.0	13.1	9
SAARCS	4.7 ± 0.5	9.2 ± 0.7	7.9 ± 5.4	3.7 ± 0.1	3.6 ± 0.2	24.2 ± 1.6	3.0 ± 0.2	59.1 ± 5.6	4.1 ± 0.0	19.1 ± 0.9	1.5 ± 0.2	8.2 ± 0.7	4.8 ± 1.1	11.8	10
TPCA-GAD	9.1 ± 2.7	8.9 ± 2.9	5.6 ± 1.0	2.6 ± 0.1	3.3 ± 0.2	21.8 ± 1.3	3.1 ± 0.3	42.6 ± 17.0	9.1 ± 0.5	15.7 ± 0.4	1.3 ± 0.0	6.2 ± 0.1	2.1 ± 0.1	10.1	13
GAD-MoRE	8.3 ± 1.3	4.2 ± 0.0	6.2 ± 0.1	3.4 ± 0.0	3.0 ± 0.3	19.4 ± 0.3	2.5 ± 0.0	60.1 ± 0.8	6.4 ± 0.2	14.6 ± 0.2	1.3 ± 0.0	6.6 ± 0.0	4.5 ± 0.1	10.8	12
NeighborDiv	5.0 ± 0.0	17.7 ± 0.3	16.2 ± 0.0	2.8 ± 0.0	3.6 ± 0.0	21.7 ± 0.0	20.7 ± 0.2	55.8 ± 0.7	6.9 ± 0.0	15.2 ± 0.1	1.5 ± 0.0	11.3 ± 0.0	3.7 ± 0.0	14.0	5
REFIGAD	16.4 ± 6.0	24.2 ± 5.3	**42.5** ± 11.8	3.9 ± 0.6	4.1 ± 0.5	22.9 ± 2.2	16.7 ± 10.7	47.2 ± 9.5	11.3 ± 2.6	19.0 ± 1.9	1.4 ± 0.2	21.7 ± 6.4	12.3 ± 5.6	**18.7**	**2**
TFM4GAD	**86.5** ± 7.2	70.7 ± 13.5	–	**5.3** ± 1.5	4.2 ± 0.7	**34.4** ± 7.2	**74.2** ± 5.3	**83.2** ± 6.6	**20.1** ± 8.1	**21.5** ± 2.6	**1.8** ± 0.5	**63.7** ± 6.0	**59.5** ± 13.3	**43.8**	**1**

Alternative multi-source results. Tables 39–42 report AUPRC under the four alternative multi-source source–target combinations.

Table 36: AUPRC (%) on unseen injected targets under the alternative single-source organic-to-injected setting (Amazon source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	CiteSeer	ACM	BlogCat.	Flickr	PubMed	CS	Photo	Avg.	Rank
ARC	47.0 ± 2.3	46.4 ± 1.1	40.8 ± 0.1	35.3 ± 0.2	38.8 ± 0.5	33.8 ± 0.2	36.5 ± 0.0	34.1 ± 1.4	39.1	4
UNPrompt	8.2 ± 1.0	5.4 ± 0.5	10.2 ± 1.5	26.8 ± 0.4	23.0 ± 1.0	5.5 ± 0.5	8.9 ± 1.5	7.6 ± 0.6	11.9	11
AnomalyGFM	6.8 ± 0.3	5.0 ± 0.4	4.4 ± 0.1	5.9 ± 0.1	6.7 ± 0.1	3.5 ± 0.1	4.5 ± 0.2	6.5 ± 0.2	5.4	13
ZeroGAD	7.4 ± 0.1	5.4 ± 0.2	OOM	11.0 ± 0.5	5.7 ± 0.1	OOM	OOM	7.7 ± 0.4	–	–
IA-GGAD	44.1 ± 1.0	48.8 ± 2.1	43.1 ± 0.6	34.7 ± 0.5	36.5 ± 0.7	32.2 ± 1.3	38.9 ± 1.4	13.7 ± 2.1	36.5	6
DR-GGAD	**55.5** ± 4.1	**62.4** ± 2.1	**50.9** ± 1.3	36.8 ± 1.0	38.2 ± 0.3	**50.6** ± 2.6	**70.3** ± 1.9	14.0 ± 1.1	**47.3**	**1**
OWLEYE	45.0 ± 1.2	45.0 ± 0.5	39.4 ± 0.2	36.0 ± 0.4	38.7 ± 0.4	30.5 ± 0.2	39.3 ± 0.2	36.7 ± 2.7	38.8	5
TA-GGAD	50.5 ± 1.1	48.6 ± 0.7	42.0 ± 0.2	36.1 ± 0.1	39.9 ± 0.2	35.6 ± 0.2	38.5 ± 0.5	28.4 ± 5.9	**39.9**	**3**
ProMoS	41.3 ± 0.9	43.3 ± 1.0	38.5 ± 0.5	35.5 ± 0.2	36.5 ± 0.2	29.2 ± 0.5	33.4 ± 0.3	23.7 ± 5.6	35.2	7
SAARCS	47.3 ± 0.9	50.8 ± 0.7	41.5 ± 0.1	35.9 ± 0.1	**40.6** ± 0.2	35.5 ± 0.2	38.5 ± 0.1	**43.9** ± 1.1	**41.7**	**2**
TPCA-GAD	5.0 ± 1.1	4.5 ± 1.0	8.1 ± 9.1	5.0 ± 0.2	5.0 ± 0.6	3.0 ± 1.8	3.9 ± 1.8	5.1 ± 0.2	5.0	14
GAD-MoRE	19.0 ± 0.8	23.4 ± 1.6	33.6 ± 0.3	**39.1** ± 0.2	31.7 ± 0.2	14.6 ± 0.2	34.3 ± 0.7	23.3 ± 0.8	27.4	8
NeighborDiv	9.8 ± 0.0	12.3 ± 0.0	8.2 ± 0.0	5.3 ± 0.0	5.0 ± 0.0	10.9 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	9.8	12
REFIGAD	29.7 ± 10.0	34.2 ± 5.7	20.4 ± 5.9	8.8 ± 1.2	12.4 ± 2.8	35.5 ± 8.1	25.4 ± 7.2	13.7 ± 3.7	22.5	9
TFM4GAD	42.1 ± 12.5	47.3 ± 7.7	10.8 ± 3.9	8.0 ± 2.4	7.6 ± 1.6	17.2 ± 9.3	10.7 ± 2.7	31.2 ± 10.8	21.9	10

Table 37: AUPRC (%) on unseen organic targets under the alternative single-source injected-to-organic setting (CiteSeer source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amazon	Amz-all	Disney	Question	Reddit	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	6.6 ± 1.2	9.8 ± 1.3	10.3 ± 1.5	6.7 ± 1.9	3.9 ± 0.1	4.3 ± 0.2	22.9 ± 1.5	2.9 ± 0.2	63.7 ± 1.2	5.1 ± 0.1	20.0 ± 1.4	1.2 ± 0.1	6.1 ± 0.2	3.0 ± 0.5	11.9	7
UNPrompt	7.2 ± 2.4	8.4 ± 1.2	8.6 ± 1.5	4.3 ± 0.4	3.3 ± 0.4	3.9 ± 0.3	19.9 ± 0.8	3.2 ± 0.4	16.0 ± 2.4	3.7 ± 0.4	14.8 ± 0.5	**2.2** ± 0.1	9.9 ± 0.9	7.8 ± 3.8	8.1	13
AnomalyGFM	4.0 ± 2.8	9.8 ± 6.7	8.6 ± 4.2	5.1 ± 1.7	3.1 ± 0.3	2.9 ± 0.4	23.5 ± 4.6	6.0 ± 1.0	10.4 ± 4.1	5.5 ± 0.1	15.8 ± 1.5	1.4 ± 0.3	8.9 ± 0.9	2.6 ± 0.1	7.7	14
ZeroGAD	7.2 ± 0.9	6.8 ± 0.9	5.7 ± 0.9	6.8 ± 0.7	OOM	4.3 ± 0.3	18.9 ± 0.3	OOM	7.8 ± 0.5	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	7.7 ± 1.4	8.1 ± 1.3	26.0 ± 3.8	5.2 ± 0.6	3.9 ± 0.1	4.2 ± 0.4	23.0 ± 1.0	2.7 ± 0.1	72.8 ± 0.3	5.7 ± 0.2	12.6 ± 0.4	1.2 ± 0.1	6.0 ± 0.1	2.9 ± 0.2	13.0	4
DR-GGAD	16.4 ± 4.2	8.5 ± 0.6	6.2 ± 0.3	5.1 ± 0.1	3.2 ± 0.1	**5.0** ± 0.2	18.8 ± 0.4	2.5 ± 0.0	36.3 ± 0.9	4.6 ± 0.2	16.7 ± 0.8	1.1 ± 0.0	7.7 ± 0.1	3.9 ± 0.1	9.7	12
OWLEYE	6.6 ± 1.0	10.5 ± 3.1	4.7 ± 0.1	5.6 ± 0.3	3.3 ± 0.1	4.0 ± 0.2	21.6 ± 1.9	2.7 ± 0.0	63.6 ± 1.4	5.1 ± 0.2	17.3 ± 0.5	1.1 ± 0.1	5.7 ± 0.0	3.5 ± 1.5	11.1	10
TA-GGAD	6.7 ± 0.7	9.2 ± 1.5	10.3 ± 1.4	5.5 ± 0.1	3.9 ± 0.1	3.6 ± 0.3	23.4 ± 1.6	2.8 ± 0.2	72.8 ± 0.3	5.1 ± 0.2	17.1 ± 2.1	1.6 ± 0.1	6.0 ± 0.1	2.9 ± 0.1	12.2	5
ProMoS	7.6 ± 0.7	11.2 ± 1.6	9.7 ± 3.2	6.5 ± 0.0	3.2 ± 0.1	4.3 ± 0.4	16.7 ± 0.6	2.6 ± 0.0	73.3 ± 0.2	4.9 ± 0.4	19.2 ± 1.9	1.0 ± 0.0	5.4 ± 0.0	2.6 ± 0.0	12.0	6
SAARCS	4.7 ± 0.5	8.7 ± 1.2	9.2 ± 0.7	6.8 ± 1.8	3.7 ± 0.1	3.6 ± 0.2	24.2 ± 1.6	3.0 ± 0.2	59.2 ± 5.4	4.1 ± 0.0	19.1 ± 0.9	1.5 ± 0.2	8.2 ± 0.7	4.8 ± 1.2	11.5	8
TPCA-GAD	3.2 ± 1.8	6.7 ± 0.3	13.1 ± 1.2	6.5 ± 0.6	2.6 ± 0.2	3.4 ± 0.2	28.5 ± 2.4	2.9 ± 0.5	40.2 ± 15.8	8.7 ± 0.8	15.7 ± 0.2	1.3 ± 0.1	7.0 ± 0.6	2.1 ± 0.2	10.1	11
GAD-MoRE	12.1 ± 1.7	7.5 ± 0.4	4.3 ± 0.0	6.2 ± 0.1	3.4 ± 0.0	3.8 ± 0.1	21.9 ± 0.4	2.5 ± 0.0	62.3 ± 0.6	6.5 ± 0.1	13.9 ± 0.2	1.2 ± 0.0	6.3 ± 0.0	4.4 ± 0.1	11.2	9
NeighborDiv	5.0 ± 0.0	7.3 ± 0.0	17.7 ± 0.3	16.2 ± 0.0	2.8 ± 0.0	3.6 ± 0.0	21.7 ± 0.0	20.7 ± 0.2	55.8 ± 0.7	6.9 ± 0.0	15.2 ± 0.1	1.5 ± 0.0	11.3 ± 0.0	3.7 ± 0.0	**13.5**	**3**
REFIGAD	19.8 ± 5.2	12.0 ± 2.5	14.9 ± 3.3	**20.6** ± 8.7	4.6 ± 0.7	4.2 ± 0.7	24.3 ± 2.3	11.7 ± 3.5	46.9 ± 6.3	8.6 ± 2.2	17.6 ± 1.8	1.4 ± 0.2	22.7 ± 7.5	16.5 ± 5.2	**16.1**	**2**
TFM4GAD	**86.5** ± 7.2	**77.5** ± 7.0	**70.7** ± 13.5	–	**5.3** ± 1.5	4.2 ± 0.7	**34.4** ± 7.2	**74.2** ± 5.3	**83.2** ± 6.6	**20.1** ± 8.1	**21.5** ± 2.6	1.8 ± 0.5	**63.7** ± 6.0	**59.5** ± 13.3	**46.4**	**1**

Table 38: AUPRC (%) on unseen injected targets under the alternative single-source injected-to-injected setting (CiteSeer source). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	ACM	BlogCat.	Flickr	PubMed	CS	Photo	Avg.	Rank
ARC	49.0 ± 1.5	40.8 ± 0.1	36.1 ± 0.3	38.7 ± 0.6	33.6 ± 0.3	36.5 ± 0.1	33.4 ± 1.0	38.3	5
UNPrompt	33.8 ± 7.6	9.6 ± 1.6	25.1 ± 3.0	24.7 ± 1.1	25.2 ± 7.1	24.3 ± 4.1	26.7 ± 5.9	24.2	10
AnomalyGFM	6.1 ± 0.7	4.3 ± 0.3	7.3 ± 0.7	6.7 ± 0.5	3.2 ± 0.4	4.3 ± 0.5	5.7 ± 0.8	5.4	14
ZeroGAD	14.5 ± 0.3	OOM	14.6 ± 1.5	5.5 ± 0.1	OOM	OOM	19.5 ± 0.8	–	–
IA-GGAD	44.6 ± 1.2	43.1 ± 0.6	35.4 ± 0.3	36.8 ± 0.6	30.4 ± 1.2	37.8 ± 1.1	17.0 ± 2.9	35.0	7
DR-GGAD	**56.1** ± 4.1	**50.7** ± 1.4	37.0 ± 1.0	38.1 ± 0.4	**52.2** ± 2.7	**69.8** ± 1.9	16.4 ± 1.3	**45.8**	**1**
OWLEYE	44.3 ± 1.4	40.1 ± 0.1	36.7 ± 0.3	39.2 ± 1.0	30.1 ± 0.3	39.7 ± 0.2	40.3 ± 2.8	38.6	4
TA-GGAD	49.7 ± 1.2	42.2 ± 0.3	36.4 ± 0.2	39.5 ± 0.4	35.7 ± 0.3	39.1 ± 0.4	31.6 ± 2.0	**39.2**	**3**
ProMoS	46.3 ± 0.6	37.7 ± 0.4	35.4 ± 0.2	36.1 ± 0.2	30.7 ± 0.4	33.9 ± 0.4	30.9 ± 7.8	35.8	6
SAARCS	47.3 ± 0.9	41.5 ± 0.1	35.9 ± 0.1	**40.6** ± 0.2	35.5 ± 0.2	38.5 ± 0.1	**43.9** ± 1.1	**40.4**	**2**
TPCA-GAD	12.6 ± 4.5	35.0 ± 8.4	4.8 ± 0.2	5.7 ± 1.5	11.0 ± 5.8	15.4 ± 4.6	5.9 ± 0.6	12.9	12
GAD-MoRE	35.9 ± 1.1	36.3 ± 0.1	**38.0** ± 0.2	32.6 ± 0.1	23.1 ± 0.8	36.0 ± 0.4	34.2 ± 0.6	33.7	9
NeighborDiv	9.8 ± 0.0	8.2 ± 0.0	5.3 ± 0.0	5.0 ± 0.0	10.9 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	9.5	13
REFIGAD	53.3 ± 9.7	44.7 ± 11.5	13.7 ± 2.6	15.2 ± 4.6	43.1 ± 10.5	62.1 ± 4.5	12.0 ± 1.8	34.9	8
TFM4GAD	42.1 ± 12.5	10.8 ± 3.9	8.0 ± 2.4	7.6 ± 1.6	17.2 ± 9.3	10.7 ± 2.7	31.2 ± 10.8	18.2	11

Table 39: AUPRC (%) on unseen organic targets under the alternative multi-source organic-to-organic setting (Amazon, Questions, and Reddit sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amz-all	Disney	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	4.8 ± 0.7	56.9 ± 4.5	6.0 ± 1.5	22.7 ± 0.9	2.9 ± 0.2	63.4 ± 1.3	6.6 ± 0.2	**21.9** ± 1.1	1.2 ± 0.1	6.1 ± 0.2	3.0 ± 0.5	17.8	5
UNPrompt	2.5 ± 0.2	**83.0** ± 0.6	6.4 ± 0.9	19.9 ± 0.9	2.6 ± 0.0	43.7 ± 3.0	7.2 ± 0.2	21.2 ± 0.4	1.1 ± 0.0	7.1 ± 0.2	2.0 ± 0.0	17.9	4
AnomalyGFM	4.2 ± 0.3	10.1 ± 3.5	12.8 ± 4.8	21.4 ± 2.9	4.0 ± 0.3	35.9 ± 2.3	5.4 ± 0.1	18.5 ± 0.8	1.0 ± 0.0	13.5 ± 1.1	2.6 ± 0.2	11.7	13
ZeroGAD	18.6 ± 2.6	20.1 ± 6.9	5.0 ± 0.1	16.6 ± 0.2	OOM	7.6 ± 0.7	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	6.7 ± 1.4	36.7 ± 3.1	5.0 ± 0.7	22.3 ± 0.5	2.6 ± 0.1	72.4 ± 0.2	6.1 ± 0.2	13.4 ± 0.4	1.2 ± 0.1	6.0 ± 0.1	2.9 ± 0.2	15.9	7
DR-GGAD	11.6 ± 2.6	12.2 ± 2.5	5.0 ± 0.1	19.9 ± 0.4	2.5 ± 0.0	36.1 ± 1.0	5.4 ± 0.3	19.9 ± 0.9	1.1 ± 0.0	7.7 ± 0.1	4.0 ± 0.1	11.4	14
OWLEYE	6.1 ± 1.5	5.0 ± 0.7	5.7 ± 0.9	22.0 ± 0.4	2.8 ± 0.2	54.1 ± 3.0	6.6 ± 0.3	19.4 ± 0.8	1.0 ± 0.0	5.4 ± 0.2	3.0 ± 0.4	11.9	11
TA-GGAD	7.7 ± 1.1	51.6 ± 7.6	5.3 ± 0.2	22.6 ± 1.0	2.7 ± 0.1	72.2 ± 0.4	6.2 ± 0.4	19.7 ± 3.2	1.6 ± 0.1	6.0 ± 0.1	2.9 ± 0.1	**18.1**	**3**
ProMoS	6.5 ± 0.4	18.0 ± 2.6	6.5 ± 0.0	18.4 ± 0.9	2.7 ± 0.0	73.7 ± 0.2	6.5 ± 0.5	20.9 ± 2.5	1.0 ± 0.0	5.4 ± 0.0	2.6 ± 0.0	14.7	8
SAARCS	4.7 ± 0.5	9.2 ± 0.7	7.1 ± 2.5	24.4 ± 1.7	3.0 ± 0.2	59.6 ± 4.8	4.1 ± 0.0	19.0 ± 0.9	1.5 ± 0.2	8.2 ± 0.7	4.7 ± 1.1	13.2	9
TPCA-GAD	6.0 ± 1.8	10.2 ± 3.2	6.2 ± 0.4	23.2 ± 1.5	2.8 ± 0.1	60.0 ± 14.3	9.1 ± 0.6	16.1 ± 0.2	1.3 ± 0.0	6.1 ± 0.1	2.1 ± 0.1	13.0	10
GAD-MoRE	4.4 ± 0.3	4.3 ± 0.1	6.2 ± 0.1	19.8 ± 0.1	2.5 ± 0.0	60.5 ± 0.8	5.6 ± 0.1	14.0 ± 0.1	1.3 ± 0.0	6.4 ± 0.0	4.1 ± 0.1	11.8	12
NeighborDiv	5.0 ± 0.0	17.7 ± 0.3	16.2 ± 0.0	21.7 ± 0.0	20.7 ± 0.2	55.8 ± 0.7	6.9 ± 0.0	15.2 ± 0.1	1.5 ± 0.0	11.3 ± 0.0	3.7 ± 0.0	16.0	6
REFIGAD	24.8 ± 11.2	25.1 ± 5.6	**38.4** ± 11.3	23.2 ± 1.5	20.7 ± 15.5	41.7 ± 10.3	12.3 ± 3.3	19.2 ± 0.9	1.4 ± 0.2	19.7 ± 5.4	17.0 ± 6.0	**22.1**	**2**
TFM4GAD	**86.5** ± 7.2	70.7 ± 13.5	–	**34.4** ± 7.2	**74.2** ± 5.3	**83.2** ± 6.6	**20.1** ± 8.1	21.5 ± 2.6	**1.8** ± 0.5	**63.7** ± 6.0	**59.5** ± 13.3	**51.6**	**1**

Table 40: AUPRC (%) on unseen injected targets under the alternative multi-source organic-to-injected setting (Amazon, Questions, and Reddit sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	CiteSeer	ACM	BlogCat.	Flickr	PubMed	CS	Photo	Avg.	Rank
ARC	46.7 ± 2.3	45.8 ± 1.0	40.8 ± 0.1	34.6 ± 0.2	38.7 ± 0.6	33.7 ± 0.3	36.5 ± 0.1	26.0 ± 5.9	37.9	4
UNPrompt	5.5 ± 0.1	3.7 ± 0.1	4.5 ± 0.4	26.7 ± 1.1	19.1 ± 1.1	3.9 ± 0.5	5.1 ± 0.9	6.4 ± 0.3	9.4	12
AnomalyGFM	6.8 ± 0.2	4.7 ± 0.1	4.6 ± 0.2	5.9 ± 0.1	7.0 ± 0.2	3.6 ± 0.1	5.0 ± 0.2	7.0 ± 0.2	5.6	13
ZeroGAD	14.1 ± 0.4	10.6 ± 0.4	OOM	7.3 ± 0.7	5.4 ± 0.1	OOM	OOM	23.8 ± 1.5	–	–
IA-GGAD	43.7 ± 1.1	48.0 ± 1.9	42.8 ± 0.5	34.0 ± 0.3	37.6 ± 0.5	31.4 ± 1.3	36.6 ± 1.3	14.3 ± 2.3	36.0	5
DR-GGAD	**53.3** ± 4.1	**60.3** ± 2.5	**50.4** ± 1.3	35.8 ± 1.0	38.0 ± 0.3	48.0 ± 2.9	**71.0** ± 2.0	17.7 ± 2.1	**46.8**	**1**
OWLEYE	41.1 ± 7.5	40.0 ± 5.8	39.4 ± 0.7	35.3 ± 0.4	38.1 ± 0.9	28.1 ± 4.2	38.6 ± 0.8	20.0 ± 8.6	35.1	7
TA-GGAD	50.5 ± 1.0	48.3 ± 0.6	41.9 ± 0.2	35.3 ± 0.1	39.9 ± 0.2	35.1 ± 0.4	38.5 ± 0.5	19.4 ± 4.1	**38.6**	**3**
ProMoS	44.6 ± 0.8	45.8 ± 0.4	38.9 ± 0.1	35.6 ± 0.1	36.5 ± 0.1	29.8 ± 0.4	33.6 ± 0.1	21.9 ± 2.8	35.8	6
SAARCS	47.3 ± 0.9	50.8 ± 0.7	41.5 ± 0.1	35.9 ± 0.1	**40.7** ± 0.2	35.5 ± 0.2	38.5 ± 0.1	**43.9** ± 1.1	**41.7**	**2**
TPCA-GAD	5.4 ± 0.6	4.6 ± 0.4	7.7 ± 2.6	4.5 ± 0.1	4.3 ± 0.1	4.6 ± 1.0	4.3 ± 0.7	4.7 ± 0.1	5.0	14
GAD-MoRE	18.9 ± 0.8	19.3 ± 1.2	31.8 ± 0.5	**38.2** ± 0.3	29.7 ± 0.2	16.2 ± 0.2	27.3 ± 0.6	22.6 ± 2.2	25.5	9
NeighborDiv	9.8 ± 0.0	12.3 ± 0.0	8.2 ± 0.0	5.3 ± 0.0	5.0 ± 0.0	10.9 ± 0.0	17.6 ± 0.0	9.5 ± 0.0	9.8	11
REFIGAD	39.0 ± 13.6	54.7 ± 13.0	26.7 ± 11.7	9.5 ± 1.6	14.0 ± 3.2	**50.7** ± 13.2	37.6 ± 14.3	11.5 ± 1.8	30.5	8
TFM4GAD	42.1 ± 12.5	47.3 ± 7.7	10.8 ± 3.9	8.0 ± 2.4	7.6 ± 1.6	17.2 ± 9.3	10.7 ± 2.7	31.2 ± 10.8	21.9	10

Table 41: AUPRC (%) on unseen organic targets under the alternative multi-source injected-to-organic setting (CiteSeer, CS, and PubMed sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Facebook	Amazon	Amz-all	Disney	Question	Reddit	Toloker	T-Fin.	Weibo	YelpChi	Yelp-all	DGraph	Elliptic	T-Social	Avg.	Rank
ARC	7.2 ± 1.5	8.0 ± 0.8	7.6 ± 0.8	7.0 ± 1.4	3.8 ± 0.1	4.2 ± 0.3	22.9 ± 1.8	2.9 ± 0.2	63.8 ± 1.2	4.7 ± 0.1	18.1 ± 1.1	1.2 ± 0.1	6.1 ± 0.2	3.0 ± 0.4	11.5	8
UNPrompt	5.2 ± 1.5	7.0 ± 0.6	5.9 ± 0.8	7.0 ± 1.2	2.7 ± 0.2	3.5 ± 0.2	21.2 ± 2.2	3.3 ± 0.5	20.0 ± 3.3	5.7 ± 0.8	12.8 ± 0.4	1.1 ± 0.1	6.9 ± 0.4	2.5 ± 0.3	7.5	14
AnomalyGFM	6.2 ± 4.6	10.8 ± 9.3	9.2 ± 6.5	11.3 ± 2.0	3.1 ± 0.1	3.0 ± 0.3	18.5 ± 2.7	6.4 ± 0.4	9.8 ± 5.8	5.5 ± 0.1	17.2 ± 2.6	1.1 ± 0.2	11.8 ± 0.5	2.6 ± 0.1	8.3	13
ZeroGAD	7.2 ± 0.9	6.8 ± 0.9	5.6 ± 0.9	6.9 ± 0.9	OOM	4.3 ± 0.3	18.9 ± 0.3	OOM	7.8 ± 0.5	OOM	OOM	OOM	OOM	OOM	–	–
IA-GGAD	8.6 ± 1.7	7.9 ± 1.3	22.9 ± 2.8	5.2 ± 0.6	3.6 ± 0.1	4.0 ± 0.3	23.4 ± 1.1	2.6 ± 0.1	72.9 ± 0.3	5.4 ± 0.2	12.5 ± 0.3	1.2 ± 0.1	6.0 ± 0.1	2.9 ± 0.2	12.8	4
DR-GGAD	16.7 ± 4.1	8.4 ± 0.6	6.1 ± 0.3	5.1 ± 0.1	3.1 ± 0.1	**4.9** ± 0.1	18.8 ± 0.4	2.5 ± 0.0	36.0 ± 1.0	4.5 ± 0.2	16.6 ± 0.8	1.1 ± 0.0	7.7 ± 0.1	3.9 ± 0.1	9.7	11
OWLEYE	5.8 ± 1.3	7.1 ± 0.7	4.5 ± 0.1	5.6 ± 0.2	2.8 ± 0.1	3.6 ± 0.2	21.3 ± 2.0	2.7 ± 0.1	63.2 ± 0.8	4.4 ± 0.2	16.0 ± 0.4	1.0 ± 0.0	5.9 ± 0.0	3.0 ± 0.5	10.5	10
TA-GGAD	8.2 ± 1.8	7.1 ± 0.5	9.1 ± 2.0	5.6 ± 0.2	3.7 ± 0.1	3.5 ± 0.3	23.8 ± 1.5	2.7 ± 0.1	72.8 ± 0.3	4.9 ± 0.2	15.5 ± 1.4	1.6 ± 0.1	6.0 ± 0.1	2.9 ± 0.1	12.0	6
ProMoS	7.1 ± 0.4	11.2 ± 2.1	9.9 ± 2.2	6.5 ± 0.0	3.3 ± 0.1	3.6 ± 0.2	17.6 ± 0.5	2.6 ± 0.0	72.7 ± 0.1	5.8 ± 0.6	20.0 ± 2.7	1.0 ± 0.0	5.4 ± 0.0	2.6 ± 0.0	12.1	5
SAARCS	4.7 ± 0.5	8.7 ± 1.2	9.2 ± 0.7	7.4 ± 2.4	3.6 ± 0.1	3.6 ± 0.2	24.4 ± 1.7	3.0 ± 0.2	59.9 ± 4.5	4.1 ± 0.0	19.0 ± 0.9	1.5 ± 0.2	8.2 ± 0.7	4.8 ± 1.1	11.6	7
TPCA-GAD	3.3 ± 1.7	6.6 ± 0.2	12.1 ± 1.1	6.7 ± 0.8	2.6 ± 0.2	3.4 ± 0.3	27.2 ± 2.6	2.7 ± 0.3	34.4 ± 10.8	9.1 ± 0.9	15.9 ± 0.1	1.3 ± 0.1	6.7 ± 0.5	2.2 ± 0.3	9.6	12
GAD-MoRE	3.9 ± 0.7	14.9 ± 4.0	4.2 ± 0.0	6.1 ± 0.1	3.5 ± 0.2	3.7 ± 0.1	26.5 ± 0.4	2.5 ± 0.0	54.2 ± 1.7	7.0 ± 0.5	15.1 ± 0.3	1.2 ± 0.0	6.5 ± 0.1	3.8 ± 0.1	10.9	9
NeighborDiv	5.0 ± 0.0	7.3 ± 0.0	17.7 ± 0.3	16.2 ± 0.0	2.8 ± 0.0	3.6 ± 0.0	21.7 ± 0.0	20.7 ± 0.2	55.8 ± 0.7	6.9 ± 0.0	15.2 ± 0.1	1.5 ± 0.0	11.3 ± 0.0	3.7 ± 0.0	**13.5**	**3**
REFIGAD	17.7 ± 4.4	12.1 ± 2.1	15.8 ± 4.5	**20.0** ± 9.5	4.5 ± 0.8	4.1 ± 0.6	23.5 ± 1.8	17.9 ± 10.0	44.5 ± 6.8	8.4 ± 2.1	17.2 ± 2.0	1.4 ± 0.2	19.4 ± 6.1	15.2 ± 6.1	**15.8**	**2**
TFM4GAD	**86.5** ± 7.2	**77.5** ± 7.0	**70.7** ± 13.5	–	**5.3** ± 1.5	4.2 ± 0.7	**34.4** ± 7.2	**74.2** ± 5.3	**83.2** ± 6.6	**20.1** ± 8.1	**21.5** ± 2.6	**1.8** ± 0.5	**63.7** ± 6.0	**59.5** ± 13.3	**46.4**	**1**

Table 42: AUPRC (%) on unseen injected targets under the alternative multi-source injected-to-injected setting (CiteSeer, CS, and PubMed sources). marks the best result on each target; , , and denote the overall first-, second-, and third-ranked methods, respectively. OOM denotes infeasible evaluation due to excessive computational or memory cost.

Method	Cora	ACM	BlogCat.	Flickr	Photo	Avg.	Rank
ARC	47.8 ± 1.1	40.8 ± 0.1	36.6 ± 0.2	38.7 ± 0.7	33.3 ± 1.3	39.5	5
UNPrompt	5.9 ± 0.5	11.1 ± 0.9	29.8 ± 1.9	26.1 ± 1.4	8.5 ± 1.8	16.3	11
AnomalyGFM	6.3 ± 0.3	4.8 ± 0.2	7.8 ± 0.4	6.9 ± 0.4	6.2 ± 0.4	6.4	14
ZeroGAD	14.5 ± 0.4	OOM	14.6 ± 1.5	5.5 ± 0.1	19.5 ± 0.8	–	–
IA-GGAD	42.8 ± 1.6	42.7 ± 0.5	36.0 ± 0.5	37.5 ± 0.5	19.4 ± 2.2	35.7	7
DR-GGAD	**56.4** ± 4.1	50.5 ± 1.4	**37.2** ± 1.0	38.2 ± 0.3	16.8 ± 1.2	**39.8**	**2**
OWLEYE	43.2 ± 1.9	40.0 ± 0.1	36.8 ± 0.3	39.1 ± 0.8	40.0 ± 2.2	**39.8**	**3**
TA-GGAD	48.4 ± 1.5	42.2 ± 0.2	36.9 ± 0.3	39.6 ± 0.4	30.7 ± 1.9	39.6	4
ProMoS	46.5 ± 0.4	38.7 ± 0.2	35.5 ± 0.2	36.4 ± 0.2	27.4 ± 3.1	36.9	6
SAARCS	47.3 ± 0.9	41.5 ± 0.1	35.9 ± 0.1	**40.7** ± 0.2	**43.9** ± 1.1	**41.9**	**1**
TPCA-GAD	12.7 ± 3.8	36.1 ± 9.5	4.6 ± 0.1	5.8 ± 1.5	5.5 ± 0.7	12.9	12
GAD-MoRE	14.3 ± 0.3	28.5 ± 0.6	35.2 ± 0.6	30.9 ± 0.4	26.3 ± 7.0	27.1	9
NeighborDiv	9.8 ± 0.0	8.2 ± 0.0	5.3 ± 0.0	5.0 ± 0.0	9.5 ± 0.0	7.6	13
REFIGAD	54.0 ± 14.0	**57.0** ± 8.9	12.6 ± 2.1	15.2 ± 5.0	10.9 ± 1.9	29.9	8
TFM4GAD	42.1 ± 12.5	10.8 ± 3.9	8.0 ± 2.4	7.6 ± 1.6	31.2 ± 10.8	19.9	10

E Additional Related Work

E.1 Graph Anomaly Detection

Graph anomaly detection (GAD) aims to identify nodes whose attributes, structural patterns, or relational behaviors deviate from the majority of a graph.

Unsupervised Graph Anomaly Detection. Early deep GAD methods mainly rely on reconstruction, assuming that regular graph patterns can be more faithfully recovered than anomalous ones. Graph autoencoders therefore reconstruct node attributes, graph structures, or their interactions and use reconstruction discrepancies for anomaly scoring Ding et al. (2019); Fan et al. (2020); Luo et al. (2022). Recent studies further refine this paradigm by moving from direct feature or adjacency reconstruction toward neighborhood-level reconstruction and by reducing the contamination of abnormal patterns during representation learning Roy et al. (2024) Despite their simplicity, reconstruction-based objectives may also reconstruct anomalous observations well when anomalies are included during training, weakening the separation between normal and abnormal nodes.

Self-supervised GAD instead constructs supervision from the graph itself and characterizes anomalies through inconsistencies between nodes and their surrounding contexts. A major line adopts contrastive learning to compare nodes with local subgraphs, neighboring nodes, or augmented graph views, where abnormal nodes are expected to exhibit weaker contextual agreement than normal ones Liu et al. (2021b); Zheng et al. (2021); Jin et al. (2021); Duan et al. (2023). Subsequent studies investigate multi-scale contextual signals, neighborhood affinity, and simplified ego–neighbor matching to obtain more direct or efficient anomaly measures Qiao & Pang (2023); Pan et al. (2023). These methods avoid explicit anomaly labels, but their effectiveness is closely related to the assumed relation between abnormality and local inconsistency, which may vary substantially across graphs.

Supervised Graph Anomaly Detection. When anomaly labels are available, GNN-based detectors can directly learn discriminative representations, but they face several characteristics that distinguish GAD from conventional node classification. One major challenge is that anomalous nodes

are typically rare and may deliberately connect to normal nodes, leading to severe class imbalance, feature or relation camouflage, and unreliable neighborhood aggregation. Prior studies address these issues through consistency-aware neighbor selection, label-aware sampling, relation-specific aggregation, and anomaly-oriented data augmentation Liu et al. (2020); Dou et al. (2020b); Liu et al. (2021a; 2022a). Other works explicitly exploit both homophilic and heterophilic relations rather than treating all neighboring interactions uniformly, improving the ability of GNNs to preserve anomaly-specific signals under mixed local structures Shi et al. (2022).

Another line studies GAD from structural and spectral perspectives. Conventional message-passing GNNs favor smooth graph signals, whereas anomalous nodes can introduce stronger high-frequency components or exhibit substantially different local homophily from normal nodes. This observation has motivated adaptive multi-frequency filtering, band-pass graph filters, and spectrum-aware neighborhood modeling Chai et al. (2022); Tang et al. (2022); Gao et al. (2023b). Structural noise and heterophily have also been addressed through graph decomposition, learnable sparsification, and task-oriented graph structure optimization Gao et al. (2023a). These approaches introduce anomaly-specific structural inductive biases beyond generic GNN aggregation, but are commonly optimized for the graph on which anomaly labels are available.

E.2 Generalist Graph Anomaly Detection

Generalist graph anomaly detection (GGAD) aims to develop a unified detector that can generalize to unseen graphs without dataset-specific retraining. Early studies establish this setting through feature alignment and neighborhood-based anomaly modeling. ARC Liu et al. (2024; 2026a) introduces in-context learning for few-shot generalization, while UNPrompt Niu et al. (2024) learns unified neighborhood prompts for zero-shot anomaly detection. AnomalyGFM Qiao et al. (2025) instead learns graph-agnostic normal and abnormal prototypes from node representation residuals, supporting both zero-shot inference and few-shot prompt tuning. Subsequent methods focus on improving robustness to cross-domain feature and structural shifts through spectral normalization, invariant representation learning, residual modeling, multi-domain pattern learning, and test-time adaptation Zheng et al. (2025); Zhang et al. (2025); Fu et al. (2026); Zheng et al. (2026); Zhang et al. (2026). Recent studies further explore adaptive multi-hop contexts, topology-aware transfer, and heterogeneous geometric representations to accommodate diverse anomaly patterns across graphs Liu et al. (2026d); Yao et al. (2026); Zhao et al. (2026). Another emerging direction reduces the dependence on source-specific anomaly supervision by learning transferable normality, constructing relational or neighborhood statistics, or leveraging pretrained general-purpose models for anomaly inference Xu et al. (2026); Wei et al. (2026); Liu et al. (2026b;c). Despite rapid progress, existing GGAD methods differ substantially in their source data, target supervision, and evaluation protocols, making their cross-graph generalization capabilities difficult to compare directly.

E.3 Graph Anomaly Detection Benchmarks

Benchmarking studies have played an important role in standardizing the evaluation of graph anomaly detection. BOND Liu et al. (2022b) establishes a comprehensive benchmark for unsupervised outlier node detection on static attributed graphs, evaluating 14 methods on nine datasets with both synthetic and organic anomalies, together with efficiency analyses. GADBench Tang et al. (2023) extends the scope to supervised node-level GAD by benchmarking 29 classical, GNN-based, and tree-ensemble models on ten real-world graphs under semi-supervised and fully supervised settings. More recent studies examine additional evaluation dimensions. Beyond static node-level detection, UB-GOLD Wang et al. (2025b) unifies unsupervised graph-level anomaly detection and graph-level out-of-distribution detection across 35 datasets, while other research extend benchmarking to temporal graphs Poštuvan et al. (2024); Hua et al. (2026).

Despite these efforts, existing benchmarks primarily evaluate dataset-specific anomaly detection under a fixed graph or task setting. They do not systematically study the generalist setting in which anomaly detectors are learned from one or multiple source graphs and subsequently deployed on previously unseen target graphs. Consequently, the rapidly growing family of GGAD methods has been evaluated under heterogeneous choices of source graphs, target graphs, supervision, and transfer protocols, making results across studies difficult to compare directly. Our benchmark complements prior GAD benchmarks by providing a unified evaluation of cross-graph generalization and by

jointly examining specialized GGAD methods, conventional GAD baselines, and general-purpose graph foundation models.

F LIMITATIONS

GGADBench focuses on node-level anomaly detection over attributed graphs and therefore does not cover other anomaly granularities, such as anomalous edges, subgraphs, or entire graphs. Although we evaluate 22 graphs with both organic and injected anomalies and consider multiple source–target settings, these datasets and transfer configurations cannot represent all real-world graph domains or anomaly patterns. In addition, the reported efficiency results are measured under a fixed hardware environment, and several methods rely on reimplementations or method-specific preprocessing, so absolute computational costs may vary across implementations and platforms.