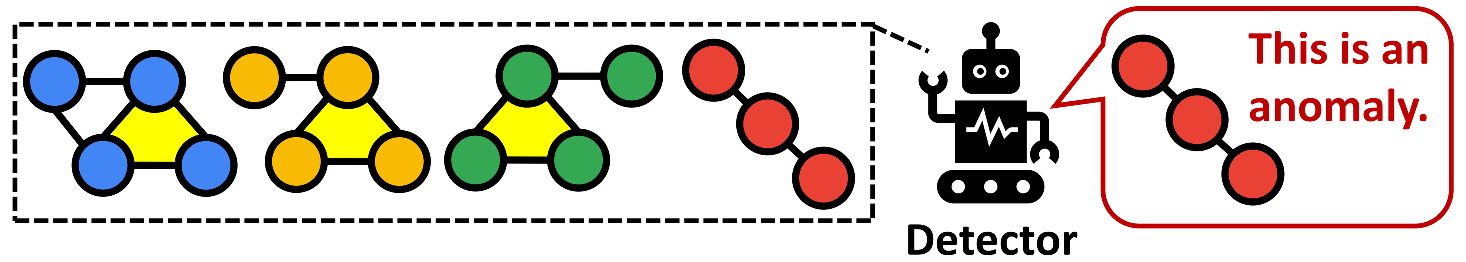
Rethinking Reconstruction-based Graph-Level Anomaly Detection: Limitations and a Simple Remedy

Sunwoo Kim — KAIST AI — kswoo97@kaist.ac.kr
Soo Yong Lee — KAIST AI — syleetolow@kaist.ac.kr
Fanchen Bu — KAIST EE — boqvezen97@kaist.ac.kr
Shinhwan Kang — KAIST AI — Shinhwan.kang@kaist.ac.kr
Kyungho Kim — KAIST AI — kkyungho@kaist.ac.kr
Jaemin Yoo — KAIST AI & EE — jaemin@kaist.ac.kr
Kijung Shin — KAIST AI & EE — kijungs@kaist.ac.kr

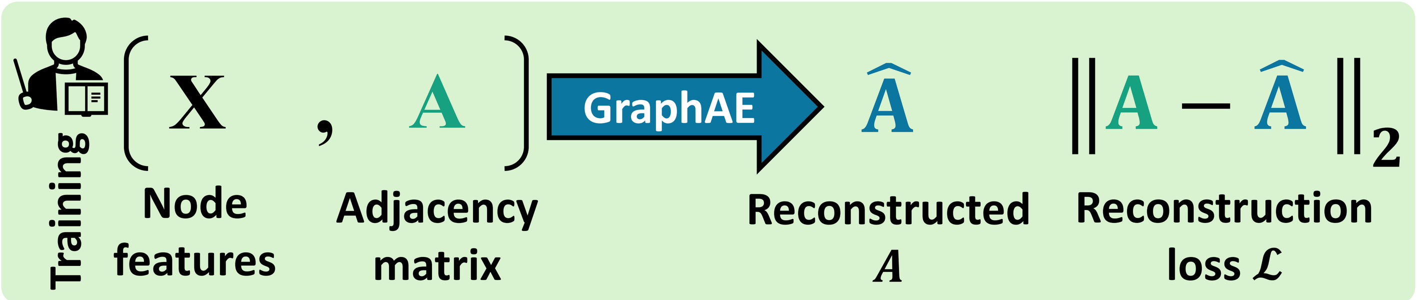
NEURAL INFORMATION PROCESSING SYSTEMS

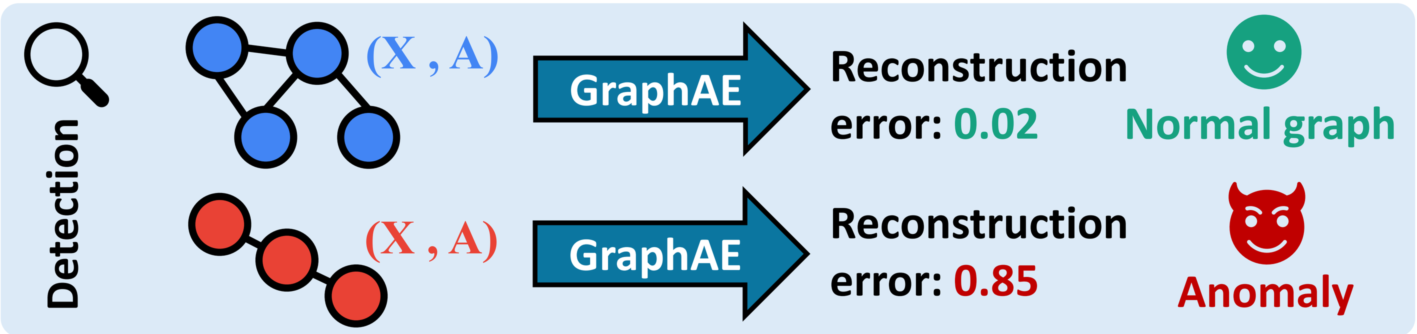
Preliminaries

- **[Main task]** Graph-level anomaly detection **(GLAD)**.
  - **[Goal]** Identifying anomalous **graphs** (graph-level task).
  - **[Application]** Brain diagnosis, drug discovery, to name a few.



- **[Representative GLAD method]** Graph autoencoders (GraphAEs).
  - **[Training]** Reconstructing the given graph's topology.
  - **[Detection]** High mean reconstruction error → Anomaly.



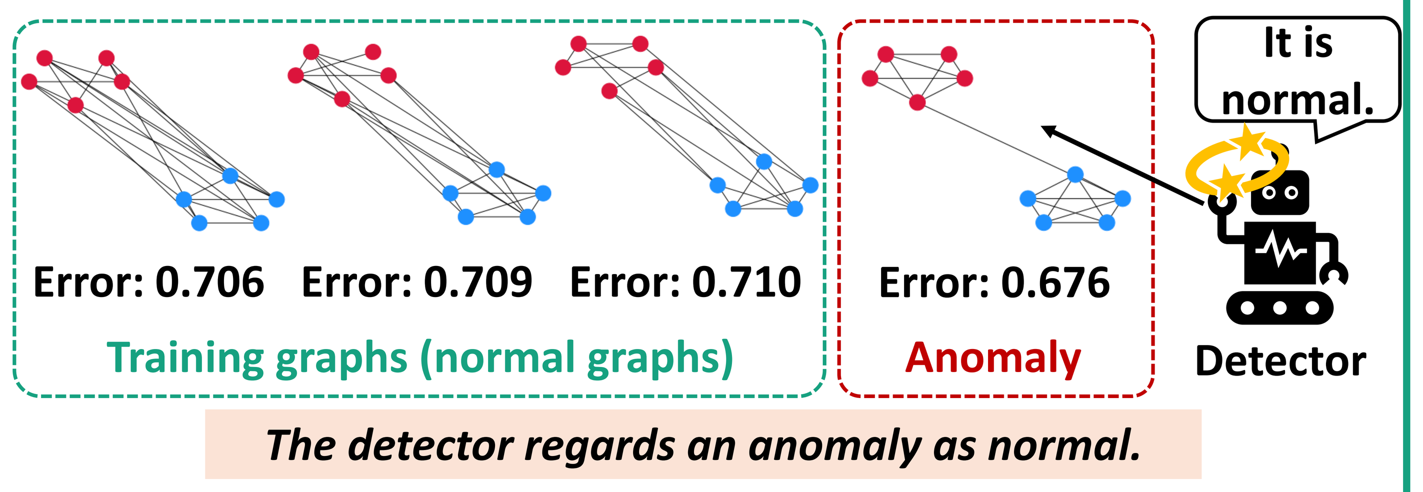


Tell me more details! Let's move on!

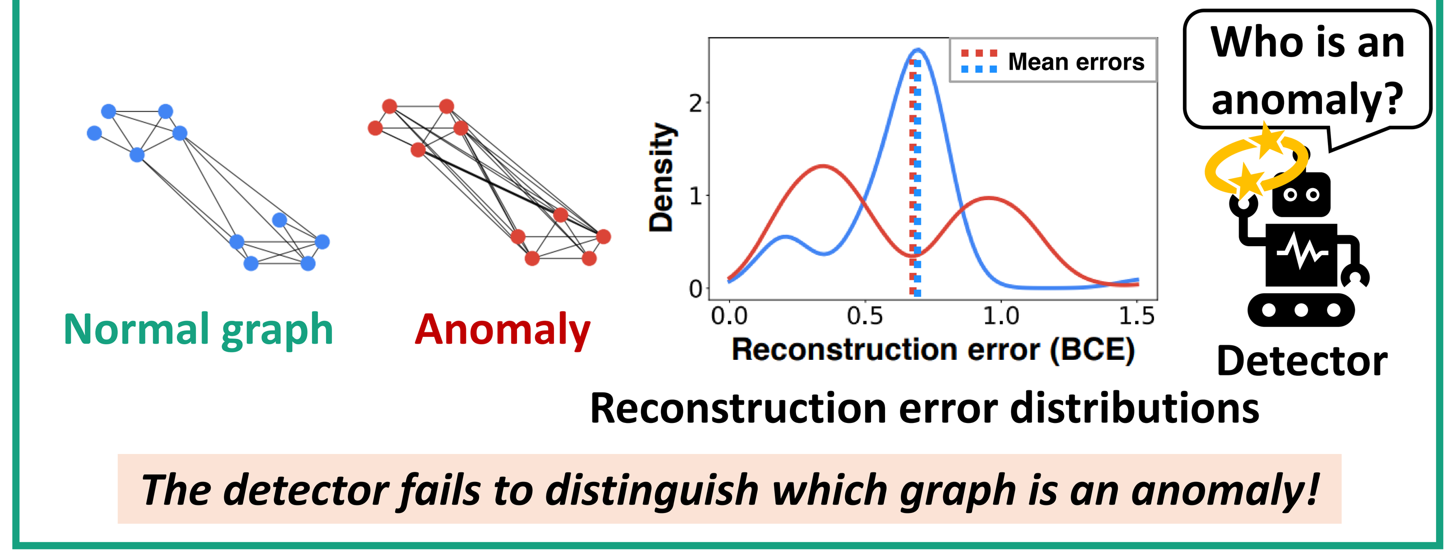
- **[What is GLAD?]** GLAD aims to find graphs with anomalous node features and/or topology compared to most graphs in the population.
  - **[Note]** Real-world graphs exhibit much diverse patterns, and GLAD primarily aims to detect anomalies in such general scenarios.
- **[What is an adjacency matrix?]** An adjacency matrix $\mathbf{A} \in \{0,1\}^{n \times n}$, where n denotes the number of nodes, represents the edge connection of a graph. If nodes i and j are linked by an edge, then $\mathbf{A}_{ij} = 1$ holds; otherwise, $\mathbf{A}_{ij} = 0$.
- **[How does GraphAE reconstruct A?]** Typically, GraphAE first uses a graph neural network to generate node embeddings $\mathbf{Z}$ (i.e., $\mathbf{Z} = \text{GNN}(\mathbf{X}, \mathbf{A})$). The reconstructed adjacency matrix $\hat{\mathbf{A}}$ is then computed as $\hat{\mathbf{A}} = \sigma(\mathbf{A}\mathbf{A}^{\mathrm{T}})$, where σ is the sigmoid function.
- **[What is mean reconstruction error?]** The mean reconstruction error is the mean of errors from all node pairs. For a graph $\mathcal{G} = (\mathbf{X}, \mathbf{A})$, its mean reconstruction error is equal to $(\mathrm{A}_{ij} - \hat{\mathrm{A}}_{ij})^2/n^2$.

Contribution 1: Limitations of GraphAEs in GLAD

- **[Limitation 1]** GraphAEs may better reconstruct an anomaly.
  - **[Detail]** Anomaly mean error < Normal graph mean error.



- **[Limitation 2]** Mean error alone may not distinguish distinct graphs.
  - **[Detail]** Anomaly mean error ≈ Normal graph mean error

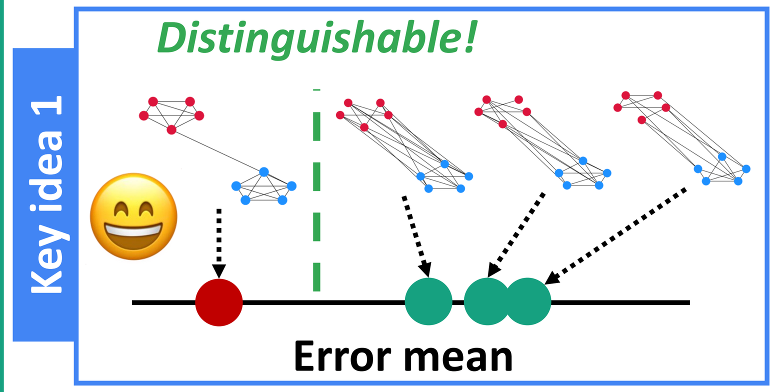


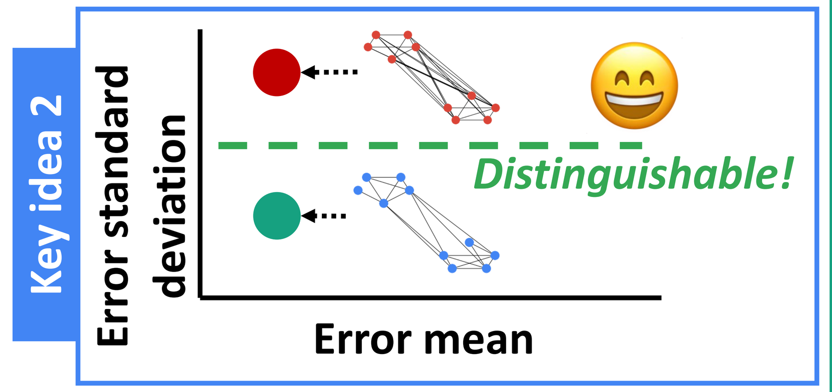
Tell me more details! Let's move on!

- **[What happens in Limitation 1?]** Our first observation is that an unseen graph may have a lower mean error than the training graphs.
  - **[Implication in GLAD]** This implies that an anomaly can have a lower mean error than the normal graphs, making GraphAE-based GLAD methods fail to detect them.
  - **[Why this happens?]** One of the reasons is that an anomaly shares the same graph pattern with normal graphs, but with stronger strength. Detailed theoretical and empirical analysis is provided in the main paper.
- **[What happens in Limitation 2?]** Our second observation is that two graphs with distinct graph structures may exhibit similar mean reconstruction errors.
  - **[Implication in GLAD]** This implies that an anomaly can have a mean error similar to that of normal graphs, again making GraphAE-based GLAD methods fail to detect them.

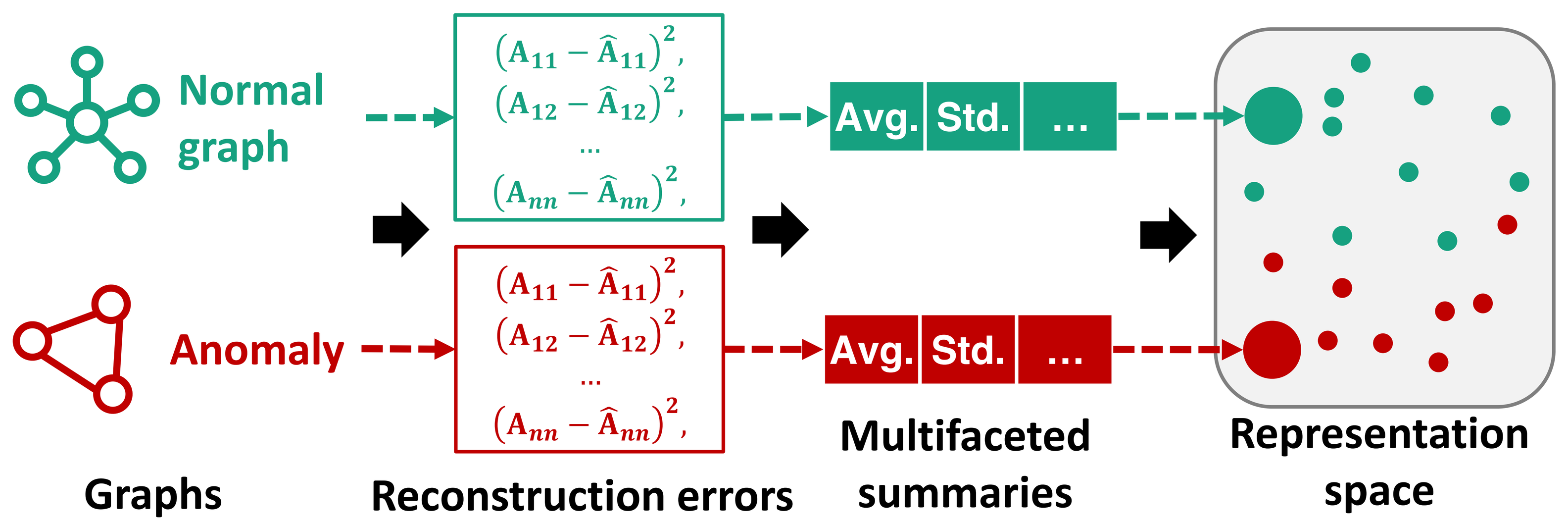
Contribution 2: MUSE, our new GLAD method

- **[Key idea]** We propose a new GLAD method, **MUSE** (**MU**ltifaceted **S**ummaries of Reconstruction **E**rrors), mitigating Limitations 1 and 2.
  - **[K1]** Use reconstruction errors as graph features (remedy of Limitation 1).
  - **[K2]** Use multifaceted (various) summary statistics, not just mean alone (remedy of Limitation 2).





- **[MUSE Overview]**



Tell me more details! Let's move on!

- **[What are multifaceted summaries?]** They are summary statistics that capture diverse aspects of error distributions, such as the mean, and standard deviation combined.
- **[How does MUSE obtain error distributions?]** MUSE first obtains node embeddings using a GIN [1] encoder. It then reconstructs the adjacency matrix and computes the error for each entry in the matrix using binary cross-entropy.
- **[How does MUSE get multifaceted summaries?]** MUSE summarizes the reconstruction error values by calculating the mean and standard deviation of the values.
- **[How does MUSE finally detect anomalies?]** After obtaining multifaceted summaries, which serve as graph features, MUSE applies a one-class classifier—an MLP-based autoencoder in this case—to these features. Graphs with a high mean reconstruction error from the MLP autoencoder are classified as anomalies.

Experiments

- **[Datasets]** Graph classification benchmark datasets, where one class is designated as normal graphs, and all other classes are treated as anomalous graphs.
- **[Baselines]** State-of-the-art GLAD methods [2, 3] and graph self-supervised learning methods.
- **[Result 1]** MUSE is accurate.

Table 1: **GLAD performance**: Mean and standard deviation of test AUROC values (×100) in the GLAD task are reported. The best and second-best performances are highlighted in green and yellow. A.R. denotes average ranking. MUSE obtains the best average ranking among 18 methods.

	Method	DD	Protein	NCI1	AIDS	Reddit	IMDB	MUTAG	DHFR	BZR	ER	AR
GLAD methods	DOMINANT-G [6]	64.3 (4.4)	55.9 (9.7)	65.5 (6.1)	80.6 (4.0)	58.6 (5.3)	60.8 (6.7)	65.0 (4.2)	56.6 (9.2)	76.2 (7.8)	58.7 (5.5)	10.7
	OCGTL [39]	74.5 (5.1)	71.0 (8.7)	61.2 (5.5)	95.3 (3.7)	69.0 (4.0)	65.8 (5.8)	64.9 (4.9)	66.5 (9.9)	71.3 (17.1)	63.0 (3.6)	6.9
	GLocalKD [34]	47.8 (8.5)	50.7 (8.5)	51.6 (5.6)	51.2 (1.2)	49.8 (4.2)	58.5 (6.7)	55.1 (4.4)	54.1 (8.1)	55.8 (16.7)	54.4 (4.4)	17.0
	GLADC [33]	52.1 (5.2)	50.7 (5.6)	51.4 (3.6)	51.4 (1.0)	52.2 (2.6)	57.7 (5.2)	53.3 (4.5)	55.8 (4.1)	59.0 (14.5)	52.8 (4.2)	16.8
	GLAM [57]	61.6 (5.2)	60.3 (5.6)	58.1 (1.9)	93.6 (2.6)	75.6 (4.0)	65.1 (3.5)	63.0 (2.0)	57.2 (2.7)	72.6 (8.9)	55.2 (2.9)	9.8
	HIMNET [38]	52.1 (3.7)	56.9 (5.8)	53.6 (4.6)	64.3 (3.2)	65.7 (2.4)	61.8 (4.3)	57.5 (2.9)	63.6 (6.7)	72.0 (9.9)	55.7 (2.8)	12.3
	SIGNET [32]	64.2 (9.3)	56.4 (6.4)	63.1 (4.0)	97.2 (1.6)	78.0 (4.4)	48.2 (4.8)	67.5 (1.6)	40.2 (5.8)	66.6 (9.5)	56.2 (4.3)	10.4
SSL-based	GraphCL-1 [53]	64.5 (3.9)	60.7 (4.2)	55.8 (3.1)	71.2 (6.6)	57.7 (5.5)	54.2 (6.2)	53.6 (2.3)	57.8 (6.7)	60.5 (9.3)	55.5 (4.1)	14.2
	GAE-1 [22]	64.7 (5.2)	61.3 (7.0)	62.5 (2.2)	86.2 (1.4)	74.8 (3.2)	63.8 (7.4)	63.2 (3.3)	56.5 (9.6)	68.5 (13.7)	60.0 (3.9)	10.3
	GraphMAE-1 [15]	56.7 (7.3)	60.5 (4.9)	53.4 (3.2)	91.8 (5.3)	72.7 (3.2)	67.0 (5.0)	62.3 (2.6)	62.2 (9.6)	70.1 (7.6)	52.2 (3.6)	10.6
	GraphCL-2 [53]	66.1 (3.0)	59.1 (5.2)	60.3 (4.4)	91.8 (3.5)	77.3 (4.1)	66.3 (5.6)	67.4 (3.3)	59.1 (4.6)	71.9 10.4	67.3 (3.4)	7.2
	GAE-2 [22]	67.2 (3.4)	62.3 (5.0)	62.4 (3.9)	85.8 (1.6)	75.3 (5.7)	66.6 (7.6)	67.3 (3.3)	60.8 (5.6)	72.0 (8.8)	65.7 (2.0)	7.0
	GraphMAE-2 [15]	68.0 (4.3)	61.2 (4.0)	68.3 (3.6)	90.8 (3.6)	75.8 (4.8)	66.7 (5.8)	68.1 (2.4)	61.4 (6.0)	72.8 (6.4)	66.2 (6.4)	5.1
Variants	MUSE w/o $\mathbb{L}_{\mathbf{X}}$	79.4 (3.7)	75.6 (3.7)	69.2 (3.7)	99.6 (0.5)	72.2 (4.0)	65.8 (5.7)	65.8 (3.1)	60.4 (6.6)	65.6 (19.4)	66.3 (3.6)	5.8
	MUSE w/o $\mathbb{L}_{\mathbf{A}}$	61.8 (7.6)	64.7 (7.1)	63.1 (3.3)	89.3 (2.8)	72.0 (4.8)	56.9 (7.1)	57.0 (3.5)	58.1 (3.1)	68.7 (14.2)	60.7 (4.0)	11.0
	MUSE w/o AVG	78.6 (4.0)	68.1 (5.5)	68.0 (2.0)	95.0 (2.6)	73.2 (6.6)	66.2 (6.5)	60.9 (3.9)	60.1 (2.4)	66.3 (13.0)	62.0 (3.5)	7.7
	MUSE w/o STD	74.3 (5.4)	74.4 (5.2)	65.2 (3.6)	98.7 (0.5)	70.5 (4.3)	70.7 (3.7)	62.0 (2.4)	62.9 (6.4)	71.3 (11.5)	66.7 (2.4)	5.6
	MUSE	80.5 (2.3)	78.4 (2.2)	71.1 (2.0)	99.7 (0.5)	78.4 (5.7)	69.2 (3.5)	67.5 (3.4)	63.8 (8.6)	69.5 (12.6)	67.9 (3.6)	2.2

- **[Result 2]** MUSE is robust against training set contamination.

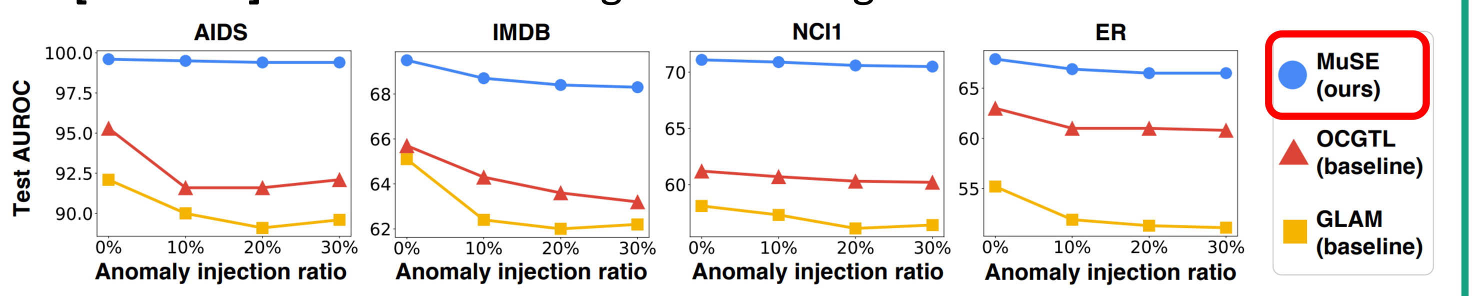
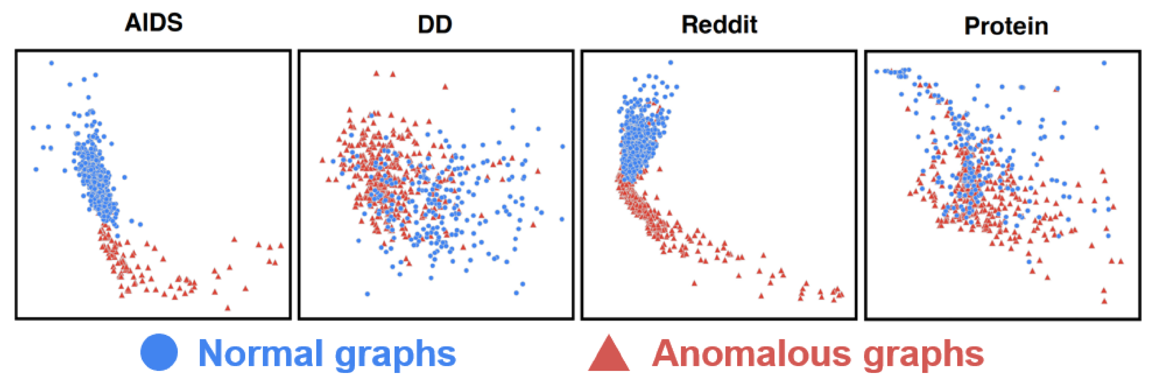


Figure 6: Comparison of the three strongest GLAD methods' robustness against training set contamination. MUSE undergoes the least performance drop among the three methods.

- **[Result 3]** MUSE well separates anomalies from normal graphs.

PCA Visualization



References

[1] Xu et al., How Powerful Are Graph Neural Networks, ICLR 2019.
[2] Qiu et al., Raising the Bar in Graph-level Anomaly Detection, IJCAI 2022.
[3] Zhao et al., Graph Anomaly Detection with Unsupervised GNNs, ICDM 2022.