NEURAL INFORMATION PROCESSING SYSTEMS

KAIST AI
Kim Jaechul Graduate School

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

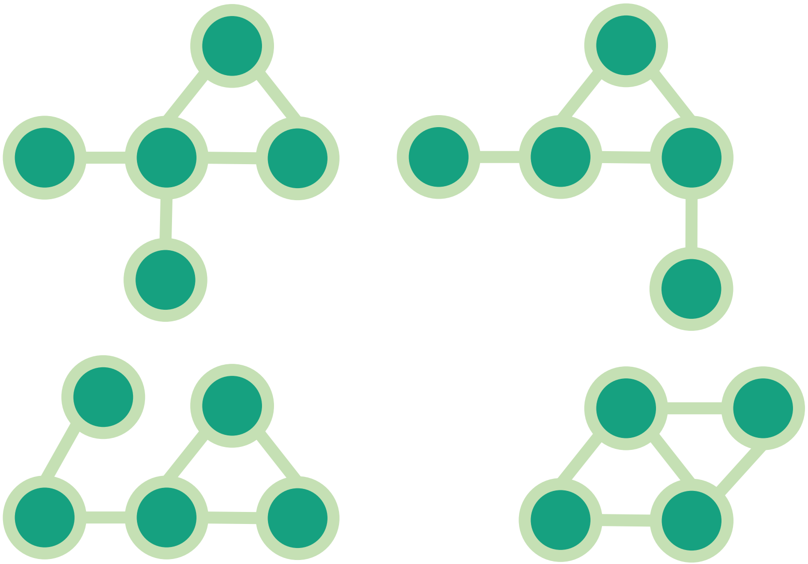
Sunwoo Kim   Soo Yong Lee   Fanchen Bu   Shinhwan Kang   Kyungho Kim   Jaemin Yoo   Kijung Shin

Roadmap

- **Overview**
- Limitations of Graph-AEs and Our Method
- Conclusions

About Graph-Level Anomaly Detection (GLAD)

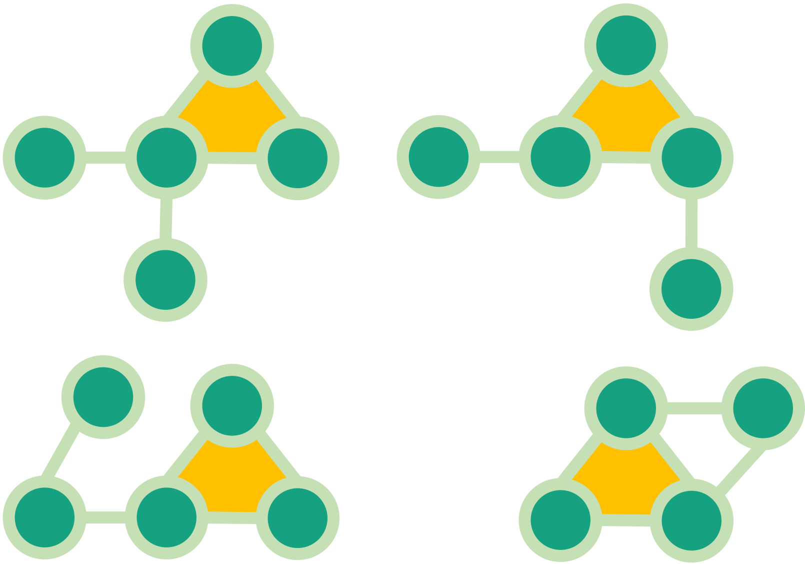
- **GLAD** aims to find graphs with anomalous node features and/or topology compared to most in the population.



Graph population

About **G**raph-**L**evel **A**nomaly **D**etection (cont.)

- **GLAD** aims to find graphs with anomalous node features and/or topology compared to most in the population.

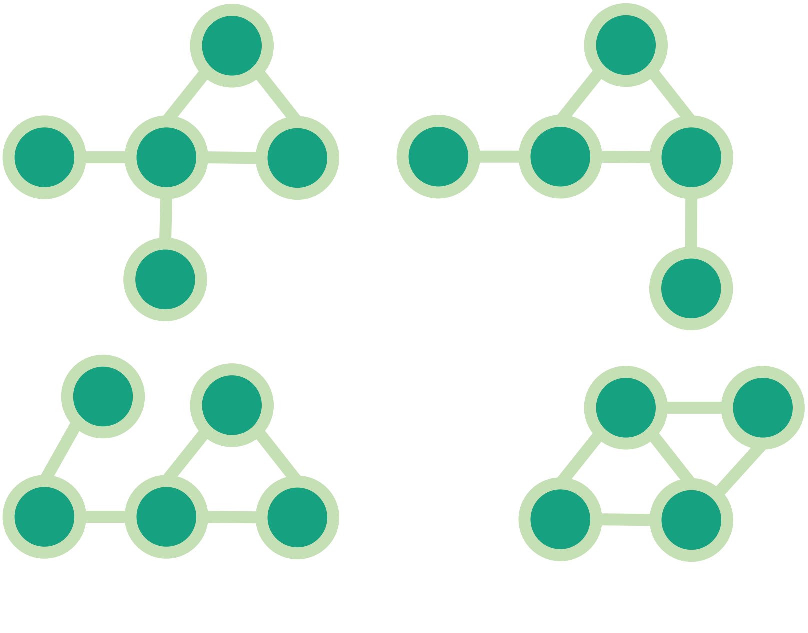


Graph population

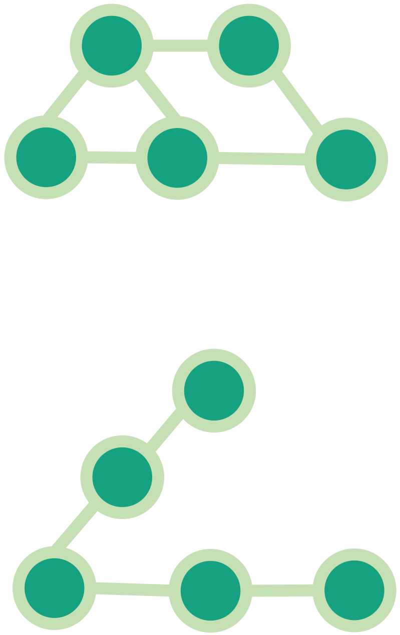
Each graph has a triangle.

About Graph-Level Anomaly Detection (cont.)

- **GLAD** aims to find graphs with anomalous node features and/or topology compared to most in the population.

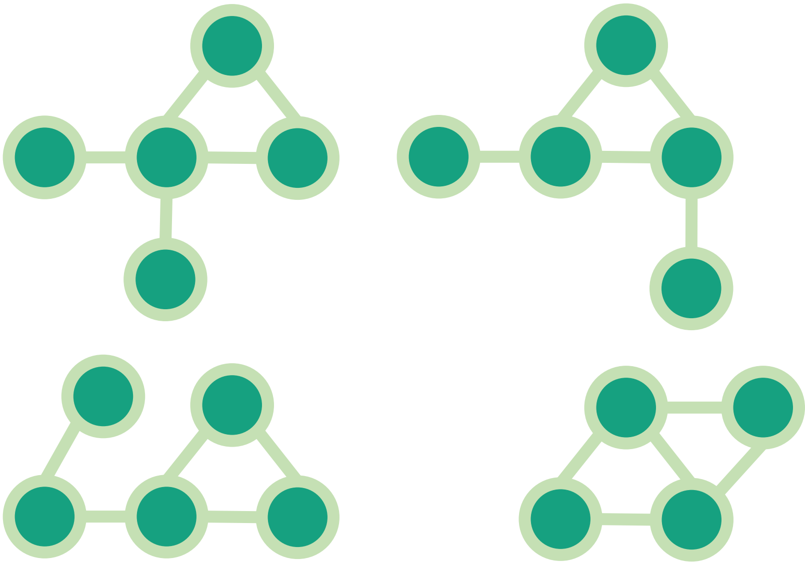


Graph population

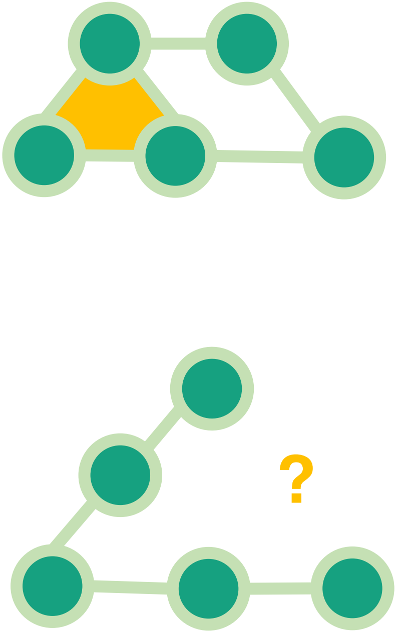


About Graph-Level Anomaly Detection (cont.)

- **GLAD** aims to find graphs with anomalous node features and/or topology compared to most in the population.



Graph population

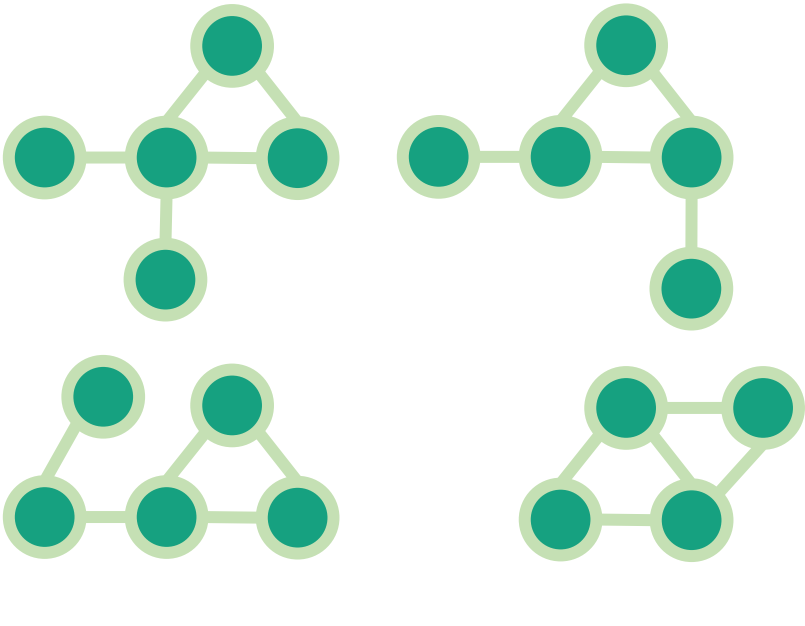


Triangle O

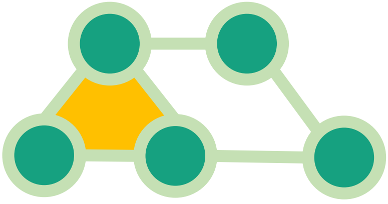
Triangle X

About Graph-Level Anomaly Detection (cont.)

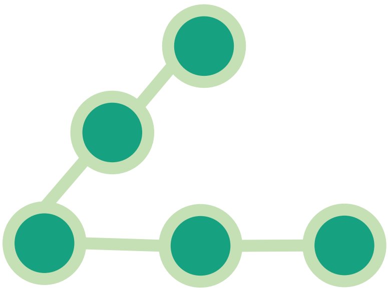
- **GLAD** aims to find graphs with anomalous node features and/or topology compared to most in the population.



Graph population



Normal



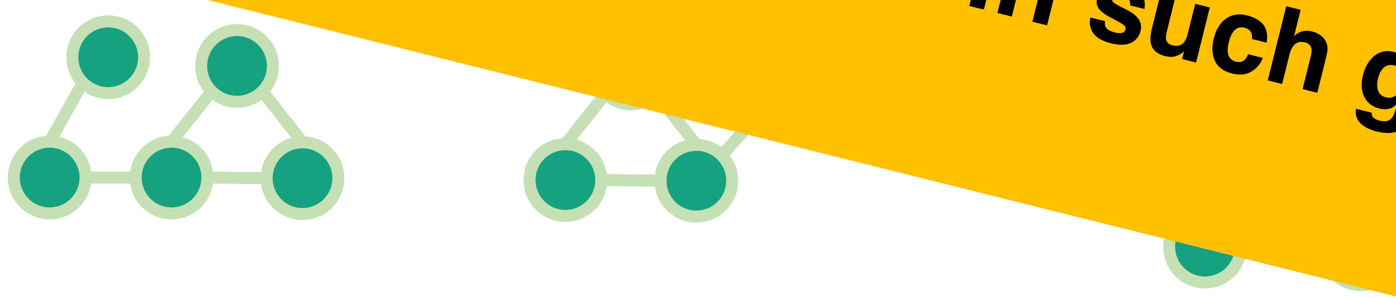
Anomaly

About Graph-Level Anomaly Detection (cont.)

- G … graphs with anomalous node features and/or

t … pulation.

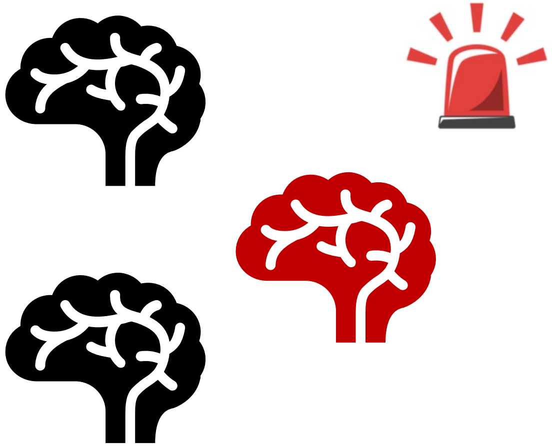
Real-world graphs exhibit diverse patterns, and GLAD primarily aims to detect anomalies in such general cases.



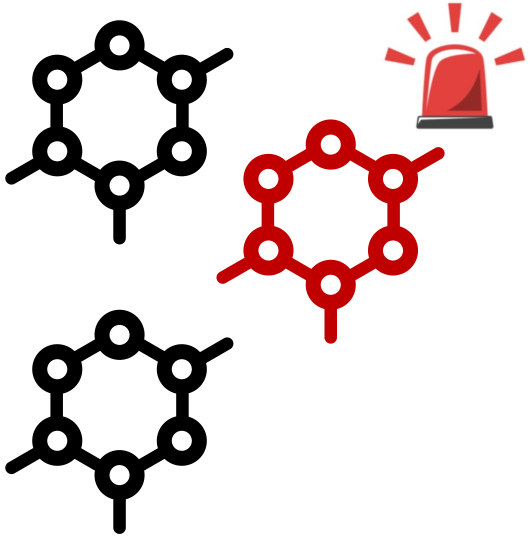
Graph population

About Graph-Level Anomaly Detection (cont.)

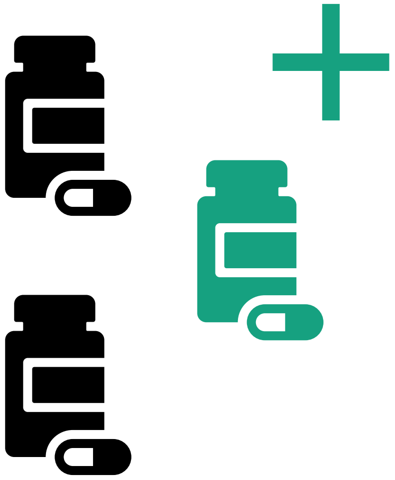
- **GLAD** have various applications.



Brain diagnosis



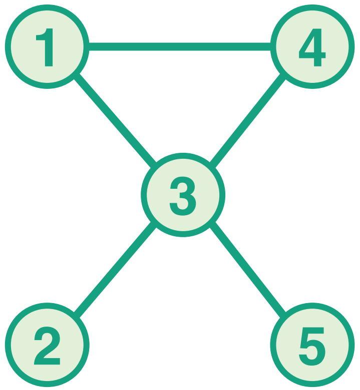
Toxic chemical detection



Drug discovery

GLAD Methods

- Many **GLAD** methods use graph auto-encoders (Graph-AEs).
  - They typically learn to **reconstruct** the given graph's topology.



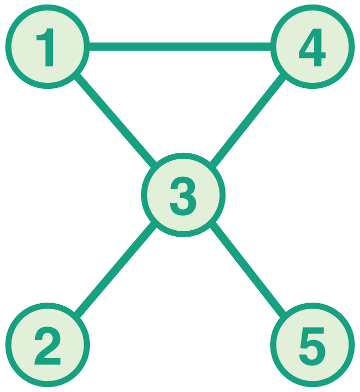
Graph topology

1	0	0
0	1	0
0	0	1
1	0	0
0	1	0

Node features

GLAD Methods (cont.)

- Many **GLAD** methods use graph auto-encoders (Graph-AEs).
  - They typically learn to **reconstruct** the given graph's topology.



Graph topology

1	0	0
0	1	0
0	0	1
1	0	0
0	1	0

Node features

$$\begin{array}{c|c} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

Adjacency matrix

GLAD Methods (cont.)

- Many **GLAD** methods use graph auto-encoders (Graph-AEs).
  - They typically learn to **reconstruct** the given graph's topology.

$$
\begin{array}{c|c}
 & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} &
\begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}
\end{array}
,
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}
\xrightarrow{\text{Graph-AE}}
\begin{array}{c|c}
 & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} &
\begin{pmatrix} 0.1 & 0.3 & 0.7 & 0.9 & 0.1 \\ 0.3 & 0.2 & 0.6 & 0.1 & 0.2 \\ 0.7 & 0.6 & 0.1 & 0.9 & 0.8 \\ 0.9 & 0.1 & 0.9 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.8 & 0.2 & 0.2 \end{pmatrix}
\end{array}
$$

Inputs of Graph-AE

Reconstructed adjacency matrix

GLAD Methods (cont.)

- Many **GLAD** methods use graph auto-encoders (Graph-AEs).
  - They typically learn to **reconstruct** the given graph's topology.

$$
\begin{array}{c c}
 & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} &
\begin{pmatrix}
0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0
\end{pmatrix}
\end{array}
,
\begin{array}{|c|c|c|}
\hline
1 & 0 & 0 \\ \hline
0 & 1 & 0 \\ \hline
0 & 0 & 1 \\ \hline
1 & 0 & 0 \\ \hline
0 & 1 & 0 \\ \hline
\end{array}
\xrightarrow{\text{Graph-AE}}
\begin{array}{c c}
 & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} &
\begin{pmatrix}
0.1 & 0.3 & 0.7 & 0.9 & 0.1 \\
0.3 & 0.2 & 0.6 & 0.1 & 0.2 \\
0.7 & 0.6 & 0.1 & 0.9 & 0.8 \\
0.9 & 0.1 & 0.9 & 0.2 & 0.2 \\
0.1 & 0.2 & 0.8 & 0.2 & 0.2
\end{pmatrix}
\end{array}
$$

Trained to make these two matrices similar!

GLAD Methods (cont.)

- Graph-AEs use **mean reconstruction error** as the anomaly score.
  - Typically, element-wise mean squared error is used.

$$\boxed{(0-0.1)^2}$$

$$\begin{pmatrix} \boxed{0} & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \qquad \begin{pmatrix} \boxed{0.1} & 0.3 & 0.7 & 0.9 & 0.1 \\ 0.3 & 0.2 & 0.6 & 0.1 & 0.2 \\ 0.7 & 0.6 & 0.1 & 0.9 & 0.8 \\ 0.9 & 0.1 & 0.9 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.8 & 0.2 & 0.2 \end{pmatrix}$$

Original adjacency matrix | **Reconstructed adjacency matrix**

GLAD Methods (cont.)

- Graph-AEs use **mean reconstruction error** as the anomaly score.
  - Typically, element-wise mean squared error is used.

$$\begin{pmatrix} 0 & \boxed{0} & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Original adjacency matrix

$$\begin{pmatrix} 0.1 & \boxed{0.3} & 0.7 & 0.9 & 0.1 \\ 0.3 & 0.2 & 0.6 & 0.1 & 0.2 \\ 0.7 & 0.6 & 0.1 & 0.9 & 0.8 \\ 0.9 & 0.1 & 0.9 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.8 & 0.2 & 0.2 \end{pmatrix}$$

Reconstructed adjacency matrix

$$(0 - 0.1)^2 + \boxed{(0 - 0.3)^2}$$

GLAD Methods (cont.)

- Graph-AEs use **mean reconstruction error** as the anomaly score.
  - Typically, element-wise mean squared error is used.

$$
\begin{pmatrix}
0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & \boxed{0}
\end{pmatrix}
\qquad
\begin{pmatrix}
0.1 & 0.3 & 0.7 & 0.9 & 0.1 \\
0.3 & 0.2 & 0.6 & 0.1 & 0.2 \\
0.7 & 0.6 & 0.1 & 0.9 & 0.8 \\
0.9 & 0.1 & 0.9 & 0.2 & 0.2 \\
0.1 & 0.2 & 0.8 & 0.2 & \boxed{0.2}
\end{pmatrix}
$$

Original adjacency matrix

Reconstructed adjacency matrix

$$(0-0.1)^2 + (0-0.3)^2 + \cdots + \boxed{(0-0.3)^2}$$

GLAD Methods (cont.)

- Graph-AEs use **mean reconstruction error** as the anomaly score.
  - Typically, element-wise mean squared error is used.

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Original adjacency matrix

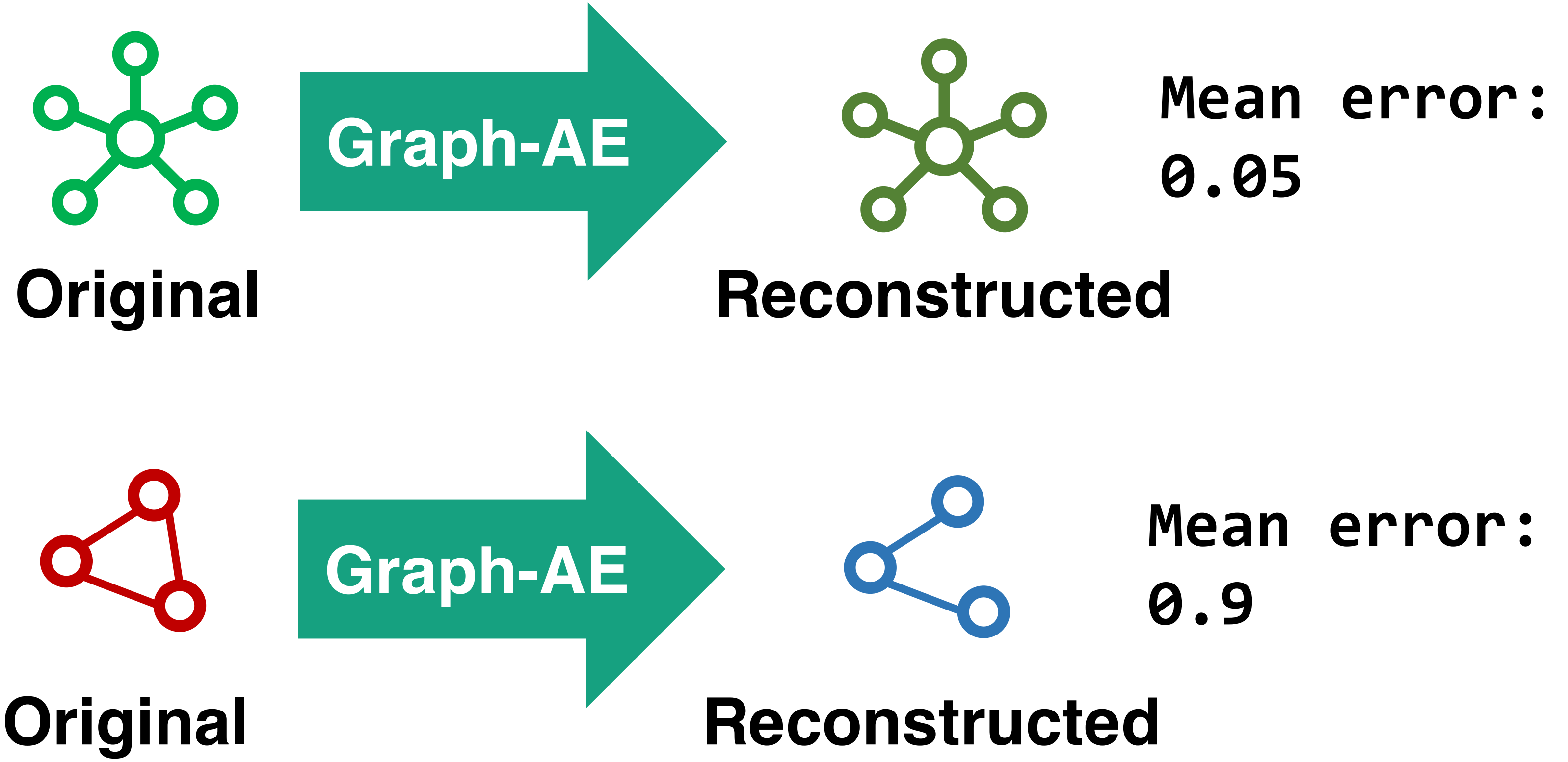
$$\begin{pmatrix} 0.1 & 0.3 & 0.7 & 0.9 & 0.1 \\ 0.3 & 0.2 & 0.6 & 0.1 & 0.2 \\ 0.7 & 0.6 & 0.1 & 0.9 & 0.8 \\ 0.9 & 0.1 & 0.9 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.8 & 0.2 & 0.2 \end{pmatrix}$$

Reconstructed adjacency matrix

$$\left.\boxed{\begin{matrix} (\mathbf{0} - \mathbf{0.1})^2 \\ + \\ (\mathbf{0} - \mathbf{0.3})^2 \\ + \\ \cdots \\ + \\ (\mathbf{0} - \mathbf{0.3})^2 \end{matrix}}\right\} \text{Mean} = \text{Anomaly score}$$

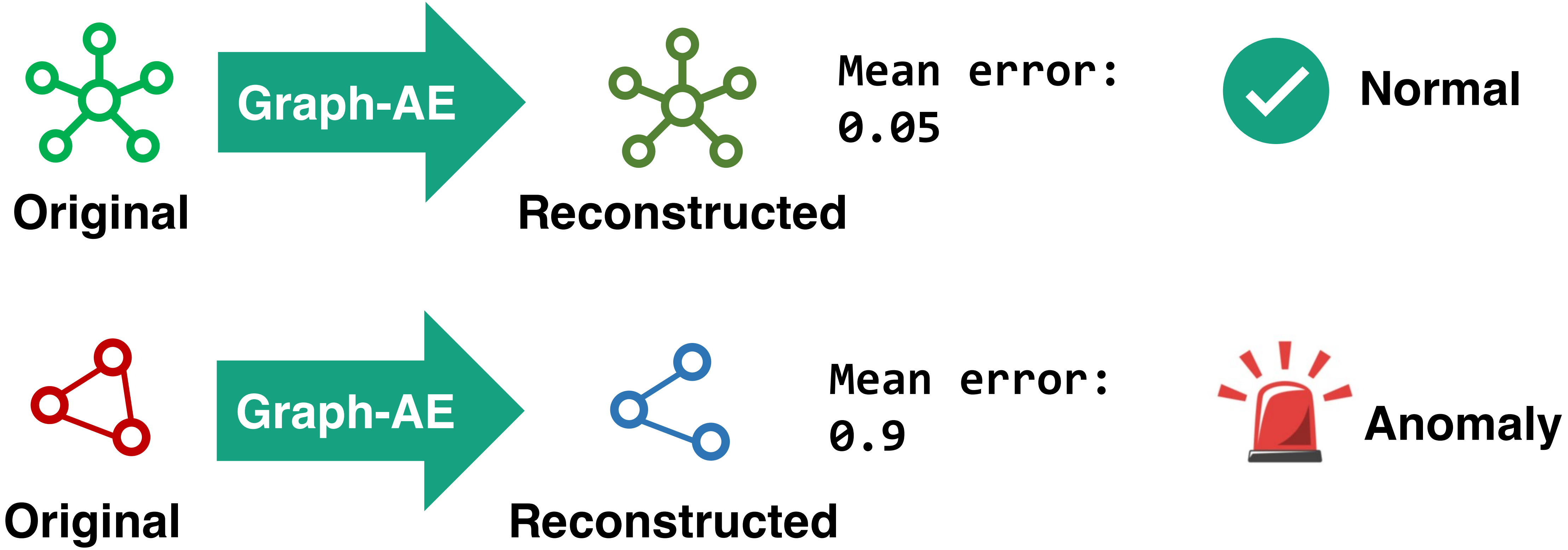
GLAD Methods (cont.)

- Graph-AEs use **mean reconstruction error** as the anomaly score.
  - High mean reconstruction errors indicate anomalous graphs.



GLAD Methods (cont.)

- Graph-AEs use **mean reconstruction error** as the anomaly score.
  - High mean reconstruction errors indicate anomalous graphs.

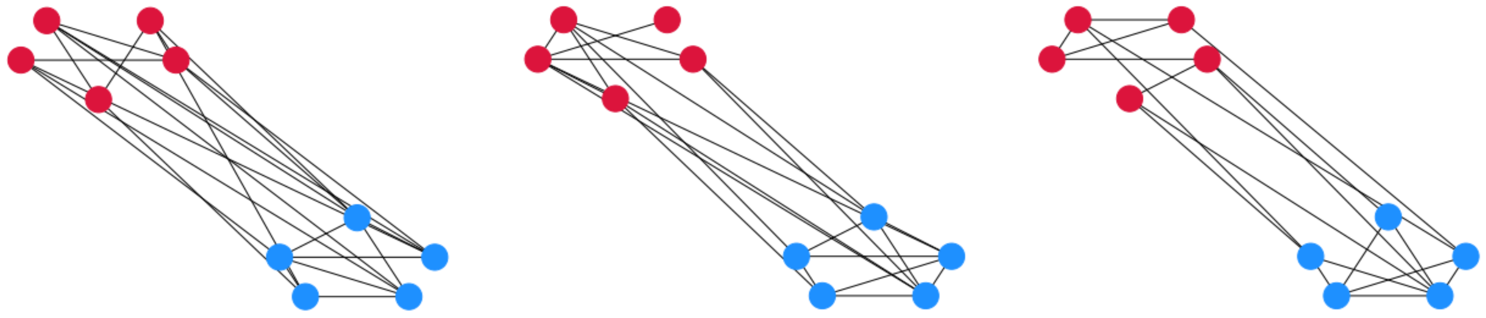


Roadmap

- Overview
- **Limitations of Graph-AEs and Our Method**
- Conclusions

Limitation 1

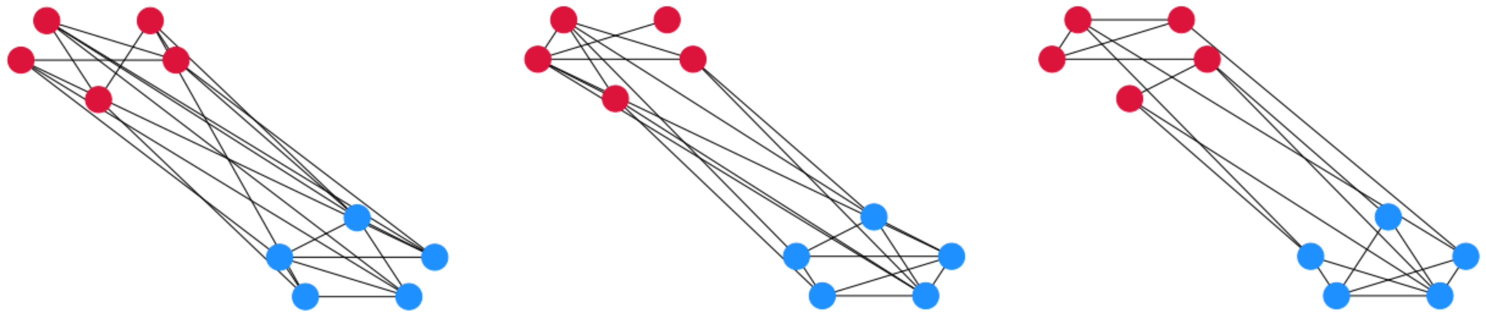
- Surprisingly, Graph-AEs can better reconstruct certain graphs that are dissimilar from the training graphs.



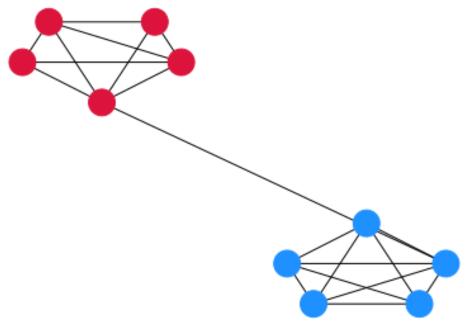
Training graphs

Limitation 1 (cont.)

- Surprisingly, Graph-AEs can better reconstruct certain graphs that are dissimilar from the training graphs.



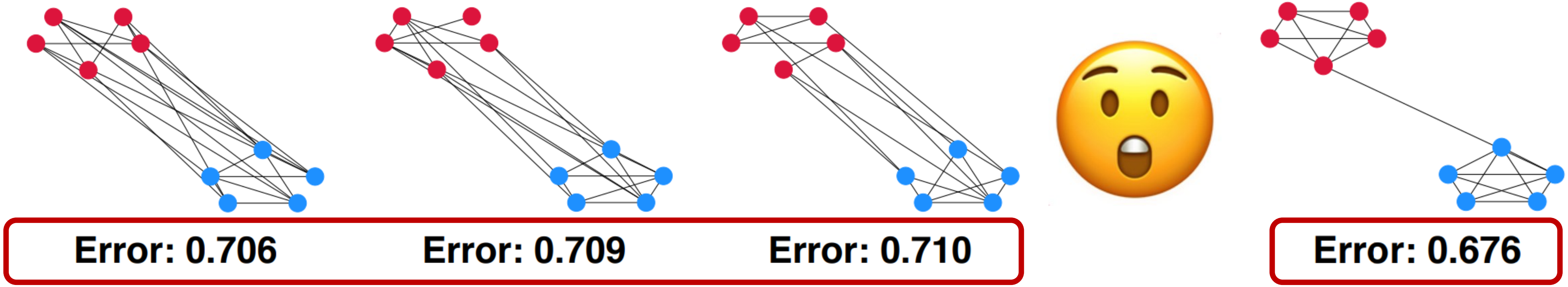
Training graphs



Unseen graph

Limitation 1 (cont.)

- Surprisingly, Graph-AEs can better reconstruct certain graphs that are dissimilar from the training graphs.



Training graphs **Unseen graph**

Limitation 1 (cont.)

- S AEs can better reconstruct certain graphs that are c

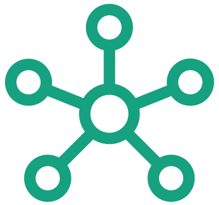
In our paper, we provide theoretical and empirical analysis of this phenomenon.

Error: 0.706 Error: 0.

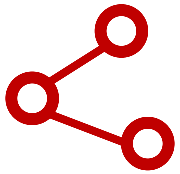
Training graphs ph

Limitation 1 (cont.)

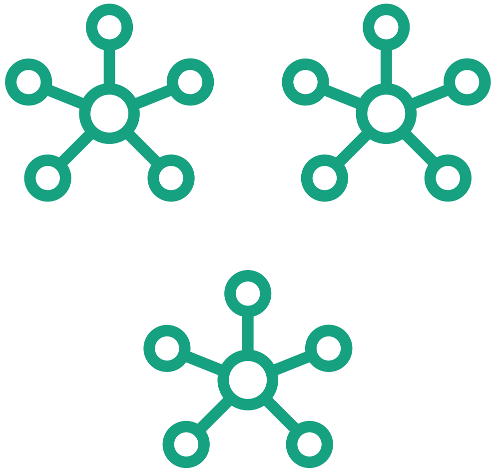
- In such cases, Graph-AEs cannot detect anomalies.



Normal graph



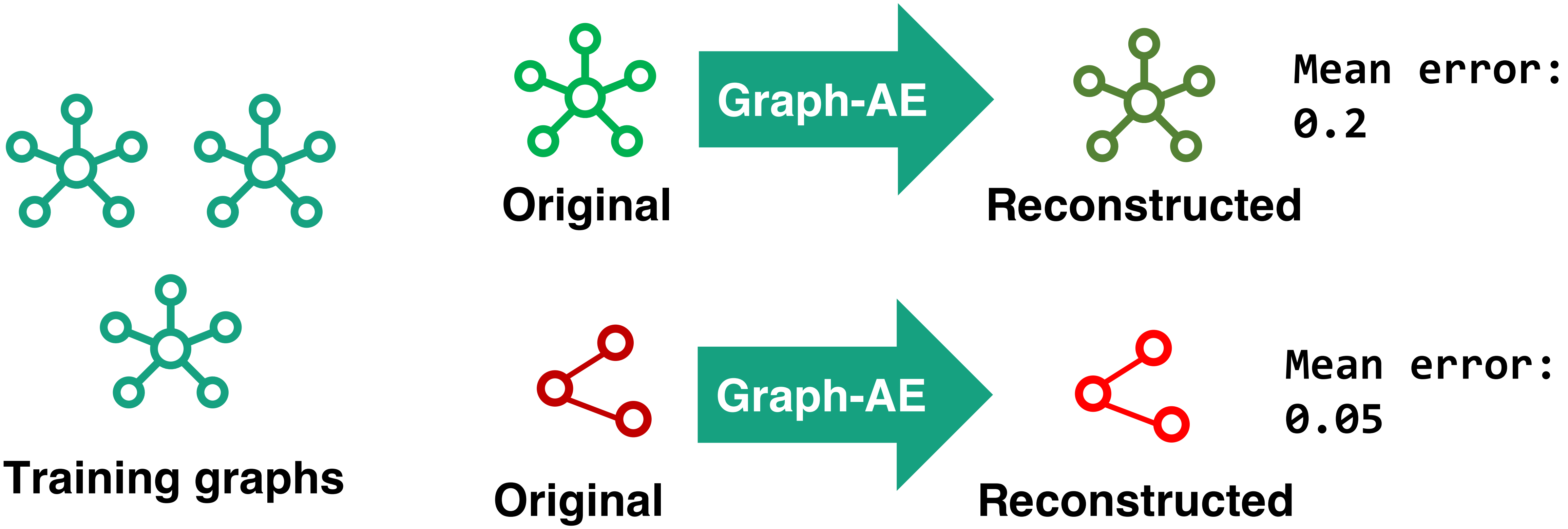
Anomalous graph



Training graphs

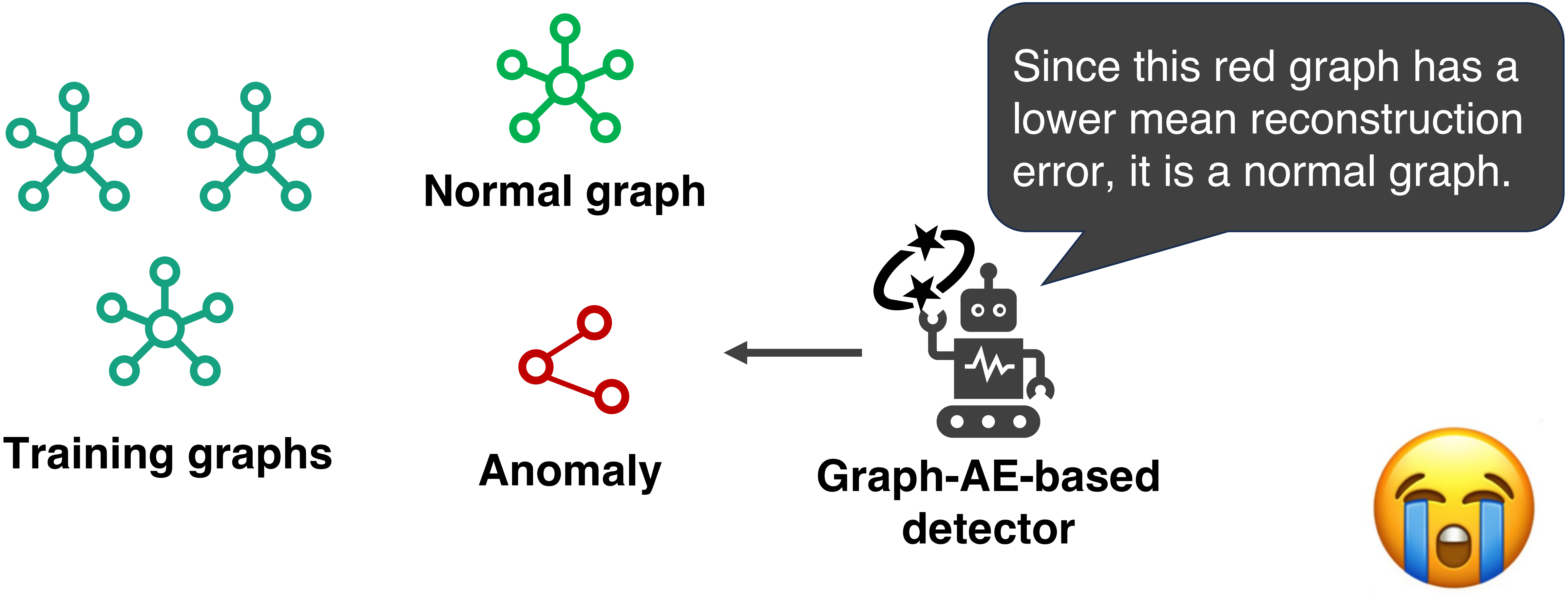
Limitation 1 (cont.)

- In such cases, Graph-AEs cannot detect anomalies.



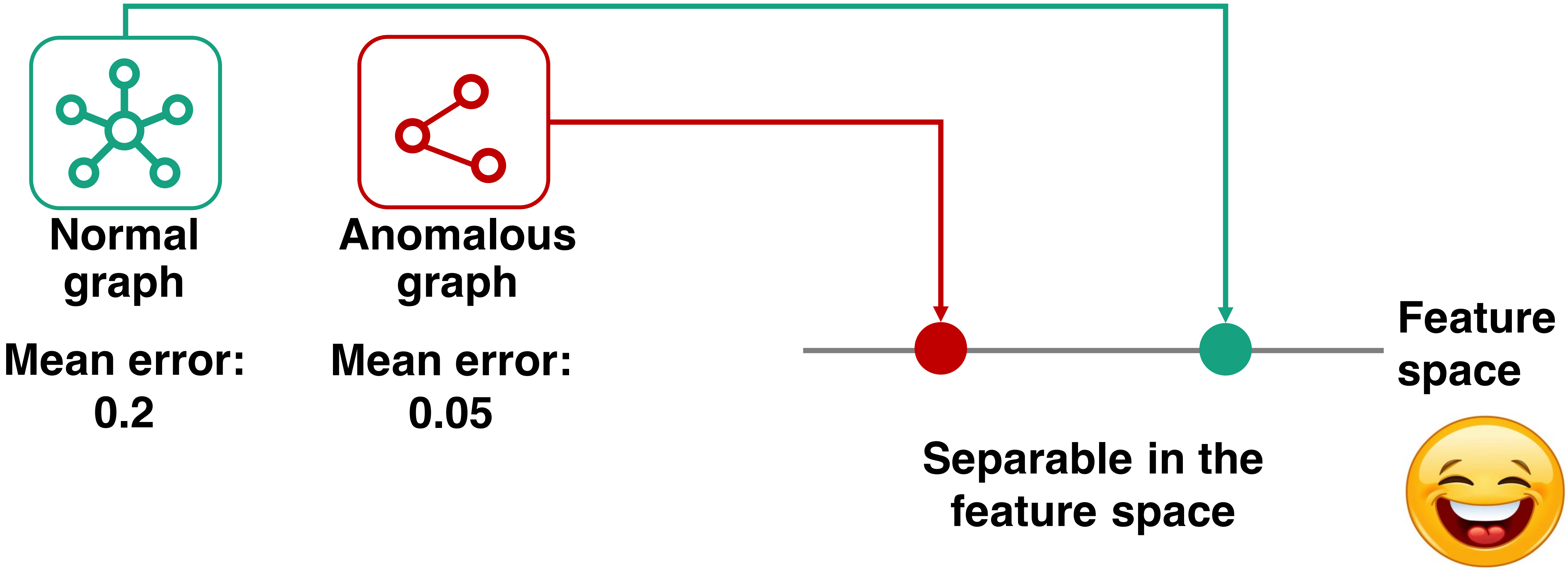
Limitation 1 (cont.)

- In such cases, Graph-AEs cannot detect anomalies.



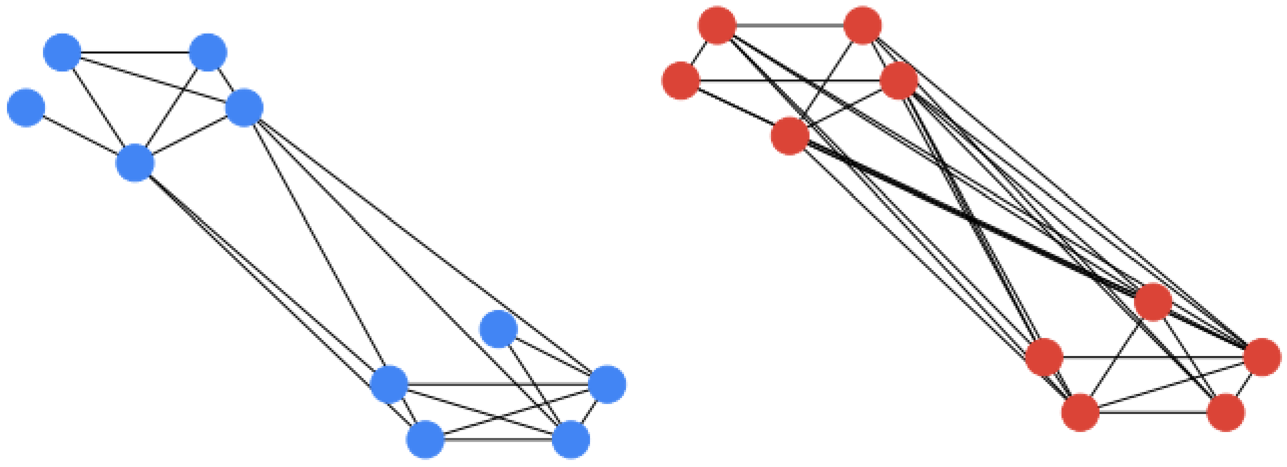
Our Remedy of Limitation 1

- We use reconstruction errors as **features** of a graph.



Limitation 2

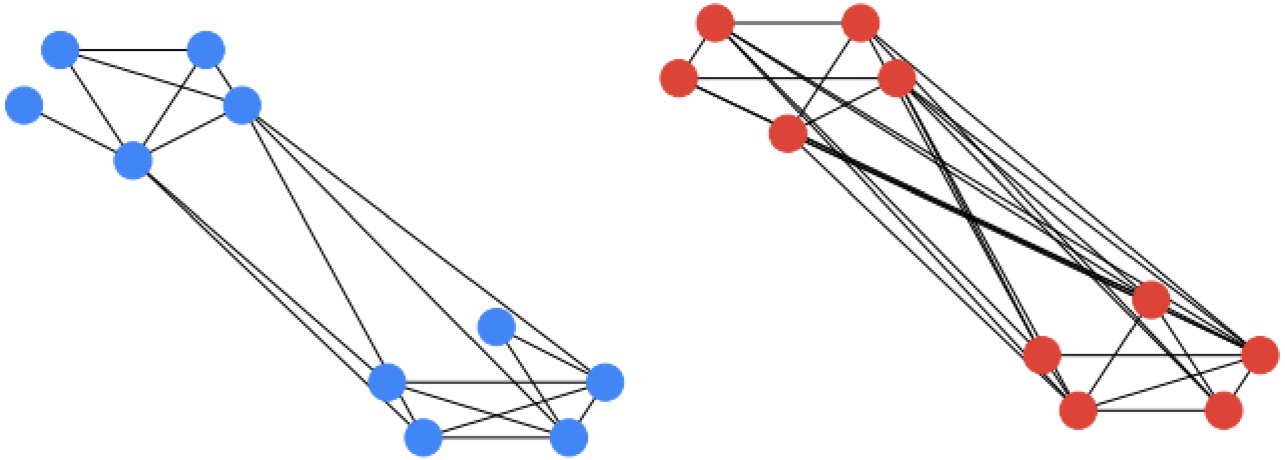
- Dissimilar graphs may have the similar mean reconstruction errors.



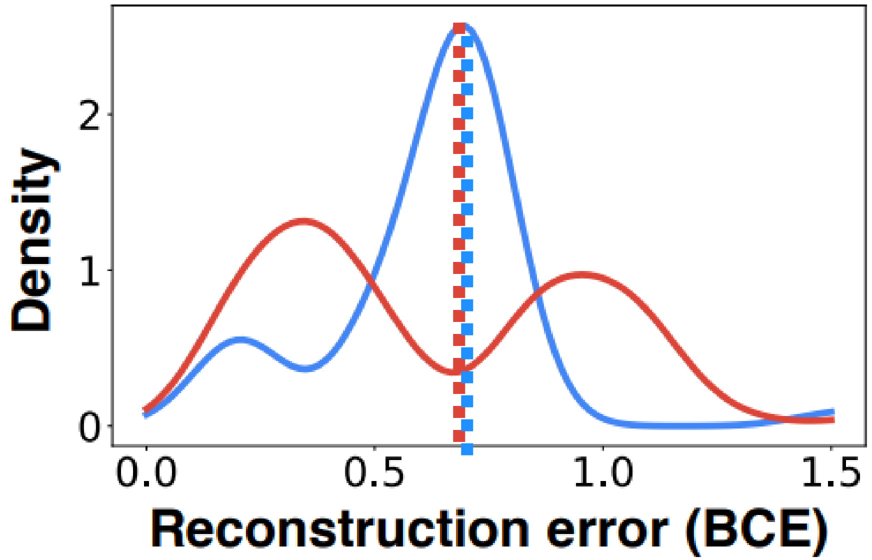
Two dissimilar graphs

Limitation 2 (cont.)

- Dissimilar graphs may have the similar mean reconstruction errors.



Two dissimilar graphs

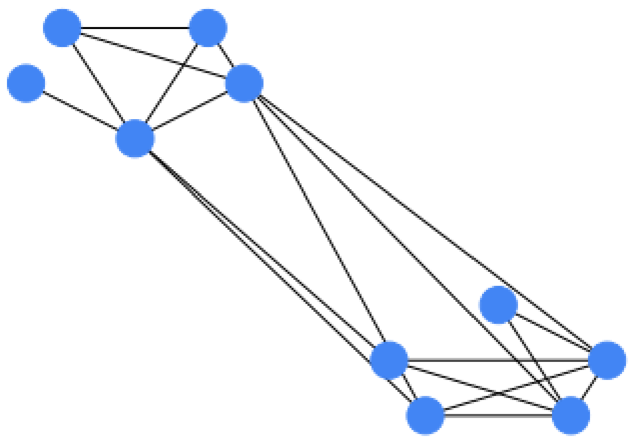


Reconstruction error distributions

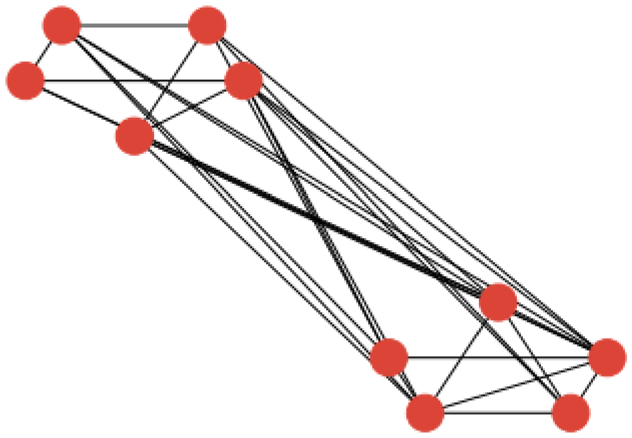
* **Mean reconstruction errors**

Limitation 2 (cont.)

- In such cases, Graph-AEs cannot detect anomalies.



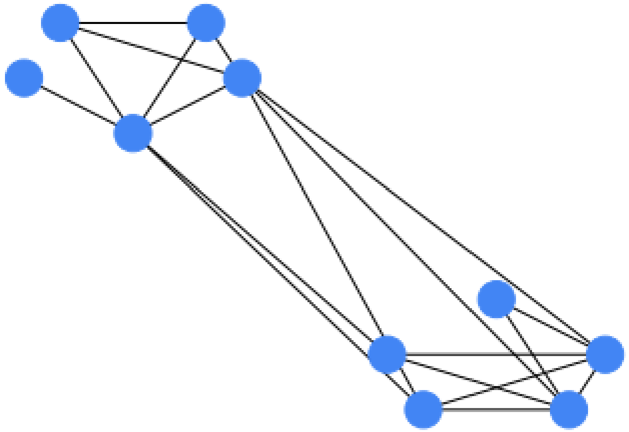
Normal graph



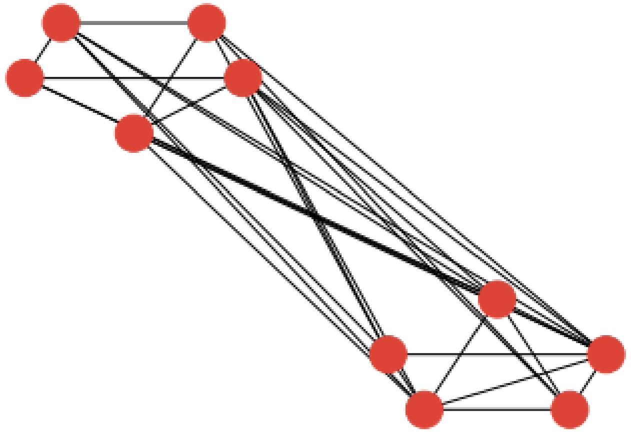
Anomalous graph

Limitation 2 (cont.)

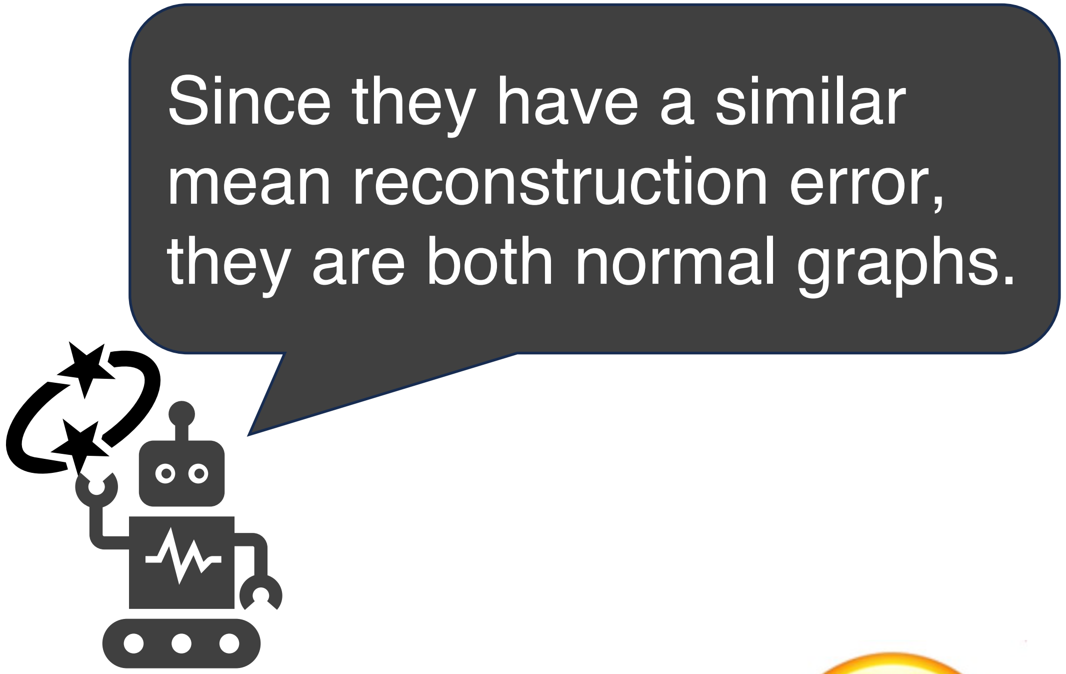
- In such cases, Graph-AEs cannot detect anomalies.



Normal graph



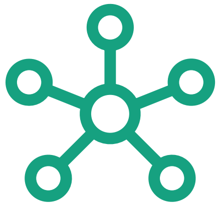
Anomalous graph



Graph-AE-based detector

Our Remedy of Limitation 2

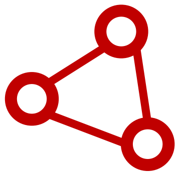
- We use **multifaceted summaries** of errors to represent a graph.



Normal graph

Mean error: 0.2

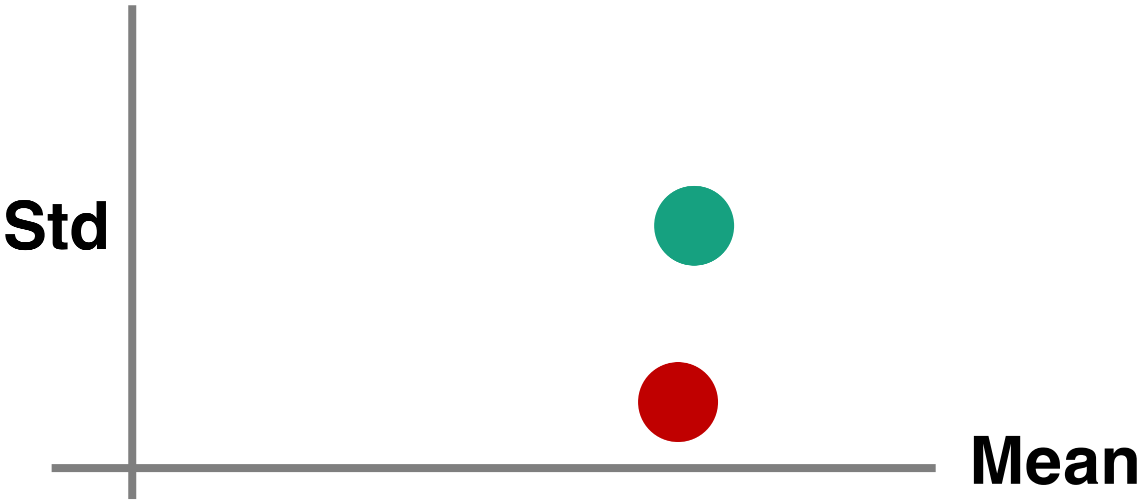
Std of error: 0.3



Anomalous graph

Mean error: 0.2

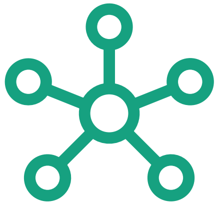
Std of error: 0.1



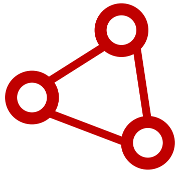
Separable in the feature space

Proposed Method: MUSE

- Proposed method: Represent a graph with its multifaceted summaries of errors, called **MUSE** (**MU**ltifaceted **S**ummaries of reconstruction **E**rrors).



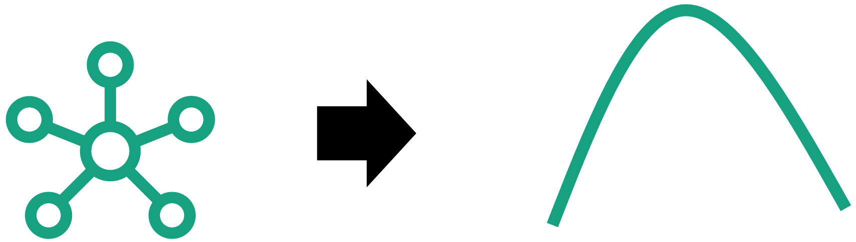
Normal graph



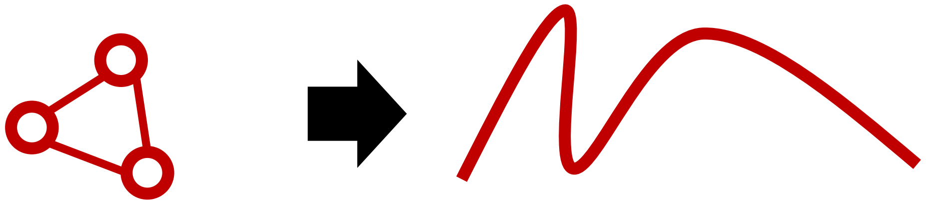
Anomaly

Proposed Method: MUSE (cont.)

- Proposed method: Represent a graph with its multifaceted summaries of errors, called **MUSE** (<u>**MU**</u>ltifaceted <u>**S**</u>ummaries of reconstruction <u>**E**</u>rrors).



Normal graph **Reconstruction error distribution**

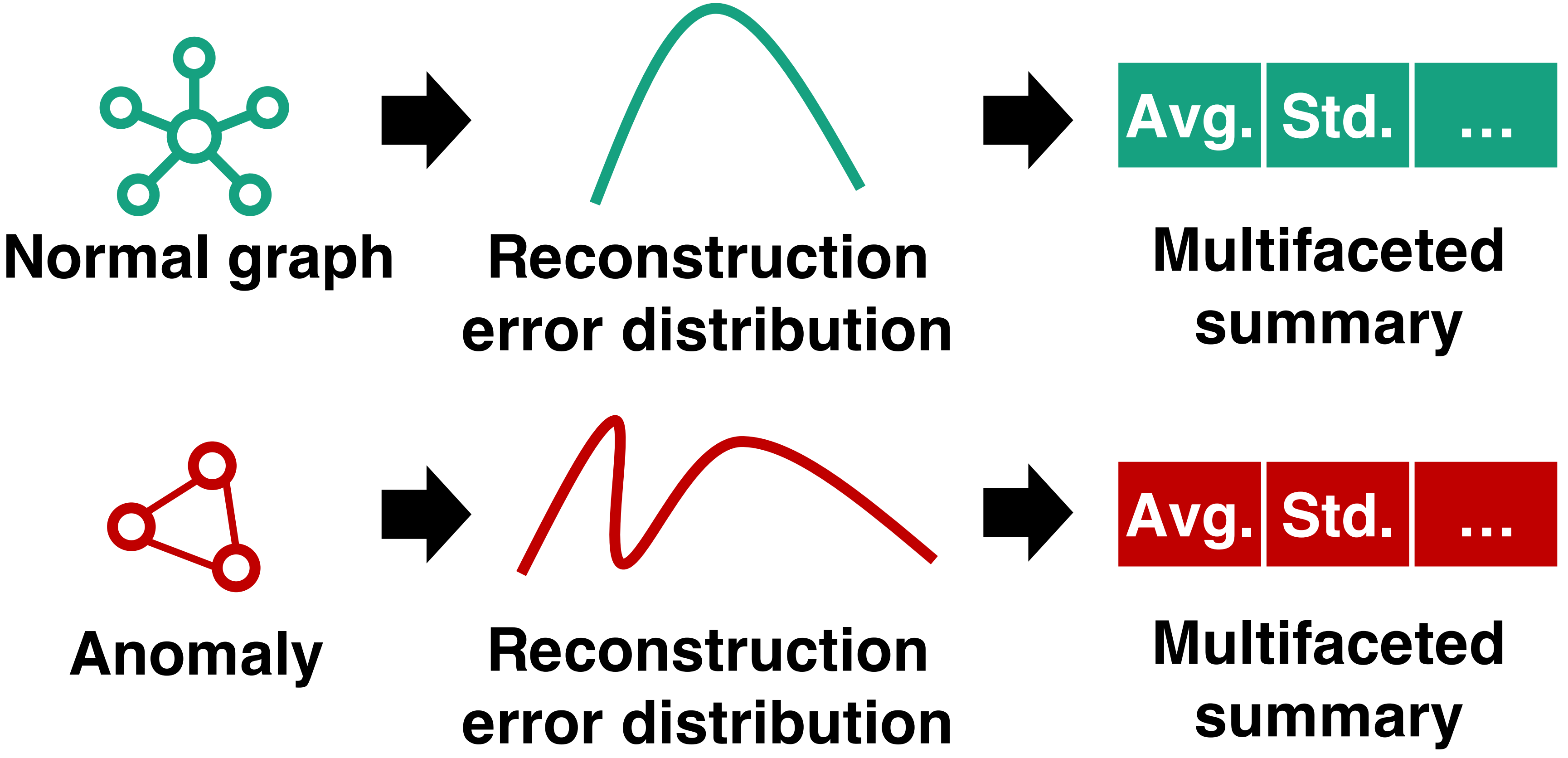


Anomaly **Reconstruction error distribution**

Obtain error distribution with graph autoencoders.

Proposed Method: MUSE (cont.)

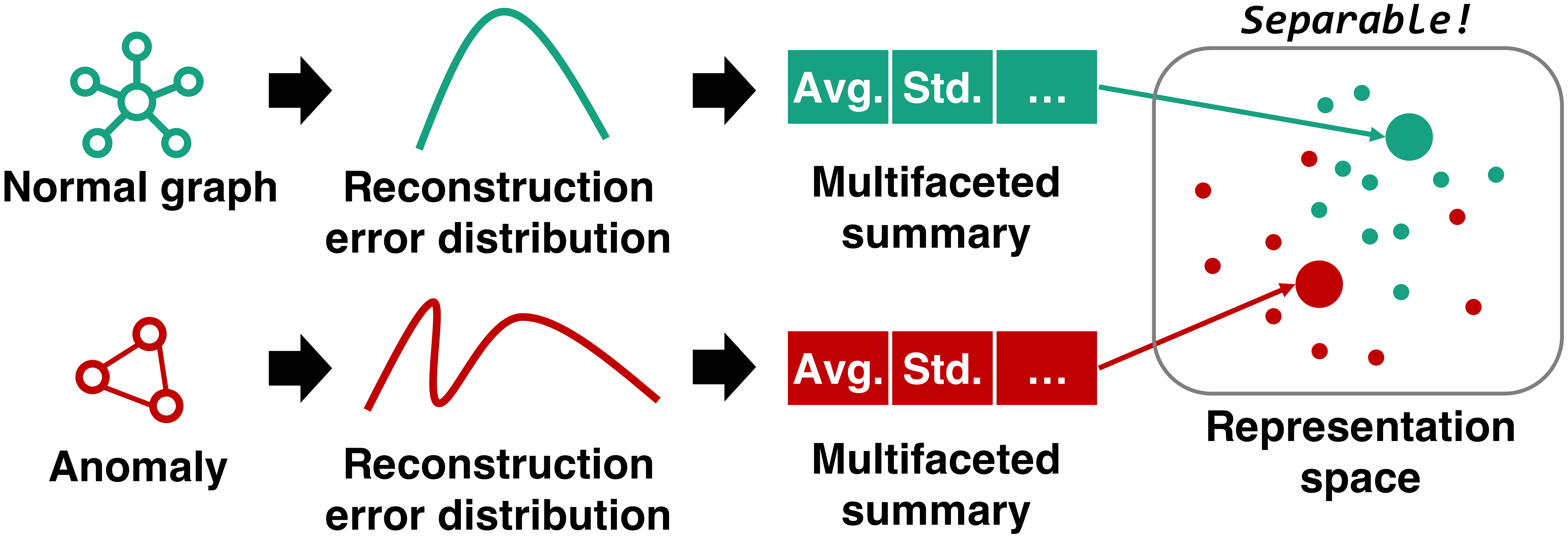
- Proposed method: Represent a graph with its multifaceted summaries of errors, called **MUSE** (**MU**ltifaceted **S**ummaries of reconstruction **E**rrors).



Obtain multifaceted summaries of error distributions

Proposed Method: MUSE (cont.)

- Proposed method: Represent a graph with its multifaceted summaries of errors, called **MUSE** (**MU**ltifaceted **S**ummaries of reconstruction **E**rrors).



Experimental results

- 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

Experimental results (cont.)

- Result 2. MUSE is **robust**.

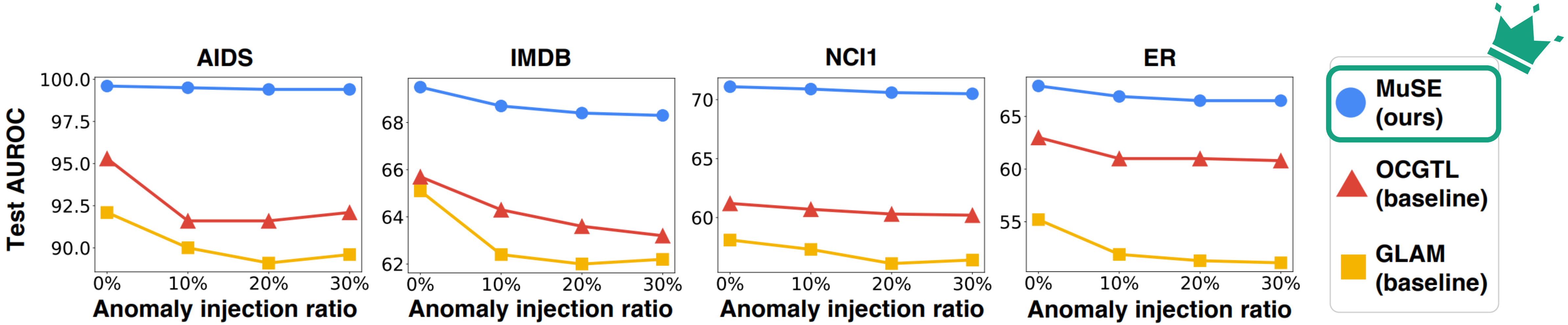
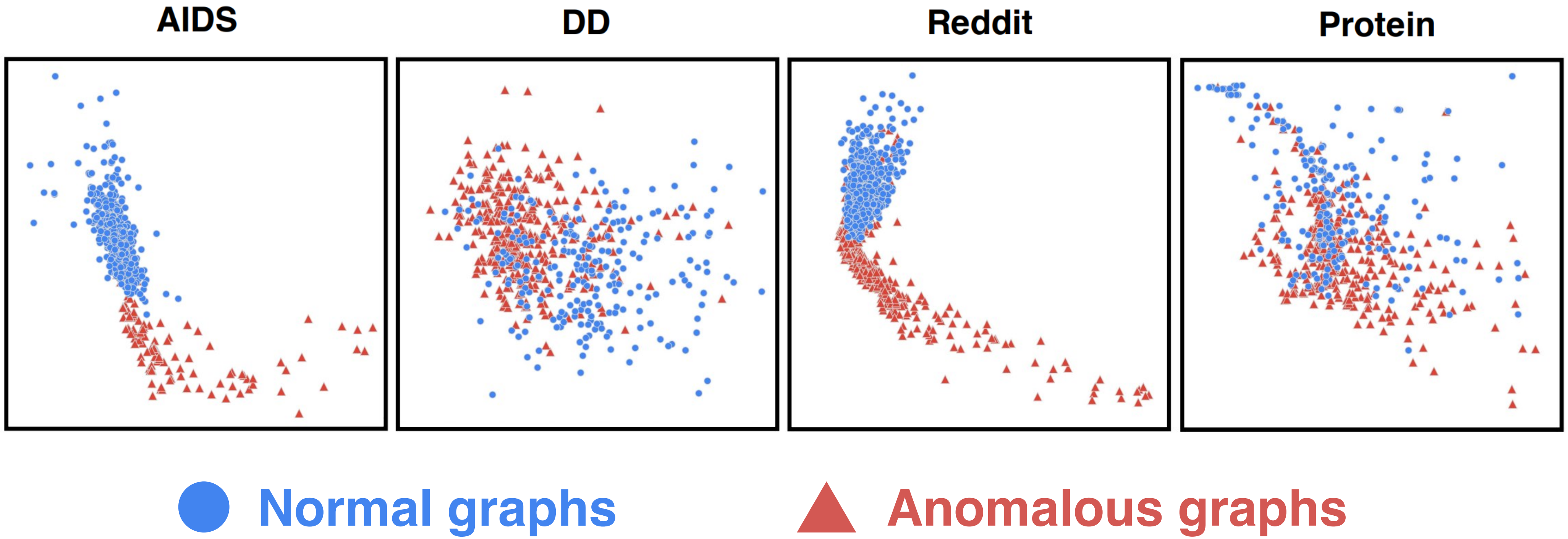


Figure 6: Comparison of the three strongest GLAD methods' robustness against training set contamination. MUSE undergoes the least performance drop among the three methods.

Experimental results (cont.)

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

PCA Visualization



Roadmap

- Overview
- Limitations of Graph-AEs and Our Method
- **Conclusions**

Conclusions

Topic: Graph-level anomaly detection (GLAD)

- **Analysis:** Limitations of graph-autoencoder-based GLAD methods.
- **Method:** MUSE, a novel GLAD method.
- **Experiments:** MUSE outperforms existing GLAD methods.

NEURAL INFORMATION PROCESSING SYSTEMS

KAIST AI
Kim Jaechul Graduate School

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

Sunwoo Kim

Soo Yong Lee

Fanchen Bu

Shinhwan Kang

Kyungho Kim

Jaemin Yoo

Kijung Shin