August 3-7, 2025
KDD2025

AnomalyGFM: Graph Foundation Model for Zero/Few-shot Anomaly Detection

Hezhe Qiao, Chaoxi Niu, Ling Chen, Guansong Pang

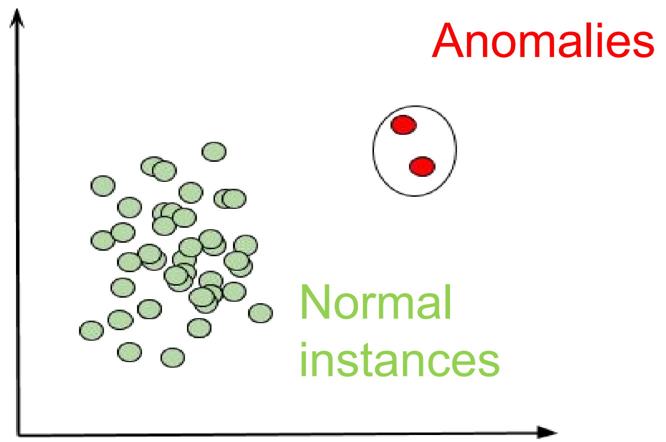
SMU SINGAPORE MANAGEMENT UNIVERSITY

UTS

Background-Introduction to Anomalies and Graph Anomalies

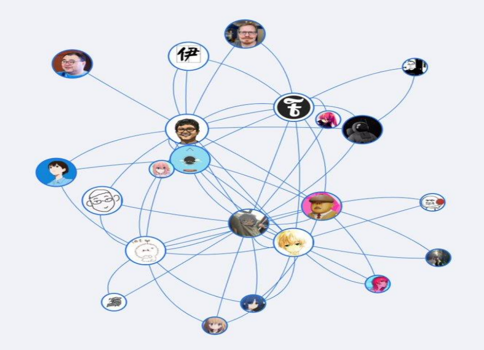
Anomalies (a.k.a., outliers, novelties): Instances that are significantly different from most of the data.

Graph Anomalies: Unexpected, irregular, or suspicious instance in graph-structured data that significantly different from most of the data.

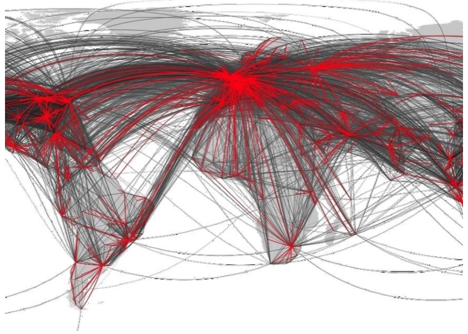


Background-Introduction to GAD

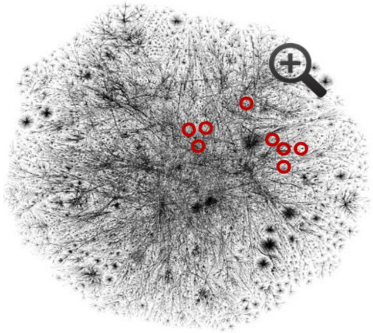
Social network

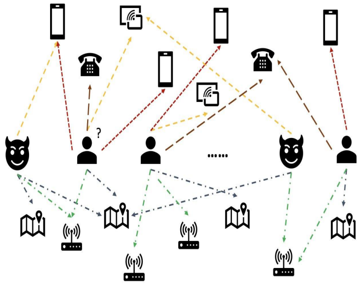


Bank transaction



Web client-server





Social spam

Financial fraud

Network intrusion

......

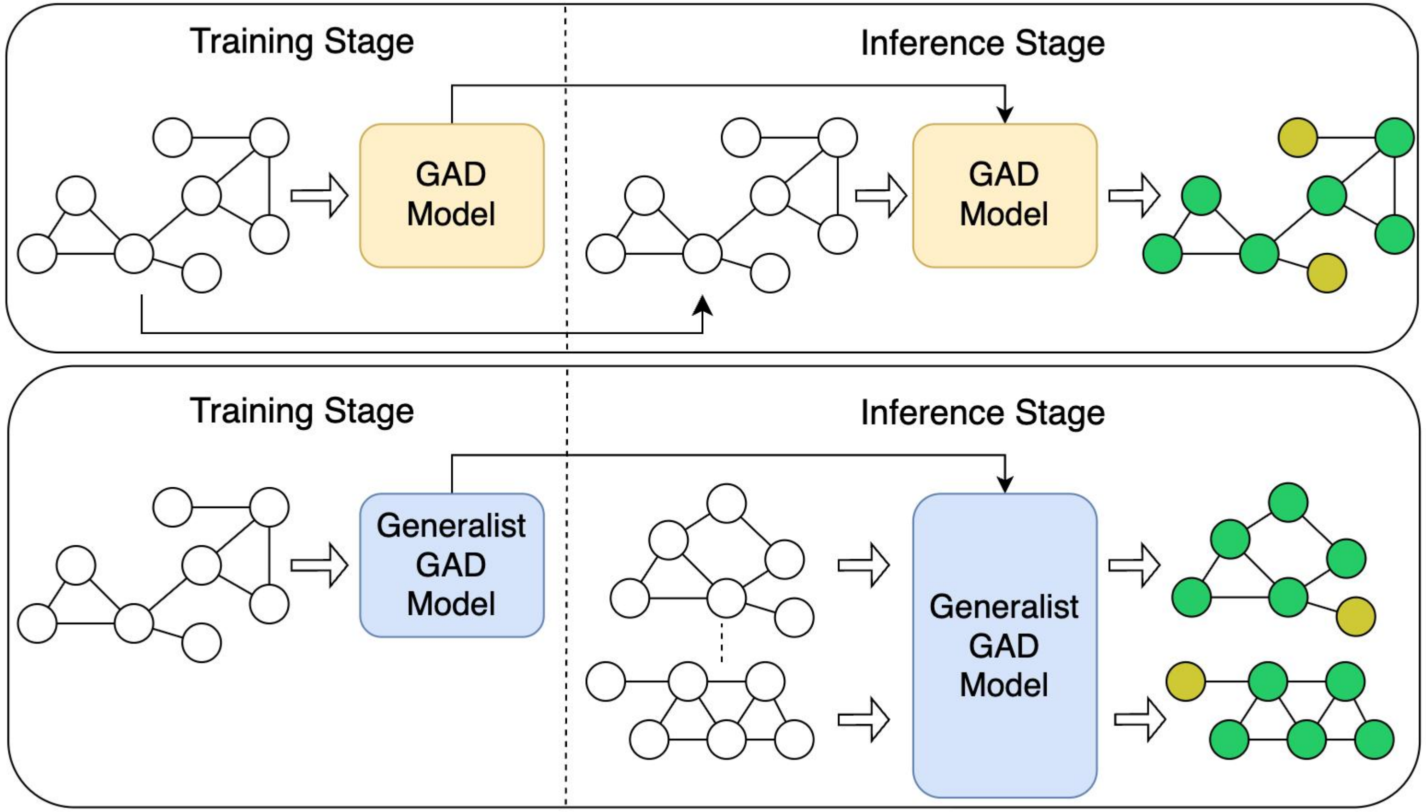
Graph Anomaly Detection (GAD), which aims to identify the rare observations in graphs.

Introduction to Generalist GAD

Generalist GAD：Pre-train a graph neural network designed to generalize across diverse graph domains and anomaly types.

Conventional GAD

Generalist GAD



Challenges of Generalist GAD

- ❖ **Generalist Node Classification**
  - ➢ Feature Heterogeneity
  - ➢ Domain Difference
  - ➢ Fine Tuning Mechanism

- ❖ **Generalist GAD**
  - Challenges of General Node Classification
  - ...
  - ➢ Abnormality/Normality Difference
  - ➢ Generalized Scoring Measurement

Motivation-Residual Feature

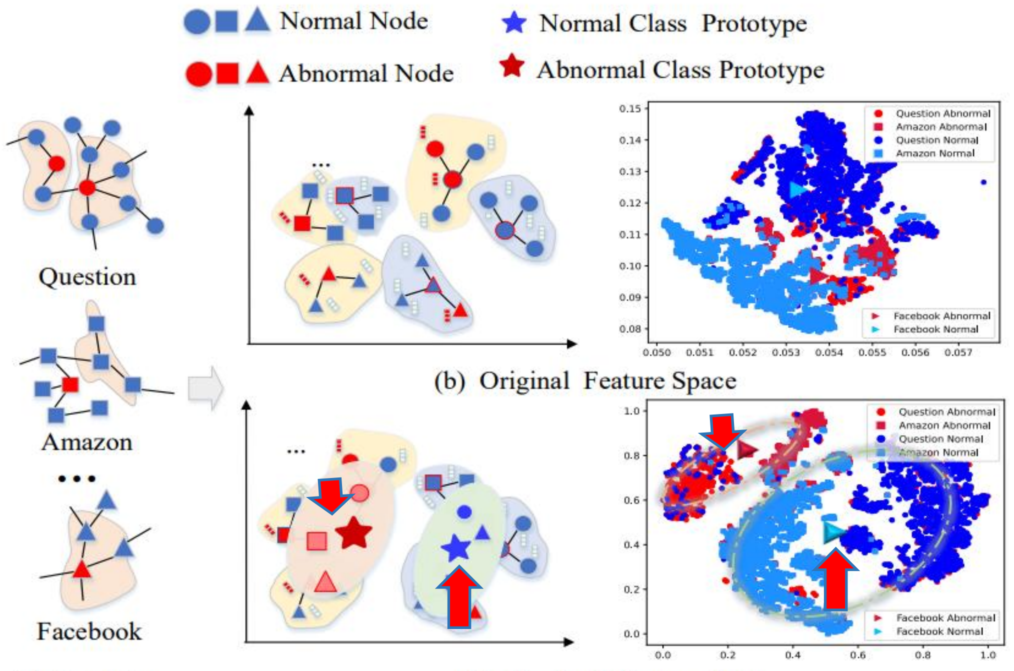
- The residual features essentially project the node information into a unified feature space where we can effectively measure the abnormality of nodes from different graphs in a consistent way.

- **Residual Feature**

$$\mathbf{r}_i = \mathbf{h}_i - \frac{1}{|\mathcal{N}(v_i)|} \sum_{v_j \in \mathcal{N}(v_i)} \mathbf{h}_j$$

- **Abnormality Measure**

$$s_i = \exp\left(\mathbf{r}_i^{\mathrm{T}} \mathbf{p}_a\right) + \beta_{\exp}\left(-\mathbf{r}_i^{\mathrm{T}} \mathbf{p}_n\right)$$



(a) Graph Set (b) Original Feature Space (c) Residual Feature Space

Preliminaries

Feature Unification

- Due to the feature dimension difference across the graphs, we need to align the node features/attributes into a shared feature space

$$\mathbf{X}^{(i)} \in \mathbb{R}^{\mathbb{N}^{(i)} \times d^{(i)}} \xrightarrow[\text{Projection}]{\text{Feature}}$$

GNN for Representation Learning

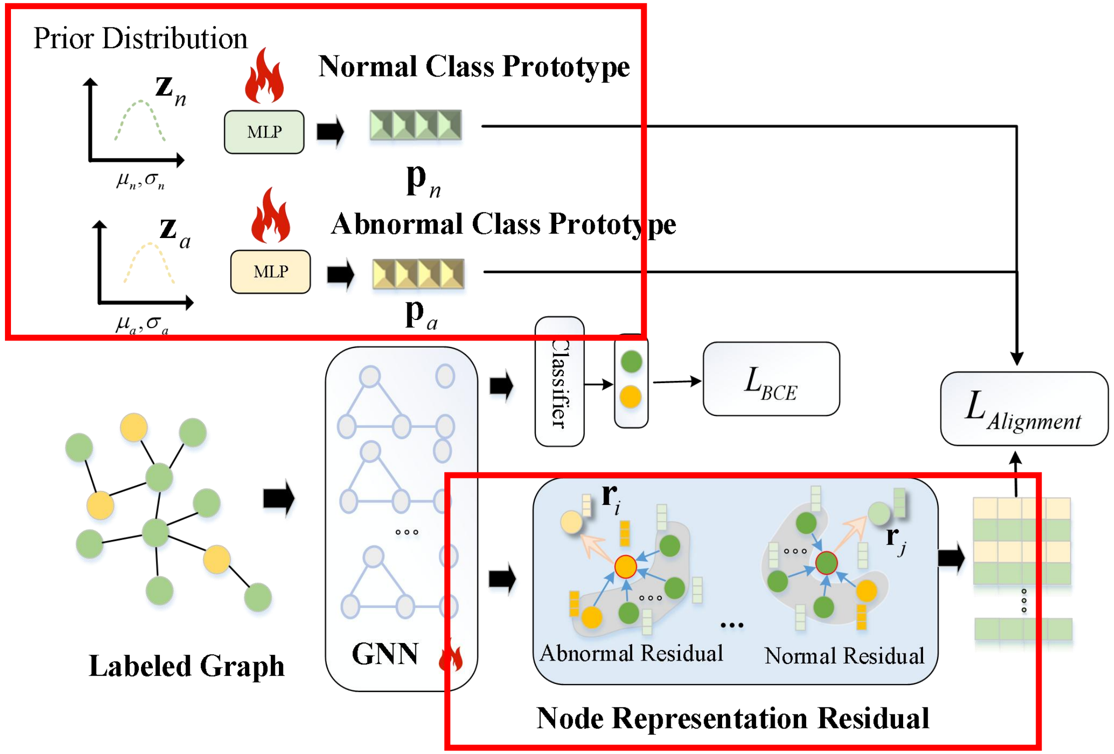
- We adopt a 2-layer GCN to model the graph due to its efficient and simple architecture.

$$\mathbf{H}^{(\ell)} = \mathrm{GNN}\left(\mathbf{A}, \mathbf{H}^{(\ell-1)}; \mathbf{W}^{(\ell)}\right)$$

AnomalyGFM-Pre-training Loss

- Learn the graph-agnostic, discriminative prototypes for the normal and abnormal classes using alignment

data-independent, learnable class prototypes



(a) Pre-training

- Supervised Loss

$$L_{BCE} = \sum_{i=1}^{|\mathcal{V}|} y_i \log(p_i) + (1 - y_i)\log(1 - p_i)$$

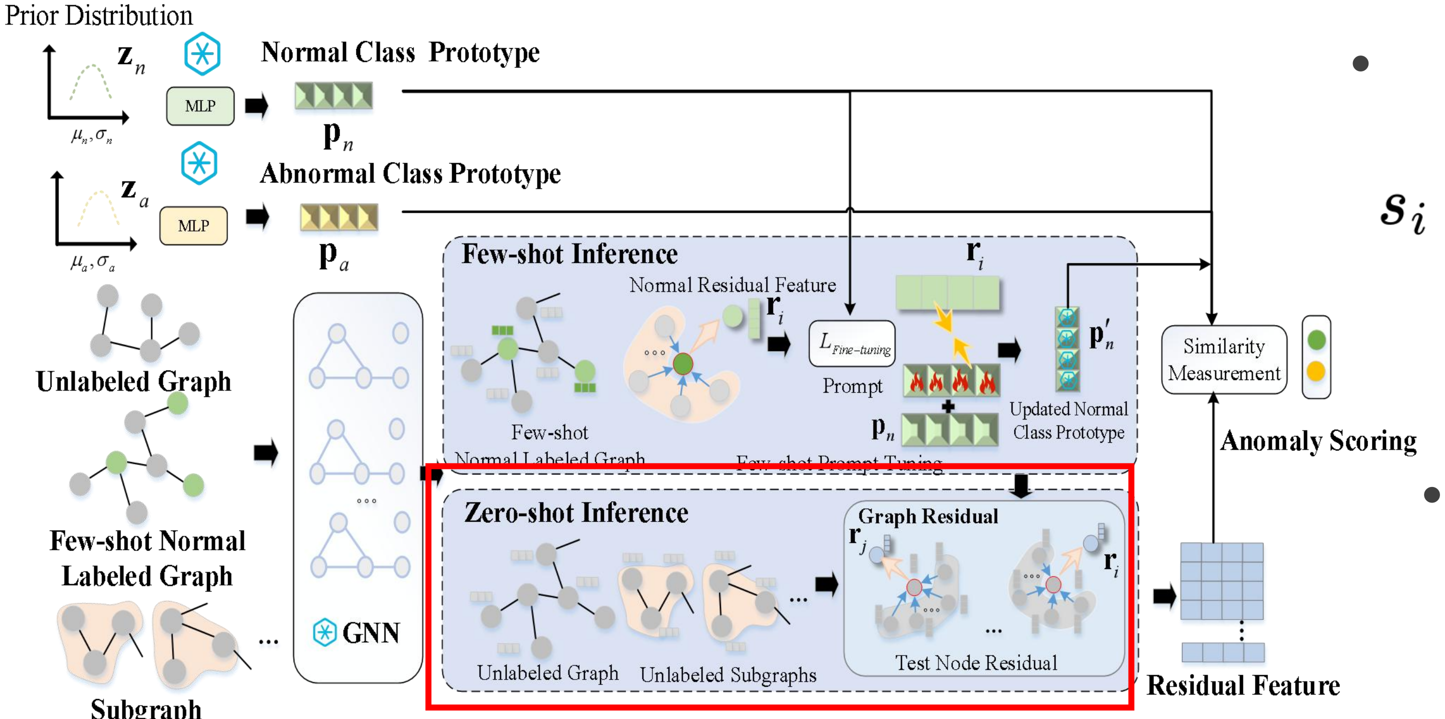
- Alignment Loss

$$L_{\text{Alignment}} = \sum_{i=1}^{|\mathcal{V}|} I_{y_{i=0}} \boxed{\|\mathbf{r}_i - \mathbf{p}_n\|_2^2} + I_{y_{i=1}} \boxed{\|\mathbf{r}_i - \mathbf{p}_a\|_2^2}$$

$$L_{\text{total}} = L_{BCE} + \alpha L_{\text{Alignment}}$$

AnomalyGFM Zero-shot Inference

- During the inference, we use the similarity between node representation and prototypes of normal and abnormal class as the anomaly score.



(b) Zero-/Few-shot Inference

- Zero-shot Inference

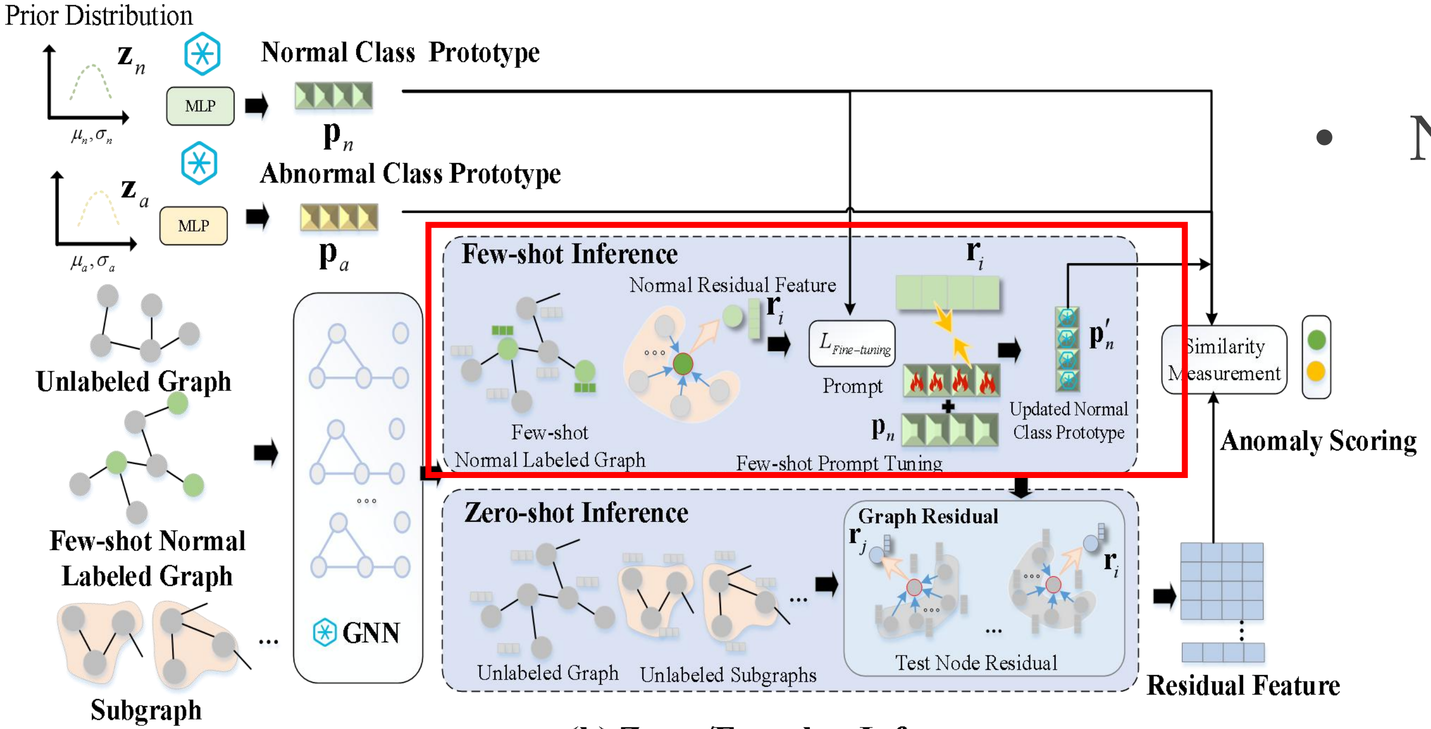
$$s_i = \exp\left(\mathbf{r}_i^{\mathrm{T}} \mathbf{p}_a\right) + \beta \exp\left(-\mathbf{r}_i^{\mathrm{T}} \mathbf{p}_n\right)$$

- Few-shot Inference?

How to exploit some labeled normal nodes ?

AnomalyGFM Few-shot Inference

- A small learnable prompt and adaptivation layer were involved into the normal class prototype to better align it with the normal node representation residuals on the test graph during a prompt fine-tuning step



(b) Zero-/Few-shot Inference

- Normal Prototype-based Fine-tuning

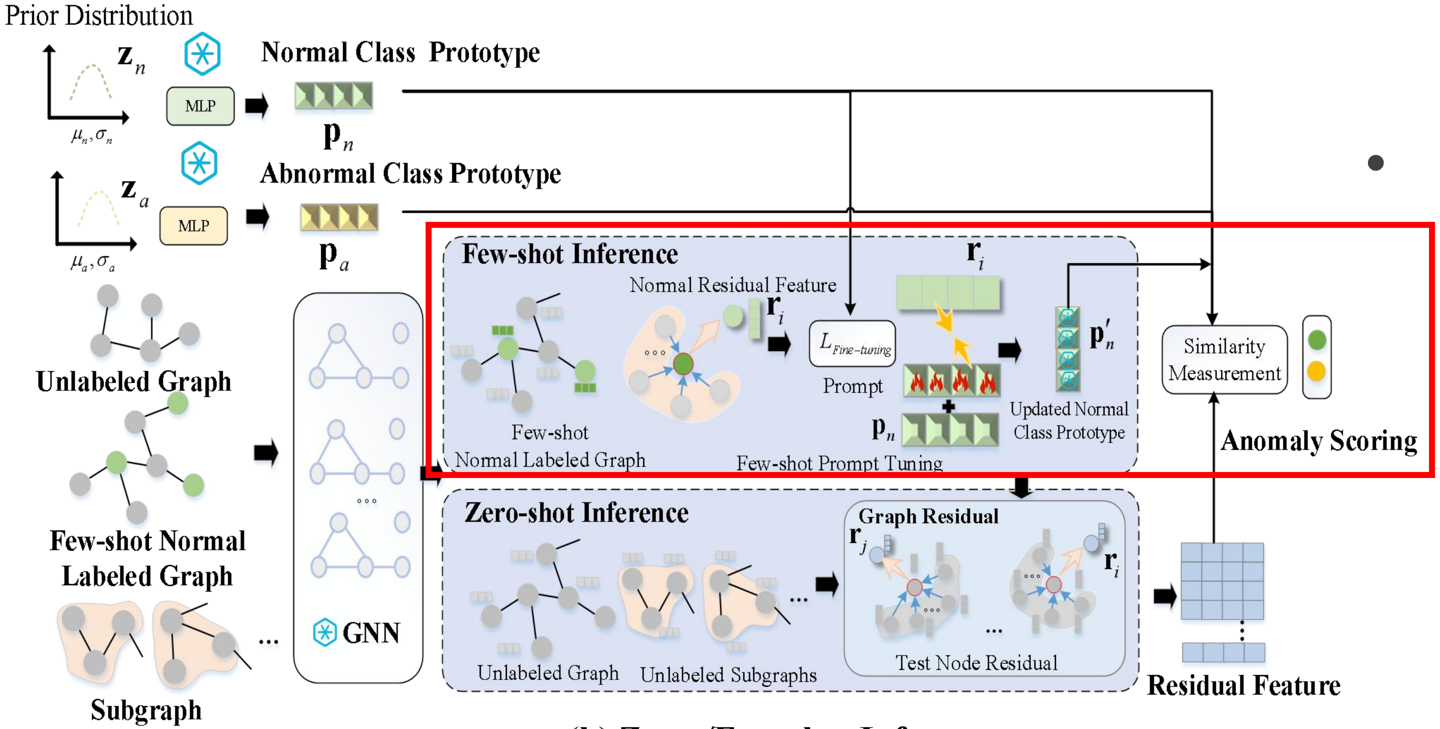
adaptation layer learnable prompt

$$\mathbf{p}'_{\mathbf{n}} = \mathbf{p}_n + \boxed{g(\mathbf{p}_n; \phi)} + \boxed{\Psi_{\mathbf{p}_n}}$$

$$L_{pt} = \sum_{i=1}^{|v_i^{\text{Test}}|} \|\mathbf{r}_i - \mathbf{p}'_n\|_2^2$$

AnomalyGFM Few-shot Inference

- During the inference, we use the similarity between node representation and abnormal class and refined prototypes of normal as the anomaly score.



(b) Zero-/Few-shot Inference

- Few-shot Inference

Refined prototype

$$s_i = \boxed{\exp\left(\mathbf{r}_i^{\mathrm{T}}\mathbf{p}_a\right)} + \beta \exp\left(-\mathbf{r}_i^{\mathrm{T}}\mathbf{p}'_n\right)$$

Datasets and Competing Methods

Dataset	Domain	#Nodes	#Edges	#Feat.	Ano.	Sim.
Facebook	Social Networks	1,081	55,104	576	27(2.49%)	0.690
Reddit	Social Networks	10,984	168,016	64	366(3.33%)	0.997
Amazon	Co-review	10,244	175,608	25	693(6.66%)	0.645
Disney	Co-purchase	124	335	28	6(4.8%)	0.804
Amazon-all	Co-review	11,944	4,398,392	25	821(6.87%)	0.645
YelpChi-all	Co-review	45,941	3,846,979	32	6,674(14.52%)	0.905
Tolokers	Work Collaboration	11,758	519,000	10	2,566(21.8%)	0.814
Question	Social Networks	48,921	153,540	301	1,460(2.98%)	0.679
Elliptic	Bitcoin Transaction	46,564	73,248	93	4,545 (9.8%)	0.356
T-Finance	Transaction Record	39,357	21,222,543	10	1,803(4.6%)	0.107
T-Social	Social Friendship	5,781,065	73,105,508	10	174,280(3.0%)	0.307

- **Unsupervised methods**: AnomalyDAE, CoLA , TAM , and GADAM;
- **Supervised methods:** GCN , GAT , BWGNN, GHRN, and XGBGraph;
- **General GFM methods:** GraphPrompt for general graph tasks, and the one for zero-shot GAD UNPrompt

Main Results —— Zero-shot Inference

- AnomalyGFM performs inference based on two discriminative graph-agonistic prototypes distilled from the node residual feature, supporting strong generalization across the datasets.

Metric	Method	Dataset									Avg.	p-value
		Reddit	Amazon	Disney	Aamzon-all	YelpChi-all	Tolokers	Question	Elliptic	T-Finance		
AUROC	Unsupervised Methods											
	AnomalyDAE (ICASSP'20)	0.5016	0.5818	0.4853	0.7228	0.5002	0.5948	0.4311	0.4197	0.2324	0.4966	0.007
	CoLA (TNNLS'21)	0.4623	0.4580	0.4696	0.4091	0.4879	0.4501	0.4945	0.5572	0.4889	0.4752	0.003
	TAM (NeurIPS'23)	0.5725	0.4720	0.4773	0.7543	0.4216	0.5351	0.5119	0.3282	0.2990	0.4857	0.003
	GADAM (ICLR'24)	0.4532	0.6646	0.4288	0.5959	0.4829	0.4832	**0.5594**	0.3922	0.1382	0.4664	0.007
	Supervised Methods											
	GCN (ICLR'17)	0.5645	0.5988	0.5000	0.7195	0.5486	0.5319	0.5161	**0.7640**	0.2345	0.5531	0.039
	GAT (ICLR'18)	0.5000	0.4981	0.5175	0.5005	0.4802	0.5030	0.4577	0.6588	0.5072	0.5136	0.007
	BWGNN (ICML'22)	0.5208	0.4769	0.6073	0.3648	0.5282	0.4877	0.4404	0.5843	0.5457	0.5062	0.003
	GHRN (WebConf'23)	0.5253	0.4560	0.5336	0.3382	0.5125	0.4860	0.4535	0.5400	0.5324	0.4863	0.003
	XGBGraph (NeurIPS'23)	0.4601	0.4179	0.6692	0.7950	0.4945	0.5462	0.5095	0.4274	0.3402	0.5177	0.003
	Generalist Methods											
	GraphPrompt (WebConf'23)	0.4677	0.4904	0.5192	0.3215	0.4976	0.4779	0.4204	0.3221	0.5405	0.4508	0.003
	UNPrompt (Arxiv'24)	0.5337	0.7525	0.6412	0.7962	0.5558	**0.6853**	0.4757	0.5901	0.2318	0.5847	0.074
	AnomalyGFM	**0.5974**	**0.8417**	**0.6751**	**0.9032**	**0.5791**	0.5843	0.5280	0.6195	**0.5614**	**0.6544**	/
AUPRC	Unsupervised Methods											
	AnomalyDAE (ICASSP'20)	0.0327	0.0833	0.0566	0.1921	0.1484	0.1876	0.0241	0.0798	0.0274	0.0924	0.003
	CoLA (TNNLS'21)	0.0391	0.0669	0.0701	0.0861	0.1466	0.0848	0.0292	0.0998	0.0430	0.0739	0.007
	TAM (NeurIPS'23)	0.0413	0.0666	0.0628	0.1736	0.1240	0.0970	0.0307	0.0697	0.0332	0.0776	0.007
	GADAM (ICLR'24)	0.0293	0.1562	0.0651	0.1595	0.1371	0.1001	0.0395	0.0733	0.0261	0.0873	0.003
	Supervised Methods											
	GCN (ICLR'17)	**0.0439**	0.0891	0.0484	0.1536	0.1735	0.1060	0.0387	**0.1963**	0.0274	0.0974	0.074
	GAT (ICLR'18)	0.0329	0.0688	0.0530	0.0696	0.1400	0.0822	0.0259	0.1366	0.0463	0.0728	0.003
	BWGNN (ICML'22)	0.0389	0.0652	0.0624	0.0586	0.1605	0.1030	0.0257	0.1158	0.0479	0.0753	0.007
	GHRN (WebConf'23)	0.0407	0.0633	0.0519	0.0569	0.1505	0.0957	0.0259	0.1148	0.0457	0.0717	0.007
	XGBGraph (NeurIPS'23)	0.0330	0.0536	0.1215	0.2307	0.1449	0.1256	0.0306	0.0816	**0.0754**	0.0996	0.027
	Generalist Methods											
	GraphPrompt (WebConf'23)	0.0334	0.0661	0.0610	0.0666	0.1542	0.2070	0.0266	0.0664	0.0492	0.0811	0.003
	UNPrompt (Arxiv'24)	0.0351	0.1602	0.1236	0.2430	0.1810	0.2219	0.0348	0.1278	0.0279	0.1283	0.003
	AnomalyGFM	0.0387	**0.5790**	**0.1242**	**0.6820**	**0.1819**	**0.2749**	**0.0397**	0.1371	0.0593	**0.2352**	/

Main Results —— Few-shot Inference

- **Few shot methods:** GPPT, GraphPrompt, and ARC

- AnomalyGFM achieves better sample-efficiency tuning due to the learning of class-level prototypes that are agnostic to different domains of graphs

Metric	Setting	Method	Reddit	Amazon	Disney	Aamzon-all	YelpChi-all	Tolokers	Question	Elliptic	T-Finance	Avg.	p-value
AUROC	1-shot	GPPT (KDD'22)	0.5000	0.5303	0.4997	0.5010	0.5000	0.5061	0.4921	0.6162	0.3647	0.5011	0.003
		GraphPrompt (WebConf'23)	0.4216	0.4882	0.4223	0.2631	0.4811	0.5328	0.4086	0.6001	0.4000	0.4464	0.004
		ARC (NeurIPS'24)	0.4899	0.4571	0.3578	0.4570	0.4910	0.4667	**0.5865**	0.2904	0.2484	0.4272	0.008
		AnomalyGFM	**0.5922**	**0.8531**	**0.6649**	**0.8972**	**0.5872**	**0.5898**	0.5303	**0.6199**	**0.5916**	**0.6584**	/
	5-shot	GPPT (KDD'22)	0.5000	0.5098	0.5000	0.5051	0.5000	0.5181	0.4959	0.5736	0.2609	0.4848	0.003
		GraphPrompt (WebConf'23)	0.4406	0.4900	0.6497	0.4726	0.5359	0.5381	0.4069	0.6012	0.4069	0.5046	0.003
		ARC (NeurIPS'24)	0.4720	0.4458	0.4435	0.4473	0.5112	0.4746	**0.5906**	0.2714	0.2168	0.4303	0.007
		AnomalyGFM	**0.6023**	**0.8600**	**0.6613**	**0.9011**	**0.5951**	**0.6095**	0.5426	**0.6119**	**0.6248**	**0.6676**	/
	10-shot	GPPT (KDD'22)	0.5000	0.5087	0.4769	0.5023	0.5000	0.4971	0.5047	0.4212	0.5539	0.4961	0.003
		GraphPrompt (WebConf'23)	0.4321	0.4906	0.6314	0.7167	0.5367	0.5329	0.3826	0.6221	0.4260	0.5301	0.007
		ARC (NeurIPS'24)	0.4867	0.4323	0.4769	0.4467	0.5145	0.4786	**0.5901**	0.2644	0.2298	0.4355	0.003
		AnomalyGFM	**0.6252**	**0.8649**	**0.6649**	**0.9215**	**0.6064**	**0.6140**	0.5611	**0.6303**	**0.6283**	**0.6796**	/
AUPRC	1-shot	GPPT (KDD'22)	0.0333	0.0766	0.0488	0.0687	0.1453	0.2204	0.0294	0.1239	0.0432	0.0877	0.003
		GraphPrompt (WebConf'23)	0.0283	0.0680	0.0486	0.0426	0.1113	0.2321	**0.0448**	0.1108	0.0302	0.0796	0.012
		ARC (NeurIPS'24)	0.0332	0.0581	0.0453	0.0590	0.1402	0.2122	0.0468	0.0701	0.0277	0.0769	0.011
		AnomalyGFM	**0.0398**	**0.5801**	**0.1223**	**0.6921**	**0.1852**	**0.2786**	0.0332	**0.1401**	**0.0601**	**0.2368**	/
	5-shot	GPPT (KDD'22)	0.0333	0.0692	0.0504	0.0716	0.1453	0.2265	0.0297	0.1127	0.0365	0.0861	0.004
		GraphPrompt (WebConf'23)	0.0285	0.0681	0.0892	0.0600	0.1661	**0.2957**	0.0327	0.1416	0.0360	0.1019	0.009
		ARC (NeurIPS'24)	0.0312	0.0571	0.0546	0.0572	0.1464	0.2150	**0.0471**	0.0726	0.0267	0.0786	0.011
		AnomalyGFM	**0.0401**	**0.5831**	**0.1257**	**0.6985**	**0.1918**	0.2866	0.0336	**0.1437**	**0.0622**	**0.2405**	/
	10-shot	GPPT (KDD'22)	0.0334	0.0691	0.0526	0.0698	0.1453	0.2178	0.0301	0.0905	0.0511	0.0844	0.004
		GraphPrompt (WebConf'23)	0.0278	0.0681	0.0848	0.1427	0.1649	**0.2922**	0.0263	0.1421	0.0382	0.1096	0.007
		ARC (NeurIPS'24)	0.0327	0.0557	0.0743	0.0583	0.1491	0.2168	**0.0463**	0.0677	0.0272	0.0809	0.011
		AnomalyGFM	**0.0444**	**0.5895**	**0.1399**	**0.7124**	**0.1990**	0.2897	0.0346	**0.1570**	**0.0644**	**0.2478**	/

Scale up to Very Large Graphs

- ❑ AnonalyGFM can effectively infer the anomaly score without considering the entire graph structure, eliminating the bottleneck of loading the full graph for GAD inference.

$$\mathbf{r}_i = \mathbf{h}_i - \frac{1}{|\mathcal{S}(v_i)|} \sum_{v_j \in \mathcal{S}(v_i)} \mathbf{h}_j$$

- ❖ Privacy-sensitive settings where we do not want to reveal the entire graph structure

Table 4: Subgraph inference on the very large-scale graphs. '/' indicates that the model cannot handle the dataset.

Metric	Method	Dataset T-Finance	Dataset T-Social
AUROC	Unsupervised Methods		
	TAM (NeurIPS'23)	0.2990	/
	GADAM (ICLR'24)	0.1382	0.5155
	Supervised Methods		
	BWGNN (ICML'22)	0.5457	0.4964
	GHRN (WebConf'23)	0.5324	0.4934
	XGBGraph (NeurIPS'23)	0.3402	0.4602
	AnomalyGFM	**0.7852**	**0.5991**
AUPRC	Unsupervised Methods		
	TAM (NeurIPS'23)	0.0332	/
	GADAM (ICLR'24)	0.0261	0.0285
	Supervised Methods		
	BWGNN (ICML'22)	0.0479	0.0301
	GHRN (WebConf'23)	0.0457	0.0303
	XGBGraph (NeurIPS'23)	0.0754	0.0305
	AnomalyGFM	**0.1059**	**0.0398**

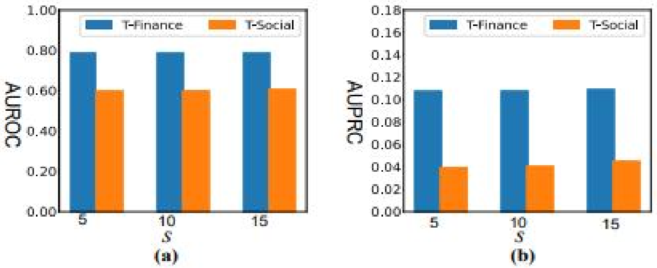
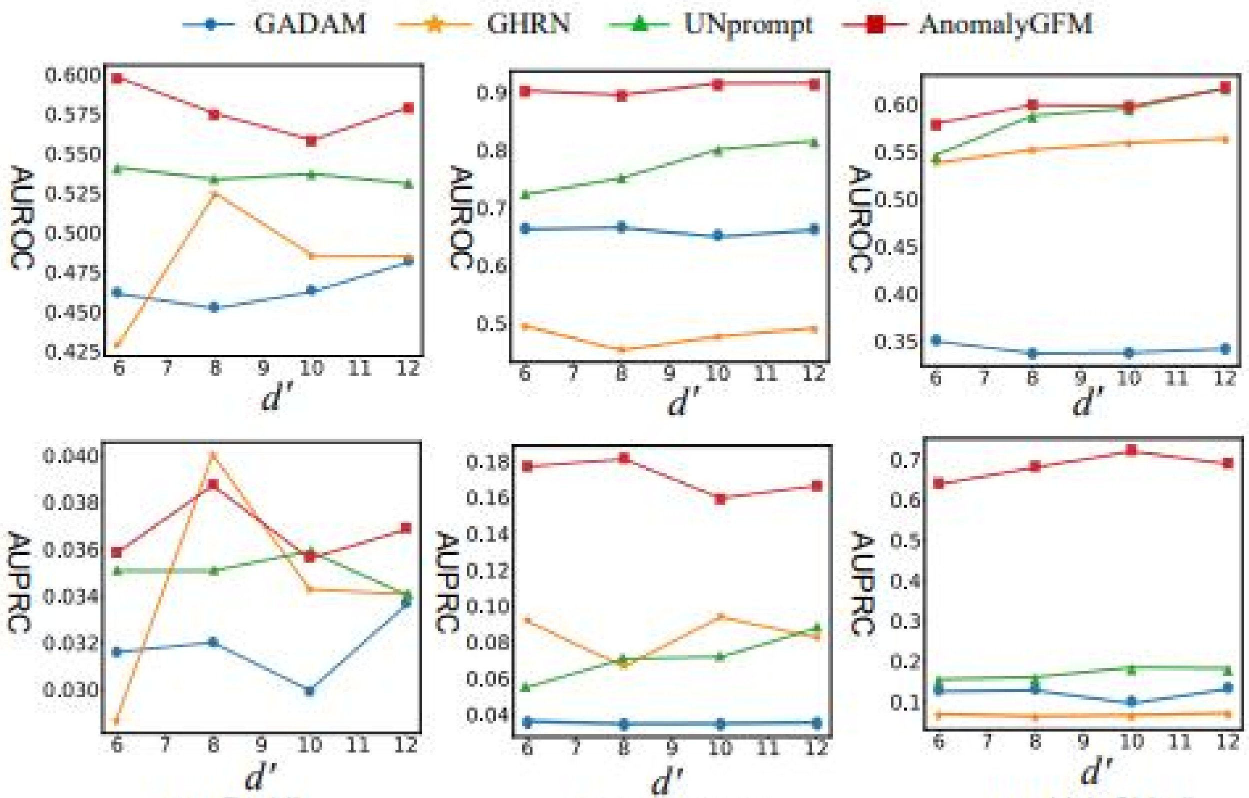


Figure 3: AnomalyGFM performance w.r.t subgraph size *s*.

Sensitivity of AnomalyGFM w.r.t common dimensionality

❖ Feature Unification

$$\mathbf{X}^{(i)} \in \mathbb{R}^{\mathbb{N}^{(i)} \times d^{(i)}} \xrightarrow[\text{Projection}]{\text{Feature}} \widetilde{\mathbf{X}}^{(i)} \in \mathbb{R}^{\mathbb{N}^{(i)} \times d'}$$



(a) Reddit (b) Amazon-all (c) YelpChi-all

Conclusion

- ❑ We build a GAD-oriented graph foundation model, AnoamlyGFM, that can work effectively under both few-shot and zero-shot scenarios.
- ❑ AnomalyGFM is pre-trained to learn discriminative and data-independent prototypes by aligning them with the graph-agnostic node representation residuals.
- ❑ This provides a consistent and identical way for abnormality measurement using the similarity between residual node representation and the learned class prototypes, facilitating the strong generalization in both zero-shot and few-shot inference.