Multi-dimensional Adaptive Mix-hop Contextual Learning Framework for Universal Graph Anomaly Detection

Zhaowei Liu[1*], **Leilei Jiang**[1], **Haitao Yang**[1]

[1]School of Computer and Control Engineering, Yantai University, Shandong, China
lzw@ytu.edu.cn, {jiang0109, yanghaitao}@s.ytu.edu.cn

Abstract

Graph Anomaly Detection focuses on identifying instances that deviate from normal patterns in graph-structured data. Although substantial progress has been made in this field, current approaches are constrained by the "one-dataset-one-model" paradigm, exhibiting limited generalization across graphs with heterogeneous feature spaces, poor adaptability in few-shot scenarios, and inefficient cross-domain deployment. To overcome these limitations, we propose SAARCS, a universal graph anomaly detection framework capable of performing anomaly detection across diverse graph datasets without requiring any target data training. SAARCS aligns feature dimensions through composite spatial smoothness, learns graph embeddings via an adaptive-hop attention encoder, and predicts node abnormality using only a small set of normal context nodes. Extensive experiments on eight real-world datasets demonstrate that our approach achieves superior performance compared to state-of-the-art baselines.

Introduction

GRAPH Anomaly Detection (GAD), as an extension of anomaly detection research to non-Euclidean spaces, aims to identify anomalous components—such as nodes, edges, or subgraphs—in a single graph or detect anomalous graphs within a graph set (Pang et al. 2021; Wang et al. 2023; He et al. 2024). Given its substantial potential across diverse real-world applications, including financial risk assessment (Li et al. 2022; Liu et al. 2023) and system fault detection (Huang et al. 2022), GAD has garnered growing research attention in recent years (Ma et al. 2021; Tang et al. 2022, 2023; Motie and Raahemi 2024; Yang, Zhao, and Shen 2025). However, the diverse types of anomalies and complex node interaction patterns in graph-structured data present significant challenges for representation in Euclidean feature spaces (Ding et al. 2019; Kong et al. 2024). As a result, traditional anomaly detection methods cannot be directly applied to graph anomaly detection, highlighting the urgent need for specialized research (Liu et al. 2021b; Wang et al. 2023). Driven by their ability to extract deep structural and feature information from graphs, Graph Neural Networks (GNNs) have recently become a dominant approach for GAD (Liu, Gao, and Ji 2020; Jin et al. 2024). Various GNN-based frameworks have been proposed, demonstrating notable advancements in this field (Ding et al. 2019; Tang et al. 2023). Many GNN-based GAD methods formulate GAD as a supervised binary classification task, leveraging specially designed GNN architectures to capture anomaly patterns (Dou et al. 2020; Li et al. 2019; Liu et al. 2021a; Li et al. 2024). Alternatively, another category of approaches focuses on the more challenging unsupervised scenario, utilizing diverse unsupervised learning objectives and frameworks to detect anomalies without relying on labeled data (Ding et al. 2019; Liu et al. 2021b; Qiao and Pang 2023; Pan et al. 2025; Duan et al. 2024).

Despite their high detection accuracy, existing GAD approaches predominantly adopt a "one model for one dataset" learning paradigm. This paradigm necessitates dataset-specific training and substantial training data for constructing individual detection models. However, this approach has several inherent limitations: (1) **Challenges in Handling Heterogeneous Feature Spaces**: Graph datasets from diverse domains vary significantly in feature dimensions, semantic meanings, and data distributions. Traditional feature alignment techniques, such as Principal Component Analysis, rely on linear transformations and local smoothness assumptions. As a result, they fail to preserve the global statistical characteristics of anomaly-sensitive features across different domains, limiting the effectiveness of cross-dataset generalization. (2) **Restricted Generalizability**: These methods require retraining for each new dataset, making it difficult to leverage cross-domain knowledge efficiently. This limitation becomes particularly problematic in scenarios with sparse data availability or strict privacy constraints, where data sharing and extensive retraining are infeasible. (3) **Inadequate Few-Shot Adaptability**: When applied to new datasets, existing GAD approaches struggle to identify dataset-specific anomaly patterns using only a limited number of normal samples. Instead, they often demand extensive data labeling or full model retraining, which increases both the time and resource costs of model deployment.

To address these limitations, this study proposes a universal graph anomaly detection framework with an integrated multi-dimensional anomaly perception mechanism, which achieves robust cross-dataset detection through three

*Corresponding author.

synergistic core components. Firstly, the feature alignment module innovatively integrates local Laplacian smoothness and global statistical features to construct a composite spatial feature selection criterion, thereby capturing both structural anomalies (e.g., local topological mutations) and numerical anomalies (e.g., global distribution shifts) simultaneously. Secondly, the graph embedding module introduces an adaptive hop-attention mechanism to dynamically aggregate residual information from multi-hop neighborhoods, enabling precise screening of key features in complex graph structures while adaptively learning high-order structural representations of nodes. Finally, the anomaly discrimination module incorporates contextual node information and leverages reconstruction errors (discrepancies between query nodes and their reconstructed counterparts) to quantify anomaly levels, generating fine-grained anomaly scores. The synergistic effect of these components enables more comprehensive capture of multi-dimensional features of anomalous nodes, significantly enhancing the performance of universal GAD tasks across diverse datasets.

Related Work

Graph Anomaly Detection. In this paper, we focus on the node-level AD on graphs and refer to it as "graph anomaly detection" following (Zheng et al. 2021; Tang et al. 2022). Shallow methods primarily rely on graph statistics and the homophily assumption (i.e., connected nodes share similar features) (Li et al. 2017; Peng et al. 2018). While highly interpretable, these approaches exhibit limited performance when handling complex topological structures.

Advanced methods predominantly employ GNNs and can be categorized into supervised approaches and unsupervised approaches (Tang et al. 2023; Ma et al. 2021). Supervised methods assume availability of both normal and anomalous node labels for training (Tang et al. 2023), typically framing GAD as a binary classification task (Li et al. 2019; Liu et al. 2021a); Unsupervised methods, requiring no labeled data and instead detecting anomalies through techniques like data reconstruction (Luo et al. 2022; Fan, Zhang, and Li 2020; He et al. 2024), contrastive learning (Liu et al. 2021b; Duan et al. 2023; Zheng et al. 2024), or auxiliary objectives (Qiao and Pang 2023; Pan et al. 2023). DOMINANT (Ding et al. 2019) employs a GCN-based autoencoder to reconstruct graph structures, identifying anomalies via reconstruction errors. AEGIS (Ding et al. 2021) introduces an unsupervised inductive framework, enabling anomaly detection for previously unseen nodes.(Qiao and Pang 2023) introduced a Truncated Affinity Maximization (TAM) framework that optimizes local node affinity on pruned graph structures, demonstrating robust performance on both synthetic datasets and real-world anomaly detection benchmarks. GAD-NR (Roy et al. 2024) detecting anomalies through reconstructed neighbor nodes. ADA-GAD (He et al. 2024) proposes a novel two-stage anomaly-denoised autoencoders for graph anomaly detection.AD-GCL (Xu et al. 2025) devises the neighbor pruning strategy to filter noisy edges for head nodes and facilitate the detection of genuine tail nodes by aligning from head nodes to forged tail nodes.

In-Context Learning. ICL is an adaptation paradigm that enables pretrained models to generalize to new domains, datasets, or tasks using few-shot context samples without requiring fine-tuning. This capability has demonstrated remarkable versatility in both large language models (LLMs) and computer vision (CV) models. Recent work has extended ICL to graph learning—for instance, PRODIGY adapts few-shot node classification via prompt graphs (Huang et al. 2023), while UniLP generalizes it to link prediction tasks (Dong et al. 2024), and (Hendel, Geva, and Globerson 2023) through graph-text alignment, structural information of graphs is encoded into natural language prompts; (Li et al. 2025) by conceptualizing graph learning as a retrieval-augmented generation (RAG) process to enhance the in-context learning capabilities of LLMs for graph-based tasks. Nevertheless, ICL for generalist graph anomaly detection (GAD) remains an open challenge, particularly in developing effective mechanisms to capture anomaly patterns through contextual learning.

Preliminaries

Notations. Consider an attributed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ consisting of n nodes and m edges. Here, $\mathcal{V} = \{v_1, \ldots, v_n\}$ denotes the node set, and $\mathcal{E}$ denotes the edge set. Node-level attributes are encapsulated in a feature matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, where the i-th row $\mathbf{X}_i$ corresponds to the feature vector of node v_i. Inter-node connectivity is captured by an adjacency matrix $\mathbf{A} \in \{0,1\}^{n \times n}$: the entry $\mathbf{A}_{ij} = 1$ if and only if nodes v_i and v_j are connected.

Conventional GAD Problem. GAD targets distinguishing abnormal nodes $\mathcal{V}_a$ from normal nodes $\mathcal{V}_n$ in a given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$. Here, $\mathcal{V}_a$ and $\mathcal{V}_n$ satisfy $\mathcal{V}_a \cup \mathcal{V}_n = \mathcal{V}$, $\mathcal{V}_a \cap \mathcal{V}_n = \emptyset$, and $|\mathcal{V}_a| \ll |\mathcal{V}_n|$. An anomaly label vector $\mathbf{y} \in \{0,1\}^n$ marks node abnormality: $\mathbf{y}_i = 1$ if and only if $v \in \mathcal{V}_a$, and $\mathbf{y}_i = 0$ if and only if $v \in \mathcal{V}_n$. Formally, GAD seeks to learn an anomaly scoring function (i.e., GAD model) $f : \mathcal{V} \to \mathbb{R}$ such that $f(v') > f(v)$ holds for all $v' \in \mathcal{V}_a$ and all $v \in \mathcal{V}_n$. In the traditional "one-model-per-dataset" GAD paradigm, the model f is optimized on a target graph dataset $\mathcal{D} = (\mathcal{G}, \mathbf{y})$, either with partial anomaly labels (supervised scenario) or without labels (unsupervised scenario). Post sufficient training, f enables anomaly identification in $\mathcal{G}$ during inference.

Generalist GAD Problem. This paper explores the generalist GAD problem. The objective is to develop a generalist GAD model that can detect abnormal nodes across diverse graph datasets from various application domains, without requiring any training on specific target data. Concretely, let $\mathcal{T}_{\text{train}} = \{\mathcal{D}^{(1)}_{\text{train}}, \ldots, \mathcal{D}^{(N)}_{\text{train}}\}$ represent a set of training datasets. Each $\mathcal{D}^{(i)}_{\text{train}} = (\mathcal{G}^{(i)}_{\text{train}}, \mathbf{y}^{(i)}_{\text{train}})$ is a labeled dataset from an arbitrary domain. Our goal is to train a generalist GAD model f on $\mathcal{T}_{\text{train}}$, such that f can identify anomalies in any test graph dataset $\mathcal{D}^{(i)}_{\text{test}} \in \mathcal{T}_{\text{test}}$. Here, $\mathcal{T}_{\text{test}} = \{\mathcal{D}^{(1)}_{\text{test}}, \ldots, \mathcal{D}^{(N')}_{\text{test}}\}$ denotes a set of testing datasets. Notably, $\mathcal{T}_{\text{train}} \cap \mathcal{T}_{\text{test}} = \emptyset$, and datasets in $\mathcal{T}_{\text{train}}$ and $\mathcal{T}_{\text{test}}$ may originate from distinct distributions and domains. Following (Liu et al. 2024), We adopt a "normal few-shot" in-

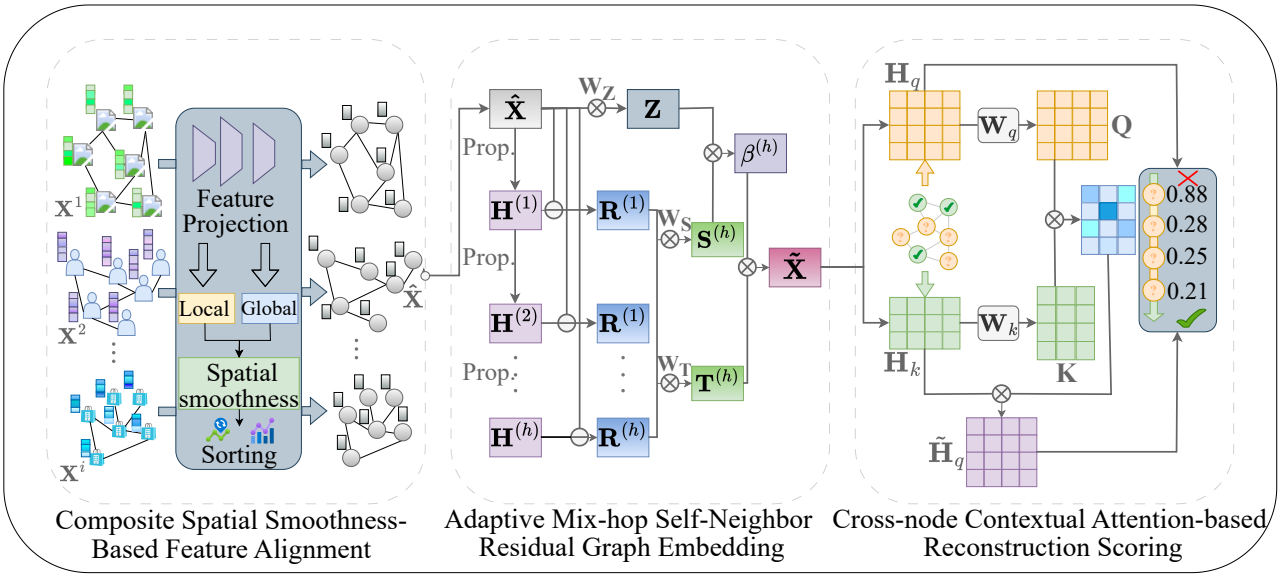


Figure 1: The overall pipeline of SAARCS, the proposed generalist GAD approach.

ference setting: for each $\mathcal{D}_{\text{test}}^{(i)}$, only a small number of n_k normal nodes ($n_k \ll n$) are accessible. The model f is then expected to predict the abnormality of the remaining nodes without retraining or fine-tuning.

Methodology

In this section, we present the general framework of our method, which can identify anomalies across different datasets without any fine-tuning and mainly consists of three modules: the Composite Spatial Smoothness Alignment Module, the Adaptive k-hop Self-neighbor Residual Module, and the Cross-Node Contextual Attention Reconstruction Scoring Module. Figure 1 illustrates the overall framework of the method. First, to align features from different datasets, we introduce composite spatial smoothness, which considers both local and global perspectives, and sort the features in the order of their sensitivity to anomalies. Next, through the Adaptive Hop Count Self-Neighborhood Residual Module, we automatically learn the residuals between nodes and their k-hop neighbors to construct node embeddings. Finally, considering the small-sample nature of some datasets, we construct the Cross-Node Contextual Attention Reconstruction Scoring Module. This module leverages cross-attention blocks to learn the reconstruction of query node embeddings based on contextual node embeddings, and then uses the drift distance between the original and reconstructed query embeddings to quantify the anomaly of each node.

Composite Spatial Smoothness-Based Feature Alignment

Graph data from diverse domains typically have different features, characterized by differences in dimensionality and unique meanings for each dimension. For example, features in a citation network usually consist of textual and meta-information associated with each paper, whereas in a social network, the features may be the profile of each user. Therefore, in the first step, we need to align the features into a shared feature space. To achieve this, we introduce the feature alignment module in SAARCS, consisting of two phases: feature projection, which aligns dimensionality, and Multi-scale feature sorting, which reorders features according to their smoothness characteristics.

Feature Projection. At the first step of SAARCS, we employ a feature projection block to unify the feature dimensionality of multiple graph datasets (Zhao et al. 2024).Specifically, given a feature matrix $\mathbf{X}^{(i)} \in \mathbb{R}^{n^{(i)} \times d^{(i)}}$ of $D^{(i)} \in T_{train} \cup T_{test}$, the feature projection is defined by a linear mapping:

$$\tilde{\mathbf{X}}^{(i)} \in \mathbb{R}^{n^{(i)} \times d_u} = \text{Proj}\left(\mathbf{X}^{(i)}\right) = \mathbf{X}^{(i)}\mathbf{W}^{(i)} \quad (1)$$

where $\tilde{\mathbf{X}}^{(i)}$ is the projected feature matrix for $D^{(i)}$, d_u is a predefined projected dimension shared across all datasets, and $\mathbf{W}^{(i)} \in \mathbb{R}^{d^{(i)} \times d_u}$ is a dataset-specific linear projection weight matrix. To maintain generality, $\mathbf{W}^{(i)}$ can be defined using commonly used dimensionality reduction approaches such as singular value decomposition(SVD) and principal component analysis(PCA).

Composite Spatial Smoothness-Based Feature Sorting. Although feature projection can align dimensionality, the semantic meaning of each projected feature across different datasets remains distinct. In this paper, we choose to align features based on their contributions to anomaly detection. To comprehensively capture anomaly features, we proposes a feature alignment method that integrates local neighborhood differences and global feature dispersion. High-frequency graph signals/heterogeneity have been proven to

play an important role in distinguishing anomalies. Therefore, we consider whether it is possible to unify them with a single metric to measure the contribution of each feature to GAD based on frequency/heterogeneity?

Considering its correlation to frequency and heterophily, in this paper, we select *Composite Spatial Smoothness* as the measure for contribution. First, the feature matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ in the input graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ (where n is the number of nodes and d is the number of feature dimensions) is standardized. For each feature dimension k, the feature is normalized to eliminate the influence of dimensions and put different feature dimensions on a unified scale and the normalized features are denoted as $\mathbf{X}'_{ik}$.

Next, two key indices are calculated respectively. On one hand, the smoothness s_k is calculated using the formula:

$$s_k = \frac{1}{|\varepsilon|} \sum_{(v_i,v_j)\in\varepsilon} (\mathbf{X}'_{ik} - \mathbf{X}'_{jk})^2 \tag{2}$$

This index characterizes the feature change between adjacent nodes. A higher value indicates a greater local difference and higher sensitivity to anomalies. On the other hand, the variance var_k is calculated as:

$$\mathrm{var}_k = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{X}'_{ik} - \bar{\mathbf{X}}'_k)^2 \tag{3}$$

which reflects the dispersion degree of features in the entire graph. A larger value implies a higher global dispersion and a greater likelihood of containing outlier features.

Subsequently, for each feature dimension, a composite spatial smoothness score is calculated through weighted fusion:

$$\mathrm{score}_k = \alpha \cdot s_k + (1-\alpha)\cdot \mathrm{var}_k \tag{4}$$

Here, $\alpha \in [0, 1]$ is a weight parameter that controls the relative importance of the local smoothness index and the global variance index. A larger value of the score indicates a higher sensitivity to anomalies.

Based on the above, given the projected features of different datasets, we can align their feature spaces by rearranging the permutation of features based on the descending order of score_k w.r.t. each projected feature. For all datasets, the feature in the first column is the one with the highest score_k, which deserves more attention by SAARCS; conversely, features with less contribution (i.e. lower score_k) are placed at the end. In this way, the GNN-based model can learn to filter graph signals with different spatial smoothness levels automatically and predict anomalies accordingly. During inference, spatial smoothness remains transferable under consistent alignment.

Adaptive Mix-hop Self-Neighbor Residual Graph Embedding

To jointly encode semantic and structural cues for every node, we introduce a GNN-based graph encoder after feature alignment; the resulting embeddings are subsequently fed to a downstream anomaly-scoring module to predict node abnormality. Nevertheless, a naive adoption of prevailing GNN architectures suffers from two intertwined limitations. First, their intrinsic low-pass filtering nature hinders the capture of high-frequency, heterophilic patterns that often accompany anomalies. Second, even GNNs specifically designed for graph anomaly detection predominantly emphasize node-level semantics while overlooking the affinity distribution within local subgraphs. Consequently, the encoder tends to overfit to dataset-specific semantic features and fails to learn the universal anomaly patterns that are shared across different datasets, which ultimately impairs its generalization capability.

To overcome the aforementioned limitations, we present an adaptive multi-hop residual graph encoder. Building upon the traditional self-neighbor residual paradigm, this module introduces a hop-wise attention mechanism that is decoupled from downstream tasks. It first extracts residual representations from neighbors at each hop and then dynamically assigns weights according to their contribution to anomaly detection. Consequently, the model amplifies high-frequency indicators of anomalies while preserving critical high-order structural patterns, enabling more accurate capture of multi-hop anomaly features of nodes.

Firstly, for the input graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \hat{\mathbf{X}})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges, and $\hat{\mathbf{X}} \in \mathbb{R}^{n\times d}$ is the node feature matrix. We perform multiple iterative propagations over the aligned feature matrix $\hat{\mathbf{X}}$, the neighbor feature representations $\mathbf{H}^h \in \mathbb{R}^{n\times d}$ of nodes at different hops $h \in \{1, 2, \ldots, k\}$ are extracted respectively.

$$\mathbf{H}^h = \tilde{\mathbf{A}}\mathbf{H}^{h-1}, \mathbf{H}^{\mathbf{0}} = \hat{\mathbf{X}} \tag{5}$$

where $\mathbf{H}^h$ is the propagated feature matrix at the h-th iteration, and $\tilde{\mathbf{A}}$ is the normalized adjacency matrix.

Then the residual information $\mathbf{R}^h = \mathbf{H}^h - \hat{\mathbf{X}}$ between nodes and their neighbors at the corresponding hops is calculated. To avoid conflicts with the parameter definitions in the third module, this module adopts an independent attention calculation system. The node feature matrix X is transformed into the query matrix $\mathbf{Z} = \hat{\mathbf{X}}\mathbf{W}_{\mathbf{Z}}$ through the learnable weight matrix $\mathbf{W}_{\mathbf{Z}} \in \mathbb{R}^{d\times d_a}$, where d_a is the dimension of the attention mechanism. Meanwhile, for each hop h, the learnable weight matrices $\mathbf{W}_{\mathbf{S}}, \mathbf{W}_{\mathbf{T}} \in \mathbb{R}^{d\times d_a}$ are used to transform the residual $\mathbf{R}^h$ into the key matrix $\mathbf{S}^h = \mathbf{R}^h\mathbf{W}_{\mathbf{S}}$ and the value matrix $\mathbf{T}^h = \mathbf{R}^h\mathbf{W}_{\mathbf{T}}$, respectively.

Based on the above, the hop-related attention scores are calculated as follows :

$$\beta^{(h)} = \mathrm{LeakyReLU}\left(\frac{\mathbf{Z}(\mathbf{S}^h)^\top}{\sqrt{d_a}}\right) \tag{6}$$

The LeakyReLU activation function is employed to enhance the model's sensitivity to key-hop information. Subsequently, the attention scores are normalized as follows to obtain the weights:

$$\hat{\beta}^{(h)} = \frac{\exp(\beta^{(h)})}{\sum_{h'=1}^{k}\exp(\beta^{(h')})}, \tag{7}$$

and the residual information of different hops is aggregated

by weighted summation as follows:

$$\tilde{\mathbf{X}} = \sum_{h=1}^{k} \hat{\beta}^{(h)} \mathbf{T}^{(h)}. \tag{8}$$

Finally, we obtain the enhanced node feature representation $\tilde{\mathbf{X}}$. This representation can adaptively highlight the multi-hop structural information valuable for anomaly detection, providing more discriminative feature inputs for subsequent anomaly scoring and detection tasks.

In summary, our graph encoder surpasses existing GNNs by employing adaptive residual operations that: (1) emphasize node-neighbor differences over ego features, enabling explicit local affinity modeling for cross-dataset anomaly detection; and (2) act as high-pass filters to capture abnormality-related high-frequency signals and multi-hop heterophily patterns.

Cross-node Contextual Attention-based Reconstruction Scoring Module

To leverage the limited normal samples (denoted as context nodes) for predicting the abnormality of remaining nodes (query nodes), we adopt an in-context learning module with a cross-attention mechanism for anomaly scoring in SAARCS. The core principle of this module is to reconstruct the embedding of each query node by fusing context node embeddings through a cross-attention block. The drift distance between the original and reconstructed embeddings of a query node then serves as an indicator of its abnormality.

Specifically, the embedding matrix is partitioned into two components based on row indices: context node embeddings $\mathbf{H}_k \in \mathbb{R}^{n_k \times d_e}$ and query node embeddings $\mathbf{H}_q \in \mathbb{R}^{n_q \times d_e}$. Then, a cross-attention block is employed to reconstruct each row of $\mathbf{H}_q$ via a linear combination of $\mathbf{H}_k$:

$$\mathbf{Q} = \mathbf{H}_q \mathbf{W}_q, \quad \mathbf{K} = \mathbf{H}_k \mathbf{W}_k \tag{9}$$

$$\tilde{\mathbf{H}}_q = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_e}}\right)\mathbf{H}_k \tag{10}$$

where $\mathbf{Q} \in \mathbb{R}^{\mathbf{n_q} \times \mathbf{d_e}}$ and $\mathbf{K} \in \mathbb{R}^{\mathbf{n_k} \times \mathbf{d_e}}$ are the query and key matrices respectively, $\mathbf{W}_q$ and $\mathbf{W}_k$ are learnable parameters, and $\tilde{\mathbf{H}}_q$ is the reconstructed query embedding matrix. Notably, unlike conventional cross-attention blocks that introduce a separate value matrix $\mathbf{V}$, this design directly multiplies the attention matrix with $\mathbf{H}_k$. This ensures $\tilde{\mathbf{H}}_q$ resides in the same embedding space as $\mathbf{H}_q$ and $\mathbf{H}_k$. Thanks to this property, given a query node v_i, we can calculate its anomaly score $f(v_i)$ by computing the L2 distance between its query embedding vector $\mathbf{H}_{q_i}$ and the corresponding reconstructed query embedding vector $\tilde{\mathbf{H}}_q$:

$$f(v_i) = d\left(\mathbf{H}q_i, \tilde{\mathbf{H}}q_i\right) = \sqrt{\sum_{j=1}^{d_e}\left(\mathbf{H}q_{ij} - \tilde{\mathbf{H}}q_{ij}\right)^2} \tag{11}$$

The cross-node contextual cttention-based reconstruction scoring framework is grounded in the assumption that normal nodes can be linearly reconstructed from context embeddings (yielding small drift distances), while anomalies exhibit larger reconstruction errors due to pattern divergence. Consequently, this module not only significantly enhances anomaly discrimination but also improves the model's few-shot learning capability.

Experiment

Experimental Setup

Datasets. To evaluate the generalizability of GAD models, we adopt a cross-domain experimental design where models are trained on one set of graph datasets and tested on entirely different datasets,including citation networks, social network and e-commerce review network. In light of this, we train models on the largest representative dataset from each domain (social, citation, e-commerce) and evaluate performance on all remaining datasets. The training datasets T_{train} comprise PubMed (Sen et al. 2008), Flickr (Tang and Liu 2009), Questions (Platonov et al. 2023), and YelpChi (McAuley and Leskovec 2013), while the testing datasets T_{test} consist of Cora (Sen et al. 2008), CiteSeer (Giles, Bollacker, and Lawrence 1998), Weibo (Zhao et al. 2020), Reddit (Kumar, Zhang, and Leskovec 2019), Amazon (Shchur et al. 2018), ACM (Tang et al. 2008), BlogCatalog (Ding et al. 2019) and Facebook (Xu et al. 2022). All the datasets contain either injected anomalies or real-world anomalies to assess model robustness across different anomaly types.

Baselines. We evaluate SAARCS against 12 representative baseline methods, systematically categorized into two groups. The first group encompasses conventional graph neural network architectures, including GCN (Kipf 2016), GAT (Velickovic et al. 2017), and GIN (Xu et al. 2018). The second group comprises state-of-the-art specialized approaches for homogeneous / heterogeneous graphs, including DOMINANT (Ding et al. 2019), CoLA (Liu et al. 2021b), BWGNN (Tang et al. 2022), GHRN (Gao et al. 2023), GRADATE (Duan et al. 2023), GAD-NR (Roy et al. 2024), ADA-GAD (He et al. 2024), AD-GCL (Xu et al. 2025), and ARC (Liu et al. 2024). Detailed implementations and configurations of all baseline methods are documented in Appendix.

Evaluation and Implementation. Following (Tang et al. 2023; Liu et al. 2024), we adopt AUROC and AUPRC as evaluation metrics for graph anomaly detection (GAD), reporting mean values with standard deviations across five independent trials. SAARCS is jointly trained on all T_{train} datasets and evaluated on each T_{test} dataset using in-context learning (default n_k =10). Feature standardization is achieved through learnable/random projection adapters between raw features and model inputs. Hyperparameter optimization employs random search with identical parameter sets applied across all test datasets to preserve generalizability, avoiding dataset-specific tuning that could compromise the generalist evaluation objective.

Model Training. For SAARCS optimization on training datasets T_{train}, we implement a marginal cosine similarity loss function that simultaneously minimizes normal node drift distances while maximizing abnormal node drift distances. The implementation involves: (1) randomly selecting n_k normal nodes as context nodes, (2) sampling balanced

sets of normal and abnormal query nodes, and (3) computing the sample-level loss for each query node v_i based on its original embedding $\mathbf{H}_{q_i}$, reconstructed embedding $\hat{\mathbf{H}}_{q_i}$, and anomaly label y_i, the sample-level loss function can be written by:

$$\mathcal{L} = \begin{cases} 1 - \cos\left(\mathbf{H}q_i, \tilde{\mathbf{H}}q_i\right), & \mathbf{if}\, \mathbf{y}_i = 0 \\ \max\left(0, \cos\left(\mathbf{H}q_i, \tilde{\mathbf{H}}q_i\right) - \epsilon\right), & \mathbf{if}\, \mathbf{y}_i = 1 \end{cases} \tag{12}$$

Main Results and Analysis

We evaluate SAARCS against 12 state-of-the-art baselines using AUROC metrics across eight benchmark datasets, with detailed results presented in Table 1. To further validate our approach, we conduct extensive ablation studies and effectiveness analyses documented in the Appendix, which include AUPRC performance scores, comparative AUROC under different spatial-feature alignment strategies, dataset-specific analysis of optimal aggregation hops, and visualization of attention weight distributions between context and query nodes. Based on these comprehensive experiments, we derive the following key observations:

We can intuitively observe that SAARCS establishes new state-of-the-art performance in the generalist GAD setting without requiring fine-tuning, achieving top results on 6/8 datasets while maintaining competitive performance on the remaining two. It outperforms the runner-up approach by 1.24%, 3.26%, 0.29%, 3.67%, 2.06%, and 1.59% on six datasets, respectively, while remaining competitive on the remaining two. Compared with traditional GNNs such as GCN, GAT, and GIN, SAARCS achieves a decisive lead. The reason is that these conventional graph neural networks essentially function as low-pass filters, which struggle to capture high-frequency and heterogeneous patterns that are crucial for anomaly detection. In contrast, SAARCS adaptively assigns residual weights between the ego node and its multi-hop neighbors. The residual operation can be viewed as a high-pass filter that emphasizes differences among nodes rather than the ego node's own features. Consequently, the learned embeddings are able to comprehensively encode anomalous signals across the graph, thereby further boosting model performance.

SAARCS demonstrates consistent superiority over state-of-the-art methods, outperforming competitors on nearly all datasets with only two minor exceptions. While prior models are tailored to individual datasets, SAARCS aligns features across different datasets before embedding learning and prioritizes anomaly-sensitive dimensions via spatial smoothness, conferring strong universality. Even compared to ARC - the only existing universal GAD baseline - SAARCS maintains superior performance. This advantage stems from composite spatial smoothness, which simultaneously captures sensitivity to both local topological discrepancies and global distributional dispersion, together with the automatic learning of the most informative hop distances. These mechanisms render SAARCS more flexible when confronting diverse anomaly distributions across datasets and endow it with markedly stronger generalization capacity.

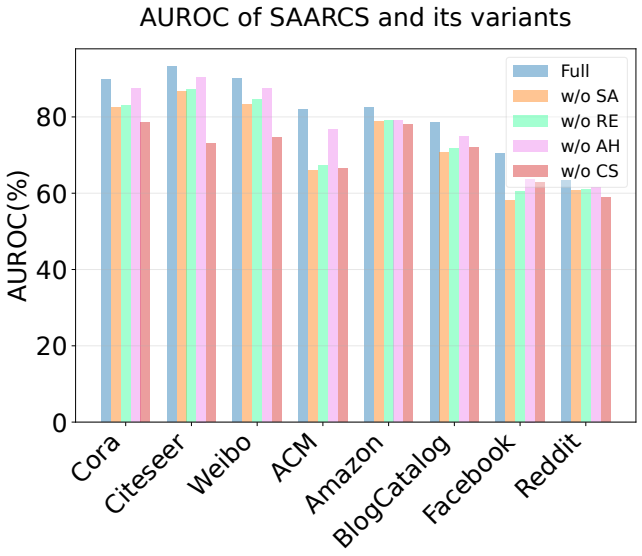


Figure 2: Ablation study results

Ablation Study

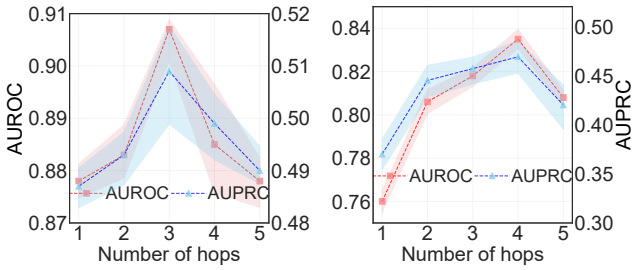
We conduct an ablation study presented in Figure 2 to evaluate the contribution of each SAARCS component by designing three variants: (1) **w/o SA**: replaces composite spatial feature alignment with random projection; (2)**w/o RE**: using GCN to replace residual graph encoder; (3)**w/o AH**: substitutes the adaptive mix-hop Self-Neighbor residual graph embedding with a fixed optimal hop-count self-neighbor residual block; (4) **w/o CS**: employs binary classification instead of cross-node contextual attention-based reconstruction scoring. The experimental results conclusively demonstrate the significant contributions of each SAARCS component. Specifically: Ablating the composite spatial smoothness alignment module causes AUROC decreases of 2%–15% across eight datasets, confirming its capability to identify features sensitive to anomaly detection; The residual operation also exerts a substantial influence on final performance.Replacing the adaptive multi-hop residual blocks with fixed-hop counterparts leads to an average 5% AUROC reduction, proving that adaptive hop selection dynamically adapts to graph topology to preserve critical anomaly signals while improving generalization; Removing the cross-node contextual attention scoring module results in 2%–8% performance degradation, verifying its effectiveness in utilizing limited contextual nodes for anomalous node prediction.

Impact of Multi-Hop Aggregation with Varying Hop Counts

In graph anomaly detection, the hop count determines the neighborhood aggregation scope, directly shaping the model's perception of both topology and features. Optimal hop counts vary across datasets; therefore, we adopt an adaptive multi-hop design and evaluate it on multiple benchmarks. As shown in Figure 3, on Cora the peak AUROC is achieved at $h = 3$. An insufficient hop count re-

Method	Cora	Citeseer	Weibo	Reddit	Amazon	BlogCatalog	ACM	Facebook
GCN	59.64 ± 8.30	60.27 ± 8.11	76.64 ± 17.69	50.43 ± 4.41	46.63 ± 3.47	47.32 ± 2.79	42.51 ± 4.68	29.51 ± 4.86
GAT	50.06 ± 2.65	51.59 ± 3.49	53.05 ± 7.48	51.71 ± 4.04	50.52 ± 17.22	52.42 ± 3.69	43.71 ± 2.51	51.88 ± 2.16
GIN	60.78 ± 0.91	55.42 ± 6.22	64.37 ± 1.09	41.75 ± 4.73	82.11 ± 0.15	41.14 ± 2.59	39.21 ± 2.97	39.62 ± 3.88
Dominant	66.53 ± 1.15	69.47 ± 2.02	89.88 ± 0.32	50.05 ± 4.92	48.94 ± 2.69	60.87 ± 3.79	51.36 ± 6.98	51.01 ± 0.78
Cola	63.29 ± 3.88	62.84 ± 9.52	26.27 ± 5.64	52.81 ± 6.69	47.40 ± 7.97	64.96 ± 1.79	55.51 ± 6.17	32.99 ± 11.68
BWGNN	76.71 ± 0.91	49.58 ± 6.22	71.76 ± 1.09	45.42 ± 4.73	76.12 ± 0.15	63.24 ± 2.59	51.44 ± 7.97	52.34 ± 3.88
GHRN	81.72 ± 0.07	45.22 ± 3.42	72.35 ± 3.35	51.73 ± 2.14	77.28 ± 0.13	61.67 ± 2.52	53.21 ± 3.64	49.92 ± 2.29
GRADATE	87.55 ± 0.39	81.38 ± 1.82	87.12 ± 0.25	52.36 ± 1.53	71.33 ± 1.04	66.98 ± 1.57	61.23 ± 4.79	64.13 ± 1.96
GAD-NR	87.55 ± 0.39	59.52 ± 0.83	87.71 ± 0.18	57.99 ± 0.33	70.63 ± 2.19	74.92 ± 1.65	69.47 ± 3.75	68.78 ± 1.66
ADA-GAD	84.73 ± 4.73	60.12 ± 6.80	$\mathbf{98.44 \pm 0.08}$	56.89 ± 2.75	83.25 ± 0.72	69.21 ± 4.59	63.75 ± 3.96	65.49 ± 3.97
AD-GCL	$\mathbf{92.37 \pm 0.15}$	89.52 ± 0.83	87.71 ± 0.18	57.99 ± 0.33	70.63 ± 2.19	68.92 ± 1.65	65.47 ± 5.34	65.88 ± 2.87
ARC	87.45 ± 0.74	90.95 ± 0.59	88.85 ± 0.14	60.04 ± 0.69	80.67 ± 1.81	74.76 ± 0.06	79.88 ± 0.28	67.56 ± 1.60
SAARCS	90.77 ± 0.81	$\mathbf{92.19 \pm 0.46}$	93.15 ± 0.21	$\mathbf{63.30 \pm 1.01}$	$\mathbf{83.54 \pm 1.32}$	$\mathbf{78.59 \pm 0.84}$	$\mathbf{81.94 \pm 1.03}$	$\mathbf{70.37 \pm 1.36}$

Table 1: AUROC on eight GAD datasets(in percent, mea±std). In each set of comparative results, the best performing method is in bold and the second-best performance is indicated with underlining.



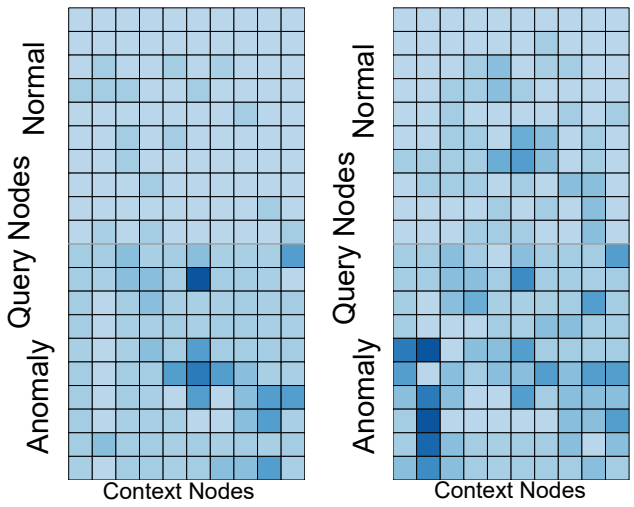
(a) Cora (b) Amazon

Figure 3: Optimal Hop-Count Results Across Different Datasets.

stricts aggregation to first-order neighbors, limiting the capture of global structural anomalies, whereas an excessive hop count introduces irrelevant neighbors and causes feature confusion, leading to significant performance degradation. On Amazon, the best performance occurs at $h = 4$: when $h = 1$, the model relies solely on local information and struggles to detect anomalies with complex relations; as h increases to 2–3, coverage of critical neighborhood structures expands and performance rises steadily; however, at $h = 5$, redundant information dominates and performance drops markedly. These observations confirm the necessity of hop-aware strategies across datasets, and SAARCS's adaptive multi-hop residual aggregation effectively identifies the optimal neighborhood residuals for each graph, thereby capturing structural anomalies with high fidelity.

Visualization

Figure 4 visualizes the attention weight distribution in SAARCS's cross-node contextual attention module, revealing distinct allocation patterns: (a) Anomalies selectively attend to only 1-2 context nodes, resulting in peripheral reconstructed embeddings; meanwhile, normal nodes distribute attention uniformly, causing their embeddings to cluster near the centroid. (b) Normal nodes follow two consistent allocation patterns, each associating with multiple context nodes. These findings demonstrate the module's adaptive capacity to diverse normal/anomaly distributions, which fundamentally contributes to the model's generalization capability.



(a) Weibo (b) Facebook

Figure 4: Visualization results.

Conclusion

This paper focuses on the emerging paradigm of universal graph anomaly detection and proposes an "all-in-one universal" detection model that does not require adjustment for specific datasets. The SAARCS framework we propose is based on an adaptive multi-hop contextual learning mechanism and can achieve real-time anomaly detection with only a small number of normal nodes as references. Comprehensive multi-domain experiments validate SAARCS's superior detection capability and cross-domain generalizability compared to state-of-the-art methods. While demonstrating promising results, SAARCS is currently confined to static graph structures and does not account for temporal evolution of graph data, extending it to dynamic topologies remains an important direction for future work. Moreover, the integration of dual attention mechanisms enhances representational capacity but also increases model complexity and computational cost, subsequent research will seek to balance detection performance with computational efficiency.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62572420, the Shandong Province Key R&D Program under Grant 2024CXGC010801, Yantai Science and Technology Plan Project under Grant 2023ZDCX001, Yantai Smart City Innovation Lab Project, School and Locality Integration Development Project of Yantai City(2022).

References

Ding, K.; Li, J.; Agarwal, N.; and Liu, H. 2021. Inductive anomaly detection on attributed networks. In *Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence*, 1288–1294.

Ding, K.; Li, J.; Bhanushali, R.; and Liu, H. 2019. Deep anomaly detection on attributed networks. In *Proceedings of the SIAM International Conference on Data Mining*, 594–602.

Dong, K.; Mao, H.; Guo, Z.; and Chawla, N. V. 2024. Universal link predictor by in-context learning on graphs. *arXiv preprint arXiv:2402.07738*.

Dou, Y.; Liu, Z.; Sun, L.; Deng, Y.; Peng, H.; and Yu, P. S. 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 315–324.

Duan, J.; Wang, S.; Zhang, P.; Zhu, E.; Hu, J.; Jin, H.; Liu, Y.; and Dong, Z. 2023. Graph anomaly detection via multi-scale contrastive learning networks with augmented view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 7459–7467.

Duan, M.; Zheng, T.; Gao, Y.; Wang, G.; Feng, Z.; and Wang, X. 2024. Dga-gnn: Dynamic grouping aggregation gnn for fraud detection. In *Proceedings of the AAAI conference on Artificial Intelligence*, 11820–11828.

Fan, H.; Zhang, F.; and Li, Z. 2020. Anomalydae: Dual autoencoder for anomaly detection on attributed networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 5685–5689.

Gao, Y.; Wang, X.; He, X.; Liu, Z.; Feng, H.; and Zhang, Y. 2023. Addressing heterophily in graph anomaly detection: A perspective of graph spectrum. In *Proceedings of the ACM Web Conference*, 1528–1538.

Giles, C. L.; Bollacker, K. D.; and Lawrence, S. 1998. CiteSeer: An automatic citation indexing system. In *Proceedings of the Third ACM Conference on Digital Libraries*, 89–98.

He, J.; Xu, Q.; Jiang, Y.; Wang, Z.; and Huang, Q. 2024. Ada-gad: Anomaly-denoised autoencoders for graph anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 8481–8489.

Hendel, R.; Geva, M.; and Globerson, A. 2023. In-context learning creates task vectors. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 9318–9333.

Huang, M.; Liu, Y.; Ao, X.; Li, K.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2022. Auc-oriented graph neural network for fraud detection. In *Proceedings of the ACM Web Conference*, 1311–1321.

Huang, Q.; Ren, H.; Chen, P.; Kržmanc, G.; Zeng, D.; Liang, P. S.; and Leskovec, J. 2023. Prodigy: Enabling in-context learning over graphs. *Advances in Neural Information Processing Systems*, 16302–16317.

Jin, M.; Koh, H. Y.; Wen, Q.; Zambon, D.; Alippi, C.; Webb, G. I.; King, I.; and Pan, S. 2024. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10466–10485.

Kipf, T. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*.

Kong, X.; Zhang, W.; Wang, H.; Hou, M.; Chen, X.; Yan, X.; and Das, S. K. 2024. Federated graph anomaly detection via contrastive self-supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*, 7931–7944.

Kumar, S.; Zhang, X.; and Leskovec, J. 2019. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1269–1278.

Li, A.; Qin, Z.; Liu, R.; Yang, Y.; and Li, D. 2019. Spam review detection with graph convolutional networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2703–2711.

Li, C.-T.; Tsai, Y.-C.; Chen, C.-Y.; and Liao, J. C. 2024. Graph neural networks for tabular data learning: A survey with taxonomy and directions. *ACM Computing Surveys*, 1–51.

Li, J.; Dani, H.; Hu, X.; and Liu, H. 2017. Radar: Residual analysis for anomaly detection in attributed networks. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2152–2158.

Li, J.; Wu, R.; Zhu, Y.; Zhang, H.; Chen, L.; and Zheng, Z. 2025. Are Large Language Models In-Context Graph Learners? *arXiv preprint arXiv:2502.13562*.

Li, R.; Liu, Z.; Ma, Y.; Yang, D.; and Sun, S. 2022. Internet financial fraud detection based on graph learning. *IEEE Transactions on Computational Social Systems*, 1394–1401.

Liu, M.; Gao, H.; and Ji, S. 2020. Towards deeper graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 338–348.

Liu, Y.; Ao, X.; Qin, Z.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2021a. Pick and choose: a GNN-based imbalanced learning approach for fraud detection. In *Proceedings of the Web Conference*, 3168–3177.

Liu, Y.; Li, S.; Zheng, Y.; Chen, Q.; Zhang, C.; and Pan, S. 2024. Arc: A generalist graph anomaly detector with in-context learning. *Advances in Neural Information Processing Systems*, 50772–50804.

Liu, Y.; Li, Z.; Pan, S.; Gong, C.; Zhou, C.; and Karypis, G. 2021b. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2378–2392.

Liu, Z.; Yang, D.; Wang, Y.; Lu, M.; and Li, R. 2023. EGNN: Graph structure learning based on evolutionary computation helps more in graph neural networks. *Applied Soft Computing*, 110040.

Luo, X.; Wu, J.; Beheshti, A.; Yang, J.; Zhang, X.; Wang, Y.; and Xue, S. 2022. Comga: Community-aware attributed graph anomaly detection. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*, 657–665.

Ma, X.; Wu, J.; Xue, S.; Yang, J.; Zhou, C.; Sheng, Q. Z.; Xiong, H.; and Akoglu, L. 2021. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 12012–12038.

McAuley, J. J.; and Leskovec, J. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd International Conference on World Wide Web*, 897–908.

Motie, S.; and Raahemi, B. 2024. Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications*, 122156.

Pan, J.; Liu, Y.; Zheng, X.; Zheng, Y.; Liew, A. W.-C.; Li, F.; and Pan, S. 2025. A label-free heterophily-guided approach for unsupervised graph fraud detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 12443–12451.

Pan, J.; Liu, Y.; Zheng, Y.; and Pan, S. 2023. Prem: A simple yet effective approach for node-level graph anomaly detection. In *IEEE International Conference on Data Mining*, 1253–1258.

Pang, G.; Shen, C.; Cao, L.; and Hengel, A. V. D. 2021. Deep learning for anomaly detection: A review. *ACM Computing Surveys*, 1–38.

Peng, Z.; Luo, M.; Li, J.; Liu, H.; Zheng, Q.; et al. 2018. ANOMALOUS: A joint modeling approach for anomaly detection on attributed networks. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 3513–3519.

Platonov, O.; Kuznedelev, D.; Diskin, M.; Babenko, A.; and Prokhorenkova, L. 2023. A critical look at the evaluation of GNNs under heterophily: Are we really making progress? In *Proceedings of the 11th International Conference on Learning Representations*.

Qiao, H.; and Pang, G. 2023. Truncated affinity maximization: One-class homophily modeling for graph anomaly detection. *Advances in Neural Information Processing Systems*, 49490–49512.

Roy, A.; Shu, J.; Li, J.; Yang, C.; Elshocht, O.; Smeets, J.; and Li, P. 2024. Gad-nr: Graph anomaly detection via neighborhood reconstruction. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 576–585.

Sen, P.; Namata, G.; Bilgic, M.; Getoor, L.; Galligher, B.; and Eliassi-Rad, T. 2008. Collective classification in network data. *AI Magazine*, 93–106.

Shchur, O.; Mumme, M.; Bojchevski, A.; and Günnemann, S. 2018. Pitfalls of graph neural network evaluation. *arXiv e-prints*, arXiv–1811.

Tang, J.; Hua, F.; Gao, Z.; Zhao, P.; and Li, J. 2023. Gadbench: Revisiting and benchmarking supervised graph anomaly detection. *Advances in Neural Information Processing Systems*, 29628–29653.

Tang, J.; Li, J.; Gao, Z.; and Li, J. 2022. Rethinking graph neural networks for anomaly detection. In *International Conference on Machine Learning*, 21076–21089.

Tang, J.; Zhang, J.; Yao, L.; Li, J.; Zhang, L.; and Su, Z. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 990–998.

Tang, L.; and Liu, H. 2009. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 817–826.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y.; et al. 2017. Graph attention networks. In *arXiv preprint arXiv:1710.10903*.

Wang, Q.; Pang, G.; Salehi, M.; Buntine, W.; and Leckie, C. 2023. Cross-domain graph anomaly detection via anomaly-aware contrastive alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 4676–4684.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How powerful are graph neural networks? In *International Conference on Learning Representations*.

Xu, Y.; Peng, Z.; Shi, B.; Hua, X.; Dong, B.; Wang, S.; and Chen, C. 2025. Revisiting Graph Contrastive Learning on Anomaly Detection: A Structural Imbalance Perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 12972–12980.

Xu, Z.; Huang, X.; Zhao, Y.; Dong, Y.; and Li, J. 2022. Contrastive attributed network anomaly detection with data augmentation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 444–457.

Yang, X.; Zhao, X.; and Shen, Z. 2025. A generalizable anomaly detection method in dynamic graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 22001–22009.

Zhao, H.; Chen, A.; Sun, X.; Cheng, H.; and Li, J. 2024. All in one and one for all: A simple yet effective method towards cross-domain graph pretraining. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4443–4454.

Zhao, T.; Deng, C.; Yu, K.; Jiang, T.; Wang, D.; and Jiang, M. 2020. Error-bounded graph anomaly loss for gnns. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 1873–1882.

Zheng, L.; Jing, B.; Li, Z.; Tong, H.; and He, J. 2024. Heterogeneous contrastive learning for foundation models and beyond. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 6666–6676.

Zheng, Y.; Jin, M.; Liu, Y.; Chi, L.; Phan, K. T.; and Chen, Y.-P. P. 2021. Generative and contrastive self-supervised learning for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 12220–12233.