Dual Encoder Contrastive Learning with Augmented Views for Graph Anomaly Detection

Nannan Wu [1,], **Hongdou Dong** [1,], **Wenjun Wang** [1,2] and **Yiming Zhao** [1]
[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]Yazhou Bay Innovation Institute, Hainan Tropical Ocean University, China
{nannan.wu, hongdou_dong, wjwang, zhaoym}@tju.edu.cn

Abstract

Graph anomaly detection (GAD), which aims to identify patterns that deviate significantly from normal nodes in attributed networks, is widely used in financial fraud, cybersecurity, and bioinformatics. The paradigms of jointly optimizing contrastive learning and reconstruction learning have shown significant potential in this field. However, when using GNNs as an encoder, it still faces the problem of over-smoothing, and it is difficult to effectively capture the fine-grain topology information of the graph. In this paper, we introduce an innovative approach: Dual Encoder Contrastive Learning with Augmented Views for Graph Anomaly Detection, named DECLARE. Specifically, the dual encoder integrates the strengths of GNNs and Graph Transformers to learn graph representation from multiple perspectives comprehensively. Although contrastive learning enhances the model's ability to learn discriminative features, it cannot directly identify anomalous patterns. To address this, the reconstruction module independently reconstructs graph structures and attributes, helping the model focus on learning the normal patterns of both structure and attributes. Extensive experimental results demonstrate that DECLARE outperforms state-of-the-art baselines across six benchmark datasets.

1 Introduction

An attribute network is a graph structure in which both nodes and edges encapsulate topological information and incorporate rich attribute data, enhancing the depth and context of the graph mining and analysis. Such networks are prevalent in a variety of domains, including social networks [Latah, 2020; Zhang *et al.*, 2019], financial networks [Motie and Raahemi, 2024], and recommendation systems [Ying *et al.*, 2018; Yu *et al.*, 2022]. A critical aspect of analyzing these networks is anomaly detection, which aims to identify instances that significantly deviate from the majority of the network's elements. Graph anomaly detection (GAD) plays a crucial role in numerous applications, including fraud detection and social spam detection [Ding *et al.*, 2019; Ma *et al.*, 2021].

The shallow anomaly detection methods, such as AMEN [Perozzi and Akoglu, 2016] and Radar [Li *et al.*, 2017], have been widely used in various applications, particularly for tasks where domain expertise can aid in crafting effective features. However, these approaches have notable limitations, especially when applied to complex and high-dimensional data. This challenge has led to a shift towards deep learning techniques that automatically extract meaningful patterns and representations. [Ding *et al.*, 2019] introduced an unsupervised autoencoder-based method that effectively detects abnormal patterns by leveraging reconstruction errors. Building on this, [Liu *et al.*, 2021] pioneered a contrastive learning framework for graph anomaly detection, which effectively utilizes local contextual information as supervision. Further advancements have been made by those who employed patch-level and context-level contrastive learning using two GNN-based models. Currently, models for graph anomaly detection ([Zheng *et al.*, 2021; Zhang *et al.*, 2022a]), which combine both reconstruction-based and contrastive-based learning techniques, have attracted significant attention due to their effectiveness in identifying anomalies.

However, this joint paradigm also has some limitations. The contrastive learning approach relies on fixed-size subgraph sampling, primarily focusing on local information while neglecting the global structure. This results in incomplete subgraph capture and the potential for abnormal nodes to undermine the reliability of neighborhood information. Additionally, the current paradigm mainly emphasizes attribute reconstruction while disregarding topological information, weakening the detection capability for topological anomalies and failing to capture the complex global relationships within the graph structure effectively. Although Graph Convolutional Networks (GCNs) [Kipf and Welling, 2016] can model both attribute features and structural information effectively, the learned representations are not always well-suited for anomaly detection. Furthermore, the lack of effective dependency modeling between distant nodes remains a significant gap in existing graph contrastive learning methods, limiting their ability to capture long-range interactions.

To address these limitations, we propose a novel method for graph anomaly detection called DECLARE (**D**ual **E**ncoder **C**ontrastive **L**earning with **A**ugmented Views for G**r**aph Anomaly D**e**tection). DECLARE integrates a dual encoder framework with multi-view contrastive learning to

detect anomalies in attributed networks. Specifically, our approach leverages a dual encoder architecture comprising a Graph Neural Network (GNN) and a Graph Transformer (GT). This dual encoder framework enables us to learn more expressive representations of both structural and attribute information within the graph. The GNN encoder focuses on learning node-level features by capturing local neighborhood information, while the Graph Transformer encoder models long-range dependencies and captures global interactions using self-attention mechanisms. This complementary encoding allows for a holistic representation of the graph that incorporates both local and global structural insights. In contrast to previous joint learning paradigms that focus solely on attribute reconstruction, our method introduces a reconstruction module that simultaneously reconstructs both structural and attribute information. This dual reconstruction process enhances the model's ability to detect a wide range of anomalies, including those related to graph structure and node attributes. Finally, an anomaly score is computed for each node based on the learned representations, facilitating the identification of anomalous nodes in the graph. Through this approach, DECLARE effectively addresses the challenges posed by previous methods, providing a more comprehensive and accurate solution for anomaly detection in attributed networks. In summary, the key contributions of this work are:

- We introduce DECLARE, a novel framework that combines a dual encoder architecture using GNN and Graph Transformer to learn expressive and informative representations of graph data, which has not been thoroughly explored in the field of Graph Contrastive Learning.
- Design a novel dual reconstruction approach that simultaneously reconstructs both graph structures and attribute information, thereby enhancing the model's anomaly detection capabilities and overall performance.
- We demonstrate the effectiveness of DECLARE through extensive experiments on benchmark datasets, showing significant improvements over state-of-the-art methods in detecting various types of graph anomalies.

2 Related Work

2.1 Graph Anomaly Detection

[Perozzi and Akoglu, 2016] proposes normality, a measure combining structure and attributes to evaluate neighborhood consistency and separability, and introduces AMEN, which outperforms traditional methods in anomaly detection on attributed graphs. [Li *et al.*, 2017] addresses anomaly detection in attributed networks by proposing a framework that models residuals of attribute coherence with network structure, effectively identifying anomalies without prior knowledge. Recently, with advancements in deep learning, Graph Neural Networks (GNNs) have been widely adopted for anomaly detection on graphs, often coupled with diverse learning paradigms to enhance their effectiveness. [Ding *et al.*, 2019] introduced a framework leveraging GCNs and autoencoders to detect anomalies by jointly evaluating reconstruction errors in the adjacency and attribute matrices, offering a solution to challenges like network sparsity and data nonlinearity.

Contrastive learning approaches often utilize the similarity between target nodes and their surrounding context for model pre-training. Building on this idea, [Liu *et al.*, 2021] developed a contrastive self-supervised learning framework specifically designed for graph anomaly detection. This framework effectively models the interactions between nodes and their subgraphs to identify anomalies. [Hu *et al.*, 2023] introduced a method for detecting anomalous nodes through subgraph-aligned contrastive learning across multiple graph views. Similarly, [Duan *et al.*, 2023a] leveraged contrastive learning to compute anomaly scores by exploring multi-scale relationships, including node-to-node, node-to-subgraph, and subgraph-to-subgraph interactions, between the original graph and its augmented version.

2.2 Graph Augmentation

Existing graph augmentation techniques encompass a variety of strategies. [You *et al.*, 2020] propose a node-dropping strategy for graph augmentation, while [Qiu *et al.*, 2020] and [Zhang *et al.*, 2023] focus on edge perturbation techniques. [Zhu *et al.*, 2021a] and [Zhang *et al.*, 2022b] suggest attribute masking approaches, and [Hassani and Khasahmadi, 2020] advocate for subgraph extraction. [Zhu *et al.*, 2020] introduce GRACE, which employs random edge perturbation and node feature masking to generate two augmented graph views. Subsequently, [Zhu *et al.*, 2021b] extend this idea with GCA, an adaptive augmentation strategy that combines structural and attribute information. [Wei *et al.*, 2023] develop GCS, which further integrates structural and semantic data for adaptive graph augmentation. [Tan *et al.*, 2024] propose CI-GCL, a framework designed to preserve graph community structures during augmentation. Despite their success, these methods often disrupt semantic relationships, potentially introducing biases into the learning process.

3 Preliminaries

In this section, we present the notations, formal definitions, and the problem statement that form the conceptual and mathematical foundation of our work.

Notaions. **(Attributed Network)** Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ denote an attributed network, where $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ represents the set of nodes and $\mathcal{E}$ is the set of edges. Here, $n = |\mathcal{V}|$ denotes the number of the nodes in the $\mathcal{G}$. Each node $v_i \in \mathcal{V}$ is associated with a feature vector $x_i \in \mathbb{R}^m (i = 1, 2, ..., n)$, where m is the dimensionality of the feature space. The attribute matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ contains the feature vectors of all nodes. Additionally, the adjacency matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is used to represent the graph structure, where $\mathbf{A}_{ij} = 1$ if there is an edge between nodes v_i and v_j, otherwise $\mathbf{A}_{ij} = 0$.

Definition 1. **(Encoders)** To effectively capture the complex structural and semantic patterns in $\mathcal{G}$, we utilize two complementary encoders: a Graph Neural Network (GNN) Encoder and a Graph Transformer (GTR) Encoder. The GNN encoder, denoted as $GNN(\cdot) \in \mathbb{R}^{n \times d}$, is used for localized aggregation, while the Graph Transformer encoder $GTR(\cdot) \in \mathbb{R}^{n \times d}$ captures long-range dependencies, where d represents the dimensionality of the hidden embeddings.

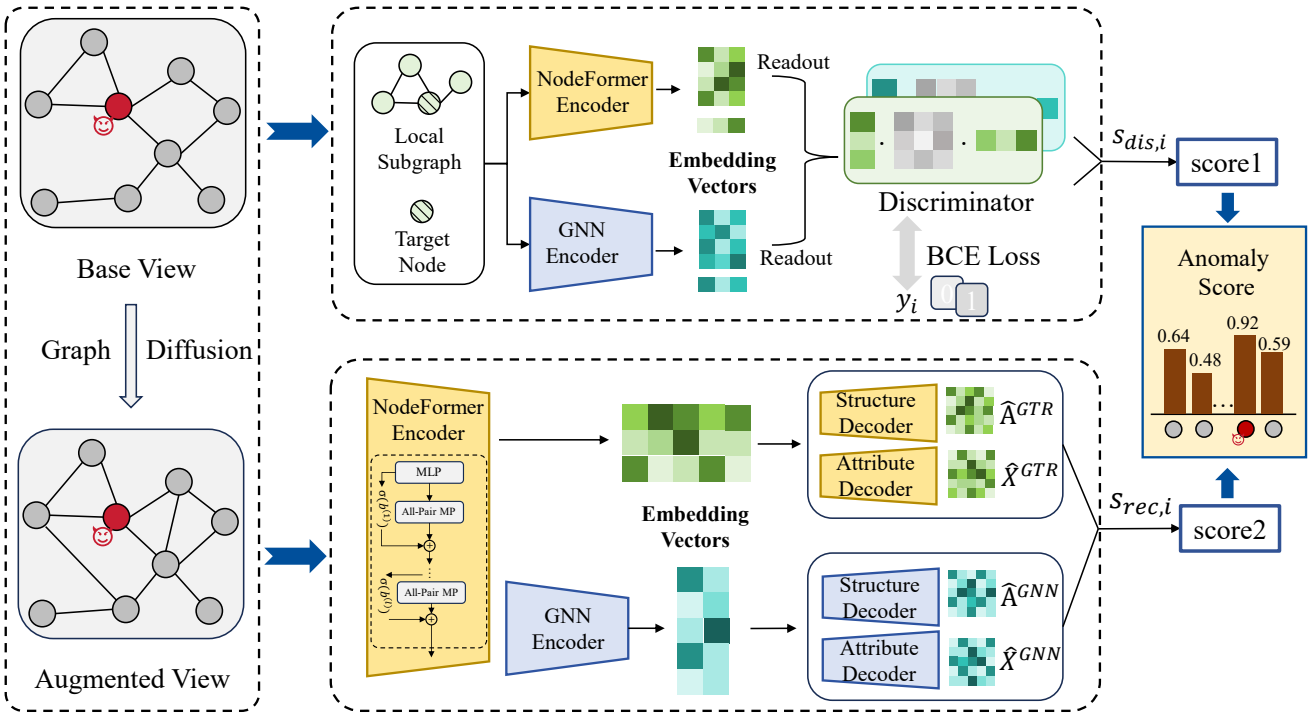


Figure 1: The proposed model consists of four key modules. The Graph Enhancement Module generates two views—base and augmented—through graph diffusion, enriching the graph's structural information. The Graph Contrastive Learning Module creates positive and negative pairs from these views, learning discriminative node representations using contrastive loss. The Graph Reconstruction Module reconstructs node attributes and graph structures, using reconstruction errors to identify anomalies. Finally, the Anomaly Scoring Module calculates anomaly scores based on reconstruction errors and contrastive learning outcomes, effectively detecting nodes that deviate from expected patterns. These modules work together to identify anomalies in graph-based data.

Problem Definition (Graph Anomaly Detection) Given an attributed network $\mathcal{G}$ with adjacency matrix $\mathbf{A}$, the goal is to learn an anomaly score function $f(\cdot)$ to measure the degree of the abnormality for each node in $\mathcal{G}$.

4 Methodology

In this section, we outline the overall framework of the proposed DECLARE method, as illustrated in Figure 1. DECLARE comprises four main modules: a graph enhancement module, a graph contrastive learning module, a graph reconstruction module, and an anomaly scoring module. The original graph is treated as the base view, while a second view is generated via graph diffusion. For each view, both the GNN and Graph Transformer (GTR) encoders process the sampled subgraph and the target node to learn informative hidden representations. The contrastive learning module identifies anomalies by distinguishing positive and negative pairs from multiple perspectives. Meanwhile, the reconstruction module independently reconstructs graph attributes and structures, with anomalies detected based on their respective reconstruction errors in both attribute and structural spaces.

4.1 Dual Encoders

The issue of smoothing in GCNs has been discussed in previous works [Chen *et al.*, 2020], highlighting the need for effective solutions. To obtain a more comprehensive graph representation, we employ dual encoders: GCN and Graph Transformer. The layer of the GCN encoder are designed to capture local graph structures, while the Graph Transformer encoder enhances the model's ability to capture long-range dependencies and intricate relationships. By combining these two encoders, we aim to overcome the smoothing problem and improve the overall graph representation. The formulation of each graph convolutional layer is given as follows:

$$\mathbf{H}^{(l+1)} = \sigma(GNN(\mathbf{H}^{(l)}, \hat{\mathbf{A}}, \mathbf{W}^{(l)})) \tag{1}$$

where $\mathbf{H}^{(l)}$ and $\mathbf{H}^{(l+1)}$ are the input and output node feature matrices at layer l, $\mathbf{W}^{(l)}$ is the learnable weight matrix, $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ is the symmetrically normalized adjacency matrix with self-loops $\tilde{\mathbf{A}}$, and $\sigma(\cdot)$ is a nonlinear activation.

Simultaneously, we use NodeFormer [Wu *et al.*, 2022] as the Transformer encoder, which can be represented as:

$$\mathbf{h}_{v_i}^{(l+1)} \approx \frac{\phi(\mathbf{q}_{v_i}/\sqrt{\tau})^\top \sum_{v=1}^{N} e^{g_{v_j}/\tau}\phi(\mathbf{k}_{v_j}/\sqrt{\tau}) \cdot \mathbf{v}_{v_j}^\top}{\phi(\mathbf{q}_{v_i}/\sqrt{\tau})^\top \sum_{w=1}^{N} e^{g_w/\tau}\phi(\mathbf{k}_w/\sqrt{\tau})} \tag{2}$$

where $\phi(\cdot)$ denotes a kernel function, the kernel function is approximated using Random Features, making the kernel mapping in Eq. 2 a random feature map from $\mathbb{R}^d$ to $\mathbb{R}^m$. We

adopt widely used Positive Random Features for this mapping: $\phi(\mathbf{x}) = \frac{exp(\frac{-||\mathbf{x}||_2^2}{2})}{\sqrt{m}}[exp(\mathbf{w}_1^T\mathbf{x}, ..., exp(\mathbf{w}_m^T\mathbf{x})]$. And $\mathbf{q}_{v_i} = W_Q^{(l)}\mathbf{h}_{v_i}^{(l)}$, $\mathbf{k}_{v_j} = W_K^{(l)}\mathbf{h}_{v_j}^{(l)}$ and $\mathbf{v}_{v_j} = W_V^{(l)}\mathbf{h}_{v_j}^{(l)}$.

4.2 Contrastive Learning with Augmented Views

Building on the effective embeddings obtained earlier, we further enhance the graph's structural representation for contrastive learning by employing graph diffusion to generate augmented views, providing a global perspective of the structure [Hassani and Khasahmadi, 2020]. The graph diffusion process is defined as:

$$\mathbf{S} = \sum_{k=0}^{\infty} \theta_k \mathbf{T}^k, \tag{3}$$

where $\mathbf{S}$ is the diffused adjacency matrix, $\{\theta_k\}_{k\geq 0}$ are non-negative weighting coefficients satisfying $\sum_{k=0}^{\infty}\theta_k = 1$, and $\mathbf{T} \in \mathbb{R}^{n\times n}$ is the row-normalized transition matrix. To power the graph diffusion process, we utilize Personalized PageRank (PPR) [Page *et al.*, 1999]. The diffusion matrix $\mathbf{S}$ can be reformulated as:

$$\mathbf{S} = \rho(\mathbf{I} - (1-\rho)\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})^{-1} \tag{4}$$

where ρ denotes the teleport probability in the random walk process. The resulting matrix $\mathbf{S}$ represents a globally diffused view of the original graph structure.

The contrastive learning framework is applied across two views—base and augmented. For each node v_i, we sample a local subgraph P_i of size t from the original graph using random walk with restart (RWR) [Tong *et al.*, 2006]. From this base view, we construct its positive pair. The augmented-view subgraph is then derived by applying the same procedure to sample a subgraph. The subgraph of v_i in both the base and augmented views serves as the positive pair, while the subgraph of another node becomes the negative pair.

To enhance the discriminative power of node representations, we mask the target node's attributes within the sampled subgraph before feeding it into both branches of the dual encoder, consisting of a shared Graph Convolutional Network ($GNN(\cdot)$) and a Graph Transformer $GTR(\cdot)$. These representations serve as the foundation for downstream contrastive and reconstruction tasks. To illustrate, the two encoders compute low-dimensional node representations as follows:

$$\mathbf{H}_i^{GNN} = GNN(P_i), \mathbf{H}_i^{GTR} = GTR(P_i) \tag{5}$$

Since the target node is masked, its representation is mapped into a shared embedding space using the GCN weights. The output is then given by:

$$\mathbf{h}_i^l = \phi\left(\mathbf{h}_i^{l-1}\mathbf{W}^{l-1}\right) \tag{6}$$

where ϕ represents an activation function such as ReLU, and $\mathbf{h}_i^{l-1}$ is the previous layer representation (either from $GNN(\cdot)$ or $GTR(\cdot)$. These steps yield the node representations $\mathbf{H}_i$ and $\tilde{\mathbf{H}}_i$ for the local and global subgraphs, respectively, and $\mathbf{h}_i$ for the target node v_i.

To ensure consistency between the node and its subgraph representation, we apply an average pooling function as the readout module to generate the subgraph-level embedding vector. For simplicity, we use $GNN(\cdot)$ as an example, where the subgraph-level embedding vector is computed by averaging the node representations:

$$\mathbf{z}_i^{GNN} = \text{Readout}(\mathbf{H}_i^{GNN}) = \frac{1}{t_i}\sum_{n=1}^{t_i}(\mathbf{H}_i^{GNN})_n \tag{7}$$

where t_i is the number of nodes in the subgraph P_i. The discriminative score for the positive pair is then computed as:

$$s_{dis,i}^{GNN} = \sigma(\mathbf{h}_i^{GNN}\mathbf{W}_s T(\mathbf{z}_i^{GNN})) \tag{8}$$

where W_s is a learnable matrix and T represents the transpose operation. A similar score $\tilde{s}_{dis,i}^{GNN}$ is calculated for the negative pair. The base-view contrastive loss for the pair $v_i, G_i, \tilde{G}_i)$ is:

$$\mathcal{L}_{\text{base}}^{GNN}(v_i) = -\frac{1}{2}\left(\log(s_{dis,i}^{GNN}) + \log(1 - \tilde{s}_{dis,i}^{GNN})\right) \tag{9}$$

A similar loss $\mathcal{L}_{aug}^{GNN}$ is computed for the augmented view. The combined intra-view contrastive learning objective is:

$$\mathcal{L}_{\text{intra}}^{GNN} = \frac{1}{2N}\sum_{i=1}^{N}\left(\mathcal{L}_{\text{base}}^{GNN}(v_i) + \mathcal{L}_{\text{aug}}^{GNN}(v_i)\right) \tag{10}$$

Next, we define the inter-view contrastive loss to align the representations across the two views. The inter-view loss is calculated as $\mathcal{L}_{\text{inter}}^{GNN} = \|s_{base,i}^{GNN} - s_{aug,i}^{GNN}\|_F^2$ where $s_{base,i}^{GNN}$ and $s_{aug,i}^{GNN}$ are the discriminative score vectors for the positive pairs in the base and augmented views, respectively. For the GNN encoder, the contrastive learning loss with augmented views is defined as:

$$\mathcal{L}_{con}^{GNN} = \mathcal{L}_{inter}^{GNN} + \mathcal{L}_{intra}^{GNN} \tag{11}$$

Similarly, the GTR encoder uses the same formulation for its contrastive learning loss.

4.3 Graph Reconstruction Module

Using the embeddings obtained from the aforementioned Dual Encoder, we perform separate reconstructions for both the structure and attributes. The structure decoder aims to reconstruct the structural information and detects structural anomalies by measuring the reconstruction error between the output and the input structure. We employ a GCN-based decoder, which can be formulated as follows:

$$\begin{aligned}\mathbf{H}^{GNN} &= GCN(P_i; \mathbf{W}_{T_{dec}}^{GNN}),\\ \widehat{\mathbf{A}}^{GNN} &= \mathbf{H}^{GNN}T(\mathbf{H}^{GNN})\end{aligned} \tag{12}$$

where $\mathbf{H}^{GNN}, \mathbf{W}_{T_{dec}}^{GNN}$ denote the potential node embedding matrix after the decoder and the weight matrix of the topology decoder. $\widehat{\mathbf{A}}^{GNN}$ denotes the reconstructed adjacency matrix obtained from the inner product of $\mathbf{H}^{GNN}$.

The attribute reconstruction decoder plays a critical role in regressing node attributes and is an effective tool for detecting contextual anomalies in graphs. A GCN-based decoder is used for this purpose. Nodes whose attributes are well-approximated by the decoder are considered less likely to

be anomalous, whereas significant reconstruction errors suggest that a node's attributes deviate from the patterns typical of normal nodes. The attribute reconstruction process is expressed as:

$$\widehat{X}_{\text{raw}}^{GNN} = GCN(P_i; \mathbf{W}_{A_{\text{dec}}}^{GNN}) \tag{13}$$

where $\widehat{X}_{raw}^{GNN}$ is the reconstructed attribute matrix, and $\mathbf{W}_{A_{dec}}^{GNN}$ represents the decoder's weight matrix. The graph reconstruction module encompasses two essential tasks: topology and attribute regression. Reconstruction errors from these tasks are combined to form the objective function, which is defined as:

$$L_{\text{rec}}^{GNN} = (1-\beta)\|\mathbf{A}^{GNN} - \widehat{\mathbf{A}}^{GNN}\|_F^2 + \beta\|\mathbf{X}_{\text{raw}}^{GNN} - \widehat{\mathbf{X}}_{\text{raw}}^{GNN}\|_F^2 \tag{14}$$

where β is a balancing parameter that adjusts the relative importance of topology and attribute reconstruction. This approach ensures a comprehensive understanding of both structural and attribute-level inconsistencies, aiding in anomaly detection on graphs.

4.4 Graph Anomaly Score Module

Anomaly detection combines scores from the graph reconstruction and discrimination modules. For node v_i, the reconstruction score is calculated as:

$$s_{\text{rec},i} = (1-\beta)\|a_i - \hat{a}_i\|_2^2 + \beta\|x_i - \hat{x}_i\|_2^2 \tag{15}$$

where $\beta \in [0,1]$ balances topology and attribute errors. Higher scores indicate greater anomaly likelihood. To complement this, we compute two discriminative scores, $s_{dis,i}^{GNN}$ and $s_{dis,i}^{GTR}$, derived from the contrastive learning objectives under the dual encoders. And the final composite anomaly score combines all three components, formulated as:

$$a_i = \alpha \cdot s_{dis,i}^{GNN} + (1-\alpha) \cdot s_{dis,i}^{GTR} + \gamma \cdot s_{rec,i} \tag{16}$$

where $\alpha \in [0,1]$ balances the contributions of the two discriminative scores, and γ controls the influence of the reconstruction-based anomaly signal. To ensure robustness against stochastic variations such as subgraph sampling and training noise, we repeat the entire scoring process over R independent runs. The final anomaly score for each node is then computed by combining the average and variability of its per-round scores:

$$\begin{aligned} \bar{a}_i &= \frac{1}{R}\sum_{r=1}^{R} a_i^{(r)}, \\ a_i &= \bar{a}_i + \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(a_i^{(r)} - \bar{a}_i\right)^2}. \end{aligned} \tag{17}$$

5 Experiment

5.1 Datasets

To evaluate our model, we use six widely recognized benchmark datasets commonly employed in graph anomaly detection. These datasets are categorized into two types: citation network datasets and social network datasets [Liu *et al.*, 2021; Zheng *et al.*, 2021]. The citation network datasets include Cora, Citeseer, ACM, and Pubmed, while the social network datasets consist of BlogCatalog and Flickr. Since ground truth anomalies are not available, we adopt the anomaly injection method proposed by [Ding *et al.*, 2019; Liu *et al.*, 2021] to simulate realistic anomalous behaviors. This approach ensures a controlled and consistent evaluation environment across all datasets. The detailed statistics of these datasets are provided in Table 1.

Dataset	Nodes	Edges	Attributes	Anomalies	Ratio
Cora	2,708	5,429	1,433	150	5.5%
Citeseer	3,327	4,723	3,703	150	4.5%
ACM	16,484	71,980	8,337	600	3.6%
Pubmed	19,717	44,338	500	600	3.0%
BlogCatalog	5,196	171,743	8,189	300	5.8%
Flickr	7,575	239,738	12,407	450	5.9%

Table 1: The statistics of the datasets.

5.2 Experimental Setup

Baselines And Evaluation Metrics

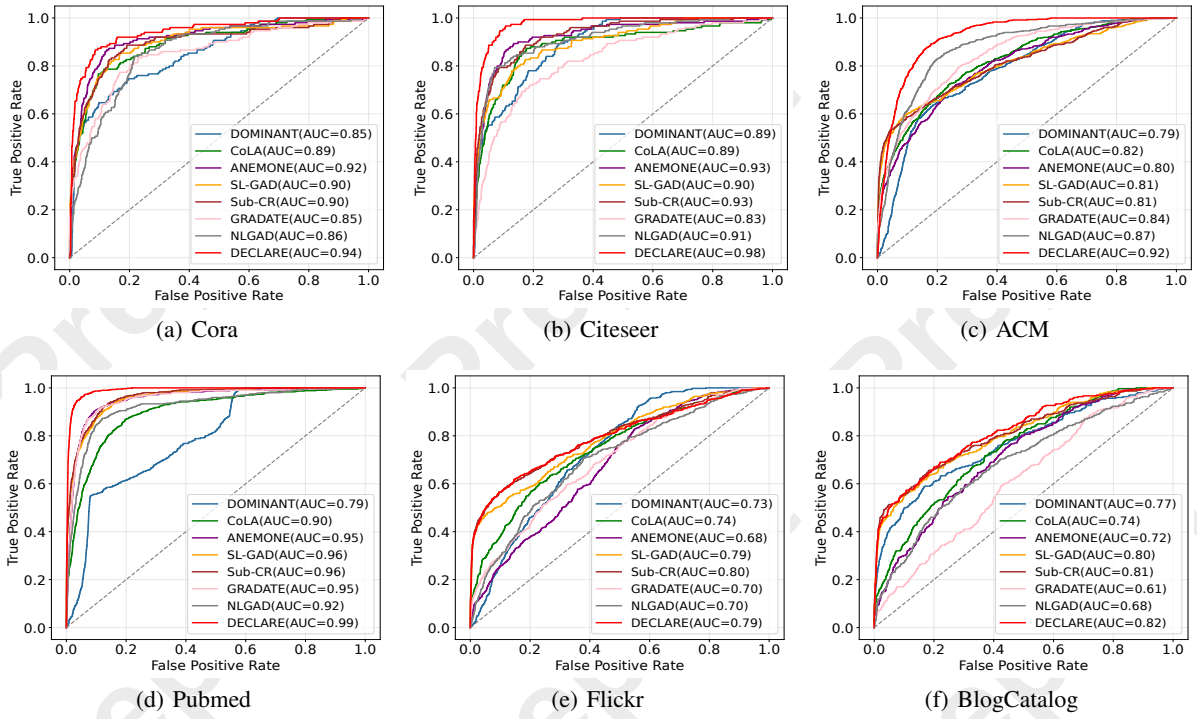
In our experiments, we compare our model with seven deep learning-based anomaly detection methods. DOMINANT [Ding *et al.*, 2019] employs a deep graph autoencoder to detect anomalous nodes by utilizing both graph structure and features. CoLA [Liu *et al.*, 2021] leverages a GNN-based contrastive learning approach at the node-subgraph level to compute anomaly scores for nodes based on positive and negative instance pairs. ANEMONE [Jin *et al.*, 2021] utilizes multi-scale patch and context-level contrastive learning with GNNs to identify anomalous nodes. SL-GAD [Zheng *et al.*, 2021] is a self-supervised method that integrates generative and structural spaces. Sub-CR [Zhang *et al.*, 2022a] is a contrastive learning-based anomaly detection framework that uses graph diffusion to augment the original graph for both inner and outer views. GRADATE [Duan *et al.*, 2023a] applies multi-view contrastive learning across node-node, node-subgraph and subgraph-subgraph relations to detect anomalies. Lastly, NLGAD [Duan *et al.*, 2023b] is a multi-scale contrastive learning network that incorporates high-confidence nodes into the normality pool, achieving superior results through training based on these nodes. All methods are comprehensively evaluated using the AUC metric to ensure a fair and reliable comparison.

Hyperparameter Settings

In this study, we set the subgraph size to 4. The GNN encoder and the attribute decoder each consist of two layers of GNN, while the structure decoder uses a single-layer GCN. The hidden layer embedding dimension is fixed to 64. We optimize the model using the Adam optimizer, with a batch size of 300 across all datasets. For the Cora, Citeseer, Pubmed, and ACM datasets, the learning rate is set to 0.001, whereas for BlogCatalog, the learning rate is 0.003, and for Flickr, it is 5e-4. The model is trained for a total of 300 epochs.

Model	Cora	Citeseer	ACM	Pubmed	Flickr	BlogCatalog
DOMINANT	0.8538	0.8938	0.7853	0.7911	0.7330	0.7653
CoLA	0.8941	0.8864	0.8224	0.8984	0.7449	0.7421
ANEMONE	0.9057	<u>0.9268</u>	0.8037	0.9536	0.6772	0.7151
SL-GAD	0.9001	0.8986	0.8120	0.9575	0.7859	0.8039
Sub-CR	0.8969	0.9295	0.8131	<u>0.9606</u>	**0.7968**	<u>0.8078</u>
GRADATE	0.8517	0.8282	0.8438	0.9525	0.6978	0.6106
NLGAD	<u>0.9075</u>	0.9140	<u>0.8741</u>	0.9228	0.7021	0.6848
DECLARE	**0.9433**	**0.9772**	**0.9292**	**0.9901**	<u>0.7933</u>	**0.8189**

Table 2: The area under the ROC curve (AUC) for anomalous node detection is reported on six benchmark datasets for DECLARE and competing methods. Boldface highlights the top-performing results, while underlining marks the runner-up scores.



(a) Cora (b) Citeseer (c) ACM

(d) Pubmed (e) Flickr (f) BlogCatalog

Figure 2: Roc curve evaluation of anomalous node detection for DECLARE and baselines.

5.3 Result And Analysis

As shown in the Table 2, DECLARE performs exceptionally well across all datasets, especially on more common datasets like Cora, Citeseer, and Pubmed, where it achieves high scores, particularly in Citeseer and Pubmed, with scores of 0.9772 and 0.9901, respectively. This indicates that DECLARE excels in these specific tasks. In contrast, DECLARE performs slightly worse on the Flickr and BlogCatalog datasets but still maintains stable performance, with scores of 0.7933 and 0.8189. Considering that these datasets usually contain more noise and complex multimodal information, DECLARE's performance remains acceptable. Cora and Citeseer are traditional citation network datasets, where DECLARE shows a clear advantage over other methods. For instance, on the Cora dataset, compared to CoLA (score 0.8941), DECLARE's score of 0.9433 represents an improvement of about 5.5%, and compared to SL-GAD (score 0.9001), the improvement is around 4.8%. On the Citeseer dataset, DECLARE scores 0.9772, which is an improvement of around 10.3% compared to CoLA (score 0.8864), and about 5.4% compared to ANEMONE (score 0.9268). On these datasets, DECLARE's performance improvement ranges from 5% to 10%, showing a significant advantage. On the Pubmed dataset, DECLARE also demonstrates substantial improvement, with a score of 0.9901, significantly outperforming other methods. Compared to CoLA (score 0.8984), DECLARE shows an improvement of around 10.2%, and compared to SL-GAD (score 0.9575), the improvement is about 3.4%, further highlighting its strong performance.

Although DECLARE's scores are relatively lower on the Flickr and BlogCatalog datasets, it still demonstrates stable performance with notable improvements. On the Flickr dataset, DECLARE scores 0.7933, which reflects an improvement of about 6.5% compared to CoLA (score 0.7449)

	Cora	Citeseer	ACM	Pubmed	Flickr	BlogCatalog
DECLARE$_{GNN}$	0.9347	0.9546	0.9038	0.9510	0.7741	0.8016
DECLARE$_{GTR}$	0.8804	0.9281	0.8714	0.7003	0.7356	0.7207
DECLARE$_{Rec}$	0.8950	0.8833	0.8624	0.9417	0.7334	0.7531
DECLARE$_{Con}$	0.9122	0.9301	0.8809	0.9603	0.7566	0.7719
DECLARE	**0.9424**	**0.9768**	**0.9292**	**0.9901**	**0.7933**	**0.8189**

Table 3: AUC Values from the Ablation Study.

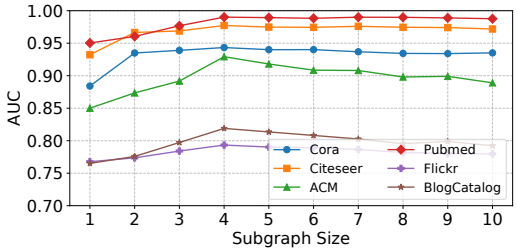
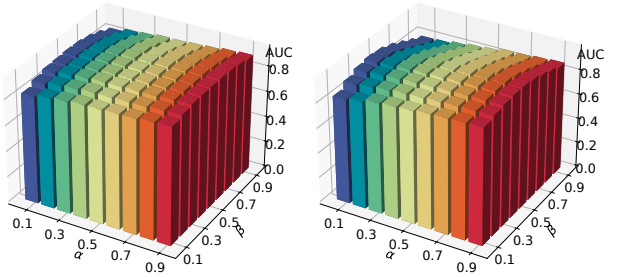


Figure 3: Subgraph size t vs. AUC across datasets.



(a) Cora (b) Citeseer

Figure 4: α and β parameter effects on AUC across datasets.

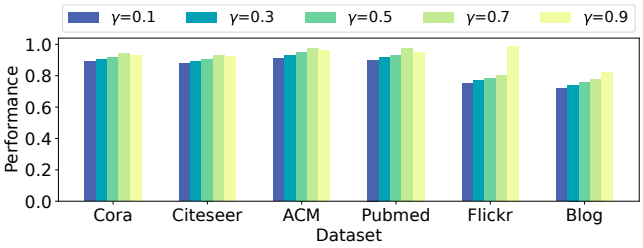


Figure 5: Trade-off parameter γ w.r.t. AUC across all datasets.

and around 13.0% compared to NLGAD (score 0.7021). On the BlogCatalog dataset, DECLARE scores 0.8189, showing an improvement of about 10.3% compared to CoLA (score 0.7421) and around 1.9% compared to SL-GAD (score 0.8039). While the improvements on these datasets are smaller, DECLARE still exhibits significant gains, with improvements ranging from 6% to 13%. However, DECLARE performs relatively weaker on complex multimodal datasets like Flickr and BlogCatalog. This is likely due to the intricate relationships between graph structures and textual features in these datasets, which DECLARE may not handle as effectively as models specifically designed for these tasks, such as NLGAD or GRADATE. Additionally, the smaller improvements observed in some datasets suggest that DECLARE's generalization ability could benefit from further targeted optimization in specific challenging scenarios.

5.4 Ablation and Analysis

The ablation experiments, summarized in Table 3, analyze the contributions of key components in DECLARE. Specifically, we conducted an ablation study on the reconstruction and contrastive learning modules, as well as on the use of GCNs and Transformers for modeling local and global structures. The results demonstrate that combining GCNs and Transformers yields better performance than using either alone. Furthermore, the results show that both the reconstruction and contrastive learning modules are essential for improving performance. Removing the reconstruction module (DECLARE$_{Rec}$) causes performance drops across all datasets, with the largest declines on BlogCatalog (9.7%) and Flickr (10.5%). Similarly, removing the contrastive learning module (DECLARE$_{Con}$) reduces performance, especially on Cora (3.3%) and Citeseer (4.7%). These findings underscore the critical role of each component and motivate further studies to refine and optimize the framework.

5.5 Sensitivity analysis

Subgraph Size. As shown in Figure 3, the AUC rises from $t = 1$ to $t = 4$, peaks at $t = 4$, and then stabilizes or slightly decreases from $t = 5$ to $t = 10$, indicating that $t = 4$ is the optimal subgraph size across datasets.

Balance Parameter. We further analyzed the impact of the balance parameters α, β, and γ on performance. As shown in Figure 4, increasing α and β generally improves performance, with notable gains when $\alpha, \beta \in [0.5, 0.9]$. In practice, we set $\alpha = [0.9, 0.9, 0.6, 0.8, 0.9, 0.9]$ and $\beta = [0.9, 0.4, 0.7, 0.8, 0.8, 0.8]$ for Cora, Citeseer, ACM, Pubmed, Flickr, and BlogCatalog, respectively. Similarly, as shown in Figure 5, performance generally improves with higher γ, with optimal values typically in the range $[0.6, 0.9]$. We observe that $\gamma = 0.7$ works best for Cora, Citeseer, ACM, and Pubmed, while $\gamma = 0.9$ yields the best results on the more complex Flickr and BlogCatalog datasets.

6 Conclusion

In this paper, we propose the DECLARE framework as a comprehensive and robust solution for graph anomaly detection. It seamlessly integrates four key modules—graph augmentation, contrastive learning, reconstruction, and anomaly scoring—that work synergistically to improve detection accuracy. By generating multiple complementary views through graph diffusion and utilizing both GNN and Graph Transformer encoders, DECLARE effectively captures rich and informative embeddings of the graph's structure and attributes. The contrastive module enhances detection by distinguishing positive and negative pairs, while the reconstruction module identifies outliers via structural and attribute reconstruction errors. Experimental results demonstrate the superior effectiveness of DECLARE across various challenging datasets.

Acknowledgments

This work was supported by the National Key R&D Program of China (No.31400), NSFC (U2468204), Key R&D and Transformation Plan of Qinghai Province (No.2022-QY-218), Science and Technology Plan of Shenzhen (KCXFZ20211020172544004), the Key R&D project in Hainan province, China (NO.ZDYF2024SHFZ051) and the Major Science and Technology Program of Yazhou Bay Innovation Institute of Hainan Tropical Ocean University (2023CXYZD001).

Contribution Statement

Nannan Wu and Hongdou Dong contribute equally to this work.

References

[Chen *et al.*, 2020] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International conference on machine learning*, pages 1725–1735. PMLR, 2020.

[Ding *et al.*, 2019] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM international conference on data mining*, pages 594–602. SIAM, 2019.

[Duan *et al.*, 2023a] Jingcan Duan, Siwei Wang, Pei Zhang, En Zhu, Jingtao Hu, Hu Jin, Yue Liu, and Zhibin Dong. Graph anomaly detection via multi-scale contrastive learning networks with augmented view. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 7459–7467, 2023.

[Duan *et al.*, 2023b] Jingcan Duan, Pei Zhang, Siwei Wang, Jingtao Hu, Hu Jin, Jiaxin Zhang, Haifang Zhou, and Xinwang Liu. Normality learning-based graph anomaly detection via multi-scale contrastive learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7502–7511, 2023.

[Hassani and Khasahmadi, 2020] Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *International conference on machine learning*, pages 4116–4126. PMLR, 2020.

[Hu *et al.*, 2023] Jingtao Hu, Bin Xiao, Hu Jin, Jingcan Duan, Siwei Wang, Zhao Lv, Siqi Wang, Xinwang Liu, and En Zhu. Samcl: Subgraph-aligned multiview contrastive learning for graph anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[Jin *et al.*, 2021] Ming Jin, Yixin Liu, Yu Zheng, Lianhua Chi, Yuan-Fang Li, and Shirui Pan. Anemone: Graph anomaly detection with multi-scale contrastive learning. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 3122–3126, 2021.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Latah, 2020] Majd Latah. Detection of malicious social bots: A survey and a refined taxonomy. *Expert Systems with Applications*, 151:113383, 2020.

[Li *et al.*, 2017] Jundong Li, Harsh Dani, Xia Hu, and Huan Liu. Radar: Residual analysis for anomaly detection in attributed networks. In *IJCAI*, volume 17, pages 2152–2158, 2017.

[Liu *et al.*, 2021] Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE transactions on neural networks and learning systems*, 33(6):2378–2392, 2021.

[Ma *et al.*, 2021] Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z Sheng, Hui Xiong, and Leman Akoglu. A comprehensive survey on graph anomaly detection with deep learning. *IEEE transactions on knowledge and data engineering*, 35(12):12012–12038, 2021.

[Motie and Raahemi, 2024] Soroor Motie and Bijan Raahemi. Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications*, 240:122156, 2024.

[Page *et al.*, 1999] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford infolab, 1999.

[Perozzi and Akoglu, 2016] Bryan Perozzi and Leman Akoglu. Scalable anomaly ranking of attributed neighborhoods. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 207–215. SIAM, 2016.

[Qiu *et al.*, 2020] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1150–1160, 2020.

[Tan *et al.*, 2024] Shiyin Tan, Dongyuan Li, Renhe Jiang, Ying Zhang, and Manabu Okumura. Community-invariant graph contrastive learning. *arXiv preprint arXiv:2405.01350*, 2024.

[Tong *et al.*, 2006] Hanghang Tong, Christos Faloutsos, and Jia-Yu Pan. Fast random walk with restart and its applications. In *Sixth international conference on data mining (ICDM'06)*, pages 613–622. IEEE, 2006.

[Wei *et al.*, 2023] Chunyu Wei, Yu Wang, Bing Bai, Kai Ni, David Brady, and Lu Fang. Boosting graph contrastive learning via graph contrastive saliency. In *International conference on machine learning*, pages 36839–36855. PMLR, 2023.

[Wu *et al.*, 2022] Qitian Wu, Wentao Zhao, Zenan Li, David P Wipf, and Junchi Yan. Nodeformer: A scalable graph structure learning transformer for node classification. *Advances in Neural Information Processing Systems*, 35:27387–27401, 2022.

[Ying *et al.*, 2018] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 974–983, 2018.

[You *et al.*, 2020] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

[Yu *et al.*, 2022] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. Are graph augmentations necessary? simple graph contrastive learning for recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 1294–1303, 2022.

[Zhang *et al.*, 2019] Yiming Zhang, Yujie Fan, Wei Song, Shifu Hou, Yanfang Ye, Xin Li, Liang Zhao, Chuan Shi, Jiabin Wang, and Qi Xiong. Your style your identity: Leveraging writing and photography styles for drug trafficker identification in darknet markets over attributed heterogeneous information network. In *The World Wide Web Conference*, pages 3448–3454, 2019.

[Zhang *et al.*, 2022a] Jiaqiang Zhang, Senzhang Wang, and Songcan Chen. Reconstruction enhanced multi-view contrastive learning for anomaly detection on attributed networks. *arXiv preprint arXiv:2205.04816*, 2022.

[Zhang *et al.*, 2022b] Yifei Zhang, Hao Zhu, Zixing Song, Piotr Koniusz, and Irwin King. Costa: covariance-preserving feature augmentation for graph contrastive learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2524–2534, 2022.

[Zhang *et al.*, 2023] Yifei Zhang, Hao Zhu, Zixing Song, Piotr Koniusz, and Irwin King. Spectral feature augmentation for graph contrastive learning and beyond. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11289–11297, 2023.

[Zheng *et al.*, 2021] Yu Zheng, Ming Jin, Yixin Liu, Lianhua Chi, Khoa T Phan, and Yi-Ping Phoebe Chen. Generative and contrastive self-supervised learning for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12220–12233, 2021.

[Zhu *et al.*, 2020] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*, 2020.

[Zhu *et al.*, 2021a] Hao Zhu, Ke Sun, and Peter Koniusz. Contrastive laplacian eigenmaps. *Advances in neural information processing systems*, 34:5682–5695, 2021.

[Zhu *et al.*, 2021b] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Graph contrastive learning with adaptive augmentation. In *Proceedings of the web conference 2021*, pages 2069–2080, 2021.