entropy

MDPI

Article

Few-Shot Graph Anomaly Detection via Dual-Level Knowledge Distillation

Xuan Li, Dejie Cheng *, Luheng Zhang, Chengfang Zhang and Ziliang Feng

National Key Laboratory of Fundamental Science on Synthetic Vision, Sichuan University, Chengdu 610065, China; lixuanlmw@stu.scu.edu.cn (X.L.); zhangluheng@sptc.edu.cn (L.Z.); chengfangzhang@scpolicec.edu.cn (C.Z.); fengziliang@scu.edu.cn (Z.F.)

* Correspondence: chengdejie@sptc.edu.cn

Abstract: Graph anomaly detection is crucial in many high-impact applications across diverse fields. In anomaly detection tasks, collecting plenty of annotated data tends to be costly and laborious. As a result, few-shot learning has been explored to address the issue by requiring only a few labeled samples to achieve good performance. However, conventional few-shot models may not fully exploit the information within auxiliary sets, leading to suboptimal performance. To tackle these limitations, we propose a dual-level knowledge distillation-based approach for graph anomaly detection, DualKD, which leverages two distinct distillation losses to improve generalization capabilities. In our approach, we initially train a teacher model to generate prediction distributions as soft labels, capturing the entropy of uncertainty in the data. These soft labels are then employed to construct the corresponding loss for training a student model, which can capture more detailed node features. In addition, we introduce two representation distillation losses—short and long representation distillation—to effectively transfer knowledge from the auxiliary set to the target set. Comprehensive experiments conducted on four datasets verify that DualKD remarkably outperforms the advanced baselines, highlighting its effectiveness in enhancing identification performance.

Keywords: anomaly detection; graph neural network; cross entropy; knowledge distillation

check for updates

Academic Editors: Irwin King and Ziqiao Meng

Received: 30 October 2024
Revised: 19 December 2024
Accepted: 25 December 2024
Published: 1 January 2025

Citation: Li, X.; Cheng, D.; Zhang, L.; Zhang, C.; Feng, Z. Few-Shot Graph Anomaly Detection via Dual-Level Knowledge Distillation. *Entropy* **2025**, *27*, 28. https://doi.org/10.3390/e27010028

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Anomaly detection in graphs intends to identify nodes that exhibit abnormal behaviors, significantly deviating from the majority of nodes [1]. This task has numerous high-impact applications across various domains, including detecting abnormal users [2,3] and identifying fraudulent behavior [4,5]. To effectively identify anomalies within graph-structured data, it is essential to develop robust classification models with high generalization capabilities. To achieve this, various techniques such as graph neural networks (GNNs) and matrix factorization have been explored for anomaly detection [6,7].

For graph anomaly detection tasks, obtaining label information is generally costly and time-consuming [1]. Therefore, existing methods are predominately developed in an unsupervised manner. However, the anomalies they identify may turn out to be data noises or uninteresting data instances due to the lack of prior knowledge on the anomalies of interest. Hence, leveraging several labeled samples as the prior information to improve anomaly detection has become a trend, since it is relatively low-cost in real-world scenarios to collect a few labels. Correspondingly, few-shot learning has been introduced to train machine learning models, such as graph convolutional networks (GCNs) [8], on

datasets with a limited number of labeled nodes for tasks like anomaly detection and node classification.

Previous few-shot methods may be divided in two primary methodologies: meta-learning-based and self-training-based approaches [9,10]. Meta-learning approaches aim to train an initialization module using prior information from an auxiliary set, employing techniques such as prototypical networks. For example, G-META [9] utilizes meta-learning to maintain the structural and feature knowledge of graphs, while GPN [11] leverages prototypical networks to compute prototypes that capture expressive node representations. Despite their effectiveness, meta-learning methods require the construction of multiple tasks to achieve generalization, which can be time-consuming. Furthermore, these methods often overlook the valuable information contained in unlabeled nodes, limiting their overall effectiveness.

To this end, self-training methods have been designed to utilize prior information from the auxiliary data with low time overhead while assigning pseudo-labels to a portion of the unannotated data of the target set. For example, IA-FSNC [12] cuts down time costs through building a single GCN based on the auxiliary set and allocates pseudo-labels to unlabeled examples with small information entropy in the target set to achieve information augmentation. However, some limitations need to be addressed. For instance, self-training few-shot methods often fail to thoroughly exploit the knowledge contained in the auxiliary set. For IA-FSNC, solely the parameters from the initial layer of the graph convolution are migrated to initialize the target set, overlooking information from other graph convolutional layers. This may lead to an insufficient transfer of information, resulting in a lack of comprehensive knowledge for the target set.

In this paper, we introduce a dual-level knowledge distillation-based graph anomaly detection approach, DualKD, which can significantly enhance the detection performance in scenarios with limited labeled samples. Our approach begins by training a teacher model to produce prediction distributions, which serve as the soft labels of samples to capture entropy-related uncertainties. Subsequently, a student network is trained to learn more detailed node features from these soft labels. The student model is then utilized for the final generalized few-shot anomaly detection, leveraging the enriched knowledge transferred from the teacher model. This knowledge transfer enables the distillation of information from auxiliary datasets, enhancing the detection accuracy on the target dataset while requiring only a few labeled samples.

To maximize the transfer of information from the auxiliary set generated by the teacher model, we devise a dual-level representation distillation mechanism that conveys data from every layer of graph convolution in the teacher model to the student model. This representation distillation consists of two processes: short representation distillation and long representation distillation. To advocate the robustness, the representation distillation loss is combined with the teacher–student distillation loss to form the final distillation loss.

Our contributions are summarized as follows:

- We introduce a dual-level knowledge distillation-based graph anomaly detection framework, DualKD, which can effectively handle scenarios with limited labeled samples.
- We design a dual-level representation distillation strategy that incorporates both short- and long-representation distillation processes to enhance the model's generalization capabilities.
- We provide experimental evidence for four datasets demonstrating that DualKD outperforms state-of-the-art approaches.

2. Related Work

This work is mainly related to three research areas: graph anomaly detection, knowledge distillation, and few-shot learning. Here, we present an overview of the most closely related works in each area.

2.1. Anomaly Detection on Graphs

Since graph-structured data are ubiquitous and have the capacity to model a wide range of real-world complex systems, identifying anomalies in graphs has drawn increased research interest [13,14]. Due to their demonstrated superior modeling power for graphs, various GNN-based methods have been proposed to detect anomalies on graphs. The pioneer used GNNs to build an autoencoder to simultaneously reconstruct the attribute and structure information, and the abnormality is evaluated by reconstruction errors [15]. Based on this framework, a tailored deep GCNN is designed to detect local, global, and structural anomalies by capturing community structure in the graph [16]. Contrastive learning and self-supervised learning [17,18] are also introduced to identify the anomalies in attributed networks [19,20]. Meta-learning and hypersphere learning are incorporated into GNNs to leverage the labeled samples for anomaly detection [21–24].

To remedy the problem that numerous neighbors with normal labels might make the anomaly representations learned by GNNs less distinguishable, multiple re-sampling (e.g., oversampling and undersampling) strategies are designed in [25–27]. Researchers also utilized re-weighting methods to assign different weights to different samples [28–30]. More recently, spectral filters and counterfactuals are explored to enhance the expressive power of GNNs for learning better anomaly representations [31–34].

2.2. Knowledge Distillation

Knowledge distillation [35] initially emerged for model compression, aiming to guide a comparatively simple student model using a well-trained teacher model characterized by a more complex structure and a greater number of parameters. Building on this, several knowledge distillation methods have been proposed for graph neural networks (GNNs). For instance, G-CRD [36] presents a novel graph contrastive representation distillation for GNNs, employing contrastive learning to align student node embeddings with teacher node embeddings in a shared representation space. GFKD [37] designs a method for knowledge distillation with GNNs that does not involve any training data. Meanwhile, GLNN [38] introduces a high-accuracy, low-delay distillation model by using the teacher model as a GNN and the student model as a multi-layer perceptron. GraphKD [39] and KD-FSNC [40] explore knowledge distillation for graph node classification, which distill auxiliary data to the student for enhancing classification performance on the target data.

2.3. Few-Shot Learning

Few-shot learning is a paradigm within deep learning designed to address the challenges associated with training models when there is only a limited amount of labeled data available for each class. Few-shot methods can be divided into two types, i.e., data-based methods and model-based methods. Data-based methods involve learning an augmentation mapping, which maps the training data to new data, and then utilizes the newly generated data to expand the training set in few-shot tasks. For example, FewGAN [41] uses generative adversarial networks to create additional samples. FEFS [42] proposes a data augmentation method based on the assumption that each data dimension is modeled by a Gaussian distribution, where categories that are alike share similar distribution characteristics. The mean and variance of the novel (target) set are adapted based on those from the base (auxiliary) set. On the other hand, model-based methods address the few-shot

learning problem by constraining the size of the hypothesis space. For instance, TRPN [43] utilizes intra-class commonality and interclass uniqueness between support samples to estimate the relationship and adjacency relationship between different support–query pairs.

Difference: Compared to the aforementioned works, our approach focuses on few-shot anomaly detection on graphs using knowledge distillation. To enhance detection performance, we propose a novel distillation framework that combines soft and hard target losses as the final objective for the student model. In addition, we introduce a dual-level knowledge transfer strategy to effectively capture information from multiple layers. Furthermore, we employ a graph attention network as the backbone architecture for both the teacher and student models.

3. Methodology

This section gives the formal expression of the proposed method and explains the functions and working mechanism of each part in the expression. As depicted in Figure 1, DualKD encompasses three key elements: a pre-trained GNN module that serves as the backbone for both teacher and student, a soft label distillation process that conveys a class relationship from the teacher to the student, and a representation distillation process that transfers information from every layer of graph convolution in the teacher to the student for learning.

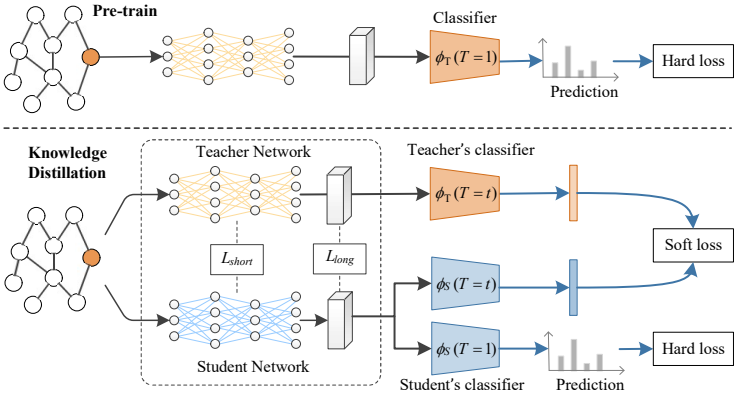

Figure 1. Overall framework of DualKD.

3.1. Pre-Training

Inspired by the advances on graph learning [44,45], we introduce the graph attention networks (GATs) [46] as the backbone, which introduces the masked attention mechanism to represent the importance of different adjacent nodes. Formally, in each layer $l-1$, node v_i integrates the features of neighboring nodes to obtain representations of layer l via:

$$\mathbf{h}_i^{(l)} = \sigma\left(\sum_{j\in\mathcal{V}_i\cup\{v_i\}} a_{ij}\mathbf{W}\cdot\mathbf{h}_j^{(l-1)}\right), \tag{1}$$

where σ refers to a nonlinear activation function (e.g., ReLU), $\mathcal{V}_i$ is the set of neighbors for v_i, and a_{ij} represents the attention coefficient between node v_i and node v_j, which can be computed as:

$$a_{ij} = \frac{\exp(\sigma(\mathrm{a}^{\mathrm{T}}[\mathbf{W}\mathbf{h}_i^{(l)} \oplus \mathbf{W}\mathbf{h}_j^{(l)}]))}{\sum_{k\in\mathcal{V}_i\cup\mathcal{V}_i'\cup\{v_i\}} \exp(\sigma(a^{\mathrm{T}}[\mathbf{W}\mathbf{h}_i^{(l)} \oplus \mathbf{W}\mathbf{h}_k^{(l)}]))}, \tag{2}$$

where $\oplus$ is the concatenation operation and attention vector a is a trainable weight vector that assigns importance to different neighbors of node v_i, allowing the model to highlight the features of the important neighboring node that is more task-relevant.

To incorporate a high-order neighborhood, multiple layers are adopted to build the graph attentive encoder:

$$\begin{aligned} \mathbf{h}_i^{(1)} &= \sigma\Big(\textstyle\sum_{j\in\mathcal{V}_i\cup\mathcal{V}_i'\cup\{v_i\}} a_{ij}^{(1)}\mathbf{W}^{(1)}\cdot\mathbf{x}_j\Big), \\ &\cdots\cdots \\ \mathbf{z}_i &= \sigma\Big(\textstyle\sum_{j\in\mathcal{V}_i\cup\mathcal{V}_i'\cup\{v_i\}} a_{ij}^{(L)}\mathbf{W}^{(L)}\cdot\mathbf{h}_j^{(l-1)}\Big), \end{aligned} \tag{3}$$

where $\mathbf{z}_i$ is the latent representation of node v_i. In this way, the graph attentive encoder is able to map the learned node representations by capturing the nonlinearity of topological structure and node attributes.

Next, the MLP with a Sigmoid function is adopted to detect anomalies. The aggregated representations $\mathbf{z}_i$ are then fed into another MLP with a Sigmoid function to compute the abnormal probability p_i. The weighted cross-entropy loss is then used for the model training:

$$\mathcal{L} = \sum_i (\varphi y_i \log(p_i) + (1-y_i)\log(1-p_i)), \tag{4}$$

where φ is the proportion of anomaly labels ($y_i = 1$) to normal labels ($y_i = 0$).

3.2. Soft Label Distillation

Once the GAT is trained using the auxiliary data, we clone the trained model and transfer it to teacher model ϕ_t and student model ϕ_s

Teacher Model: The teacher model is designed to extract comprehensive structural insights from the input data and deliver precise predictions. Thanks to its extensive training and high capacity, the teacher model effectively generalizes, setting a benchmark for developing a more resilient student model [39]. Specifically, the teacher model is utilized to produce soft targets, which are probability distributions across classes, generated by applying a high temperature to the softmax function. Unlike hard labels, these soft targets provide richer information, revealing the relative likelihoods among different classes. The teacher model typically uses a cross-entropy loss, which strives to maximize likelihoods belonging to the correct class. This approach fine-tunes the model parameters to align the predicted probabilities as closely as possible with the actual labels. The objective of the teacher model can be represented:

$$L_{\text{teacher}} = -\sum_i y_i \log(p_i), \tag{5}$$

where y_i stands for the real annotation; p_i describes the inferred likelihood of category i. By leveraging the graph attention network and temperature scaling, the teacher model can produce well-calibrated soft labels that accurately reflect the underlying distribution of anomalies.

Student model: The student model aims to maintain high prediction accuracy, even when trained on a more limited dataset. By leveraging the soft targets provided by the teacher, the student model can advocate its generalization and more efficiently consider the inherent characteristics within the dataset. Training the student includes adopting a combination of two losses: the cross-entropy loss regarding soft targets as well as the cross-entropy loss regarding hard targets. For the former, the objective is to align the student forecasts with the teacher soft labels using the cross-entropy loss of the soft labels and the predicted probabilities of the student. This specific loss function is formulated as:

$$L_{\text{soft}} = -\sum_i p_i \log(q_i), \tag{6}$$

where p_i denotes the soft target probability generated by the teacher model, while q_i represents the predicted probability from the student model, both calculated at an elevated temperature setting.

Aside from learning from soft labels, the student is also learned to predict the actual hard class labels. The cross-entropy loss with hard targets employs the standard softmax function (temperature set to 1) to calculate the loss between the true labels and forecasted probabilities. This loss is defined as:

$$L_{\text{hard}} = -\sum_i y_i \log(q_i), \tag{7}$$

where y_i stands for the real annotation. q_i represents the inferred probabilities from the student model at a normal temperature.

As a result, the ultimate loss of the student denotes a blend of the soft and hard target losses:

$$L_{\text{SLD}} = \alpha L_{\text{soft}} + (1-\alpha) L_{\text{hard}}, \tag{8}$$

where α serves as a weighting factor that balances the contributions of the soft target loss and the hard target loss. Typically, this parameter is adjusted to place greater emphasis on the soft targets, ensuring that the student effectively involves the subtle information distilled from the teacher model.

3.3. Representation Distillation

Through knowledge distillation, the student model can leverage the soft labels generated by the teacher model, which encapsulate the relative relationships between classes to adjust the learning weights based on the knowledge extracted from the teacher model for the purpose of graph anomaly detection. To fully exploit the information in the auxiliary set and ensure that the student model captures more detailed node features and achieves better generalization for few-shot anomaly detection, we present the short and long representation distillation ways. This ensures the effective transfer of information from all graph convolutional layers of the teacher model to the student model, thereby improving the student model's performance in graph anomaly detection.

Short representation distillation. For graph anomaly detection tasks, the homophilic neighbors tend to belong to the same category [47,48]. Hence, we minimize the embedding distance (i.e., maximizing the embedding similarity) of two homophilic adjacent nodes in the student model. To this end, we first identify the homophilic nodes and then build the shot distillation loss. Similarly to [47], we adopt GPNN [49] to detect homophilic nodes. Concretely, we leverage a pointer network to compute attention vectors (scores) and then select the most relevant nodes from neighborhoods according to these scores. Since the attention scores can denote the relevant relationship with a given node [49], we use them to identify homophilic nodes: two nodes are regarded as homophilic if they have higher attention scores than a threshold. The optimal threshold can be determined by the existing training data.

Furthermore, we utilize the knowledge of the teacher to enforce this constraint. To achieve this, we ensure that the local structure between each pair of homophilic nodes (such as the *i*- and *j*-th nodes) in the teacher is maintained in the student. Based on this, we define the short representation distillation loss as follows:

$$\begin{aligned}\mathcal{L}_{\text{short}} = \sum_{i\in\mathcal{C}} \sum_{j\in\mathcal{N}^h(i)} & \exp\left(-\frac{1}{\sigma^2}\left\|\phi_T^i(\mathbf{A},\mathbf{X}) - \phi_T^j(\mathbf{A},\mathbf{X})\right\|_2^2\right) \\ & \times \left\|\phi_S^i(\mathbf{A},\mathbf{X}) - \phi_S^j(\mathbf{A},\mathbf{X})\right\|_2^2,\end{aligned} \tag{9}$$

where $\mathcal{C}$ denotes the target set, $N^h(i)$ represents the first-order homophilic neighborhood set of the i-th node, and $\mathbf{A}$ and $\mathbf{X}$ refer to the adjacent and attribute matrices, individually. $\phi_T^i(\mathbf{A}, \mathbf{X})$ and $\phi_S^i(\mathbf{A}, \mathbf{X})$ denote the representations of the i-th node generated by the teacher and student, respectively.

Equation (9) enforces a substantial penalty when the embeddings of two adjacent nodes (i.e., the i-th and j-th nodes in the teacher model) are positioned too closely in the student model. In essence, minimizing Equation (9) guarantees that, if the i-th and j-th nodes are homophilic neighbors in the teacher model, they will likewise be adjacent in the student model. This equation enables the transfer of information—particularly the local structure of each node's embedding across all layers of the graph convolutional network in the teacher model—ensuring that the representations of adjacent nodes in the student model remain similar. As a result, short representation distillation preserves the local structure of the teacher model, effectively guiding the training of the student model.

Long representation distillation. Following the local structure preservation achieved through the short representation distillation, we introduce long representation distillation to maintain the global structure, leveraging information from the teacher model. The preservation of both short and long structures offers complementary insights [40,50]. Specifically, we begin by extracting node embeddings in the target set using the teacher. In order to maintain the global structural information, we then compute the mean squared error between each node's representation in the teacher and its corresponding one in the student. The long representation distillation loss is consequently defined as follows:

$$\mathcal{L}_{\text{long}} = \|\phi_S(\mathbf{A}, \mathbf{X}) - \phi_T(\mathbf{A}, \mathbf{X})\|_2^2. \tag{10}$$

Equation (10) leverages the information from each graph convolutional tier in the teacher to maintain the holistic structural information from the target set. Consequently, each node's embedding in the student model closely matches the one of the corresponding item from the teacher. This holistic structural alignment addresses the challenge posed by the limited availability of training samples (i.e., labeled data on the target set) in the student.

Representation distillation loss. We combine the short objective function in Equation (9) with the long objective function in Equation (10) to formulate the final loss for our designed approach:

$$\mathcal{L}_{\text{RDL}} = (1 - \beta)\mathcal{L}_{\text{short}} + \beta\mathcal{L}_{\text{long}}, \tag{11}$$

where β serves as a superparameter that balances the knowledge derived from both partial and holistic structural information. Equation (11) proposes a couple of distilled losses to maintain the partial and holistic structures within the target data. This method strengthens the resilience of the student model by minimizing the influence of poorly trained nodes while efficiently leveraging information from the auxiliary dataset.

In our developed distillation approach, short representation distillation capitalizes on the similarity of embeddings at each convolution tier within the teacher to preserve the local structural information of the target data. In contrast, long distillation employs the representations of all nodes across every convolution tier in the teacher to maintain the global structural information of the target data.

3.4. Overall Objective Function

To effectively leverage both class knowledge and representation information, we introduce a unified loss function that merges the knowledge distillation loss with the repre-

sentation distillation loss. This integration creates a more powerful distillation framework. The resulting combined loss function is formulated as follows:

$$\mathcal{L} = \gamma \cdot \mathcal{L}_{\text{SLD}} + (1 - \gamma) \cdot \mathcal{L}_{\text{RDL}}, \tag{12}$$

where γ stands for a balancing factor to adjust the relative significance of the two loss components. This term guarantees that the student tightly emulates the teacher decision-making process. By employing this loss function, the distillation process not only preserves the classification strengths of the teacher but also enables the student to acquire richer and more comprehensive node representations. Overall, this integrated loss function effectively harnesses both knowledge and representation distillation, facilitating notable performance enhancements in the student model, even with limited training samples.

For computational efficiency, the graph convolution operation in the pre-training phase, based on GATs, has a per-layer time complexity of $O(L \cdot (E + V))$, where V and E are the number of nodes and edges, and L is the number of layers. During the soft label distillation phase, the cross-entropy loss computation for the teacher and student models has a complexity of $O(n)$, where n is the number of samples. In the representation distillation phase, both short and long representation distillation methods have a complexity of $O(V \cdot d)$, as they involve embedding comparisons between neighboring nodes or between teacher and student models, where d is the embedding dimension. The overall objective function combines the soft label distillation loss $(\mathcal{L}SLD)$ and representation distillation loss $(\mathcal{L}RDL)$, resulting in a total time complexity of $O(L \cdot (E + V) + n + V \cdot d)$. In practice, the dominant factors are the graph size (E, V) and embedding dimension d.

4. Experiment

In this section, we perform empirical evaluations to demonstrate the effectiveness of the proposed DualKD. We mainly investigate the efficacy of the proposed model, ablation study and the role of auxiliary network number.

4.1. Experimental Setup

Datasets. We conduct experiments on two types of datasets: Ground-truth anomaly graphs: Amazon is a co-purchase network and Yelp is a transaction network, both of which have ground-truth labels of the anomalies; injected anomaly graphs: PubMed and Reddit are two citation networks, with injected anomaly labels. Attribute and structural anomalies are injected into these two datasets using the injection methods of previous studies [15,51]. Table 1 summarizes the statistics of each dataset.

- **Amazon** [16] is collected from Amazon.com and contains product reviews across various categories. The reviewers are classified into two classes, abnormal (reviewers with suspicious review patterns) and normal (reviewers with regular review patterns) according to the Amazon anti-fraud detection algorithm. We select products in the same category to construct each network, where nodes represent reviewers and there is a link between two reviewers if they have reviewed the same product. We apply the bag-of-words model on top of the textual contents to obtain the attributes of each node.
- **Yelp** [52] is collected from Yelp.com and contains reviews for restaurants in several states of the U.S. where the restaurants are organized by ZIP codes. The reviewers are classified into two classes, abnormal (reviewers with only filtered reviews) and normal (reviewers with no filtered reviews) according to the Yelp anti-fraud filtering algorithm. We select restaurants in the same location according to ZIP codes to construct each network where nodes represent reviewers and there is a link between two reviewers if

they reviewed the same restaurant. We apply the bag-of-words model on top of the textual contents to obtain the attributes of each node.

- **PubMed** [53] is a citation network where nodes represent scientific articles related to diabetes and edges are citations relations. Node attribute is represented by a TF/IDF-weighted word vector from a dictionary which consists of 500 unique words. We randomly partition the large network into non-overlapping sub-networks of similar size.
- **Reddit** [54] is collected from an online discussion forum where nodes represent threads and an edge exits between two threads if they are commented on by the same user. The node attributes are constructed using the averaged word embedding vectors of the threads. Similarly, we extract non-overlapping sub-networks from the original large network for our experiments.

Table 1. Statistics of evaluation datasets. r_1 denotes the ratio of labeled anomalies to the total anomalies and r_2 is the ratio of labeled anomalies to the total number of nodes.

Datasets	Amazon	Yelp	PubMed	Reddit
# nodes	3200	4872	3675	15,860
# edges	29,000	43,728	8895	136,781
# features	8000	10,000	500	602
# anomalies	160	223	201	796
r_1 (avg.)	5.00%	4.48%	4.97%	1.26%
r_2 (avg.)	0.25%	0.21%	0.27%	0.063%

Note that, except the Amazon and Yelp dataset, we are not able to access ground-truth anomalies for PubMed and Reddit. Thus, following the works [15,34], we refer to two anomaly injection methods [55,56] to inject a combined set of anomalies (i.e., structural anomalies and attribute anomalies) by perturbing the topological structure and node attributes of the original network, respectively. To inject structural anomalies, we adopt the approach used by [55] to generate a set of small cliques since a small clique is a typical abnormal substructure in which a small set of nodes are much more closely linked to each other than average [57]. Accordingly, we randomly select c nodes (i.e., clique size) in the network and then make these nodes fully linked to each other. By repeating this process K times (i.e., K cliques), we can obtain $K \times c$ structural anomalies. In our experiment, we set the clique size c to 15. In addition, we leverage the method introduced by [56] to generate attribute anomalies. Specifically, we first randomly select a node i and then randomly sample another 50 nodes from the network. We choose the node j whose attributes have the largest Euclidean distance from node i among the 50 nodes. The attributes of node i will then be replaced with the attributes of node j.

Baselines. We assess the identification results of our DualKD against the following baselines:

- **Autoencoder** [58] is an unsupervised deep autoencoder model which introduces an anomaly regularizing penalty based upon L1 or L2 norms.
- **Radar** [59] is an unsupervised method that detects anomalies on an attributed network by characterizing the residuals of attribute information and its coherence with network structure.
- **DOMINANT** [15] is a GCN-based autoencoder framework which computes anomaly scores using the reconstruction errors from both network structure and node attributes.
- **DeepSAD** [22] is a deep learning approach for general semi-supervised anomaly detection. In our experiment, we leverage the node attribute as the input feature.
- **SemiGNN** [28] is a semi-supervised GNN model, which leverages the hierarchical attention mechanism to better correlate different neighbors and different views.

- **BWGNN** [31] is a method equipped with spectral and spatial localized band-pass filters that can better address the right-shift phenomenon in graph anomalies.
- **Meta-GDN** [21] is a new family of graph neural network that not only leverages a small number of labeled anomalies to enforce statistically significant deviations between abnormal and normal nodes on a network, but also incorporates a cross-network meta-learning algorithm to enable few-shot network anomaly detection.
- **CAGAD** [47] is a data augmentation-based method for graph anomaly detection that can produce counterfactual augmented data to enhance detection performance.

Experiment setting. For each dataset, we extract 5 networks, among which 4 networks are considered as the auxiliary networks and 1 for the target network. For each one, we adopt 10 labeled anomalous nodes for model training. We use the following metrics to conduct a comprehensive evaluation of the performance of different anomaly detection methods:

- **AUC-ROC** is widely used in previous anomaly detection research [15,59]. Area under the curve (AUC) is interpreted as the probability that a randomly chosen anomaly receives a higher score than a randomly chosen normal object.
- **AUC-PR** is the area under the curve of precision against recall at different thresholds, and it only evaluates the performance on the positive class (i.e., abnormal objects). AUC-PR is computed as the average precision as defined in [60,61].
- **Precision@K** is defined as the proportion of true anomalies in a ranked list of K (e.g., 10) objects. We obtain the ranking list in descending order according to the anomaly scores that are computed from a specific anomaly detection algorithm.

4.2. Overall Detection Performance

In the experiments, we evaluate the performance of the proposed DualKD by comparing with the included baseline methods. We present the evaluation results with respect to AUC-ROC and AUC-PR in Table 2. Accordingly, we have the following observations: (1) In terms of AUC-ROC and AUC-PR, our approach DualKD outperforms all the other compared methods by a significant margin. This indicates that our method can effectively distill knowledge from the teacher model to the student and improve detection performance. (2) Unsupervised methods (e.g., DOMINANT, Radar) are not able to leverage supervised knowledge of labeled anomalies and therefore have limited performance. Semi-supervised methods (e.g., DeepSAD, SemiGNN) also fail to deliver satisfactory results. The possible explanation is that they require a relatively large number of labeled data, which makes them less effective in our evaluation.

Table 2. Performance comparison results with respect to AUC-ROC, AUC-PR, and Precision@K on four datasets.

Methods	Amazon			Yelp			PubMed			Reddit		
	A-ROC	**A-PR**	**P@K**	**A-ROC**	**A-PR**	**P@K**	**A-ROC**	**A-PR**	**P@K**	**A-ROC**	**A-PR**	**P@K**
Autoencoder	0.4211	0.0539	0.4125	0.3756	0.0423	0.3874	0.5755	0.1873	0.5678	0.5183	0.0712	0.5027
Radar	0.4814	0.0753	0.4547	0.4602	0.0623	0.4929	0.5286	0.1156	0.5891	0.5039	0.0665	0.4614
DOMINANT	0.4507	0.0683	0.4421	0.3657	0.0412	0.5859	0.5842	0.2365	0.4896	0.7223	0.3477	0.6234
DeepSAD	0.3987	0.0575	0.3674	0.3973	0.0465	0.4057	0.4213	0.0483	0.4295	0.2987	0.0483	0.2561
SemiGNN	0.4726	0.0635	0.4489	0.4023	0.0414	0.5117	0.4583	0.0352	0.4645	0.5518	0.0855	0.5223
Meta-GDN	0.5026	0.0654	0.5549	0.4978	0.0585	0.5542	0.5238	0.3066	0.5317	0.5196	0.1349	0.5749
BWGNN	0.5106	0.0788	0.6554	0.5784	0.1095	0.5574	0.6497	0.3379	0.6186	0.7357	0.3572	0.7129
CAGAD	0.7894	0.1213	0.7823	0.6783	0.1327	0.6557	0.7367	0.4386	0.6534	0.8113	0.3794	0.7412
DualKD	**0.8103**	**0.1394**	**0.7998**	**0.7246**	**0.1757**	**0.6745**	**0.7614**	**0.4854**	**0.6543**	**0.8987**	**0.3959**	**0.7649**

4.3. Ablation Study

Here, we examine the influence of various components in our designed model. There are mainly two components in DualKD: the soft label distillation and the representation

distillation. Hence, we consider the following variants: (1) *DualKD-Base* is the balanced backbone model, which is the basic learning framework that removes the soft label distillation and the representation distillation. (2) *DualKD-Soft* is the learning framework incorporating the soft label distillation. (3) *DualKD-Repr* is the learning framework with the representation distillation. (4) *DualKD-Full* is our proposed model fully involving these components.

The experimental results on the four datasets are presented in Figure 2. We summarize the observations from this figure as follows. First, *DualKD-Full* performs the best while *DualKD-Base* performs the worst, suggesting that the main components we designed can substantially enhance the detection results. Second, *DualKD-Soft* outperforms *DualKD-Base* by a certain margin, which validates the effectiveness of the proposed DualKD in completing knowledge transfer for node anomaly detection. Third, *DualKD-Repr* outperforms *DualKD-Soft*, which is primarily because the balanced backbone network with representation distillation in DualKD is able to effectively utilize node information and learn highly expressive node representations. These suggest that our designed method can efficiently distill the information from the auxiliary data to the target data.

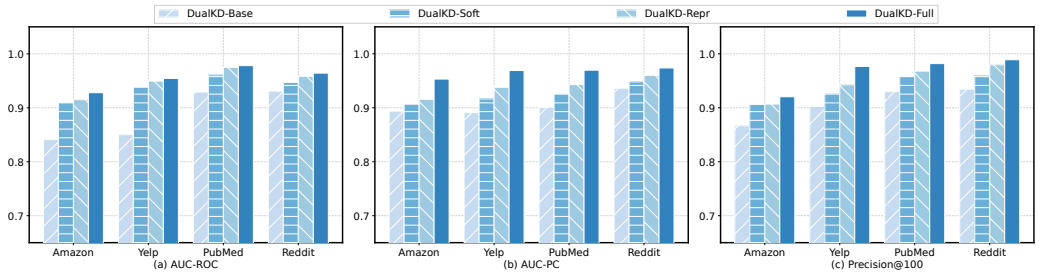

Figure 2. Detection performance of DualKD and its variants.

4.4. Role of Labeled Samples

In order to verify the effectiveness of DualKD in few-shot as well as one-shot network anomaly detection, we evaluate the performance of DualKD with different numbers of labeled anomalies on the target network (i.e., 1-shot, 3-shot, 5-shot, and 10-shot). For these settings, we adjust the batch size to 2, 4, 8, and 16, respectively. Also, we keep the number of labeled anomalies on auxiliary networks as 10. Table 3 summarizes the AUC-ROC/AUC-PR performance of DualKD under different few-shot settings. By comparing the results in Tables 2 and 3, we can see that even with only one labeled anomaly on the target network (i.e., one-shot), DualKD can still achieve good performance and outperform the baseline methods. In the meantime, we can clearly observe that the performance of DualKD increases with the growth of the number of labeled anomalies, which demonstrates that DualKD can be better fine-tuned on the target network with more labeled examples.

Table 3. Few-shot performance evaluation of DualKD with respect to AUC-ROC, AUC-PR, and Precision@K on four datasets.

Setting	Amazon			Yelp			PubMed			Reddit		
	A-ROC	**A-PR**	**P@K**	**A-ROC**	**A-PR**	**P@K**	**A-ROC**	**A-PR**	**P@K**	**A-ROC**	**A-PR**	**P@K**
1-shot	0.7810	0.1071	0.7784	0.7022	0.1569	0.7123	0.7421	0.4623	0.7731	0.8208	0.3801	0.7954
3-shot	0.7991	0.1193	0.7915	0.7094	0.1642	0.7375	0.7483	0.4681	0.7845	0.8289	0.3867	0.8026
5-shot	0.8056	0.1269	0.7842	0.7173	0.1694	0.7537	0.7537	0.4746	0.8092	0.8342	0.3897	0.8283
10-shot	0.8103	0.1394	0.7998	0.7246	0.1757	0.7745	0.7614	0.4854	0.7843	0.8426	0.3959	0.8949

4.5. Sensitivity and Robustness Analysis

In this section, we further analyze the sensitivity and robustness of the proposed DualKD. By providing different numbers of auxiliary networks during training, the model sensitivity results with respect to AUC-ROC are presented in Figure 3. Specifically, we can clearly find that (1) as the number of auxiliary networks increases, DualKD achieves constantly stronger performance on the four datasets. It shows that more auxiliary networks can provide better knowledge during the training process, which is consistent with our intuition; (2) DualKD can still achieve a relatively good performance when training with a small number of auxiliary networks (e.g., p = 2), which demonstrates the strong capability of its base model. For instance, on the Yelp data, the AUC-ROC performance only decreases by 0.042 when the amount of auxiliary networks is reduced from $p = 6$ to $p = 2$.

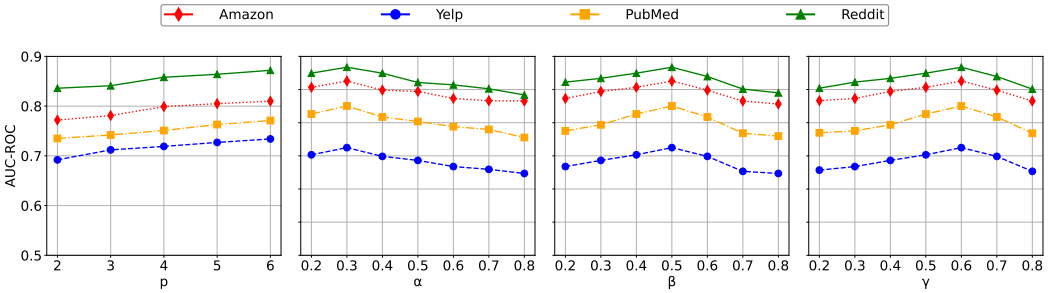

Figure 3. Sensitivity analysis for the number of auxiliary networks P and the weight α, β, and γ.

The parameters α, β, and γ are used to balance the importance of the soft and hard target losses, the short and long objective information, and the soft label and the representation distillation losses, individually. To examine their impact, we analyze different values of α, β, and γ, and present the results in Figure 3. The data show that model performance gradually improves as γ increases from a lower value, followed by a relatively sharp decline as γ continues to rise. This trend suggests that, initially, a higher γ enhances the influence of the knowledge distillation loss ($\mathcal{L}_{\mathrm{SLD}}$), enabling the student model to better replicate the decision-making process of the teacher model, thus boosting overall performance. However, as γ becomes too large, the representation distillation loss ($\mathcal{L}_{\mathrm{RDL}}$) is de-emphasized, which may hinder the model's ability to effectively learn node representations, leading to a decrease in performance. It is evident that knowledge distillation progressively contributes to performance enhancement, but if its weight is too dominant, it can significantly impair the model's representation learning capabilities. The peak performance is observed at $\gamma = 0.6$, indicating that this balance point between knowledge and representation distillation enables the model to achieve optimal results. Additionally, the results for α and β exhibit similar trends, indicating that properly considering the information, the soft and hard target losses, and the short and long objective information, is crucial to obtain better performance.

5. Conclusions

In this paper, we proposed a dual-level knowledge distillation-based graph anomaly detection framework, DualKD, to address the challenge of few-shot anomaly detection in graphs. We introduced a soft label loss which adopts the teacher model to generate soft labels that encapsulate the relative relationships between normal and anomalous nodes. We designed a representation distillation approach including both short and long representation distillation, which ensures the effective transfer of information from all graph convolutional layers of the teacher model to the student model, improving the robustness and performance of the latter. Through extensive experiments, we demonstrated the

effectiveness of DualKD in utilizing the knowledge from the teacher model to enhance the detection performance in few-shot scenarios.

Author Contributions: Conceptualization: X.L. and L.Z.; Methodology: X.L. and L.Z.; Data Curation: X.L.; Writing—Original Draft: X.L.; Writing—Review Editing: D.C.; Resources: D.C. and C.Z.; Software: L.Z.; Validation: L.Z. and Z.F.; Visualization: C.Z.; Writing—Reviewing Editing: C.Z.; Investigation: Z.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Scientific Research Fund of Key Lab of Internet Natural Language Processing of Sichuan Provincial Education Department (Grant No. INLP202302).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data available upon request due to restrictions, e.g., privacy or ethical. The data presented in this study are available upon request from the corresponding author.

Acknowledgments: The authors would like to thank all anonymous reviewers and editors for their helpful suggestions for the improvement in this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ma, X.; Wu, J.; Xue, S.; Yang, J.; Zhou, C.; Sheng, Q.Z.; Xiong, H.; Akoglu, L. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 12012–12038. [CrossRef]
2. Shen, X.; Lv, W.; Qiu, J.; Kaur, A.; Xiao, F.; Xia, F. Trust-Aware Detection of Malicious Users in Dating Social Networks. *IEEE Trans. Comput. Soc. Syst.* **2023**, *10*, 2587–2598. [CrossRef]
3. Yang, Y.; Xu, Y.; Sun, Y.; Dong, Y.; Wu, F.; Zhuang, Y. Mining Fraudsters and Fraudulent Strategies in Large-Scale Mobile Social Networks. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 169–179. [CrossRef]
4. Cui, J.; Yan, C.; Wang, C. ReMEMBeR: Ranking metric embedding-based multicontextual behavior profiling for online banking fraud detection. *IEEE Trans. Comput. Soc. Syst.* **2021**, *8*, 643–654. [CrossRef]
5. Hu, W.; Yang, Y.; Wang, J.; Huang, X.; Cheng, Z. Understanding electricity-theft behavior via multi-source data. In Proceedings of the Web Conference, Taiwan, China, 20–24 April 2020; pp. 2264–2274.
6. Li, X.; Xiao, C.; Feng, Z.; Pang, S.; Tai, W.; Zhou, F. Controlled graph neural networks with denoising diffusion for anomaly detection. *Expert Syst. Appl.* **2024**, *237*, 121533. [CrossRef]
7. Peng, Z.; Luo, M.; Li, J.; Liu, H.; Zheng, Q. ANOMALOUS: A Joint Modeling Approach for Anomaly Detection on Attributed Networks. In Proceedings of the International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; Volume 18, pp. 3513–3519.
8. Ju, W.; Yi, S.; Wang, Y.; Xiao, Z.; Mao, Z.; Li, H.; Gu, Y.; Qin, Y.; Yin, N.; Wang, S.; et al. A survey of graph neural networks in real world: Imbalance, noise, privacy and ood challenges. *arXiv* **2024**, arXiv:2403.04468.
9. Huang, K.; Zitnik, M. Graph meta learning via local subgraphs. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 5862–5874.
10. Xiao, C.; Chen, S.; Zhou, F.; Wu, J. Self-supervised few-shot time-series segmentation for activity recognition. *IEEE Trans. Mob. Comput.* **2022**, *22*, 6770–6783. [CrossRef]
11. Ding, K.; Wang, J.; Li, J.; Shu, K.; Liu, C.; Liu, H. Graph prototypical networks for few-shot learning on attributed networks. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Galway, Ireland, 19–23 October 2020; pp. 295–304.
12. Wu, Z.; Zhou, P.; Wen, G.; Wan, Y.; Ma, J.; Cheng, D.; Zhu, X. Information Augmentation for Few-shot Node Classification. In Proceedings of the IJCAI, Vienna, Austria, 23–29 July 2022; pp. 3601–3607.
13. Gao, Y.; Wang, X.; He, X.; Liu, Z.; Feng, H.; Zhang, Y. Addressing heterophily in graph anomaly detection: A perspective of graph spectrum. In Proceedings of the ACM Web Conference, Austin, TX, USA, 30 April–4 May 2023; pp. 1528–1538.
14. He, L.; Xu, G.; Jameel, S.; Wang, X.; Chen, H. Graph-Aware Deep Fusion Networks for Online Spam Review Detection. *IEEE Trans. Comput. Soc. Syst.* **2023**, *10*, 2557–2565. [CrossRef]
15. Ding, K.; Li, J.; Bhanushali, R.; Liu, H. Deep anomaly detection on attributed networks. In Proceedings of the SIAM International Conference on Data Mining, Calgary, AB, Canada, 2–4 May 2019; pp. 594–602.
16. Luo, X.; Wu, J.; Beheshti, A.; Yang, J.; Zhang, X.; Wang, Y.; Xue, S. Comga: Community-aware attributed graph anomaly detection. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Tempe, AZ, USA, 21–25 February 2022; pp. 657–665.

17. Ju, W.; Wang, Y.; Qin, Y.; Mao, Z.; Xiao, Z.; Luo, J.; Yang, J.; Gu, Y.; Wang, D.; Long, Q.; et al. Towards Graph Contrastive Learning: A Survey and Beyond. *arXiv* **2024**, arXiv:2405.11868.
18. Xiao, C.; Han, Y.; Yang, W.; Hou, Y.; Shi, F.; Chetty, K. Diffusion Model-Based Contrastive Learning for Human Activity Recognition. *IEEE Internet Things J.* **2024**, *11*, 33525–33536. [CrossRef]
19. Zhang, J.; Wang, S.; Chen, S. Reconstruction Enhanced Multi-View Contrastive Learning for Anomaly Detection on Attributed Networks. In Proceedings of the International Joint Conference on Artificial Intelligence, Vienna, Austria, 23–29 July 2022; pp. 2376–2382.
20. Zheng, Y.; Jin, M.; Liu, Y.; Chi, L.; Phan, K.T.; Chen, Y.P.P. Generative and Contrastive Self-Supervised Learning for Graph Anomaly Detection. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 12220–12233. [CrossRef]
21. Ding, K.; Zhou, Q.; Tong, H.; Liu, H. Few-shot Network Anomaly Detection via Cross-network Meta-learning. In Proceedings of the Web Conference, Ljubljana Slovenia, 19–23 April 2021; pp. 2448–2456.
22. Ruff, L.; Vandermeulen, R.A.; Görnitz, N.; Binder, A.; Müller, E.; Müller, K.; Kloft, M. Deep Semi-Supervised Anomaly Detection. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020; pp. 1–23.
23. Kumagai, A.; Iwata, T.; Fujiwara, Y. Semi-supervised anomaly detection on attributed graphs. In Proceedings of the International Joint Conference on Neural Networks, Virtual, 18–22 July 2021; pp. 1–8.
24. Zhou, S.; Huang, X.; Liu, N.; Tan, Q.; Chung, F.L. Unseen Anomaly Detection on Networks via Multi-Hypersphere Learning. In Proceedings of the SIAM International Conference on Data Mining, Alexandria, VA, USA, 28–30 April 2022; pp. 262–270.
25. Dou, Y.; Liu, Z.; Sun, L.; Deng, Y.; Peng, H.; Yu, P.S. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In Proceedings of the ACM International Conference on Information and Knowledge Management, Virtual, 19–23 October 2020; pp. 315–324.
26. Liu, Y.; Ao, X.; Qin, Z.; Chi, J.; Feng, J.; Yang, H.; He, Q. Pick and choose: A GNN-based imbalanced learning approach for fraud detection. In Proceedings of the Web Conference, Ljubljana Slovenia, 19–23 April 2021; pp. 3168–3177.
27. Liu, Z.; Dou, Y.; Yu, P.S.; Deng, Y.; Peng, H. Alleviating the inconsistency problem of applying graph neural network to fraud detection. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 1569–1572.
28. Wang, D.; Lin, J.; Cui, P.; Jia, Q.; Wang, Z.; Fang, Y.; Yu, Q.; Zhou, J.; Yang, S.; Qi, Y. A semi-supervised graph attentive network for financial fraud detection. In Proceedings of the SIAM International Conference on Data Mining, Calgary, AB, Canada, 2–4 May 2019; pp. 598–607.
29. Cui, L.; Seo, H.; Tabar, M.; Ma, F.; Wang, S.; Lee, D. Deterrent: Knowledge guided graph attention network for detecting healthcare misinformation. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, 23–27 August 2020; pp. 492–502.
30. Liu, C.; Sun, L.; Ao, X.; Feng, J.; He, Q.; Yang, H. Intention-aware heterogeneous graph attention networks for fraud transactions detection. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Singapore, 14–18 August 2021; pp. 3280–3288.
31. Tang, J.; Li, J.; Gao, Z.; Li, J. Rethinking graph neural networks for anomaly detection. In Proceedings of the International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022; pp. 21076–21089.
32. Chai, Z.; You, S.; Yang, Y.; Pu, S.; Xu, J.; Cai, H.; Jiang, W. Can Abnormality be Detected by Graph Neural Networks? In Proceedings of the International Joint Conference on Artificial Intelligence, Vienna, Austria, 23–29 July 2022; pp. 1945–1951.
33. Xiao, C.; Pang, S.; Tai, W.; Huang, Y.; Trajcevski, G.; Zhou, F. Motif-Consistent Counterfactuals with Adversarial Refinement for Graph-Level Anomaly Detection. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 3518–3526.
34. Xiao, C.; Xu, X.; Lei, Y.; Zhang, K.; Liu, S.; Zhou, F. Counterfactual graph learning for anomaly detection on attributed networks. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 10540–10553. [CrossRef]
35. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. In Proceedings of the Annual Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014.
36. Joshi, C.K.; Liu, F.; Xun, X.; Lin, J.; Foo, C.S. On representation knowledge distillation for graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *35*, 4656–4667. [CrossRef]
37. Deng, X.; Zhang, Z. Graph-free knowledge distillation for graph neural networks. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, Montreal, QC, Canada, 19–27 August 2021; pp. 2321–2327.
38. Zhang, S.; Liu, Y.; Sun, Y.; Shah, N. Graph-less Neural Networks: Teaching Old MLPs New Tricks Via Distillation. In Proceedings of the Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, 25–29 April 2022; pp. 1–21.
39. Wang, J.; Zhou, M.; Zhang, S.; Gong, Z. Generalized Few-Shot Node Classification with Graph Knowledge Distillation. *IEEE Trans. Comput. Soc. Syst.* **2024**. [CrossRef]
40. Wu, Z.; Mo, Y.; Zhou, P.; Yuan, S.; Zhu, X. Self-Training Based Few-Shot Node Classification by Knowledge Distillation. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; pp. 15988–15995.

41. Li, Y.; Xu, F.; Lee, C.G. Self-supervised metalearning generative adversarial network for few-shot fault diagnosis of hoisting system with limited data. *IEEE Trans. Ind. Inform.* **2022**, *19*, 2474–2484. [CrossRef]
42. Yang, S.; Liu, L.; Xu, M. Free Lunch for Few-shot Learning: Distribution Calibration. In Proceedings of the 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, 3–7 May 2021; pp. 1–13.
43. Ma, Y.; Bai, S.; An, S.; Liu, W.; Liu, A.; Zhen, X.; Liu, X. Transductive Relation-Propagation Network for Few-shot Learning. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, Yokohama, Japan, 7–15 January 2020; pp. 804–810.
44. Ju, W.; Yi, S.; Wang, Y.; Long, Q.; Luo, J.; Xiao, Z.; Zhang, M. A survey of data-efficient graph learning. In Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, Jeju, Republic of Korea, 3–9 August 2024; pp. 8104–8113.
45. Ju, W.; Fang, Z.; Gu, Y.; Liu, Z.; Long, Q.; Qiao, Z.; Qin, Y.; Shen, J.; Sun, F.; Xiao, Z.; et al. A comprehensive survey on deep graph representation learning. *Neural Netw.* **2024**, *173*, 106207. [CrossRef] [PubMed]
46. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; pp. 1–12.
47. Xiao, C.; Pang, S.; Xu, X.; Li, X.; Trajcevski, G.; Zhou, F. Counterfactual Data Augmentation with Denoising Diffusion for Graph Anomaly Detection. *IEEE Trans. Comput. Soc. Syst.* **2024**, *11*, 7555–7567. [CrossRef]
48. Zhuo, W.; Liu, Z.; Hooi, B.; He, B.; Tan, G.; Fathony, R.; Chen, J. Partitioning message passing for graph fraud detection. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
49. Yang, T.; Wang, Y.; Yue, Z.; Yang, Y.; Tong, Y.; Bai, J. Graph pointer neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 28 February–1 March 2022; pp. 8832–8839.
50. Wang, H.; Nie, F.; Huang, H. Globally and locally consistent unsupervised projection. In Proceedings of the AAAI Conference on Artificial Intelligence, Québec City, QC, Canada, 27–31 July 2014; Volume 28, pp. 1–6.
51. Liu, Y.; Li, Z.; Pan, S.; Gong, C.; Zhou, C.; Karypis, G. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 2378–2392. [CrossRef]
52. Rayana, S.; Akoglu, L. Collective opinion spam detection: Bridging review networks and metadata. In Proceedings of the 21th ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, 10–13 August 2015; pp. 985–994.
53. Sen, P.; Namata, G.; Bilgic, M.; Getoor, L.; Galligher, B.; Eliassi-Rad, T. Collective classification in network data. *AI Mag.* **2008**, *29*, 93–106. [CrossRef]
54. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
55. Ding, K.; Li, J.; Liu, H. Interactive anomaly detection on attributed networks. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, Melbourne, VIC, Australia, 11–15 February 2019; pp. 357–365.
56. Song, X.; Wu, M.; Jermaine, C.; Ranka, S. Conditional anomaly detection. *IEEE Trans. Knowl. Data Eng.* **2007**, *19*, 631–645. [CrossRef]
57. Skillicorn, D.B. Detecting anomalies in graphs. In Proceedings of the 2007 IEEE Intelligence and Security Informatics, New Brunswick, NJ, USA, 23–24 May 2007; IEEE: New York, NY, USA, 2007; pp. 209–216.
58. Zhou, C.; Paffenroth, R.C. Anomaly detection with robust deep autoencoders. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 665–674.
59. Li, J.; Dani, H.; Hu, X.; Liu, H. Radar: Residual analysis for anomaly detection in attributed networks. In Proceedings of the IJCAI, Melbourne, Australia, 19–25 August 2017; Volume 17, pp. 2152–2158.
60. Schütze, H.; Manning, C.D.; Raghavan, P. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, UK, 2008; Volume 39.
61. Pang, G.; Shen, C.; Van Den Hengel, A. Deep anomaly detection with deviation networks. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 353–362.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.