Subtractive Aggregation for Attributed Network Anomaly Detection

Shuang Zhou, Qiaoyu Tan, Zhiming Xu, Xiao Huang, Fu-lai Chung
{csszhou,xiaohuang,cskchung}@comp.polyu.edu.hk,{qiaoytan,zhiming.xu}@polyu.edu.hk
The Hong Kong Polytechnic University
China

ABSTRACT

Attributed network anomaly detection has become an essential tool in various networked systems such as social media and e-commerce businesses. It aims to detect nodes that significantly deviate from their corresponding background. In conventional anomaly detection, the background is defined as the vast majority. But in networks, anomalies can be local and look normal when compared with the majority. While several efforts have explored to consider communities as the background, it remains challenging to learn suitable communities for effective anomaly detection. Also, the patterns of anomalies are unknown and it is nontrivial to define criteria of anomalies. To bridge the gap, in this paper, we argue that, by using appropriate models, it is sufficient to simply consider neighbor nodes as the background to detect anomalies. Correspondingly, we propose a novel abnormality-aware graph neural network (AAGNN). It utilizes subtractive aggregation to represent each node as the deviation from its neighbors (the background). Normal nodes with high confidence are employed as labels to learn a tailored hypersphere as the criterion of anomalies. Experiments demonstrate that AAGNN surpasses state-of-the-art methods significantly.

KEYWORDS

anomaly detection; attributed networks; graph neural networks

ACM Reference Format:
Shuang Zhou, Qiaoyu Tan, Zhiming Xu, Xiao Huang, Fu-lai Chung. 2021. Subtractive Aggregation for Attributed Network Anomaly Detection. In *Proceedings of the 30th ACM Int'l Conf. on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3459637.3482195

1 INTRODUCTION

Attributed network anomaly detection (ANAD) has attracted increasing attention from both industry and academia, since attributed networks are widely witnessed in various real-world systems, including social media [10, 23], molecular networks [27], and academic networks [24]. In these systems, a small number of anomalies (e.g., fraudsters and terrorists) could cause serious issues. The goal of ANAD aims to automatically identify anomalies, whose patterns or behaviors significantly deviate from background [1, 5, 14].

CIKM '21, November 1–5, 2021, Virtual Event, Australia.

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482195

There are two main difficulties in developing an effective ANAD model. First, differ from the anomalies in other domains (e.g., multi-dimensional data, image), where the background is usually defined as the majority instances [17], network anomalies are more complex with local patterns [14]. These local anomalies often align with the vast majority yet deviate from the local community, making them hard to be detected in practice. Although several efforts have been made to identify suitable community as background [8, 20], it is a non-trivial task to effectively devise a good community. Second, it is hard to characterize the properties of abnormal instances in practice since they usually present unknown patterns [17]. To this end, existing efforts [13, 18] assort to measure nodes' pattern deviation via reconstruction residuals so as to characterize abnormality. However, residual-based methods may be sub-optimal in identifying local anomalies due to their reliance on all instances. Thus, defining a suitable background that is easy to obtain and could present local abnormal patterns clearly still remains an open problem.

Inspired by the fact that anomalies are rare and tend to link with normal nodes in networks [6, 17], a natural idea is to simply employ a node's local neighborhood structure as the background to measure its pattern deviation. Specifically, we can first leverage the advanced graph neural networks [12, 25] (GNNs) to map normal and abnormal nodes into different latent areas by considering their local neighborhood communities, and then adopt a classifier to discriminate the representations of normal nodes and anomalies.

The challenges of developing such an intuitive solution are twofold. The first is how to effectively determine the pattern deviation of nodes based on neighborhood community. Directly applying established GNNs architectures to discriminate the difference between normal nodes and anomalies by leveraging their community structures is problematic. It is mainly because GNNs assume that nodes linked together tend to be similar, while in ANAD setting, we hypothesize that the degree of deviation for normal nodes and anomalies compared with their neighborhood are different. Second, since anomalies are rare while the majority of samples are normal in practice, how to define an effective learning objective to leverage such imbalanced data samples is another challenge. Although traditional training objectives (i.e., cross-entropy loss) can be directly applied, it may be unsuitable as it omits the extremely imbalanced data distribution [1]. Several recent efforts [3, 7, 19] propose to minimize reconstruction error for model training, but reconstruction-based objectives do not directly target for anomaly detection and lead to sub-optimal performance [16]. Thus, a tailored learning objective is needed to effectively train ANAD model.

To address the aforementioned issues, in this paper, we propose a simple yet effective Abnormality-Aware Graph Neural Network framework, termed as AAGNN, for ANAD. The core ideas are twofold. First, it utilizes a subtractive aggregation approach to represent

each node by considering the pattern deviation between itself and its neighborhood community. In this way, the generated node representations are anomaly-distinguishable to some extend. This is because the representations of normal nodes tend to be similar as they have slight deviation with their neighbors, while anomalies tend to deviate from their neighbors significantly. As a result, normal nodes tend to locate together in the latent space, while abnormal ones reside in different areas. Second, it devises a novel hypersphere learning objective to effectively distinguish the normal and abnormal samples by enclosing normal nodes within a hypersphere while retaining anomalies outside of the hypersphere in the latent space. Normal nodes with high confidence are employed as labels to train the model. After training, AAGNN can learn compact normal patterns and reasonably infer the nodes far away from the hypersphere center (i.e., significantly deviate from the normal patterns) as anomalies (shown in Figure 4 (b)).

Our main contributions are summarized as follows:

- We propose a novel graph neural network (AAGNN) to effectively characterize node abnormality by measuring its pattern deviation with corresponding neighborhood.
- We devise an effective hypersphere learning objective to optimize the proposed AAGNN model.
- Empirical results on three real-world datasets demonstrate the superiority of AAGNN over the state-of-the-art methods in detecting abnormal nodes in networks.

2 PROBLEM DEFINITION

In this paper, we use bold uppercase letters for matrices (e.g., $\mathbf{X}$), bold lowercase letters for vectors (e.g., $\mathbf{h}$), lowercase letters for scales (e.g., s) and calligraphic fonts to denote set (e.g., $\mathcal{V}$). Notably, we are given an attributed network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ with n nodes as input, where $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ denotes the set of nodes ($|\mathcal{V}| = n$) and $\mathcal{E}$ is the set of edges. Each node is associated with an attribute vector $\mathbf{x}_i \in \mathbb{R}^f$, in which f is the feature dimension. $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{n \times f}$ presents the feature matrix for all nodes.

Definition 1. Attributed Network Anomaly Detection. Given an attributed network, the goal is to measure nodes' anomaly score, such that anomalies that significantly deviate from the background would have large anomaly scores and be detected.

3 AAGNN MODEL DESIGN

In this section, we present the main component of the proposed AAGNN model and describe the model learning process for ANAD. An overview of AAGNN is shown in Figure 1.

3.1 Abnormality-aware Graph Neural Layer

Our AAGNN consists of a single abnormality-aware graph neural layer, which is tailored for ANAD. It leverages subtractive aggregation to measure nodes' pattern deviation with the neighborhood (i.e., characterize node abnormality). Intuitively, if a node owns rare patterns or presents strange behaviors, its pattern deviation with the neighborhood is significant; whereas normal nodes generally present similar patterns with their neighborhood, thus owning trivial abnormality.

For a given node i, the input of the layer is node attribute vector $\mathbf{x}_i \in \mathbb{R}^f$, the output is an abnormality-aware node representation

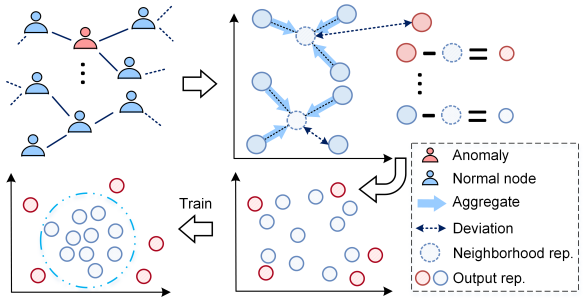

Figure 1: Overview of AAGNN model. It utilizes subtractive aggregation to represent each node as the pattern deviation from its neighborhood. Since normal nodes own trivial deviation (i.e., low abnormality), their output representations are mapped closely; whereas anomalies present significant deviation (i.e., large abnormality), their output representations are mapped to faraway regions. After training, normal nodes are further enclosed to discriminate anomalies. (Red color denotes abnormal, blue color denotes normal.)

$\mathbf{h}_i \in \mathbb{R}^d$. To that end, firstly, a shared linear mapping function $\phi(\cdot; \mathbf{W})$ parameterized by $\mathbf{W} \in \mathbb{R}^{f \times d}$ is applied to all the nodes to get low-dimensional representation $\mathbf{z}_i \in \mathbb{R}^d$. Then, the abnormality-aware node representation is computed by using its own representation subtract the aggregated representation from the corresponding k-hop neighborhood. Formally, AAGNN learns the abnormality-aware representation of node i by:

$$\begin{aligned} \mathbf{z}_i &= \phi(\mathbf{x}_i; \mathbf{W}), \\ \mathbf{h}_i &= \sigma\left(\mathbf{z}_i - \textsc{Aggregate}\left(\mathbf{z}_j, \forall j \in \mathcal{N}_i^k\right)\right), \end{aligned} \tag{1}$$

where $\mathcal{N}_i^k$ denotes the k-hop neighborhood of node i, $\mathbf{x}_j$ is the attribute vector of a neighboring node j, and $\sigma(\cdot)$ is a non-linear activation function. We further propose two different aggregators to aggregate node information from their k-hop neighborhood.

Mean aggregator. It aggregates the k-hop neighborhood information by simply averaging the representations:

$$\mathbf{h}_i = \sigma(\mathbf{z}_i - \frac{1}{|\mathcal{N}_i^k|} \sum_{j \in \mathcal{N}_i^k} \mathbf{z}_j). \tag{2}$$

Attention aggregator. We also examine attention mechanism for information aggregation, which is a single-layer neural network parameterized by a weight vector $\mathbf{a} \in \mathbb{R}^{2d}$. For a node i, the attention score α_{ij} indicates the similarity between node i and node j:

$$\alpha_{ij} = \frac{exp(LeakyReLU(\mathbf{a}^T[\mathbf{z}_i \oplus \mathbf{z}_j]))}{\sum_{j \in \mathcal{N}_i^k} exp(LeakyReLU(\mathbf{a}^T[\mathbf{z}_i \oplus \mathbf{z}_j]))}, \tag{3}$$

where $\oplus$ denotes vector concatenation. The final abnormality-aware representation for node i is generated by:

$$\mathbf{h}_i = \sigma(\mathbf{z}_i - \sum_{j \in \mathcal{N}_i^k} \alpha_{ij} \mathbf{z}_j). \tag{4}$$

With subtractive aggregation, the two AAGNN variants incorporate both attributes and network structure to measure the pattern deviation between nodes and their neighborhood, and ultimately generate abnormality-aware node representations. Because normal nodes, which own trivial pattern deviation with neighborhood, are

mapped closely together; whereas anomalies, which present significant deviation (i.e., large abnormality), are mapped to faraway regions in the latent space (shown in Figure 1).

Algorithm 1 Abnormality-aware Graph Neural Network AAGNN.

Input: Attributed network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, training epochs T, weight decay λ.

Output: A node list with anomaly score.

1: Randomly initialize AAGNN, based on Eq. 1 to get $\mathbf{h}$;
2: Calculate the hypersphere center $\mathbf{c} = \frac{1}{|\mathcal{V}|}\sum_{i\in\mathcal{V}} \mathbf{h}_i$;
3: Estimate the distance between all nodes and the center vector via $\|\mathbf{h}_j - \mathbf{c}\|_2^2, \forall j \in \mathcal{V}$, and label $p\%$ of the nodes with smallest distances as pseudo-labeled normal data $\mathcal{S}$;
4: Randomly split $\mathcal{S}$ into training set $\mathcal{R}$ and validation set $\mathcal{D}$ with appropriate proportion (e.g., 3:2);
5: Take the rest of nodes as testing set $\mathcal{T} = \mathcal{V} - \mathcal{S}$;
6: **while** $t < T$ **do**
7: Minimize $\mathcal{L} = \frac{1}{|\mathcal{R}|}\sum_{i\in\mathcal{R}} \|\mathbf{h}_i - \mathbf{c}\|_2^2 + \frac{\lambda}{2}\|\Theta\|_F^2$;
8: Update Θ by using stochastic gradient descent;
9: Evaluate model performance on the validation set $\mathcal{D}$;
10: **end while**
11: Compute anomaly scores based on Eq. 6 for the testing set $\mathcal{T}$.

3.2 Hypersphere Learning Objective

In previous section, we introduce how to use AAGNN to obtain abnormality-aware node representation. The follow-up question is how to train our model in order to facilitate anomaly detection task. Following common protocol [21], we optimize our model by designing a simple hypersphere learning objective. The intuitive idea is to encode normal instances within a hypersphere while retaining anomalies outside the hypersphere by jointly learning the network parameters and minimizing the volume of the data description hypersphere.

Formally, the hypersphere learning objective is defined as:

$$\min_{\Theta} \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{h}_i - \mathbf{c}\|_2^2 + \frac{\lambda}{2}\|\Theta\|_F^2, \tag{5}$$

where $\mathbf{c} \in \mathbb{R}^d$ is hypersphere center obtained by averaging all the normal node representations, Θ denotes all the learnable parameters. The first term is a quadratic loss for penalizing the distance of every node representation to the center $\mathbf{c}$. The second term is a network weight decay regularizer with hyperparameter $\lambda > 0$. When minimizing the objective function, the hypersphere is contracted. By training the model with normal nodes, it will extract the common patterns of normal nodes such that the description boundary of normal nodes can be obtained and the anomalies can be discriminated. Specifically, for a given test node i, we can naturally define the anomaly score s by the distance of the node representation to the hypersphere center in the latent space, i.e.,

$$s(i) = \|\mathbf{h}_i - \mathbf{c}\|_2^2. \tag{6}$$

3.3 Model Learning

Ideally, the proposed learning objective encourages learning a hypersphere by enclosing normal nodes to discriminate anomalies.

Table 1: Data statistics of the three real-world attributed networks with injected anomalies.

Dataset	# nodes	# edges	# attributes	# anomalies
BlogCatalog	5,196	171,743	8,189	300
Flickr	7,575	239,738	12,047	450
PubMed	19,797	44,338	500	600

To this end, it requires labeled normal instances for model training. However, such a requirement is not available in practice, since it is time-consuming and expensive to filter the normal data out, especially when the data is in large scale. To address this issue, we propose to use pseudo-labeled normal data for model training, without requiring any ground-truth. It is worth to note that the pseudo-labeled normal instances are allowed to contain false positive samples, as long as the majority of them are normal.

In particular, the pseudo-labeled normal data can be obtained by the virtue of AAGNN. Specifically, we find that, even after random initialization, AAGNN can generate anomaly-distinguishable node representations, i.e., normal node representations generally are close to each other, whereas anomaly representations tend to deviate far away from the normal ones in the latent space. Inspired by this observation, we propose to generate pseudo-labeled normal data with two steps: (1) We simply average all the nodes' representations and adopt the averaged result as the hypersphere center $\mathbf{c}$; (2) We calculate the Euclidean distance between all nodes and the center point, and then take $p\%$ of nodes with smallest distances as the pseudo-labeled normal instances for training, while using the remained nodes for testing. After training, we make use of Eq. 6 to calculate the anomaly score for the testing set. The detailed algorithm is summarized in Algorithm 1.

4 EXPERIMENTS

We evaluate AAGNN on several datasets and aim to answer three questions. **Q1:** Compared with the state-of-the-art anomaly detection models, can AAGNN achieve better performance? **Q2:** What are the impacts of hyperparameters on AAGNN? **Q3:** How much does each component of AAGNN contribute?

Datasets. We employ three real-world attributed networks (BlogCatalog, Flickr, and PubMed) that have been widely used in prior researches [4, 11, 15, 26] to evaluate model performance. Due to the shortage of ground-truth anomalies, we follow the perturbation scheme introduced in [4, 22] to inject a combined set of anomalies (i.e., structural anomalies and contextual anomalies) for each dataset. The statistics of the three datasets are shown in Table 1.

Baselines. We compare our methods with five anomaly detection baselines, including residual analysis method (Radar [13]), community analysis method (MADAN [9]), and deep learning methods (DOMINANT [3], AnomalyDAE [7], and AEGIS [2]). For all baselines, we retain the settings described in the corresponding papers.

Implementation Details. We implement two AAGNN model variants (AAGNN-M and AAGNN-A) by adopting the mean aggregator and the attention aggregator, respectively. Both variants consider nodes' one-hop neighborhood to capture abnormality. The representation dimension is set as 256. We set $p = 50$ (see Step 3 of Algorithm 1) to obtain the pseudo-labeled normal data for model training, and consider the remaining nodes as testing set.

Table 2: **Performance comparison results w.r.t. AUC and AUPR on three datasets.**

Methods	BlogCatlog		Flickr		PubMed	
	AUC	AUPR	AUC	AUPR	AUC	AUPR
Radar	0.6970	0.2851	0.6985	0.2747	0.6688	0.2971
MADAN	0.7273	0.2780	0.7104	0.2749	0.6467	0.1132
Dominant	0.7416	0.3381	0.7382	0.3245	0.7460	0.2997
AnomalyDAE	0.7427	0.3381	0.7381	0.3246	0.7497	0.3376
AEGIS	0.7429	0.3397	0.7381	0.3242	0.7732	0.3735
AAGNN-M	0.8171	0.4336	**0.8456**	**0.4224**	0.8459	0.3988
AAGNN-A	**0.8184**	**0.4354**	0.8299	0.4209	**0.8564**	**0.4281**

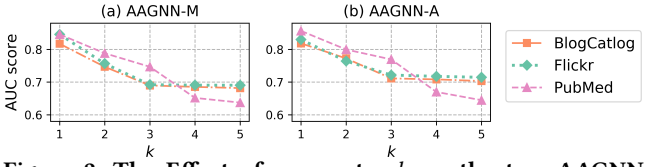

Figure 2: The Effect of parameter k on the two AAGNN model variants.

4.1 Evaluation

To answer **Q1**, we report the results of all the methods in terms of AUC and AUPR scores in Table 2. We have the following observations: (1) The proposed AAGNN models outperform all the baselines on the three datasets. In particular, compared with the best results of the baselines, our framework obtains a significant improvement, e.g., about 9.1% on AUC and 9.6% on AUPR on average. The main reason is that our AAGNN model can produce abnormality-aware representation and further map the normal node representations close to each other while mapping the anomalies far away, thus discriminating normal nodes with anomalies. (2) The AAGNN model variant with the attention-aggregator achieves better performance in two datasets. The reason might be that it leverages attention mechanism and attaches more weights on the similar neighbors w.r.t. the target node when aggregating neighborhood information. It helps the normal nodes own smaller pattern deviation with their neighborhood (i.e., smaller abnormality), thus being discriminated from anomalies. (3) Deep learning methods outperform the conventional anomaly detection methods as they are adept at dealing with the high-dimensional node attributes and the sparse, complex network structures.

4.2 Parameter Analysis

We now explore the impact of different k-hop neighborhood towards AAGNN in order to answer **Q2**. Specifically, we vary k from 1 to 5 and summarize the results w.r.t. AUC score in Figure 2.

We can see that the performance of the two AAGNN variants decreases when the k value increases. It can be explained by the Homophily principle, i.e., nodes tend to be more similar with their one-hop neighbors than those far-away neighbors. Therefore, the pattern deviation (i.e., abnormality) of normal nodes is relatively smaller when only considering one-hop neighborhood, which enables the normal nodes to be more distinguishable with anomalies. So considering effectiveness, we recommend setting k a small value (e.g.,1) to capture node abnormality for ANAD.

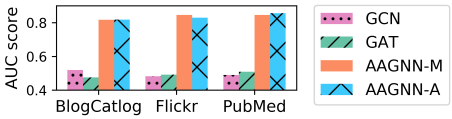

Figure 3: Ablation study to analyze the effect of abnormality-aware representation.

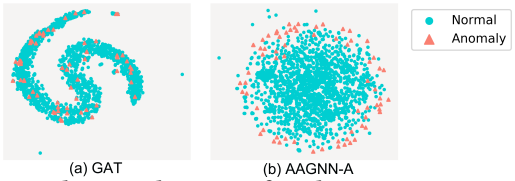

Figure 4: The visualization of node representations from the trained models. The proposed model can generate abnormality-aware representations such that the anomalies and normal nodes are distinguishable in latent space.

4.3 Ablation Study

The core component of AAGNN is the abnormality-aware graph neural layer, a tailored feature extractor to obtain abnormality-aware representation. To analyze its effect (answer **Q3**), we conduct an ablation study by comparing it with other GNNs while keeping the same training process. On top of AAGNN-M and AAGNN-A, we adopt GCN [12] and GAT [25] to obtain node representation for model training. The results are shown in Figure 3.

It is obvious that the two model variants based on GCN and GAT cannot attain satisfactory performance. It is mainly because general GNNs can hardly characterize node abnormality [2]. In contrast, AAGNN can measure nodes' pattern deviation with their neighborhood to characterize abnormality, such that normal nodes have similar representations (i.e., mapped closely to each other in the latent space), while anomalies own different representations from the normal (i.e., mapped to different regions with the normal nodes in the latent space). To support our claims, we provide a visualization of the node representations from the trained GAT-based model and AAGNN-A model in Figure 4. Specifically, for each model, we randomly select 2000 node representations from the Flickr dataset. We can observe that (1) for GAT, the representations of normal nodes and anomalies are mixed indistinguishably, making it hard to identify anomalies; (2) But the node representations from our model are anomaly-distinguishable. In short, the above results verify the superiority of abnormality-aware representation for ANAD.

5 CONCLUSION

In this paper, we propose a novel model called Abnormality-Aware Graph Neural Network (AAGNN) for attributed network anomaly detection. AAGNN characterizes node abnormality with subtractive aggregation to represent each node as the pattern deviation from its neighborhood. The learned node representations are anomaly-distinguishable. We further equip AAGNN with a hypersphere learning objective to enable unsupervised anomaly detection. After training, normal nodes representations are encouraged to cluster around a hypersphere center, whereas anomaly representations remain outside of the hypersphere. Through empirical evaluations, we demonstrate that AAGNN outperforms state-of-the-art methods significantly.

REFERENCES

[1] Leman Akoglu, Hanghang Tong, and Danai Koutra. 2015. Graph based anomaly detection and description: a survey. *Data mining and knowledge discovery* 29, 3 (2015), 626–688.

[2] Kaize Ding, Jundong Li, Nitin Agarwal, and Huan Liu. 2020. Inductive anomaly detection on attributed networks. In *29th International Joint Conference on Artificial Intelligence, IJCAI 2020*. International Joint Conferences on Artificial Intelligence, 1288–1294.

[3] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. 2019. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 594–602.

[4] Kaize Ding, Jundong Li, and Huan Liu. 2019. Interactive anomaly detection on attributed networks. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 357–365.

[5] Kaize Ding, Qinghai Zhou, Hanghang Tong, and Huan Liu. 2021. Few-shot Network Anomaly Detection via Cross-network Meta-learning. In *Proceedings of the Web Conference 2021*. 2448–2456.

[6] Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 315–324.

[7] Haoyi Fan, Fengbin Zhang, and Zuoyong Li. 2020. AnomalyDAE: Dual autoencoder for anomaly detection on attributed networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5685–5689.

[8] Jing Gao, Feng Liang, Wei Fan, Chi Wang, Yizhou Sun, and Jiawei Han. 2010. On community outliers and their efficient detection in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 813–822.

[9] Leonardo Gutiérrez-Gómez, Alexandre Bovet, and Jean-Charles Delvenne. 2020. Multi-scale anomaly detection on attributed networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 678–685.

[10] Xia Hu, Jiliang Tang, and Huan Liu. 2014. Online social spammer detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 28.

[11] Xiao Huang, Jundong Li, and Xia Hu. 2017. Label informed attributed network embedding. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. 731–739.

[12] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*.

[13] Jundong Li, Harsh Dani, Xia Hu, and Huan Liu. 2017. Radar: Residual Analysis for Anomaly Detection in Attributed Networks.. In *IJCAI*. 2152–2158.

[14] Ninghao Liu, Xiao Huang, and Xia Hu. 2017. Accelerated Local Anomaly Detection via Resolving Attributed Networks.. In *IJCAI*. 2337–2343.

[15] Ninghao Liu, Qiaoyu Tan, Yuening Li, Hongxia Yang, Jingren Zhou, and Xia Hu. 2019. Is a single vector enough? exploring node polysemy for network embedding. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 932–940.

[16] Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. 2021. Anomaly Detection on Attributed Networks via Contrastive Self-Supervised Learning. *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[17] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. 2021. Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–38.

[18] Zhen Peng, Minnan Luo, Jundong Li, Huan Liu, and Qinghua Zheng. 2018. ANOMALOUS: A Joint Modeling Approach for Anomaly Detection on Attributed Networks.. In *IJCAI*. 3513–3519.

[19] Zhen Peng, Minnan Luo, Jundong Li, Luguo Xue, and Qinghua Zheng. 2020. A Deep Multi-View Framework for Anomaly Detection on Attributed Networks. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[20] Bryan Perozzi, Leman Akoglu, Patricia Iglesias Sánchez, and Emmanuel Müller. 2014. Focused clustering and outlier detection in large attributed graphs. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1346–1355.

[21] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. 2018. Deep one-class classification. In *International conference on machine learning*. PMLR, 4393–4402.

[22] Xiuyao Song, Mingxi Wu, Christopher Jermaine, and Sanjay Ranka. 2007. Conditional anomaly detection. *IEEE Transactions on knowledge and Data Engineering* 19, 5 (2007), 631–645.

[23] Qiaoyu Tan, Ninghao Liu, and Xia Hu. 2019. Deep representation learning for social network analysis. *Frontiers in big Data* 2 (2019), 2.

[24] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 990–998.

[25] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations* (2018).

[26] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. 2016. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*. PMLR, 40–48.

[27] Xiang Yue, Zhen Wang, Jingong Huang, Srinivasan Parthasarathy, Soheil Moosavinasab, Yungui Huang, Simon M Lin, Wen Zhang, Ping Zhang, and Huan Sun. 2020. Graph embedding on biomedical networks: methods, applications and evaluations. *Bioinformatics* 36, 4 (2020), 1241–1251.