

# ANOMALYDAE: DUAL AUTOENCODER FOR ANOMALY DETECTION ON ATTRIBUTED NETWORKS

Haoyi Fan<sup>1</sup>, Fengbin Zhang<sup>1</sup>, Zuoyong Li<sup>2</sup>

<sup>1</sup>School of Computer Science and Technology,  
Harbin University of Science and Technology, Harbin 150080, China.

<sup>2</sup>Fujian Provincial Key Laboratory of Information Processing and Intelligent Control,  
Minjiang University, Fuzhou 350121, China.  
{isfanhy, zhangfengbin}@hrbust.edu.cn, fzulzytdq@126.com.

## ABSTRACT

Anomaly detection on attributed networks aims at finding nodes whose patterns deviate significantly from the majority of reference nodes, which is pervasive in many applications such as network intrusion detection and social spammer detection. However, most existing methods neglect the complex cross-modality interactions between network structure and node attribute. In this paper, we propose a deep joint representation learning framework for anomaly detection through a dual autoencoder (AnomalyDAE), which captures the complex interactions between network structure and node attribute for high-quality embeddings. Specifically, AnomalyDAE consists of a structure autoencoder and an attribute autoencoder to learn both node embedding and attribute embedding jointly in latent space. Moreover, attention mechanism is employed in structure encoder to learn the importance between a node and its neighbors for an effective capturing of structure pattern, which is important to anomaly detection. Besides, by taking both the node embedding and attribute embedding as inputs of attribute decoder, the cross-modality interactions between network structure and node attribute are learned during the reconstruction of node attribute. Finally, anomalies can be detected by measuring the reconstruction errors of nodes from both the structure and attribute perspectives. Extensive experiments on real-world datasets demonstrate the effectiveness of the proposed method.

**Index Terms**— Anomaly detection, attributed networks, dual autoencoder, graph signal processing

## 1. INTRODUCTION

Attributed networks [1] are ubiquitous in the real world such as social networks [2], communication networks [3], and product co-purchase networks [4], in which each node is associated with a rich set of attributes or characteristics, in addition to the raw network topology.

Anomaly detection on attribute networks aims at finding nodes whose patterns or behaviors significantly deviate from the reference nodes, which has a broad impact on various domains such as network intrusion detection [5], system fault diagnosis [6], and social spammer detection [7]. Recently, there is a growing interest in researches about anomaly detection on attributed networks. Some of them study the problem of community-level anomalies detection by comparing the current node with other reference nodes within the same community [8] or measuring the quality of connected subgraphs [9]. Some of them conduct anomaly analysis through subspace selection of node feature [10, 11]. While some recent residual analysis based methods attempt to find anomalies by assuming that anomalies cannot be approximated from other reference nodes [12, 13].

Although above mentioned algorithms had their fair share of success, these methods either suffer from severe computational overhead caused by shallow learning mechanisms and subspace selection, or neglect the complex interactions between nodes and attributes by only learning the representations for nodes [14], while interactions between two different modality sources are of great importance for anomaly detection task to capture both structure and attribute induced anomalies. To alleviate the above-mentioned problems, in this paper, we propose a deep joint representation learning framework for **anomaly** detection through a **dual autoencoder (AnomalyDAE)**, which captures the complex interactions between the network structure and node attribute for high-quality embeddings. Different from [14], which employs one single graph convolutional network (GCN) [15] based encoder for node embedding, **AnomalyDAE consists of a structure autoencoder and an attribute autoencoder to learn the latent representation of nodes and attributes jointly by reconstructing the original network topologies and node attributes respectively.** Then, anomalies in the network are detected by measuring the reconstruction errors of nodes from both the structure and attribute perspectives.

In sum, the main contributions of this paper are as follows:

Corresponding authors: Fengbin Zhang, Zuoyong Li.

**Table 1. Notations.**

Notation	Description
$\mathcal{G}$	Attributed network.
$\mathcal{V}$	A set of nodes in network.
$\mathcal{A}$	A set of node attributes in network.
$\mathcal{E}$	A set of edges in network.
$M$	The number of nodes.
$N$	The dimension of attribute.
$D$	The dimension of embedding.
$\mathbf{A} \in \mathbb{R}^{M \times M}$	Adjacency matrix of a network.
$\mathbf{X} \in \mathbb{R}^{M \times N}$	Attribute matrix of all nodes.
$\mathbf{Z}^{\mathcal{V}} \in \mathbb{R}^{M \times D}$	Latent embedding of nodes.
$\mathbf{Z}^{\mathcal{A}} \in \mathbb{R}^{N \times D}$	Latent embedding of attributes.

- We propose a deep joint representation learning framework for anomaly detection on attributed networks via a dual autoencoder where the complex cross-modality interactions between the network structure and node attribute are captured, and the anomalies are measured from both the structure and attribute perspectives.
- We conduct extensive experiments on multiple real-world datasets, and the results show that AnomalyDAE consistently outperforms state-of-the-art deep model significantly, with up to 22.32% improvement in terms of the ROC AUC score. The source codes<sup>1</sup> are publicly available.

## 2. NOTATIONS AND PROBLEM STATEMENT

In this section, we formally define the frequently-used notations and the studied problem. The notations used in this paper are summarized in Table 1.

**Definition 1. Attributed Network**  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{X}\}$  is defined as an undirected graph with  $M=|\mathcal{V}|$  nodes and  $|\mathcal{E}|$  edges, each of nodes is associated with a  $N=|\mathbf{X}|$  dimension attribute.

**Problem 1.** Given an attributed network  $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{X}\}$ , our goal is to detect the nodes that are rare and differ significantly from the majority of the reference nodes in terms both the structure and attribute information of the nodes. More formally, we aim to learn a score function  $f : \mathcal{V}_i \mapsto y_i \in \mathbb{R}$ , to classify sample  $x_i$  based on the threshold  $\lambda$ :

$$y_i = \begin{cases} 1, & \text{if } f(\mathcal{V}_i) \geq \lambda, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

where  $y_i$  denotes the label of sample  $x_i$ , with 0 being the normal class and 1 the anomalous class.

<sup>1</sup><https://github.com/haoyfan/AnomalyDAE>

## 3. METHOD

In this section, we introduce the proposed AnomalyDAE in detail. As shown in Fig. 1, AnomalyDAE is an end-to-end joint representation learning framework that consists of a structure autoencoder for network structure reconstruction, and an attribute autoencoder for node attributes reconstruction. Take the learned node embedding from the structure encoder and the learned attribute embedding from the attribute encoder as inputs, the interactions between the network structure and the node attribute are jointly captured by both structure decoder and attribute decoder during the training. Finally, anomalies in the network can be measured by the reconstruction errors of network structure and node attribute.

### 3.1. Structure Autoencoder

In order to obtain sufficient representative high-level node features, structure encoder firstly transforms the original observed node attribute  $\mathbf{X}$  into the low-dimensional latent representation  $\tilde{\mathbf{Z}}^{\mathcal{V}}$ , which is shown as follows:

$$\tilde{\mathbf{Z}}^{\mathcal{V}} = \sigma(\mathbf{X}\mathbf{W}^{\mathcal{V}^{(1)}} + \mathbf{b}^{\mathcal{V}^{(1)}}) \quad (2)$$

where  $\sigma(\bullet)$  is the activation function such as ReLU or Tanh,  $\mathbf{W}^{\mathcal{V}^{(1)}} \in \mathbb{R}^{D_{in}^{\mathcal{V}^{(1)}} \times D_{out}^{\mathcal{V}^{(1)}}}$  and  $\mathbf{b}^{\mathcal{V}^{(1)}} \in \mathbb{R}^{D_{out}^{\mathcal{V}^{(1)}}}$  are the weight and bias learned by encoder,  $D_{in}^{\mathcal{V}^{(1)}}$  and  $D_{out}^{\mathcal{V}^{(1)}}$  are the dimensionalities of  $\mathbf{X}$  and  $\tilde{\mathbf{Z}}^{\mathcal{V}}$ , respectively.

Given the transformed node embedding  $\tilde{\mathbf{Z}}^{\mathcal{V}}$ , a graph attention layer [16] is then employed to aggregate the representation from neighbor nodes, by performing a shared attentional mechanism on the nodes:

$$e_{i,j} = \text{attn}(\tilde{\mathbf{Z}}_i^{\mathcal{V}}, \tilde{\mathbf{Z}}_j^{\mathcal{V}}) = \sigma(\mathbf{a}^T \cdot [\mathbf{W}^{\mathcal{V}^{(2)}} \tilde{\mathbf{Z}}_i^{\mathcal{V}} \parallel \mathbf{W}^{\mathcal{V}^{(2)}} \tilde{\mathbf{Z}}_j^{\mathcal{V}}]) \quad (3)$$

where  $e_{i,j}$  is the importance weight of node  $\mathcal{V}_i$  to node  $\mathcal{V}_j$ ,  $\text{attn}(\bullet)$  denotes the neural network parameterized by weights  $\mathbf{a} \in \mathbb{R}^D$  and  $\mathbf{W}^{\mathcal{V}^{(2)}} \in \mathbb{R}^{\frac{D}{2} \times D_{out}^{\mathcal{V}^{(1)}}}$  that shared by all nodes,  $\parallel$  denotes the concatenate operation. Then, the final importance weight  $\gamma_{i,j}$  is normalized through the softmax function:

$$\gamma_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k \in \mathcal{N}_i} \exp(e_{i,k})} \quad (4)$$

where  $\mathcal{N}_i$  denotes the neighbors of node  $\mathcal{V}_i$ , which is provided by adjacency matrix  $\mathbf{A}$ , and its final embedding  $\mathbf{Z}_i^{\mathcal{V}}$  can be obtained by weighted sum based on the learned importance weights as follows:

$$\mathbf{Z}_i^{\mathcal{V}} = \sum_{k \in \mathcal{N}_i} \gamma_{i,k} \cdot \tilde{\mathbf{Z}}_k^{\mathcal{V}} \quad (5)$$

Finally, structure decoder takes the final node embeddings  $\mathbf{Z}^{\mathcal{V}}$  as inputs to decode them for reconstruction of the original

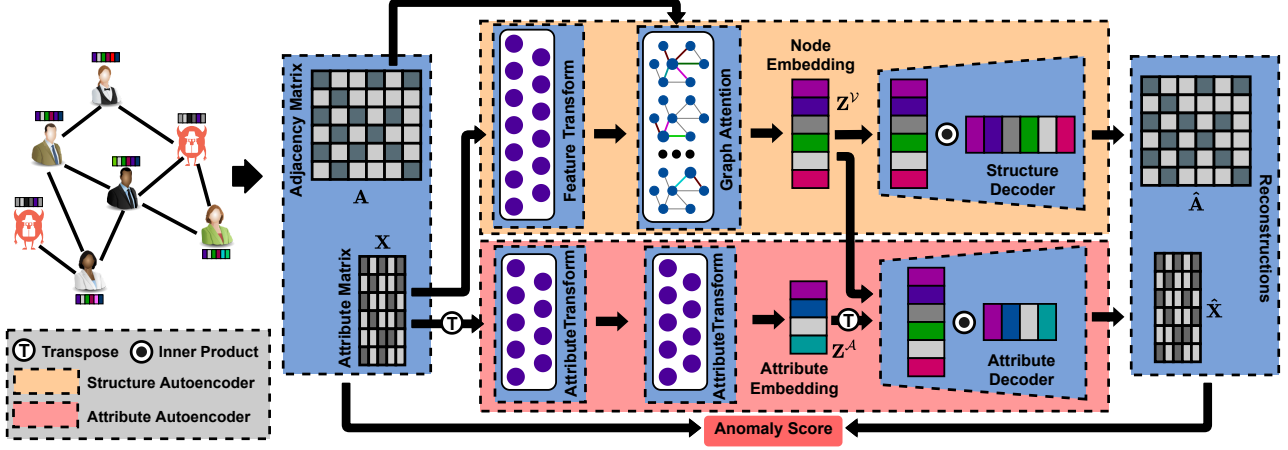


Fig. 1. The framework of the proposed AnomalyDAE.

network structure:

$$\hat{\mathbf{A}} = \text{Sigmoid}(\mathbf{Z}^{\mathbf{V}}(\mathbf{Z}^{\mathbf{V}})^{\mathbf{T}}) \quad (6)$$

where  $\text{Sigmoid}(\bullet)$  is the sigmoid activation function. Here, inner product between two node embeddings is performed to estimate the probability of a link between two nodes:

$$p(\hat{\mathbf{A}}_{i,j} = 1 | \mathbf{Z}_i^{\mathbf{V}}, \mathbf{Z}_j^{\mathbf{V}}) = \text{Sigmoid}(\mathbf{Z}_i^{\mathbf{V}}(\mathbf{Z}_j^{\mathbf{V}})^{\mathbf{T}}) \quad (7)$$

### 3.2. Attribute Autoencoder

In the attribute encoder, two non-linear feature transform layers are employed to map the observed attribute data to the latent attribute embedding  $\mathbf{Z}^{\mathbf{A}}$ , which can be formulated as follows:

$$\tilde{\mathbf{Z}}^{\mathbf{A}} = \sigma((\mathbf{X})^{\mathbf{T}}\mathbf{W}^{\mathbf{A}(1)} + \mathbf{b}^{\mathbf{A}(1)}) \quad (8)$$

$$\mathbf{Z}^{\mathbf{A}} = \tilde{\mathbf{Z}}^{\mathbf{A}}\mathbf{W}^{\mathbf{A}(2)} + \mathbf{b}^{\mathbf{A}(2)} \quad (9)$$

where  $\mathbf{W}^{\mathbf{A}(1)} \in \mathbb{R}^{D_{in}^{\mathbf{A}(1)} \times D_{out}^{\mathbf{A}(1)}}$ ,  $\mathbf{b}^{\mathbf{A}(1)} \in \mathbb{R}^{D_{out}^{\mathbf{A}(1)}}$ ,  $\mathbf{W}^{\mathbf{A}(2)} \in \mathbb{R}^{D_{out}^{\mathbf{A}(1)} \times D}$  and  $\mathbf{b}^{\mathbf{A}(2)} \in \mathbb{R}^D$  are the weights and biases learned by two layers,  $D_{in}^{\mathbf{A}(1)}$ ,  $D_{out}^{\mathbf{A}(1)}$ , and  $D$  are the dimensionalities of  $(\mathbf{X})^{\mathbf{T}}$ ,  $\tilde{\mathbf{Z}}^{\mathbf{A}}$  and  $\mathbf{Z}^{\mathbf{A}}$ , respectively.

Finally, attribute decoder takes both the node embeddings  $\mathbf{Z}^{\mathbf{V}}$  learned by structure encoder, and the attribute embeddings  $\mathbf{Z}^{\mathbf{A}}$  as inputs for decoding of the original node attribute:

$$\hat{\mathbf{X}} = \mathbf{Z}^{\mathbf{V}}(\mathbf{Z}^{\mathbf{A}})^{\mathbf{T}} \quad (10)$$

in which, the interactions between network structure and node attribute are jointly captured. Different from the structure decoder, in the attribute decoder, no activation function is utilized here for the arbitrary-valued attribute.

In AnomalyDAE, the computational complexity of structure autoencoder and attribute autoencoder are  $O(MD +$

$ED + M^2)$  and  $O(ND + NM)$  respectively, where  $M$  is the number of nodes,  $E$  is the number of edge,  $N$  is the dimension of attribute,  $D$  is the dimension of embedding.

### 3.3. Loss function

The training objective of AnomalyDAE is to minimize the reconstruction errors of both network structure and node attribute:

$$\mathcal{L}_{rec} = \alpha \|(\mathbf{A} - \hat{\mathbf{A}}) \odot \boldsymbol{\theta}\|_F^2 + (1 - \alpha) \|(\mathbf{X} - \hat{\mathbf{X}}) \odot \boldsymbol{\eta}\|_F^2 \quad (11)$$

where  $\alpha$  is the parameter which control the trade off between structure reconstruction and attribute reconstruction.  $\odot$  is the Hadamard product,  $\boldsymbol{\eta}$  and  $\boldsymbol{\theta}$  are defined as follows:

$$\theta_{i,j} = \begin{cases} 1 & \text{if } \mathbf{A}_{i,j} = 0, \\ \theta & \text{otherwise.} \end{cases}, \eta_{i,j} = \begin{cases} 1 & \text{if } \mathbf{X}_{i,j} = 0, \\ \eta & \text{otherwise.} \end{cases} \quad (12)$$

where  $\eta > 1$  and  $\theta > 1$ , which are used to impose more penalty to the reconstruction error of the non-zero elements due to some missing edges or attributes in the real world.

### 3.4. Anomaly Detection

Inspired by the motivation that the pattern of abnormal nodes deviate from the majority of other nodes in either structure or attribute, the anomaly score  $\mathcal{S}_{\mathbf{V}_i}$  of node  $\mathbf{V}_i$  is defined as the reconstruction error from both network structure and node attribute perspective:

$$\mathcal{S}_{\mathbf{V}_i} = \alpha \|(\mathbf{A}_i - \hat{\mathbf{A}}_i) \odot \boldsymbol{\theta}_i\|_F^2 + (1 - \alpha) \|(\mathbf{X}_i - \hat{\mathbf{X}}_i) \odot \boldsymbol{\eta}_i\|_F^2 \quad (13)$$

Based on the measured anomaly scores, the threshold  $\lambda$  in Eq. 1 can be determined according to distribution of scores, e.g. the nodes of top-k scores are classified as anomalous nodes.

**Table 2.** Statistics of the used Real-World datasets.

Database	# $\mathcal{V}$	# $\mathcal{E}$	# $\mathcal{A}$	# Anomalies
BlogCatalog	5,196	171,743	8,189	300
Flickr	7,575	239,738	12,047	450
ACM	16,484	71,980	8,337	600

## 4. EXPERIMENTS

### 4.1. Experimental Setup

Three commonly used real-world datasets [14] are used in this paper to evaluate the proposed method, including BlogCatalog, Flickr, and ACM. The statistics of datasets are shown in Table 2. In the experiment, we train AnomalyDAE with 100, 100, 80 iterations for BlogCatalog, Flickr, and ACM respectively. Adam [17] algorithm is utilized for optimization with learning rate as 0.001. The embedding dimension is set as 128 for all datasets. The parameters  $(\alpha, \eta, \theta)$  are empirically set as (0.7, 5, 40), (0.9, 8, 90), (0.7, 3, 10) for BlogCatalog, Flickr, and ACM respectively.

### 4.2. Result Analysis

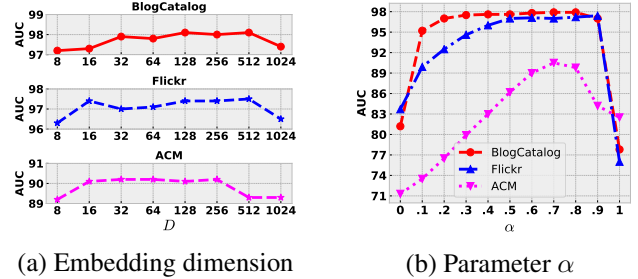
#### 4.2.1. Performance Evaluation

We compare AnomalyDAE with the state-of-the-art methods including LOF [18], SCAN [19], AMEN [8], Radar [12], Anomalous [13], and Dominant [14]. The AUC scores (the Area Under a receiver operating characteristic Curve) for anomaly detection are reported in Table 3.

The experimental results show that the proposed AnomalyDAE significantly outperforms all baselines in various datasets. The performance of AnomalyDAE is much higher than traditional anomaly detection methods, i.e., AnomalyDAE outperforms LOF by 48.66%, SCAN by 70.54%, and AMEN by 44.44% respectively on AUC for BlogCatalog dataset, because of that LOF and SCAN consider only network structure or node attribute, and AMEN is designed for anomalous neighborhoods rather than node itself. Besides, for Flickr dataset, AnomalyDAE increases the AUC score by 24.36% compared with Radar and 25.63% compared with Anomalous, this is because that residual analysis or CUR decomposition [20] based methods are not only sensitive to network sparsity, but with limited learning ability on large graph. Compared with more recent Dominant, AnomalyDAE achieves gains by 19.68%, 22.32%, 15.11% on BlogCatalog, Flickr, and ACM datasets respectively. Although GCN based encoder in Dominant is capable of learning discriminative node embeddings by aggregating neighbor features, the single graph encoder cannot jointly capture the complex interactions between network structure and node attribute, while AnomalyDAE employs two separate encoders for jointly learning of node embedding and attribute embedding respectively, which considers the complex modality interactions of

**Table 3.** AUC scores of all methods on three datasets.

Method	BlogCatalog	Flickr	ACM
LOF [18]	49.15	48.81	47.38
SCAN [19]	27.27	26.86	35.99
AMEN [8]	53.37	60.47	72.62
Radar [12]	71.04	72.86	69.36
Anomalous [13]	72.81	71.59	71.85
Dominant [14]	78.13	74.9	74.94
AnomalyDAE	<b>97.81</b>	<b>97.22</b>	<b>90.05</b>

**Fig. 2.** Parameter sensitivity for anomaly detection.

network structure and node attribute.

#### 4.2.2. Parameter Sensitivity

In this section, we investigate the parameter sensitivity of different numbers of the embedding dimension  $D$  and the trade-off parameter  $\alpha$  for anomaly detection. The experiment results are shown in Fig. 2. We can see that a relative high dimension such as 128 or 256 facilitates high performance because higher dimensional embeddings are capable of encoding more information. However, the dimension with too low value or too high value would degrade the performance because of the weak modeling ability or suffering from overfitting. In terms of  $\alpha$ , considering only attribute reconstruction ( $\alpha=0$ ) or structure reconstruction ( $\alpha=1$ ) would result in poor performance, which demonstrates the importance of the interactions between the network structure and node attribute on attributed network for anomaly detection.

## 5. CONCLUSION

In this paper, we study the problem of anomaly detection on attributed network by considering the complex modality interaction between network structure and node attribute. To cope with this problem, we propose a deep joint representation learning framework for anomaly detection via a dual autoencoder. By introducing two separate autoencoders for jointly learning of node embedding and attribute embedding, AnomalyDAE performs better than current state-of-the-art methods on multiple real-world datasets.

## 6. REFERENCES

- [1] Jundong Li, Ruocheng Guo, Chenghao Liu, and Huan Liu, “Adaptive unsupervised feature selection on attributed networks,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2019, pp. 92–100.
- [2] Lizi Liao, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua, “Attributed social network embedding,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 12, pp. 2257–2270, 2018.
- [3] Diogo Pernes and Jaime S. Cardoso, “Spamhmm: Sparse mixture of hidden markov models for graph connected entities,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2019.
- [4] Chuan Shi, Binbin Hu, Wayne Xin Zhao, and S Yu Philip, “Heterogeneous information network embedding for recommendation,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 2, pp. 357–370, 2018.
- [5] Qi Ding, Natallia Katenka, Paul Barford, Eric Kolaczyk, and Mark Crovella, “Intrusion as (anti) social communication: characterization and detection,” in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 886–894.
- [6] Wei Cheng, Kai Zhang, Haifeng Chen, Guofei Jiang, Zhengzhang Chen, and Wei Wang, “Ranking causal anomalies via temporal and dynamical analysis on vanishing correlations,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 805–814.
- [7] Shobeir Fakhraei, James Foulds, Madhusudana Shashanka, and Lise Getoor, “Collective spammer detection in evolving multi-relational social networks,” in *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining*. ACM, 2015, pp. 1769–1778.
- [8] Bryan Perozzi and Leman Akoglu, “Scalable anomaly ranking of attributed neighborhoods,” in *Proceedings of the 2016 SIAM International Conference on Data Mining*. SIAM, 2016, pp. 207–215.
- [9] Bryan Perozzi and Leman Akoglu, “Discovering communities and anomalies in attributed graphs: Interactive visual exploration and summarization,” *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 12, no. 2, pp. 24, 2018.
- [10] P. I. Snchez, E. Mller, F. Laforet, F. Keller, and K. Bhm, “Statistical selection of congruent subspaces for mining attributed graphs,” in *2013 IEEE 13th International Conference on Data Mining*, Dec 2013, pp. 647–656.
- [11] Bryan Perozzi, Leman Akoglu, Patricia Iglesias Sánchez, and Emmanuel Müller, “Focused clustering and outlier detection in large attributed graphs,” in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 1346–1355.
- [12] Jundong Li, Harsh Dani, Xia Hu, and Huan Liu, “Radar: Residual analysis for anomaly detection in attributed networks,” in *IJCAI*, 2017, pp. 2152–2158.
- [13] Zhen Peng, Minnan Luo, Jundong Li, Huan Liu, and Qinghua Zheng, “Anomalous: A joint modeling approach for anomaly detection on attributed networks,” in *IJCAI*, 2018, pp. 3513–3519.
- [14] 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*. SIAM, 2019, pp. 594–602.
- [15] Thomas N Kipf and Max Welling, “Semi-supervised classification with graph convolutional networks,” *International Conference on Learning Representations (ICLR)*, 2017.
- [16] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio, “Graph attention networks,” 2018, accepted as poster.
- [17] Diederick P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” in *International Conference on Learning Representations (ICLR)*, 2015.
- [18] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander, “Lof: identifying density-based local outliers,” in *ACM sigmod record*. ACM, 2000, vol. 29, pp. 93–104.
- [19] Xiaowei Xu, Nurcan Yuruk, Zhidan Feng, and Thomas AJ Schweiger, “Scan: a structural clustering algorithm for networks,” in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2007, pp. 824–833.
- [20] Michael W Mahoney and Petros Drineas, “Cur matrix decompositions for improved data analysis,” *Proceedings of the National Academy of Sciences*, vol. 106, no. 3, pp. 697–702, 2009.