DeepAD: a Joint Embedding Approach for Anomaly Detection on Attributed Networks

Dali Zhu[1,2], Yuchen Ma[1,2], and Yinlong Liu[1,2]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{zhudali,mayuchen,liuyinlong}@iie.ac.cn

Abstract. Detecting anomalies in the attributed network is a vital task that is widely used, ranging from social media, finance to cybersecurity. Recently, network embedding has proven an important approach to learn low-dimensional representations of vertexes in networks. Most of the existing approaches only focus on topological information without embedding rich nodal information due to the lack of an effective mechanism to capture the interaction between two different information modalities. To solve this problem, in this paper, we propose a novel deep attributed network embedding framework named DeepAD to differentiate anomalies whose behaviors obviously deviate from the majority. DeepAD (i) simultaneously capture both of the highly non-linear topological structure and node attributes information based on the graph convolutional network (GCN) and (ii) preserve various interaction proximities between two different information modalities to make them complement each other towards a unified representation for anomaly detection. Extensive experiments on real-world attributed networks demonstrate the effectiveness of our proposed anomaly detection approach.

Keywords: Anomaly Detection · Attributed Networks · Autoencoder · Graph Convolutional Network.

1 Introduction

Networks have become an important tool in many real-world applications to represent complex information systems such as social networks, transportation networks, and communication networks. In these networks, attributed networks have become a hot topic of research. Different from traditional plain networks where only the topological structure information is utilized, attributed networks also associated with rich features or attributes, which enrich the knowledge inside representations for network analysis. For example, in social networks, in addition to friend relationships, rich profile information is also an important attribute for describing user characteristics; in online shopping networks, purchase records associated with reviews provide valuable information. Studies from social science have revealed the influence of interaction between the attributes of nodes and their structures [21, 24]. Going through these insights, we can discover deeper patterns from attributed networks.

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

Anomaly detection plays a vital role in many information systems to achieve secure cyberspace. It aims to identify rare instances that do not conform to the expected patterns of majority [1]. Recently, there is emerging research of anomaly detection focusing on attributed networks due to the potential rich information contained in the attributed network. However, how to model network structure information and rich semantic nodal information into a unified representation is still a challenging problem.

Conventional anomaly detection methods mainly focus on exploiting the structure of the network to find patterns and spot anomalies such as structural-based or community-based [3] methods. Besides, feature-based methods assume that complex anomalies only exist in a subset according to node features. Unfortunately, existing efforts usually focus on either topological information or attribute information without insight into the complex interaction between those two different types of modalities. Moreover, other methods employing shallow models can hardly capture the highly non-linear [27] property of the attributed network. To address the problems as mentioned above, inspired by graph convolutional network (GCN) [16], we resort to embedding the input topological structure as well as nodal attributes seamlessly into a unified representation through stacking GCN layers. Meanwhile, the proposed model enforces the learned node representation to preserve various proximities. Then we aim to spot anomalies leveraging by the reconstruction errors both from the two kinds of modalities. The contributions of this paper are listed as follow:

- We propose a novel joint embedding approach DeepAD modeled by graph autoencoder DeepAD to capture the underlying high non-linearity in both topological structure and nodal attributes and detect anomalies according to the reconstruction errors.
- DeepAD jointly preserves the first-order, high-order, and cross-modal proximities in original networks towards a unified complementary representation.
- Experimental results show that DeepAD outperforms several state-of-art methods on benchmark datasets.

2 Related Work

2.1 Graph Based Anomaly Detection

Typically, graph-based anomaly detection methods are broadly divided into three classes: (1)*Structure-based methods* (2) *Community-based methods* and (3)*Feature-based methods* [3]. Structure-based methods mainly aim to identify substructures or subgraphs in the graph that are rare structurally, therefore anomalies can be sought out as the inverse of frequent subgraphs [22]. CODA [9] is one of the attempts that simultaneously finds communities as well as spots community anomalies using Markov random fields. OddBall [2] extracts features and finds patterns based on the ego-network of the graph to spot anomalous nodes. Community-based methods aim to find dense group nodes in the graph and spot anomalies that have connections across communities. One of the types

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

of them, LOF [5], computes the local density deviation of a given data point concerning its neighbors. The main idea behind feature-based methods is that similar graphs should share the same properties, such as degree distribution, diameter, eigenvalues [14]. Recently, residual analysis [18] shows its effectiveness for anomaly detection in a more general way. However, those shallow models failed to model the underlying high non-linearity information of attributed networks into a unified complementary representation.

2.2 Deep Network Embedding

Network embedding aims to learn low-dimensional vector representations for nodes of the network, which preserves structure information and properties of graphs. With the increasing research on deep learning, a vast amount of deep models have been proposed towards various learning tasks. For plain networks, DNGR [6] utilizes deep autoencoder to capture network's non-linearity, and SDNE [27] further preserves the first-order and second-order proximity. LANE [12] jointly combine the label, attribute, and structure information into embedding. Besides, DANE [8] adopts two deep autoencoders to train structure and attributed features separately while keeping the representation consistency and complementary. Recently, Kipf and Welling [16] propose graph convolutional network (GCN) model for attributed networks that simultaneously encode the structure and attribute information into latent space and further employ it on a variational auto-encoder architecture [17]. Our model took inspiration from these methods.

3 Problem Definition

We define the anomaly detection problem in attributed networks with first-order proximity, high-order proximity, and cross-modal proximity.

Definition 1. *(Attributed Network Embedding) An attributed network is denoted as* $G = (A, X)$ *with* n *nodes, where* $A = [a_{i,j}] \in \mathbb{R}^{n \times n}$ *is the adjacency matrix and* $X = [x_{i,j}] \in \mathbb{R}^{n \times m}$ *is the attribute matrix. Each node is associated with a nodal attributes row vector* $\mathbf{x}_i \in \mathbb{R}^m (i = 1, \ldots, n)$. $a_{i,j} = 1$ *represents there is a link between the* i^{th} *node and the* j^{th} *node. Otherwise,* $a_{i,j} = 0$. *The objective of network embedding is to map the topological structure and nodal attributes into a representation space* $H \in \mathbb{R}^{n \times d}$ *through a mapping function* $f : \{A, X\} \rightarrow H$. *Note that,* $d \ll |A|$ *is the dimension of representation space.*

Network embedding aims to preserve the intrinsic information of the network into a low-dimensional representation space. To perform the embedding appropriately for anomaly detection task, we define three proximities to preserve local proximity, global proximity and interaction proximity respectively.

Definition 2. *(First-Order Proximity) The first-order proximity describes the pairwise similarity between two nodes. For each pair of nodes,* $a_{i,j} > 0$ *indicates*

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

there exists first-order proximity between them. Otherwise, if no interaction is observed, the first-order proximity is 0.

Generally speaking, the first-order proximity is the most direct expression in a network. For example, people who are friends with each other in social media tend to share a common characteristic. Because of this importance, it is necessary to preserve the first-order proximity, which can be viewed as local proximity. However, due to the sparsity and incompleteness of the real-world network, it is not sufficient only considering the first-order proximity to represent the network. Therefore, we introduce high-order proximity to characterize the global proximity of the network to compensate for this problem.

Definition 3. *(High-Order Proximity) The high-order proximity describes the neighborhood similarity between two nodes. Given an attributed network* $G = (A, X)$*, let* $M = \left(G^1 + G^2 + \ldots, +G^k\right)$ *denotes the high-order proximity, where* G^k *is the* k^{th}*-order node proximity information propagation through the embeddings. Then the high-order proximity between* v_i^k *and* v_j^k *is determined by* M_i^t *and* M_j^t*.*

Intuitively, two nodes are similar if they share similar neighbors. For example, in a citation network, documents are similar if they are surrounded by similar citations, even if they are not referencing to each other [19]. Since the topological structure and nodal attributes are two different information modalities in the same network, to make them complement each other towards a unified informative representation of the same network [11], the cross-modal proximity is essential to be preserved.

Definition 4. *(Cross-Modal Proximity) The cross-modal proximity describes the similarity between nodes according to their structure and attribute information. Given an attributed network* $G = (A, X)$*, the cross-modal proximity of two nodes* v_i *and* v_j *is determined by* (A_i, X_i) *and* (A_j, X_j)*.*

Definition 5. *(Anomaly Detection) The task of anomaly detection is to find the node instances that are rare and significantly different from the majority of the reference nodes according to their anomalous scores.*

4 The DeepAD Model

According to the previous analysis, three challenges remain for anomaly detection on attributed networks to achieve desired results:

(1)*Network sparsity*: Many real-world networks tend to be so sparse that the utilization of limited observed node interactions severely restricts the performance of network analysis.

(2)*High non-linearity*: The underlying structure of topological structure and nodal attributes is often highly non-linear and hence cannot be accurately captured by linear models [20].

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

(3)*Proximity preservation*: The combination of the two information modalities describes different aspects of the network information. How to preserve complex interaction proximity and complement each other to form a unified information representation is still a thorny problem.

To address the challenges above, we present a novel deep joint model approach *DeepAD* for anomaly detection, as shown in Figure 1. The network structure and the nodal attributes embedded through a joint framework modeled by GCN into the same representation space. In order to preserve the complex interaction between two information sources, we add constraints to refine the representation. And then, we make use of the reconstruction errors as a measure to spot anomalies. Details are introduced as follows.

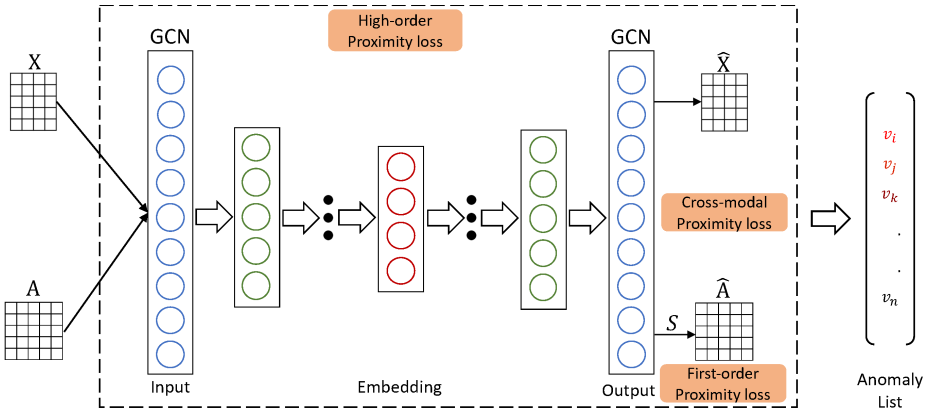


Fig. 1. The model takes the adjacency matrix A and the attribute matrix X as inputs, representing the topological structure and the nodal attributes respectively.

4.1 Framework

DeepAD embeds the input data through an autoencoder to capture the highly non-linear information simultaneously from network structure and nodal attributes. Autoencoder has proven a powerful deep learning model to learn the complex latent representation of data for various applications [13]. The primary target of autoencoder can be reduced to solving the following optimization problem:

$$\min_{\theta} \sum_{\phi \in \Phi_{tar}} \mathcal{L}\left(\psi_{dec}\left(\psi_{enc}\left(X_{\phi}\right)\right), \phi|\theta\right) \tag{1}$$

where Φ_{tar} is the target information that the embedding layers expect to preserve, and $X \in \mathbb{R}^n$ denotes the input data involved in ϕ. The encoder ψ_{enc} maps data into representation vectors, and decoder ψ_{dec} reconstructs the original data from the representation space. θ denotes the model parameters in encoders and

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

decoders. The parameters are trained by minimizing the loss function described above, thereby preserving the desired network information Φ_{tar} in the network.

To capture the complex interaction of the topological structure and nodal attributes, inspired by the significant performance improvement of graph convolutional network (GCN) [16] in the analysis of non-Euclidean structured data such as graphs and manifolds, we use GCN layers as encoder which is defined as:

$$H^{(k)} = \sigma\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(k-1)}W^{(k-1)}\right) \tag{2}$$

where H^{k-1} is the input for the embedding layer $k-1$, and $H^0 = X$. $\widetilde{A} = (I + A)$ is the adjacency matrix with added self-connections. I is the identity matrix, and $\widetilde{D}$ is the diagonal matrix of $\widetilde{A}$. $\sigma(\cdot)$ denotes a non-linear activation function such as ReLU or sigmoid. W^{k-1} is a matrix of filter parameters which are shared for all input nodes. It is worth noting that unlike autoencoders that explicitly treat each node's neighbor as features to embed into a latent space separately with the nodal attributes as in SDNE and DANE, GCN implicitly applies the local neighborhood links on each encoding layer as pathways to aggregate embeddings from neighbors [10]. Given the input attribute matrix X, the convolutional layers iteratively aggregate embeddings of neighbors as well as its own to capture a higher-order node proximity information of which both the topological structure and nodal attributes are preserved. Let $A_N = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ denotes the normalized adjacency, the encoder can be formed as:

$$\begin{aligned} H^{(1)} &= \sigma\left(A_N H^{(0)} W^{(0)}\right) \\ Z = H^{(k)} &= \sigma\left(A_N H^{(k-1)} W^{(k-1)}\right), k = 2, ..., K \end{aligned} \tag{3}$$

Therefore, the embeddings Z is the desired low-dimensional representation of the attributed network. Correspondingly, there will be k layers in the decoder and the output $\widehat{X}$ is the reconstruction of attribute matrix. Furthermore, according to [17], the reconstructed adjacency matrix $\widehat{A}$ can be calculated as $\widehat{A} = \mathcal{S}\left(HH^{\top}\right)$ where $\mathcal{S}(x)$ is the sigmoid function. To maximize the information propagation, we choose the last H to reconstruct the adjacency matrix.

4.2 Loss Function

As aforementioned analysis, nodes with similar features are more likely to be connected in attributed networks. The **first-order proximity** can be regarded as the supervised information to restrict the similarity of a pair of nodes in the latent representations. Inspired by the idea of Laplacian eigenmaps (LE) [4], we introduce a penalty term to constrain the local proximity when similar nodes are mapped away from each other in the latent representations. The loss function is shown as follows:

$$\mathcal{L}_f = \sum_{i,j=1}^{n} \hat{a}_{i,j} \left\| \mathbf{h}_i^{(K)} - \mathbf{h}_j^{(K)} \right\|_2^2 = \sum_{i,j=1}^{n} \hat{a}_{i,j} \left\| \mathbf{h}_i - \mathbf{h}_j \right\|_2^2 \tag{4}$$

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

where $\mathbf{h}_i^{(K)}$ and $\mathbf{h}_j^{(K)}$ are the row vector of the hidden layer matrix $H^{(K)}$ and $\hat{a}_{i,j} \in \widehat{A}$ indicates whether there exists a connection between nodes v_i and v_j. The loss function can be reformulated as the following term:

$$\mathcal{L}_f = \sum_{i,j=1}^{n} \hat{a}_{i,j} \left\| \mathbf{h}_i - \mathbf{h}_j \right\|_2^2 = 2tr\left(H^T L H\right) \tag{5}$$

where $L = D - \widehat{A}$, $D \in \mathcal{R}^{n*n}$ is a diagonal matrix of $\widehat{A}$, and $D_{i,i} = \sum_j \hat{a}_{i,j}$.

The **high-order proximity** refers to how similar the neighborhood information of a pair of nodes is. With the iteration of convolutions, the higher-order neighborhood information is embedded into the latent space. As SDNE proved, the constraints on reconstruction can enforce the neural network to capture the data manifold smoothly, thereby preserve the proximity among a wider range of samples. To preserve this proximity, we minimize reconstruction loss as follows:

$$L_h = R_x + \alpha R_a = \sum_{i=1}^{n} \left\| \hat{\mathbf{x}}_i - \mathbf{x}_i \right\|_2^2 + \alpha \sum_{i=1}^{n} \left\| \hat{\mathbf{a}}_i - \mathbf{a}_i \right\|_2^2 \tag{6}$$

where R_x and R_a represent the reconstruction error of the attribute matrix and adjacent matrix respectively. Specifically, if the neighborhood information is similar between two nodes, after minimizing the L_h, the learned representation H_i and H_j will also be similar. According to [7], anomalies are more difficult to reconstruct than normal nodes since their information representation does not conform to the patterns of the majority. Therefore, a larger reconstruction error indicates a higher probability of anomalies.

It's not only necessary to preserve the network proximity separately, but also essential to discover the implicit relationship since the topological structure and nodal attributes are two interdependent information modalities to describe the network. To make those two modalities complement each other, we preserve the **cross-modal proximity** by maximizing their interaction likelihood estimation as follows:

$$L_c = \prod_{i,j}^{n} p_{i,j}^{I_{i,j}} \left(1 - p_{i,j}\right)^{1 - I_{i,j}} \tag{7}$$

where $p_{i,j}$ is the joint distribution of two modalities which can be defined as $p_{ij} = \mathcal{S}\left(\mathbf{a}_i \mathbf{h}_j\right)$. Furthermore, $I_{i,j}$ indicates whether $\mathbf{a}_i$ and $\mathbf{h}_j$ are from the same node. $I_{ij} = 1$ if $i = j$. Otherwise $I_{ij} = 0$. So, the loss function can be defined in the form of the negative log-likelihood as follows:

$$L_c = -\sum_{i} \{\log p_{ii} - \sum_{j \neq i} \log\left(1 - p_{i,j}\right)\} \tag{8}$$

The objective function of Eq.(8) constrains $\mathbf{a}_i$ and $\mathbf{h}_j$ as consistent as possible when they belong to the same node while separating them from each other when they come from different nodes, resulting in sufficient complementary interactions of two modalities. To simplify the calculation, pairwise nodes with similar

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

first-order proximity need not be separated from each other, because representation $\mathbf{a}_i$ and $\mathbf{h}_j$ should also be similar. Therefore, the objective function can be revised as follows:

$$L_c = -\sum_i \{\log p_{ii} - \sum_{\hat{a}_{i,j}=0} \log(1 - p_{i,j})\} \tag{9}$$

As shown in Fig. 1, in order to simultaneously preserve the three proximities, we propose a semi-supervised framework which jointly combines Eq.(5), Eq.(6), and Eq.(8). The overall objective function is shown as follows:

$$\begin{aligned} L =& L_f + L_{hx} + \alpha L_{ha} + L_c \\ =& \sum_{i,j=1}^{n} \hat{a}_{i,j} \left\| \mathbf{h}_i - \mathbf{h}_j \right\|_2^2 + \sum_{i=1}^{n} \left\| \hat{\mathbf{x}}_i - \mathbf{x}_i \right\|_2^2 + \alpha \sum_{i=1}^{n} \left\| \hat{\mathbf{a}}_i - \mathbf{a}_i \right\|_2^2 \\ & - \sum_i \{\log p_{ii} - \sum_{\hat{a}_{i,j}=0} \log(1 - p_{i,j})\} \end{aligned} \tag{10}$$

4.3 Anomaly Detection

By minimizing the loss function, DeepAD can iteratively learn the representations of input attributed network until the objective function converges. With a Xavier Initialization, the model parameters can be optimized by using stochastic gradient descent. After a certain number of iterations, as in [7] and [23], the reconstruction error can be directly applied to rank the abnormality of nodes. Thus the anomaly score of each node v_i can be calculated as follows:

$$socre(\mathbf{v}_i) = \left\| \hat{\mathbf{x}}_i - \mathbf{x}_i \right\|_2^2 + \alpha \left\| \hat{\mathbf{a}}_i - \mathbf{a}_i \right\|_2^2 \tag{11}$$

As a result, we can calculate the ranking of anomalies according to the corresponding abnormal scores. The higher the score, the more likely the instance is to be considered an anomaly.

4.4 Complexity Analysis

The complexity of graph convolutional network is dominated by the computation of $\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}XW$ whose complexity is $\mathcal{O}(ncdh)$ [16], where n is the number of nodes, c is the average degree of network which is usually a constant in real-world applications, d is the number of feature dimensions on the attributed network and h is the number of feature maps of W. In this way, nc represents the number of non-zero elements in A so that $\tilde{A}X$ can be efficiently calculated using sparse-dense matrix multiplications. The complexity of Eq.(5) is $\mathcal{O}(ncd)$ [27] while the complexity of Eq.(9) is $\mathcal{O}\left(n^2\right)$, thus the overall complexity of the model is $\mathcal{O}\left(ncd\left(H + n^2\right)\right)$ where H is the sum of h in all layers.

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

5 Experiments

5.1 Datasets

We choose three benchmark datasets[1]: Cora, Citeseer, and PubMed. These three datasets are paper citation networks. The nodes and edges of each network denote documents and reference links, respectively. The attribute of each node is the bag-of-words feature vectors of each document. In order to obtain a ground truth of anomalies in the above datasets, we refer to two widely used methods [25, 26] to generate a combined set of anomalies from both the topological structure and nodal attributes perspectives for each dataset. In real-world scenarios, the small clique is a typical substructure created by anomalous activity [25]. Therefore we randomly select m nodes and connect them to each other to form a dense clique, iteratively repeat this process until n cliques are generated, and all the mn nodes in cliques are considered as anomalies. Then, we inject the same number of anomalies from the attribute perspective. Similarly, we randomly select mn nodes from the network, shuffle their attribute values to generate anomalous nodes, while the topological relationship remains unchanged. The details of dataset statistics are summarized in Table 1.

Table 1. Description of Benchmark Datasets

Dataset	Cora	Citeseer	PubMed
# Nodes	2780	3327	19717
# Edges	5278	4732	44338
# Attributes	1433	3703	500
# Anomalies	10%	10%	10%

5.2 Baseline Algorithms

We choose four contrast algorithms as baselines. The details are as follows:

- LOF [5] detects anomalies which have a substantially lower density and only considers attribute information.
- SCAN [28] detects anomalies at the structural level and only considers structure information.
- CODA [9] detects anomalies based on community detection within a unified probabilistic model.
- Radar [18] detects anomalies whose behavior obviously deviates from the majority according to the residuals of attribute information and its coherence with network information.

[1] https://github.com/kimiyoung/planetoid/tree/master/data

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

5.3 Evaluation Metrics

We select *AUC*, *precision*@K and *recall*@K to evaluate the performance. Their definition is listed as follows:

- AUC: AUC (Area Under ROC Curve) is a performance measurement for classification problems. Higher the AUC, better the model is at distinguishing between normal and anomalous
- Precision@K: We evaluate the proportion of true anomalies that are discovered in the top K ranked nodes.
- Recall@K: It measures the percentage of true anomalies selected out of all the ground truth anomalies.

5.4 Parameter Settings

The architecture of our approach varies with different datasets. The dimension of each layer is summarized in Table 2. All the neural networks have three layers, and the dimension of the last encoder layer is the same.

We use ReLU as the activation function and optimize the loss function with Adam algorithm [15]. The learning rate is set to 0.025. The hyper-parameter α is tuned with grid search on each dataset. For the rest baselines, their settings are set as described in the original papers.

Table 2. Neural Network Structures

Dataset	# nodes in each layer
Cora	1433-200-100
Citeseer	3703-500-100
PubMed	500-200-100

5.5 Experiment Results

The experimental results in terms of AUC values are presented in Figure 2, and the precision and recall results are reported in Table 3. The results of SCAN and CODA are not included in Table 3 since they are cluster-based methods that are incapable of providing an accurate ranking list for all nodes. From the evaluation results, we can see that the dual-modality information-based model (Radar, DeepAD) is superior to the conventional methods (LOF, SCAN, and CODA) merely based either on attribute information or structure information. However, through the comparison of the residual-based model Radar and DeepAD, we can observe that there is a significant increase in each metric. Figure 3 shows the result of five anomaly detection models on the Citeseer dataset. When the ratio of anomalies increased, our proposed DeepAD can still maintain high AUC values. The main reasons may be as follows: (1) We employ a deep neural network model based on graph autoencoder, which breaks through the limitation

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

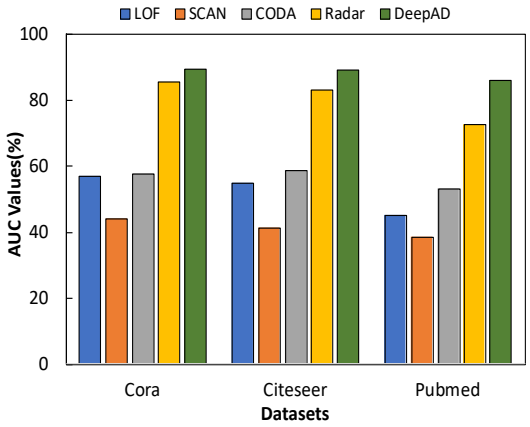


Fig. 2. Anomaly detection results by different methods.

Table 3. *precision*@K and *recall*@K on three datasets for anomaly detection.

Precision@K									
	Cora			Citeseer			PubMed		
K	50	100	200	50	100	200	50	100	200
LOF	0.480	0.375	0.314	0.525	0.462	0.410	0.080	0.075	0.053
Radar	0.786	0.770	0.756	0.780	0.765	0.726	0.575	0.583	0.560
DeepAD	**0.820**	**0.796**	**0.743**	**0.812**	**0.785**	**0.752**	**0.652**	**0.610**	**0.580**
Recall@K									
	Cora			Citeseer			PubMed		
K	50	100	200	50	100	200	50	100	200
LOF	0.060	0.095	0.120	0.065	0.087	0.115	0.008	0.012	0.016
Radar	0.090	0.204	0.250	0.086	0.180	0.294	0.052	0.095	0.186
DeepAD	**0.116**	**0.235**	**0.384**	**0.095**	**0.205**	**0.265**	**0.061**	**0.105**	**0.236**

of shallow mechanisms to handle the network sparsity issue and capture the high non-linearity information both from the topological structure and nodal attributes in attributed networks. (2) To further capture the complex interaction between two different modalities, we propose various proximities to preserve the implicit proximity make them complement each other towards a unified representation. Among the results, DeepAD outperforms other baselines on all benchmark datasets, which demonstrate the effectiveness of our proposed anomaly detection approach on attributed networks. Our future work will focus on how to develop a deep anomaly detection model robust to noise.

5.6 Parameter Sensitivity

There are several parameters in our proposed DeepAD framework, we investigate the impact of the number of the embedding dimension and the value of

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

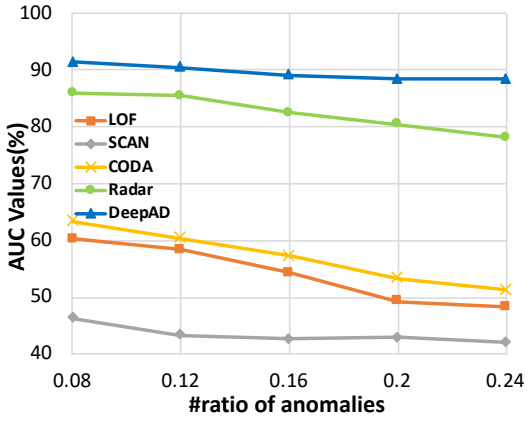
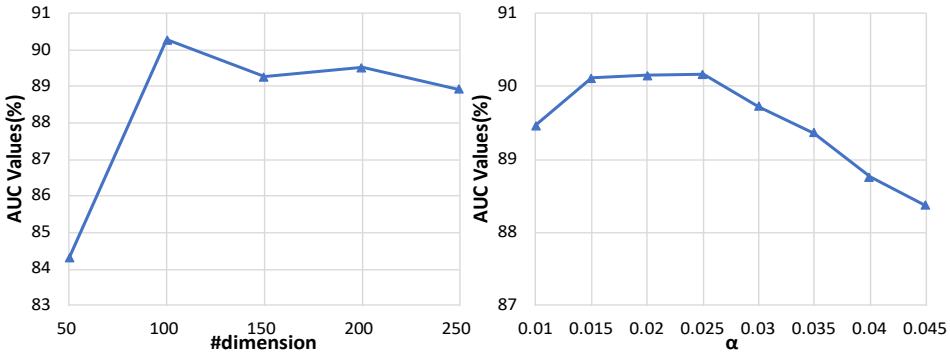


Fig. 3. AUC of anomaly detection on Citeseer dataset

hyper-parameter α on Citeseer dataset with 400 injected anomalies and report the performance variance results in Figure 4. Figure 4(a) reports the performance of DeepAD w.r.t the dimension of the embedding layer. It can be shown that performance improves as the dimension increases. However, when the dimension continues to increase beyond 100, the performance no longer improves or even drops. The possible reason is that too large dimension of embedding will introduce noise so as to influence the latent representations. The hyper-parameter α balances the impact of three proximities on model training and anomaly scores computation. The results in Figure 4(b) indicate that it is necessary to find a balance between those proximities to achieve better performance, and the best choice of α is 0.025.



(a) #dimension (b) α

Fig. 4. Sensitivity w.r.t dimension and the value of α

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

6 Conclusion

In this paper, we propose a joint embedding approach for anomaly detection on attributed networks, namely DeepAD. Specifically, to capture the highly non-linear information in network topological structure and nodal attributes. We design a graph convolutional network (GCN) based deep autoencoder model. To further address the complex interaction problem, we jointly preserve the first-order, high-order, and cross-modal proximity to make two types of information complement each other towards a unified representation. By jointly optimizing them in the deep model, the reconstruction errors are them employed to spot anomalies. The experimental results demonstrate the effectiveness of our approach to anomaly detection compared with state-of-art methods.

Acknowledgement

This work was supported by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDC02040300.

References

1. Aggarwal, C.C.: Outlier analysis. In: Data mining. pp. 237–263. Springer (2015)
2. Akoglu, L., McGlohon, M., Faloutsos, C.: Oddball: Spotting anomalies in weighted graphs. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 410–421. Springer (2010)
3. Akoglu, L., Tong, H., Koutra, D.: Graph based anomaly detection and description: a survey. Data mining and knowledge discovery **29**(3), 626–688 (2015)
4. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation **15**(6), 1373–1396 (2003)
5. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD international conference on Management of data. pp. 93–104. ACM (2000)
6. Cao, S., Lu, W., Xu, Q.: Deep neural networks for learning graph representations. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
7. Chen, J., Sathe, S., Aggarwal, C., Turaga, D.: Outlier detection with autoencoder ensembles. In: Proceedings of the 2017 SIAM international conference on data mining. pp. 90–98. SIAM (2017)
8. Gao, H., Huang, H.: Deep attributed network embedding. In: IJCAI. vol. 18, pp. 3364–3370 (2018)
9. Gao, J., Liang, F., Fan, W., Wang, C., Sun, Y., Han, J.: 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. pp. 813–822. ACM (2010)
10. Hu, X., Tan, Q., Liu, N.: Deep representation learning for social network analysis. Frontiers in Big Data **2**, 2 (2019)
11. Huang, X., Li, J., Hu, X.: Accelerated attributed network embedding. In: Proceedings of the 2017 SIAM international conference on data mining. pp. 633–641. SIAM (2017)

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22

12. Huang, X., Li, J., Hu, X.: Label informed attributed network embedding. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. pp. 731–739. ACM (2017)
13. Jiang, W., Gao, H., Chung, F.l., Huang, H.: The l2, 1-norm stacked robust autoencoders for domain adaptation. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
14. Kang, U., Papadimitriou, S., Sun, J., Tong, H.: Centralities in large networks: Algorithms and observations. In: Proceedings of the 2011 SIAM international conference on data mining. pp. 119–130. SIAM (2011)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
16. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
17. Kipf, T.N., Welling, M.: Variational graph auto-encoders. arXiv preprint arXiv:1611.07308 (2016)
18. Li, J., Dani, H., Hu, X., Liu, H.: Radar: Residual analysis for anomaly detection in attributed networks. In: IJCAI. pp. 2152–2158 (2017)
19. Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. Journal of the American society for information science and technology **58**(7), 1019–1031 (2007)
20. Luo, D., Nie, F., Huang, H., Ding, C.H.: Cauchy graph embedding. In: Proceedings of the 28th International Conference on Machine Learning (ICML-11). pp. 553–560 (2011)
21. McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: Homophily in social networks. Annual review of sociology **27**(1), 415–444 (2001)
22. Noble, C.C., Cook, D.J.: Graph-based anomaly detection. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 631–636. ACM (2003)
23. Sakurada, M., Yairi, T.: Anomaly detection using autoencoders with nonlinear dimensionality reduction. In: Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis. pp. 4–11 (2014)
24. Shalizi, C.R., Thomas, A.C.: Homophily and contagion are generically confounded in observational social network studies. Sociological methods & research **40**(2), 211–239 (2011)
25. Skillicorn, D.B.: Detecting anomalies in graphs. In: 2007 IEEE Intelligence and Security Informatics. pp. 209–216. IEEE (2007)
26. Song, X., Wu, M., Jermaine, C., Ranka, S., et al.: Conditional anomaly detection. IEEE Trans. Knowl. Data Eng. **19**(5), 631–645 (2007)
27. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1225–1234. ACM (2016)
28. Xu, X., Yuruk, N., Feng, Z., Schweiger, T.A.: Scan: a structural clustering algorithm for networks. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 824–833. ACM (2007)

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50417-5_22