Deep attributed network embedding by mutual information maximization

Wujun Tao¹*, Yu Ye² and Bailin Feng³

¹Computer science, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, China
²Computer science, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, China
³Computer science, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, China

*Corresponding author’s e-mail: wujuntao@std.uestc.edu.cn

Abstract. There is a growing body of literature that recognizes the importance of network embedding. It intends to encode the graph structure information into a low-dimensional vector for each node in the graph, which benefits the downstream tasks. Most of recent works focus on supervised learning. But they are usually not feasible in real-world datasets owing to the high cost to obtain labels. To address this issue, we design a new unsupervised attributed network embedding method, deep attributed network embedding by mutual information maximization (DMIM). Our method focuses on maximizing mutual information between the hidden representations of the global topological structure and the node attributes, which allows us to obtain the node embedding without manual labeling. To illustrate the effectiveness of our method, we carry out the node classification task using the learned node embeddings. Compared with the state-of-the-art unsupervised methods, our method achieves superior results on various datasets.

1. Introduction

Network embedding is important for a wide range of scientific and industrial processes. While maintaining the metric proximity among nodes, for each node, it aims to learn a low-dimensional vector. The learned representations can be applied as latent features for following various tasks, including node classification [1], graph classification and link prediction [2].

Early works for unsupervised network embedding rely on walk-based objective [2, 3, 4, 5], sometimes further simplified to reconstruct adjacency information [1, 6] or node attributes [7]. They intend to make node representations are close if nodes are close in the input graph. However, these works only utilize the topological structure while discarding node’s rich attributes. The node attributes contain rich information, which can indicate the different relationships between nodes. Currently, the dominant algorithms founded on graph neural networks(GNN)[1] utilize more expressive graph convolutional encoders compared with conventional methods. One of the representative methods is GraphSAGE [5], which combines graph convolutional encoder and DeepWalk-like objectives. Recent work DGI [8] first applies contrastive learning on the graph, which magnifies mutual information between local node representations and global graph representations. However, GNN based methods rely on aggregated-based scheme, which can’t employ deep network layers and apply to large-scale network. Due to its over-smoothing problem, too deep network architecture cannot be used.
To overcome above challenges, we design a new unsupervised learning method, deep attributed network embedding by mutual information maximization (DMIM). The mutual information maximization between two types of hidden representations of graph topological structure and of node’s rich attributes can encourage the model to better fuse these two types information. The following is a summary of the major contributions:

- We propose an unsupervised network embedding method to capture the network topological structure and node attributes of arbitrary size network;
- The proposed method can employ more deep network layers that GNN based methods.
- We reveal that the node embeddings derived from the proposed model can obtain superior node classification results compared with other unsupervised methods on various datasets;

2. Materials and Methods
Motivated by the core idea of contrastive learning, we design a new deep unsupervised attributed network embedding method, DMIM, to pick up topological structure information and node’s rich attributes, and dig up the node representations. Our method is on the basis of the maximization of mutual information between the hidden representations of node attributes and of the topological structure from the same node. The proposed method first utilizes random walks to extract the global topological structure, then obtains two types of hidden representation from the same node. At last fusion these two types of hidden representations on the basis of mutual information maximization.

2.1. Problem definition
Given $G = \{V, A, X\}$ indicated an attributed network, and in the network the number of nodes is $N = |V|$. $A \in \mathbb{R}^{N \times N}$ is the adjacency matrix with size $N \times N$, $A_{ij} = 1$ if node $v_i$ is connected with node $v_j$, otherwise $A_{ij} = 0$. $X \in \mathbb{R}^{N \times m}$ is the attributed matrix. In detail, $X_i$ is the attributes of the $i$-th node, which is the $i$-th row of $X$. Attributed network embedding intends to learn the low-dimensional embedding, as denoted by $Z^{N \times d}$, such that $Z$ can preserve the network structure and node’s rich features.

2.2. DMIM

2.2.1. Global topological structure
In that the adjacency matrix can only represent the local topological structure, we extract the global topological structure $S \in \mathbb{R}^{N \times N}$ founded on random walks. In detail, we first use random walks to collect enough node sequences. Then we extract the global context of the node from these node sequences. Specifically, if two nodes are similar, they are more likely to appear in the context of each other. Due to the above operations, we get the global topological structure of each node.

2.2.2. Autoencoder
For capturing the highly non-linear structure, we use autoencoder to restore the global structure and node attributes respectively.

$$h_i = \sigma(W^{(1)}x_i + b^{(1)}), \hat{x}_i = \sigma(W^{(2)}h_i + b^{(2)})$$  \hspace{1cm} (1)

Here $x_i \in \mathbb{R}^m$ is the attribute of the $i$-th node, $h_i \in \mathbb{R}^d$ is the hidden representation of the attribute of the $i$-th node from the encoder, and $\hat{x}_i \in \mathbb{R}^m$ is the reconstructed attribute of the $i$-th node from the decoder. $\sigma(.)$ denotes the non-linear activation function. We get the loss function $L_1$ for attribute reconstruction using matrix form, as follow:

$$L_1 = \|X - \hat{X}\|$$  \hspace{1cm} (2)

Similarly, for the global topological structure, we use another autoencoder to reconstruct it to obtain the hidden representation from the global topological.

$$f_i = \sigma(W^{(3)}s_i + b^{(3)}), \hat{s} = \sigma(W^4f_i + b^4)$$  \hspace{1cm} (3)

$$L_2 = \|S - \hat{S}\|$$  \hspace{1cm} (4)
Here $f_i \in \mathbb{R}^{d''}$ is the hidden representation of the global topological structure of the i-th node from the encoder.

Additionally, we reconstruct the adjacency matrix to ensure locality. Specifically, if two nodes are close, their hidden representations are also close, and vice versa.

$$L_3 = \|HH^T - A\| + \|FF^T - A\|$$  \hspace{1cm} (5)

Here, $H \in \mathbb{R}^{N \times d'}$ is hidden representations of the node attributes, and $F \in \mathbb{R}^{N \times d''}$ is the hidden representations of the global topological structure.

### 2.2.3. Fusion strategy

For each node, we have two types of hidden representations, which come from the topological structure and node attributes respectively. These hidden representations don’t contain information from each other. We fusion these two types of hidden representations on the basis of mutual information maximization. Specifically, a discriminator is introduced to help teach the encoders, which helps our model to yield the satisfied node representations. The discriminator teaches the encoders to magnify the mutual information between the hidden representations of the topological structure and of the node’s rich attributes for each node.

$$L_4 = \frac{1}{n + m} \left( \sum_{i=1}^{n} E_{(H,F)} \left[ \log D(h_i, f) \right] + \sum_{j=1}^{m} E_{(H,F)} \left[ \log \left( 1 - D(h_i, f) \right) \right] \right)$$  \hspace{1cm} (6)

Here a discriminator $D : \mathbb{R}^{d'} \times \mathbb{R}^{d''} \rightarrow \mathbb{R}$, is applied to represent the probability scores of the topology-attribute pair. By combining the hidden representation of node attribute with another hidden representation from other nodes, we draw the negative samples.

### 2.2.4. Object function

Therefore, to get the network embedding, we optimize the following objection function jointly:

$$L = \alpha (L_1 + L_2) + \beta L_3 + \gamma L_4$$  \hspace{1cm} (7)

Here, $\alpha, \beta, \gamma$ is the hyperparameter. By minimizing this objection function, we can obtain $h_i$ and $f_i$, then we concatenate them as the final low-dimensional representation of the node.

### 3. Results & Discussion

We employ the learned node embeddings on node classification tasks to evaluate our method. The classification outcomes are compared with other methods. We use a linear evaluation of the learned node embeddings, as is usual practice[1, 5]. On three well-known datasets, we employ these node embeddings to optimize a logistic regression model to handle multiclass node classification tasks.

#### 3.1. Dataset & baselines

Cora, Citeseer and Pubmed are three typical citation network benchmark datasets that we use. In all of these datasets, nodes are bag of words representations and edges are citation relations. A class label is assigned to each node. For comparison, we use Label Propagation(LP) [9] and DeepWalk [3], two state-of-the-art unsupervised approaches. We also present the results of training logistic regression using solely natural input features, as well as the concatenation of DeepWalk embeddings and the natural features of node. In addition to unsupervised methods, we also use strong supervised methods for comparison: Planetoid [10] and FastGCN [11].

#### 3.2. Node classification

Table 1 shows the node classification results on three benchmark datasets. After 20 runs of training, we provide the mean classification accuracy on the test dataset of our approach, and we reuse the metrics from Kipf & Welling [1] for the performance of DeepWalk, as well as Label Propagation (LP) [9] and Planetoid [10]. We also reuse the metrics already in Jiao & Xiong [12] for the performance of FastGCN
[11]. For all the there datasets, we set output dimension to 200 for our method. As per the outcomes, our model outperforms DeepWalk in node classification and is comparable to supervised learning methods.

| Method                    | Cora         | Citeseer     | Pubmed       |
|---------------------------|--------------|--------------|--------------|
| Raw features              | 47.9% ± 0.4% | 47.9% ± 0.4% | 047.9% ± 0.4% |
| LP                        | 68%          | 45.3%        | 63.0%        |
| DeepWalk                  | 67.2%        | 43.2%        | 65.3%        |
| DeepWalk + features       | 70.7% ± 0.6% | 51.4 ± 0.5%  | 74.3% ± 0.9% |
| Planetoid                 | 75.7%        | 64.7%        | 77.2%        |
| FastGCN                   | 78.0% ± 2.1% | 63.5% ± 1.8% | 74.4% ± 0.8% |
| DMIM(ours)                | 78.5% ± 0.8% | 64.5% ± 0.9% | 74.5% ± 1.8% |

4. Conclusions

In this paper, we propose a novel unsupervised learning method, deep attributed network embedding by mutual information maximization (DMIM), to learn node embeddings. The mutual information maximization between two types of hidden representations of graph topological structure and of node attributes can encourage the model to better fuse these two types information. This unsupervised learning model can learn node embeddings that are task-independent. In addition, it can employ deep network layers and apply to large-scale datasets than GNN based methods. We perform node classification tasks with the learned node embeddings to illustrate the model’s effectiveness. The results show that our unsupervised model outperforms DeepWalk in classification and achieves comparable results with supervised learning approaches on several tested datasets.

Acknowledgments

I would like to thank everyone who assisted me with this thesis during the writing process.

References

[1] Kipf, T.N., Welling, M. (2016) Semi-supervised classification with graph convolutional networks. J. arXiv preprint arXiv:1609.02907.
[2] Grover, A., Leskovec, J. (2016) node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. San Francisco. pp. 855-864.
[3] Perozzi, B., Al-Rfou, R., Skiena, S. (2014) Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. New York. pp. 701-710.
[4] Tang, J., Qu, M., Wang, M., Yan, J., Mei, Q.Z. (2015) Line: Large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web. Florence. pp. 1067-1077.
[5] Hamilton, W., Ying, Z.T., Leskovec, J. (2017) Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems. Long Bench. pp. 1024-1034.
[6] Duran, A.G., Niepert, M. (2017) Learning graph representations with embedding propagation. In: Advances in Neural Information Processing Systems. Long Bench. pp. 5119-5130.
[7] Gao, H., Huang, H. (2018) Deep Attributed Network Embedding. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. Stockholm. pp. 3364-3370.
[8] Velickovic, P., Fedus, W., Hamilton, W.L., Lio, P., Bengio, Y., Hjelm, R.D. (2019) Deep Graph infomax. In: Seventh International Conference on Learning Representations. New Orleans.
[9] Zhu, X.J., Gahramani, Z., Lafferty, J.D. (2003) Semi-supervised learning using gaussian fields and harmonic functions. In: Proceedings of the Twentieth International conference on Machine learning. Washington. pp. 912-919.
[10] Yang, Z., Cohen, W.W., Salakhutdinov, R. (2016) Revisiting semi-supervised learning with graph embedding. In: International conference on machine learning. New York. pp. 40-48.

[11] Chen, J., Ma, T., Cao, X. (2018) FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling. In: Sixth International Conference on Learning Representations. Vancouver.

[12] Jiao, Y., Xiong, Y., Zhang, J., Zhang, Y., Zhang, T., Zhu, Y. (2020) Sub-graph Contrast for Scalable Self-Supervised Graph Representation Learning. 2020 IEEE International Conference on Data Mining. Sorrento. pp. 222-231.