2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

Local and Global Information Preserved Network Embedding

Yao Ma
Data Science and Engineering Lab
Michigan State University
East Lansing, MI 48824, USA
mayao4@msu.edu

Suhang Wang
Data Mining and Machine Learning Lab
Arizona State University
Tempe, AZ 85281, USA
suhang.wang@asu.edu

Jiliang Tang
Data Science and Engineering Lab
Michigan State University
East Lansing, MI 48824, USA
tangjili@msu.edu

***Abstract*—Networks such as social networks, airplane networks, and citation networks are ubiquitous. To apply advanced machine learning algorithms to network data, low-dimensional and continuous representations are desired. To achieve this goal, many network embedding methods have been proposed recently. The majority of existing methods facilitate the local information i.e. local connections between nodes, to learn the representations, while neglecting global information (or node status), which has been proven to boost numerous network mining tasks such as link prediction and social recommendation. In this paper, we study the problem of preserving local and global information for network embedding. In particular, we introduce an approach to capture global information and propose a network embedding framework LOG, which can coherently model LOcal and Global information. Experiments demonstrate the effectiveness of the proposed framework.**

***Index Terms*—network, global information, embedding**

I. Introduction

Networks, such as social networks, airplane networks, and citation networks, are ubiquitous and important in our daily life. Learning meaningful node representation is important for network analysis tasks such as link prediction and node classification. Network embedding, which aims to learn low-dimensional and continuous node representations by preserving certain properties of the network, has attracted increasing attention in recent years [1]–[4]. The majority of existing network embedding algorithms exploits local information to learn node representations. By exploiting the local information, the learned vector representation of nodes captures certain local properties of the network, which has been demonstrated to advance many network analysis tasks such as link prediction [3], [4] and node classification [1], [4].

Despite the local information, global information (or node status) is another easily accessible but important information. Node status, which reflects where a node stands in the entire network, has different meanings under different "context". For example, in the web networks, status can indicate a relevancy ranking of webpages [5]; and in online social networks, it denotes reputations of users [6]. Most of the existing network embedding algorithms only facilitate local information by treating global status of each node equally. However, it is natural that nodes are of different global status. Many network based tasks such as web search [5] and social network recommendation [7] have been proven to be advanced by exploiting the global status. Therefore, global status can provide complementary information in addition to the local structure and incorporating global information has great potential to help learn better representation. However, the work on exploiting both local and global information is rather limited. Therefore, in this paper, we study a new problem of investigating both local and global information for network representation learning. The main contributions of the paper are summarized as follows:

- We propose a principled way to model global information for network representation learning;
- We propose a novel network embedding framework LOG, which integrates local and global information into a coherent model;
- We conduct experiments on real-world datasets to demonstrate the effectiveness of the proposed framework.

IEEE/ACM ASONAM 2018, August 28-31, 2018, Barcelona, Spain

II. Problem Statement

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a network, where $\mathcal{V} = \{v_1, \ldots, v_N\}$ denotes the set of N nodes and $\mathcal{E} = \{e_1, \ldots, e_M\}$ represents the set of M edges between these nodes. Furthermore, let $\mathbf{t} \in \mathbf{R}^{N\times 1}$ be the status scores to denote the global information for the N nodes. With the aforementioned notations and definitions, the problem under study is formally stated as:

Given a network $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ and the global information $\mathbf{t}$, we aim to learn the node representation matrix $\mathbf{U} \in \mathbf{R}^{N\times d}$ by preserving the local structure information as well as the global information, where d is the embedding dimension. Mathematically, the problem is written as:

$$Q(\mathcal{G}, \mathbf{t}) \rightarrow \mathbf{U} \tag{1}$$

where Q is the learning algorithm we will investigate.

III. The Proposed Framework

In this section, we introduce the proposed model. Our model consists of two components – 1) the component to preserve the global information and 2) the component to preserve the local information. We first describe the component to preserve the global information, which leads to a new algorithm GINE. We then introduce the component to preserve the local information and finally integrates both parts as the proposed framework LOG with both local and global information.

A. Global Information Preserved Network Embedding.

In this subsection, we introduce a new embedding algorithm GINE which can preserve the global information. The global status can be computed in various ways [5], [8], [9]. In this work, we use Pagerank [5]. The status rankings have been widely used in real-world applications instead of the status scores; hence, we first get the status rankings of nodes based on their status scores and then preserve the status rankings for network embedding.

To preserve the global ranking, we model this problem as a maximum likelihood problem. We maximize the probability that the ranking of the learned statuses for the nodes $\{v_1, \ldots, v_N\}$ follow the original Pagerank ranking $\{r_1, \ldots, r_N\}$. For convenience, we use the ranking as the index for the nodes, that is, we use $\{v_{(r_1)}, \ldots, v_{(r_N)}\}$ to represent the nodes. Intuitively, if we can preserve the relative ranking of all pairs of nodes, we can maintain the whole ranking for all nodes. We approximate this probability as

$$p_{global} = \prod_{1 \leq i < j \leq N} p(v_{(i)}, v_{(j)}); \tag{2}$$

where $p(v_{(i)}, v_{(j)})$ is the probability that node $v_{(i)}$ is ranked before node $v_{(j)}$

$$p(v_{(i)}, v_{(j)}) = \sigma(f(\mathbf{u}_i) - f(\mathbf{u}_j)), \tag{3}$$

where $f(\cdot)$ is the mapping function that maps the node representation to the status and $\sigma(\cdot)$ is the sigmoid function $\sigma(x) = 1/(1 + e^{-x})$. Different mapping functions can be chosen. In this paper, we adopt a linear function $f(\mathbf{u}) = \mathbf{w}^T\mathbf{u}$, where $\mathbf{w}$ is the parameter of the function.

The representations and the mapping function $f(\cdot)$ can be learned by minimizing the negative logarithm of (2)

$$\mathbf{U}, \mathbf{w} = \arg\min_{\mathbf{U},\mathbf{w}} - \sum_{1 \leq i < j \leq N} \log(p(v_{(i)}, v_{(j)})) \tag{4}$$

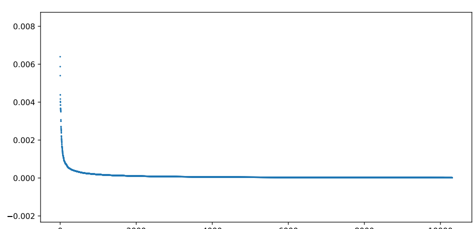


Fig. 1: Descending ordered pagerank score of nodes in BlogCatalog.

1) Reduce Computational Cost.: It is computational expensive to optimize (4) w.r.t $\mathbf{U}$ and $\mathbf{w}$ as there are $N(N-1)/2$ pairs in total. To accelerate the training speed, we relax the global ranking constraint. Our relaxation comes from the observation from the distribution of status scores. Figure 1 plots the Pagerank value of the BlogCatalog network [10] where x-axis is the ranking of nodes and y-axis is the PageRank score. The difference of the status scores of many nodes with different rankings, especially in the long tail, is very small. It is not useful to address the difference between these rankings. Therefore, we first transform the global status scores into status levels by splitting the scores into K equal size bins; and then we assume nodes in the same status level share the same status ranking. Instead of rankings 1 to N, now we have K ranking levels. Then the global ranking constraint is relaxed to preserve status levels of nodes.

To maintain the order between the nodes from different levels, we randomly generate a set of lists $\mathcal{L} = \{l_i\}_{i=1}^L$. Each list $l = \{v_{[1]}, v_{[2]} \ldots, v_{[K]}\}$ consists of K nodes, one from each level and the subscript of $v_{[i]}$ represents the level. The probability the ranking of all the lists hold can be modeled as:

$$p_{global} = \prod_{l \in \mathcal{L}} p(l); \tag{5}$$

where $p(l)$ is the probability that the order in list l is maintained, which is defined as

$$p(l) = \prod_{1 \leq i < j \leq K} p(v_{[i]}, v_{[j]}) \tag{6}$$

With this relaxation, the objective function can be formalized as follows

$$O_1(\mathbf{U}, \mathbf{w}) = -\sum_{l \in \mathcal{L}} \sum_{1 \leq i < j \leq K} \log(p(v_{[i]}, v_{[j]})) \tag{7}$$

where $\mathbf{U}_l$ is representations for the nodes in list l.

To optimize (7), we adopt the Stochastic Gradient Decent method.

B. Local and Global Information Preserved Embedding.

In this subsection, we first briefly describe the local information preserved embedding model and then introduce the proposed framework LOG, which can learn node representation preserving both local and global information.

1) Preserving Local Information: To learn node representation that can preserve local information, we follow word2vec [11]. The skip-gram model proposed in word2vec predicts surrounding context words given a center word. In the network setting, we view the nodes in a network as the "words". Then, for a node v, we regard all the nodes connected to v as the surrounding context "words", which can be denoted as $N(v)$. The probability that $N(v)$ is the surrounding "context" of node v can be modeled as:

$$p(N(v)|v) = \prod_{v_j \in N(v)} p(v_j|v); \tag{8}$$

where $p(v_j|v)$ can be modeled using a softmax function as suggested in [11].

$$p(v_j|v) = \frac{\exp(\mathbf{u}^T\mathbf{u'}_j)}{\sum_{v_i \in \mathcal{V}} \exp(\mathbf{u}^T\mathbf{u_i}')}; \tag{9}$$

where $\mathbf{u}$ and $\mathbf{u}'$ are the source and target representations for node v as suggested in [11].

To learn the node representations, we maximize the probability that $N(v)$ is the "context" of node v for all the nodes $v \in \mathcal{V}$ with respect to the node representations:

$$P_{local} = \prod_{v \in \mathcal{V}} p(N(v)|v). \tag{10}$$

The objective function to be minimized is the negative logarithm of (10)

$$O_2(\mathbf{U}) = -\sum_{v \in \mathcal{V}} \sum_{v_j \in N(v)} \log p(v_j|v) \tag{11}$$

2) The LOG Framework.: To learn the representations preserving both the local and global information, the global information needs to be incorporated into the local information preserved embedding model. We combine the two objective (11) and (7) as:

$$O(\mathbf{U}, \mathbf{U}', \mathbf{w}, \mathbf{w}') = -\sum_{v \in \mathcal{V}} \sum_{v_j \in N(v)} \log p(v_j|v) \quad (12)$$
$$- \lambda \sum_{l \in L} \sum_{1 \le i < j \le K} \log(\sigma(f(\mathbf{u}_{[i]}, \mathbf{u}'_{[i]}) - f(\mathbf{u}_{[j]}, \mathbf{u}'_{[j]}))$$

where $\lambda \in [0, 1]$ is a hyperparameter which indicates the importance of the global information.

The node representations $\mathbf{U}, \mathbf{U}'$ and the parameters $\mathbf{w}, \mathbf{w}'$ can be obtained by minimizing the objective function (12). Note that, the global part of this objective function can be used to extend some existing network embedding methods such as LINE to incorporate the global information.

Since for each node, we have two representations, hence, we need to slightly modify the mapping function f as

$$f(\mathbf{u}, \mathbf{u}') = \mathbf{w}^T \mathbf{u} + \mathbf{w}'^T \mathbf{u}' \quad (13)$$

3) An Optimization Method for LOG.: We observe that the minimization of the first term of (12) is computational expensive due to the summation over the whole set of nodes $\mathcal{V}$ in (9). Thus, we adopt the negative sampling approach proposed in [11] to solve the computational issue in (9). By using the negative sampling method, we replace each $\log p(v_i|v)$ with

$$g(v, v_i) = \log \sigma(\mathbf{u}^T \mathbf{u}'_i) + \sum_{n=1}^{N_e} \log \sigma(-\mathbf{u}^T \mathbf{u}'_{\{n\}}); \quad (14)$$

where $\sigma(x)$ is the sigmoid function and Ne is the number of negative samples. The negative samples are sampled from the node set $\mathcal{V}$ according to the noise distribution $P(v) \sim d_v^{3/4}$ as proposed in [11], where, d_v is the in-degree of the node v. The parameters $\mathbf{U}, \mathbf{U}'$, $\mathbf{w}$ and $\mathbf{w}'$ can be learned by minimizing (12) with negative sampling using Stochastic Gradient Decent method.

IV. Experiments

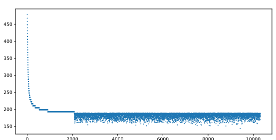
In this section, we conduct experiments to verify the effectiveness of the proposed algorithms GINE and LOG. We first show that the embeddings learned by GINE and LOG can preserve the global information. Then, we perform link prediction to verify that including the global information can help learn better representations. We conduct experiments on two network datasets, which have been widely used to evaluate network embedding algorithms [1], [12].

- **BlogCatalog** [10] is a network of social relationships provided by blogger authors. This network consists of $10,312$ nodes and $333,983$ edges.
- **Flickr** [10] is a network of contacts between users from the photo sharing website Flickr. This network consists of $80,513$ nodes and $5,899,882$ edges.

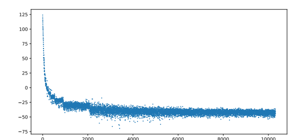
In our experiments, we set the number of levels K to 60.

A. Validating Global Information Preserving.

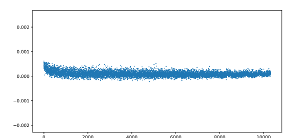
To show that the representations learned by our algorithms can preserve the global information, we visualize the global status of all the nodes in the network. The scatter plot of the PageRank score of BlogCatalog has been shown in Figure 1. In this figure, the nodes are arranged in a descending order based on the PageRank score. For the plots of the algorithms, we also arrange the nodes in this order. Figures 2a, 2b, 2c and 2d show the global status preserved by GINE, LOG, DeepWalk and LINE respectively. DeepWalk and LINE do not learn a mapping function during the representation learning, therefore, we perform a linear regression using the learned representations and the original PageRank scores and then use this linear model to calculate the global status for each node. We only show the results in the BlogCatalog network as the



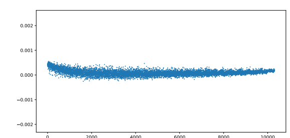
(a) GINE



(b) LOG with $\lambda = 0.5$



(c) DeepWalk



(d) LINE

Fig. 2: Global Status Preserved by Different Methods

performance in the Flickr network are similar. As shown in Figure 2a, GINE can preserve the global status quite well, and the difference between nodes in different levels are clear. LOG can also roughly preserve the ranking information, the trend has been preserved, and the difference between levels can also be observed clearly. Figure 2c and 2d do not show a specific trend, which means the representations learned by DeepWalk and LINE cannot preserve the global information.

B. Link Prediction.

In this subsection, we perform link prediction task to further evaluate the effectiveness of the learned representations.

1) Experiments Setting: In the link prediction task, a certain fraction of edges are removed from the network, we are supposed to use the remained network to predict whether the "missing" edges exist.

We set up two groups of experiments where 20% and 80% of edges are removed, while at least one edge for each node remains in the network. After removing the edges, we use the remained network to learn the node representations. Then, to perform link prediction, we use the element-wise addition of the two node representations as the representation of the edge.

To form the training set, we put all the edges remained in the network as positive samples and then sample an equal number of non-connected node pairs as negative samples. The testing set is formed in a similar way. After forming the training set and testing set, we train a binary classifier using logistic regression on the training set and perform link prediction on the testing set. In this work, we use AUC as the metrics to evaluate the link prediction performance.

DataSet		BlogCatalog		Flickr	
	% removed edges	20%	80%	20%	80%
AUC	LOG(0.3)	**0.9473**	**0.9491**	**0.9113**	**0.9115**
	LINE	0.9398	0.9284	0.8805	0.9064
	DeepWalk	0.9382	0.9139	0.8896	0.7836
	HFB	0.8693	0.8249	0.8635	0.7852
	GINE	0.9193	0.9007	0.8591	0.8746
	LOG(0)	0.9387	0.9275	0.8881	0.8883

TABLE I: Link Prediction Performance

2) Performance Comparison.: To evaluate the performance of our algorithms, we compare them with the following representative baselines:

- **LINE** [2] is a network embedding algorithm which can preserve the first-order and second-order proximity.
- **DeepWalk** [1] facilitates local information obtained from random walk to perform network embedding.
- **Handcrafted Feature Based(HFB)** consists of the number of common neighbors, Jaccard coefficient and preferential attachment value [13] as the representations for a node pair.
- **LOG(0)** is a variant of our method, where we set $\lambda = 0$ so that only local information is used to learn the representations.

For all the methods except for HFB, we set the dimension to 128. The experiment results are shown in Table I. For LOG, we set $\lambda = 0.3$, so we denote it as LOG(0.3) in Table I. The following observations can be made from the results

- All embedding algorithms obtain better performance than HFB, which supports the advantage of the automated representation learning algorithms for networks.
- GINE provides reasonable results, which shows the importance and effectiveness of the global information.
- LOG(0.3) consistently outperforms LOG(0), which further proves the importance of the Global information.
- LOG(0.3) outperforms all the baselines in each setting of the two datasets, which shows that (1) local and global information are complementary; and (2) incorporating them can lead to better presentation learning.

V. Related Work

In this section, we briefly review some works related to our problem. Network embedding algorithms, which aim to learn low-dimensional node representation are attracting increasing attention recently. Inspired by word2vec [11], DeepWalk [1] and LINE [2] are proposed. node2vec [4] further extends DeepWalk by introducing parameters to allow biased random walk to explore the neighborhood of nodes. struc2vec [14] learns the node representation from a different perspective and it tries to preserve the structural identity between nodes. In [3], a signed network embedding algorithm SiNE is proposed based on the notion that a user should be closer to their "friend" than their "enemy". Two recent surveys [15], [16] give a comprehensive overview of network embedding algorithms. However, most of these existing methods cannot preserve the global information. In this paper, we propose a model which can preserve the local information as well as the global information.

VI. Discussion and Future Work

In this paper, we propose a principal way to model global information for network embedding. Based on this, we further propose a novel network embedding framework LOG, which can preserve both local and global information. The results of experiments show that LOG can well preserve the global information.The performance of link prediction further demonstrates that the inclusion of the global information can lead to better representation learning.

In this paper, we use a linear function to model the global information. However, more complex functions can be adopted. The global status score we use in this paper is PageRank, while other global status scores could also be used. We will investigate these different possibilities in the future.

References

[1] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 701–710.

[2] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[3] S. Wang, J. Tang, C. C. Aggarwal, Y. Chang, and H. Liu, "Signed network embedding in social media," in *Proceedings of SDM*, 2017, pp. 327–335.

[4] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 855–864.

[5] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.

[6] P. Massa, "A survey of trust use and modeling in real online systems," in *Trust in E-services: Technologies, Practices and Challenges*. IGI Global, 2007, pp. 51–83.

[7] J. Tang, H. Gao, X. Hu, and H. Liu, "Exploiting homophily effect for trust prediction," in *Proceedings of the sixth ACM international conference on Web search and data mining*. ACM, 2013, pp. 53–62.

[8] K. Okamoto, W. Chen, and X.-Y. Li, "Ranking of closeness centrality for large-scale social networks," *Lecture Notes in Computer Science*, vol. 5059, pp. 186–195, 2008.

[9] M. E. Newman, "A measure of betweenness centrality based on random walks," *Social networks*, vol. 27, no. 1, pp. 39–54, 2005.

[10] R. Zafarani and H. Liu, "Social computing data repository at ASU," 2009. [Online]. Available: http://socialcomputing.asu.edu

[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[12] D. Wang, P. Cui, and W. Zhu, "Structural deep network embedding," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 1225–1234.

[13] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *journal of the Association for Information Science and Technology*, vol. 58, no. 7, pp. 1019–1031, 2007.

[14] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 385–394.

[15] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," *arXiv preprint arXiv:1709.05584*, 2017.

[16] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *arXiv preprint arXiv:1705.02801*, 2017.