2019 Scientific Conference on Network, Power Systems and Computing (NPSC 2019)

Overlapping Community Discovery Algorithm Based on Label Propagation

Zhang Meng
School of Computer Science
Nanjing University of Posts and Telecommunications
Nanjing 210023, China
zmnupt@163.com

Li Lingjuan
School of Computer Science
Nanjing University of Posts and Telecommunications
Nanjing 210023, China
Jiangsu Key Laboratory of Big Data Security &
Intelligent Processing
Nanjing 210023, China
lilj@njupt.edu.cn

***Abstract*—Based on the label propagation algorithm, the SLPA discovers the overlapping communities in the network through the dynamic process of interaction between Speaker and Listener. The time complexity is approximately linear. However, there is randomness in the process of label propagation, and the initialization of node labels takes a lot of resources when it is applied to large-scale networks. To solve the above problems, an overlapping community discovery algorithm LP-OCD based on label propagation is designed by improving SLPA. The algorithm pre-processes the network with the K-shell decomposition algorithm to remove the edge layer nodes before each node memory initializes the label. In the label propagation phase, the randomness of the algorithm is reduced by improving Speaking and Listening strategies. Labels of all edge layer nodes in the post-processing phase are determined by the information of their neighbors. Experimental results on social networks and synthetic networks show that the LP-OCD algorithm not only has approximately linear time complexity, but also significantly improves the quality of the overlapping communities discovered.**

Keywords—overlapping community; SLPA; label propagation; K-shell decomposition algorithm

I. Introduction

The social relationship between people in the real world, the biological relationship between cells and the link structure between the World Wide Web can be represented by complex network models. In such networks, there is a community structure. The nodes within the same community are closely connected, while the nodes between different communities are sparsely connected. In many networks, communities are intersected, that is a node may belong to multiple communities. For example, in social networks, a teacher may participate in multiple research projects. In a biomolecular network, a single gene often participates in the expression of multiple biological functions. Therefore, revealing the overlapping community structure in the network is of great significance for understanding the network structure and discovering the potential relationships in the network.

In recent years, researchers have proposed a number of overlapping community discovery algorithms, including: complete subgraph based algorithms CPM [1]and EAGLE, label propagation based algorithms COPRA [2]and SLPA [3], local optimization based algorithms DEMON and OSLOM. Although these algorithms can discover overlapping community structures, they have certain defects. For example, CPM and EAGLE have higher time complexity. Although SLPA and COPRA have lower time complexity, the results are not stable enough.

This paper draws on the idea of SLPA to further improve efficiency. By introducing K-shell decomposition algorithm to deal with edge layer nodes, improving Speaking strategy and Listening strategy, this paper designs a label-based propagation method that can quickly and stably discovery overlapping communities LP-OCD (Overlapping Community Discovery Algorithm Based on Label Propagation).

II. Analysis of SLPA

A. Principle of SLPA Algorithm

Speaker-listener Label Propagation Algorithm (SLPA) is an extension of Label Propagation Algorithm (LPA) [4]. In LPA, each node only has one label. During the label updating process, each node selects the label that appears most frequently in the neighbor node according to (1) to update the label:

$$l_i = arg\ \max_{l} \sum_{j \in N(i)} \delta(l_j, l) \tag{1}$$

Where l_i represents the label of the node i to be updated, $N(i)$ represents the set of neighbor nodes of the node i, l_j represents the label of the neighbor node j of the node i, and $\delta(l_j,l)$ is a Kronecker delta.

SLPA introduces Speaker and Listener to simulate human communication behavior. The algorithm sets two parameters, the maximum number of iterations T and the threshold r. During the operation of SLPA (T, r) algorithm, the historical label sequence of each node in the refresh iteration process will be recorded. When the maximum number of iterations is reached, each node will save a sequence of length T. The probability of each label occurrence in the sequence represents the degree of attribution of the current node. By setting the threshold r, the label with a probability less than r in each node history label sequence is deleted, and the label with a probability greater than r is retained. The general steps of the SLPA are as follows:

1) Initialization phase: The node id is used to initialize the memory of each node.

2) Label propagation phase: Repeat the following steps until the maximum number of iterations is reached:

- Select a node as the Listener.
- Each neighbor node of the selected node randomly selects a tag from its memory.
- The Listener adopts an asynchronous update policy and selects the label with the highest frequency in the label set of the neighbor node as the current update label, and loads it into the memory.

3) Post-processing phase: Post-processing is performed using the label information in the node memory.

B. Problems with SLPA

Although SLPA can discover overlapping communities with low time complexity, it still has the following disadvantages:

1) In the label initialization phase: SLPA initializes each node memory with the node id. By analyzing the results of SLPA, it can be found that the number of label types that all nodes have at the end is much smaller than the number of label types initially allocated, and most of the labels will disappear during the iterative update process. It takes a lot of time and space when applied to a large-scale network.

2) In the label propagation phase: The Listener node is selected each time according to a random sequence, and the Speaking policy randomly selects a label from the label memory of the neighbor node, while Listening policy selects the label with the highest number of occurrence times as the update label from the label set of the neighbor node, if there are multiple labels with the most number of times, randomly select one label. These stochastic strategies affect the stability of the algorithm and reduce the accuracy of the algorithm.

III. Design of Overlapping Community Discovery Algorithm Based on Label Propagation LP-OCD

In order to reduce the randomness and resource consumption of SLPA and improve the efficiency, accuracy and stability of overlapping community discovery, overlapping community discovery algorithm based on label propagation LP-OCD designed in this paper retains the linear time complexity of SLPA algorithm. Firstly, the K-shell decomposition algorithm is used to pre-process the network, remove the nodes with less influence on the edge layer, and reduce the number of label allocation in the initialization phase. Secondly, in the label update phase, the nodes are updated in descending order of their influence. The Speaking strategy of each neighbor node is changed to select the label with the highest occurrence in the memory, and the Listening strategy of each node to be updated is changed to select the label with the greatest influence in the neighbor node to reduce randomness. Finally, the labels of the edge layer nodes are determined by their neighbors.

A. Design of Node Comprehensive Influence and Its Calculation Method

There are many methods to evaluate the influence of nodes in complex networks. Because the K-shell decomposition algorithm proposed by Kitsak [5] has $O(n)$ time complexity and can accurately measure the global influence of nodes in networks. This paper refers to this method and improves it.

Assuming that there are no isolated nodes in the network, the general steps of K-shell algorithm are as follows: Firstly, delete all nodes with a degree equal to 1 in the network. If a new node with degree 1 appears in the deletion process, continue to delete until there are no nodes with degree 1 in the network. At this time, the K-shell value of these deleted nodes is 1. Then, all the nodes with degree 2 in the network are deleted in the same way. This is repeated until the K-shell value of all nodes in the network is determined. The larger the K-shell value is, the more core the node is located in the network, the greater its influence will be.

However, K-shell is a coarse-grained method to calculate the influence of nodes. Nodes in the same layer are given the same K-shell value, and their influence cannot be distinguished. So this paper further integrates the value of node normalization degree that can reflect the local information of the node and comprehensively considers the node influence, which is called the comprehensive influence of nodes.

The comprehensive influence of nodes is calculated as follows:

$$CI(i) = Ks(i) + \frac{d(i)}{max\{d(k) \mid k \in V\}} \quad (2)$$

Where $Ks(i)$ represents the K-shell value of node i. $d(i)$ represents the degree of node i. V represents the node in the network. $CI(i)$ represents the comprehensive influence of node i.

B. Pre-processing Phase of LP-OCD Algorithm

The node with the smallest K-shell value in the network is called the edge layer node, which is represented by $brink(i)$. Its definition is as follows:

$$brink(i) = node(Ks = min(Ks)) \quad (3)$$

Where Ks represents the K-shell value of node.

C. Label Propagation Phase of LP-OCD Algorithm

LP-OCD algorithm firstly selects Listener nodes in descending order of influence to improve the stability of the community discovery results when the algorithm sets a small memory. Secondly, the Speaking policy of each neighbor node of the Listener nodes is to select the label with the highest occurrence in memory. Finally, considering that the larger the influence of the neighbor node is, the more the number of the same label in the neighbor node is, the easier it is to propagate the label to the Listener node. This paper combines the number of label occurrences in neighbor nodes with the influence of neighbor nodes to examine the comprehensive influence of the label.

The formula for calculating the comprehensive influence of labels is as follows:

$$Influence(l) = \sum_{i \in N^l(x)} CI(i) \quad (4)$$

Where $N^l(x)$ represents a set of neighbor nodes whose node x is labeled l.

The LP-OCD algorithm reduces the randomness of the algorithm by improving the update order of Listener nodes and the Speaking and Listening strategies.

D. Post-processing Phase of LP-OCD Algorithm

At the end of the algorithm, according to the threshold r, the label whose number of occurrences of the different label in each node's memory is less than r is deleted. For the edge layer nodes that have been deleted, the labels are determined by the label with the largest influence in the neighbor nodes.

If the neighbor nodes have no influence (ie, the neighbor node are also the edge layer nodes), then they are determined by the nodes with the label in the neighbor nodes. Finally, nodes with the same label belong to the same community. If a node contains multiple labels, it is an overlapping node and belongs to multiple communities.

E. Time Complexity Analysis of LP-OCD Algorithm

Assuming that there are n nodes in the network, the time complexity of the network pre-processing phase and post-processing phase are both $O(n)$. In the label propagation phase, the number of external loops is T, which is a small constant, and the inner loop is controlled by n. For the Speaking strategy, the label with the highest number of occurrences is selected from the memory, and the time complexity is $O(T)$. For the Listening strategy, the time required to select a label from the neighbor nodes' label set is $O(k)$, where k *is* the network average degree, so the time complexity of label propagation phase is $O(Tn(T+k)) \sim O(n)$. Therefore, the total time complexity of the LP-OCD algorithm is close to $O(n)$, which inherits the advantage that the time complexity of the SLPA is approximately linear.

IV. Experiment and Results Analysis

In order to verify the performance of the LP-OCD algorithm, the comparison experiments between LP-OCD algorithm and classical CPM algorithm, as well as COPRA and SLPA, which are both extensions of LPA, are done.

A. Evaluation Indicators of Experimental Results

1) Modularity: In this paper, the modularity Q_{ov} [6]proposed by Nicosia et al. is used to evaluate the results of overlapping community discovery. The definition is as follows:

$$Q_{ov} = \frac{1}{m}\sum_{c\in C}\sum_{i,j\in V}\left[\beta_{l(i,j),c}A_{ij} - \frac{\beta^{out}_{l(i,j),c}k_i^{out}\beta^{in}_{l(i,j),c}k_j^{in}}{m}\right] \quad (5)$$

Where A_{ij} represents the adjacency matrix of the network. C represents the overlapping community set. $l(i,j)$ represents the edge from node i to node j. k_i^{out}, k_j^{in} respectively represent the output degree of node i and the input degree of node j. $\beta_{l(i,j),c}$ represents the degree of membership of the edge $l(i,j)$ belonging to the community $c(c \in C)$. $\beta^{out}_{l(i,j),c}k_i^{out}$ and $\beta^{in}_{l(i,j),c}k_i^{in}$ respectively represent the mathematical expectation that the starting point and the end point belong to the degree of membership of the community $c(c \in C)$. The value of modularity Q_{ov} is 0 to 1, and the value is closer to 1, the better the quality of the communities discovered.

2) Normalized mutual information: Normalized mutual information(*NMI*) is a community quality evaluation indicator based on information theory, which can be used to measure the similarity between the known community structure and the community structure discovered by the algorithm. This paper uses the *NMI* indicator proposed by Lancichinetti [7]for evaluating overlapping communities, which is defined as follows:

$$NMI(X \mid Y) = 1 - \frac{H(X \mid Y) + H(Y \mid X)}{2} \quad (6)$$

Where X represents the set of real communities. Y represents the set of communities discovered by the algorithm. $H(X \mid Y)$ represents the normalized conditional entropy of X on Y. The value of *NMI* is 0 to 1. The closer the value is to 1, the higher the consistency between the community structure discovered by the algorithm and the real community structure is.

B. Experimental Data

1) Social network: The experiment uses four social network datasets, namely the Zachary Karate Club Network, the Lusseau Dolphin Social Network, the American College Football League Schedule Network and the American Political Books Network. The specific information is shown in Table I.

TABLE I. Social Network

Network	Nodes	Edges
Karate	34	78
Dolphins	62	159
Football	115	616
Polbooks	105	441

2) Synthetic network: We use the LFR benchmark network generation tool [8]to generate two sets of synthetic networks with 5000 nodes. The specific parameters are shown in Table II.

TABLE II. Synthetic Network

Network	k	maxk	minc	maxc	mu	on	om
G1	10	50	20	100	0.1	500	2~8
G2	10	50	20	100	0.3	500	2~8

Where *k* represents the average value of the network. *maxk* represents the maximum value of the nodes of the network. *minc* and *maxc* respectively represent the minimum and maximum values of the number of nodes in the community. *mu* is a mixed parameter, which represents the proportion of the number of edges connecting different community nodes in the total number of edges of the network. Different mu values can be set to test the performance of the algorithm. *on* represents number of overlapping nodes; *om* represents the number of communities to which the overlapping nodes belong. The larger the *om*, the more difficult the algorithm discovers.

C. Social Network Experiment Results

In the initial phase of the LP-OCD algorithm, the four real networks are pre-processed first. According to the improved K-shell algorithm, the nodes with the smallest influence on the outermost layer of the network are stripped. After pre-processing, the Karate network nodes are reduced by 2.9%, the Dolphins network nodes by 14.5%, Football network nodes by 0.8%, the Polbooks network nodes by 1.9%.

The LP-OCD algorithm is compared with the CPM algorithm, the SLPA and the COPRA respectively, in which the SLPA and COPRA run 100 times to obtain the mean value. The results are shown in Table III. It can be seen from Table 3 that the LP-OCD algorithm has the highest value of Q_{ov} on the four social networks, and the standard deviation (std) is the lowest, which indicates that the LP-OCD algorithm discovers that the quality and stability of the overlapping community discovered by the LP-OCD algorithm are better than SLPA, COPRA and CPM algorithm.

TABLE III. SOCIAL NETWORK COMMUNITY DISCOVERY RESULTS

Network	Algorithm	Q_{ov}	std	Parameter
Karate	LP-OCD	0.66	0.20	r=0.45
	SLPA	0.65	0.21	r=0.33
	COPRA	0.44	0.21	v=3
	CPM	0.52	—	k=3
Dolphins	LP-OCD	0.74	0.03	r=0.35
	SLPA	0.71	0.04	r=0.45
	COPRA	0.69	0.05	v=4
	CPM	0.66	—	k=3
Football	LP-OCD	0.70	0.01	r=0.4
	SLPA	0.70	0.01	r=0.45
	COPRA	0.69	0.03	v=2
	CPM	0.64	—	k=4
Polbooks	LP-OCD	0.83	0.005	r=0.35
	SLPA	0.80	0.01	r=0.45
	COPRA	0.82	0.05	v=2
	CPM	0.79	—	k=3

D. Synthesis Network Experimental Results

The values of *NMI* are compared for the LP-OCD, SLPA, COPRA, and CPM algorithm, where the threshold *r* of LP-OCD and SLPA is set to the corresponding value when their mean values of *NMI* are the maximum, and the results are shown in Fig.1. It can be seen that as the mixed parameter and the number of communities of each node increase, the difficulty of community discovery also increases, and the values of *NMI* of the four algorithms generally decrease, which is also in line with the actual situation. Compared with COPRA and SLPA, the *NMI* values of LP-OCD and COPRA decrease more smoothly and have better robustness, and the LP-OCD algorithm is obviously superior to other algorithms in high-hybrid networks. While SLPA has fluctuations, which is due to the random phenomena of Speaking and Listening strategies in SLPA.

V. CONCLUSION

In order to effectively discover overlapping communities, this paper improves the SLPA and designs an algorithm LP-OCD based on label propagation, which can quickly and stably discover overlapping communities. The algorithm first pre-processes the network, removes a small number of influential edge layer nodes in the network without affecting the quality of the overlapping community. Secondly, it improves the randomness of the Speaking and Listening strategies of the SLPA, and updates the nodes in a certain order. Finally, it updates the labels of the edge layer nodes. Experiments show that compared with SLPA, COPRA and CPM algorithm, LP-OCD algorithm can effectively improve the stability and accuracy of overlapping community discovery results.

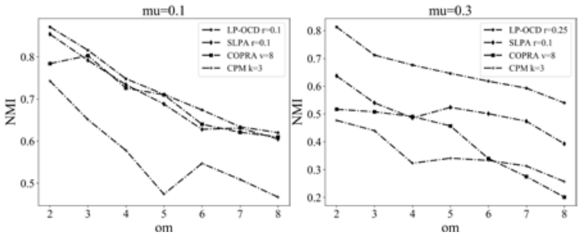

Figure 1 The values of *NMI*.

REFERENCES

[1] Palla G, Derényi I, Farkas I, et al. Uncovering the overlapping community structure of complex networks in nature and society [J]. Nature, 2005, 435(7043): 814-818.

[2] Steve G. Finding overlapping communities in networks by label propagation [J]. New Journal of Physics, 2010, 12(10): 103018.

[3] Xie J, Szymanski B K, Liu X. SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process [C] //2011 IEEE 11th International Conference on Data Mining Workshops. IEEE Computer Society, 2011: 344-349.

[4] Raghavan U N, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks [J]. Physical Review E, 2007, 76(3): 036106.

[5] Kitsak M, Gallos L K, Havlin S, et al. Identification of influential spreaders in complex networks [J]. Nature Physics, 2010, 6(11): 888-893.

[6] Nicosia V, Mangioni G, Carchiolo V, et al. Extending the definition of modularity to directed graphs with overlapping communities [J/OL]. Jounral of Statistical Mechanics:Theory and Experiment, 2009, 2009(3): P03024.

[7] Lancichinetti A, Fortunato S, Kertész J. Detecting the overlapping and hierarchical community structure in complex networks [J]. New Journal of Physics, 2009, 11(3): 033015.

[8] Lancichinetti A, Fortunato S. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities [J]. Physical Review E, 2009, 80(1): 016118.