TELKOMNIKA, Vol. 11, No. 8, August 2013, pp. 4484~4490
e-ISSN: 2087-278X

Community Detection Algorithm based on Neighbor Similarity

Jianjun Cheng, Hong Xu, Mahmud Gaybullaev, Mingwei Leng, Xiaoyun Chen*
School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China
Corresponding author, e-mail: chengjianjun@lzu.edu.cn, hxu11@lzu.edu.cn, chenxy@lzu.edu.cn

Abstract

Many complex networks have displayed the community structures, and the detection of community structure can give insights into the structural and functional information of these complex networks. In this paper, we proposed a neighbor similarity based new algorithm for community structure detection, in which we only consider the similarities between a node and its unclassified neighbors in the breadth-first traversal order, without considering other nodes influences; we take this node as a father node and its neighbors as the children nodes, to find out those children nodes which should belong in the same community with their father node. Then these children nodes are processed in the same way as their father node recursively, until the termination condition is reached. The most prominent property of our algorithm is that it has near liner time complexity, and furthermore it is a deterministic algorithm. We have tested our algorithm on several real networks, compared with some other algorithms, and the results have manifested that our algorithm outperforms the previous algorithms significantly.

Keywords*: community detection, networks, neighbor similarity, breadth-first traversal*

1. Introduction

In recent years, many scientific systems can always be represented as complex networks [1-4], e.g., the Internet and world wide web is a network formed by web pages and hyperlinks, the social network represents the relationships among people, and the food web represents the relationships among predators and preys, *etc*. In these networks, the nodes represent the objects or entities in the systems, and the edges represent the relationships or connections between the objects or entities. These networks may have some interesting characteristics, one of the most common and prominent property is its *community structure*. Although, to the concept of community, there is no unified definition at present yet [5-7], most of the researchers have reached a consensus that communities in a network indicate groups of the nodes, such that the nodes within a group are connected more often than those across different groups [8-10].

To detect the community structure of a network is of great significance, because the community structure of a network can give us some insights into the structural and functional information of the network. So the study of community detection has aroused many researchers' interests and attentions, and many algorithms have been brought out in the last decades [5], such as edge betweenness based method [8, 11], modularity optimization methods [12], LPA (Label Propagation Algorithm) and its variations [13-16], *etc*. Many of the aforementioned methods have a relative high computation demand, thus can not be used to deal with very large networks; Compared with these algorithms, LPA (Label Propagation Algorithm) has near linear time complexity, but it is not a deterministic algorithm.

To solve these problems, in this paper, we propose a neighbor similarity based algorithm (NSA), to identify the community structure in the network. The most prominent property of our algorithm is that it also has near liner time complexity, and furthermore it is a deterministic algorithm.

The rest of this paper is organized as follows: section II is the introduction of some related work for community detection; the proposed algorithm based on neighbor similarity is elaborated in section III; section IV is the experiments and results analysis; conclusions is arranged in section V.

Received November 17, 2012; Revised April 10, 2013; Accepted May 19, 2013

2. Related Work

Many algorithms have been proposed in the past years. Among them, the most famous one is the GN algorithm originated by Girvan and Newman [11]. It repeatedly calculates the betweenness for all the edges in the network, removes the edge with the highest betweenness from the network, until all the edges are removed. Along with the GN algorithm, [11] also proposed a concept named as "*modularity*" *Q*, to measure the goodness of a community structure. Although the GN algorithm has many successful applications, but its computation demand is too high to be used in very large networks.

To increase the computation efficiency, Newman has proposed a fast algorithm based on the idea of modularity optimization (Fast*Q*) [12]. In the algorithm, each node of the network is considered as a community initially, and then the algorithm chooses two communities to merge into one iteratively, until all the nodes are merged into the same community. In the process, each merge should result in the greatest increase (or smallest decrease) of modularity *Q*. The outputs of both the GN and FAST*Q* are dendrograms to depict the community structures in the networks. Each level of the dendrogram represents a community structure, and the best community structure can be pursued by seeking the maximal value of modularity.

LPA (Label Propagation Algorithm) [13] is a near linear time complexity algorithm for community detection proposed by Raghavan *et al*. The main idea of this algorithm is: if a given node x has k neighbors $x_1, x_2, ...x_k$ and each neighbor of the node has a label indicating the community in which it should belong, then the node x updates its label to the one most of its neighbors have. The process continues iteratively until every node has the label carried by most of its neighbors. In this way, the labels are propagated in the whole network, and at the end of propagation, a group of nodes have the same label form a community. The disadvantage of LPA is that it is sensitive to the label updating order of the nodes, *i.e.*, it is not a deterministic algorithm. That means running the algorithm many times against a given networks, the outputs of LPA might not be identical. But LPA has its near linear time complexity, its computation demand is far lower than the GN algorithm. Just because of this, many improvements and variants have been brought out based on the LPA. Among them, *LPAm* [14] is a representative that modified the label updating rule of LPA to pursue the maximal modularity of the community division.

3. Neighbor Similarity based Algorithm for Community Detection

To detect the community structure in a network, we often can exploit some information based on the topology of the network. For example, in social networks, one people might know clearly something about his acquaintances (there are edge connections between them, so the people and his acquaintances are neighbors each other), but he cannot know the counterpart of a stranger (the people and the stranger are not neighbors). Inspired by this phenomenon, we proposed a neighbor similarity based algorithm (NSA) to detect the community structure from a network in this paper.

In NSA, we only make utilization of the relationship between every node and its unclassified neighbors in the breadth-first traversal order, to determine whether the node and some of its neighbors should belong in the same community, without considering influences of any other nodes. The basic idea of NSA is simple, for each node, we call it a father node and take its unclassified neighbors as its child (ren) nodes. If the relationship between the node NA and NB is father node and child node, and the similarity between NA and NB is greater than a given threshold τ , the child node NB is inserted into its father node's community.

The concept of similarity between a node and his father node is very important to the algorithm. Any form of similarity measure can be employed; matched with the basic idea of NSA, in this paper, we only utilize some numerical values associated with a node and its father node to compute the similarity between them; these numerical values are the degree of the node, the degree of its father node, and the number of the common neighbors of the node and its father node, respectively. The proposed similarity measure between two nodes in a network is formulated in the form of definition following.

Definition: (Similarity between nodes) the similarity between two connected nodes i and j is computed as the following formula:

$$similarity \quad (i, j) = \frac{(k_i - 1) \times (k_j - 1)}{(k_i + k_j - n(i, j) - 2)^2}$$

Where, $n(i, j)$ is the number of the common neighbors between node i and node j; k_i and k_j are the degrees of node i and node j, respectively.

NSA is a two-stage algorithm. In the first stage, a similarity threshold τ is employed, and the node set of the network is divided into some groups correspond to the mediate communities according to the value of τ; The second stage is an optimization stage, the nodes in some of the mediate communities whose node number is less than the given threshold θ are redistributed into other communities. Here, the majority voting strategy is employed to determine which community a node should redistributed into.

Algorithm I
Input: An undirected and unweighted graph $G(V, E)$
A similarity threshold τ
Output: A community structure of Graph $G(V, E)$
1. $C = \emptyset$; // C is the set of communities
2. **While** ($V \neq \emptyset$)
3. $v = \arg\max_{x \in V}(\text{degree}(x))$
4. $U = \{v\}$; // U is the newly created community
5. $F = \{v\}$; // F is the set of father nodes
6. **While** ($F \neq \emptyset$)
7. **For** each v in F do
8. **For** each u in $\text{unclassifiedchildren}(v)$ do
9. **If** $simlarity(v,u) \geq \tau$ do
10. $U = U \bigcup \{u\}; F = F \bigcup \{u\}; V = V/\{u\};$
11. **End if**
12. **End for**
13. $F = F/\{v\}; V = V/\{v\};$
14. **End for**
15. **End while**
16. $C = C \bigcup \{U\};$
17. **End while**
18. Output C;

Figure 2. The First Stage of NSA

The pseudo-code of the first stage of NSA is depicted as algorithm I in Figure 1. It is an iterative process, and in each iteration, we select the node v with the largest degree from the unclassified nodes set V, insert it into the set U, that means a new community has been created; At the same time, we insert the selected node to the set F, that means it is now a father node. Taking this setting as the tipping point, the newly created community begins to expand: for each node v in the set F, and each unclassified child node u of v, if the similarity between v and u is larger than the similarity threshold τ, the child node u and its father node v should belong in the same community, so the child node u is inserted into the community U. And the relationship between u and its children should be considered, i.e., the node u should be a father node now, so we also insert it into the set F. After it is processed, the node v is deleted from the sets V and F. At the end of each iteration, all the nodes in the set U comprise a community; this process is repeated until every node is assigned to a corresponding community. And the set C contains the mediate community structure.

After the process of the first stage, the mediate community structure can be obtained. However, in the experiments, we have found that some of the mediate communities are too small, so that every one of them can not be held as a separate community, so the mediate community structure need to be optimized; and the optimization process is carried out in the second stage.

Compared with the first stage, the second stage is simple and intuitive. For each of the mediate communities acquired from the first stage, if the number of nodes in the community is less than or equal to the given threshold θ, then every node t in the community is reassigned to the community which contains most neighbors of the node t. The pseudo-code is listed as algorithm II in Figure 2, and it is almost self-explanatory.

Algorithm II
Input: Mediate community structure C from the first stage;
Output: The final community structure;
1. $count$ =1;
2. **While** $count < \theta$
3. **For** each community $C_i \in C$
4. **If** $|C_i| \leq count$ do
5. **For** each node t in C_i do
6. assign t to the community which has more than $count$ nodes and contains most neighbors of t
7. **End for**
8. **End if**
9. delete C_i from C;
10. **End for**
11. $count$ ++;
12. **End while**
13. Output C;

Figure 2. The Second Stage of NSA

A simple analysis can reveal the complexity of NSA. In the first stage, every node is processed as a father node once, so the time complexity of the algorithm seems to be $O(n)$, where, n is the number of nodes in the network. But in each iteration, we need to select the node v with the largest degree (line 3 in the algorithm I), the cost of this operation self is $O(n)$, so the time complexity of the first stage is $O(kn)$, where, k is the number of communities extracted from the network. To the second stage, it is obviously that its time complexity is $O(n)$. So, the computation complexity of NSA is $O(kn)$. In reality, the number of communities is far less than the number of nodes in the network, i.e., $k << n$, so NSA has near linear time complexity.

4. Experiments

4.1. Thresholds and Modularity

In the first stage of NSA, the algorithm needs to use the similarity threshold τ, and τ works as a parameter. So we need to determine the value of τ. It is obviously, $\tau \in [0,1]$. Different values of τ indicate different community structures, that is, correspond to different values of modularity. We choose the value of τ which indicates the maximal value of modularity as the optimal value of τ.

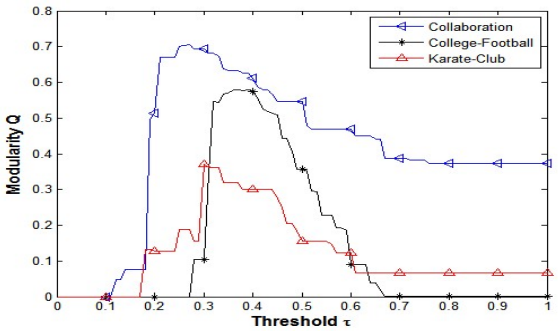

Figure 3. Curves of Threshold τ and Modularity

In the experiments, the value of τ is assigned to be 0 at the beginning, then it is increased by 0.01 at each time until it equals to 1; And we execute the first stage 100 times on each network. The curves in Figure 3 illustrate the relationship between τ and the modularity of community structure for the three networks after running the first stage, respectively. It is clearly that the optimal value of τ for Zachary's Karate Club network is 0.3, for American College Football network is 0.36, and for Santa Fe Institute collaboration network is 0.27, respectively. From our experiments, it seems the optimal value of τ should be in the range of [0.25, 0.38], but this range is only an empirical interval, maybe need to be verified further in the future.

Having determined the value of τ, we need to determine the value of θ used in the second stage of NSA. Just like the method of determining the optimal value of τ in the first stage, we also take the value of θ which results in the maximum of modularity as the optimal value of threshold θ. It is easy to see that the value of θ should be an integer, so we assign 1 to θ at the beginning, and increase it by 1 each time, until the value of modularity reaches to 0. Generally, the value of modularity will increase along with the increase of θ at the beginning, and then it will decrease after it reaches its maxima. And the value of θ at the peak point is what we need, the relationship between θ and modularity is illustrated in Fig.4. We can see clearly that the maxima of modularity in Figure 4 are greater than the counterparts in Figure 3 on all the three networks. So, after the optimization of second stage, the quality of community structure is improved.

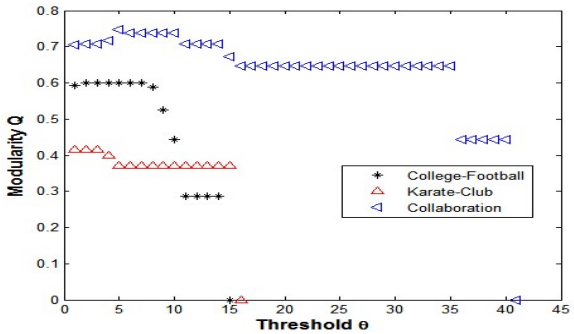

Figure 4. Scatters of Threshold θ and Modularity

4.2. Aanlysis of Experimental Results

We have tested NSA on three real networks; they are Zachary's Karate Club network, American College Football network and Santa Fe Institute collaboration network, respectively. The true community structures of these three networks have been known a priori, and they are illustrated in the Figure 5, Figure 6 and Figure 7, respectively.

The community structure identified by NSA on Zachary's Karate Club network is the same as the true community structure that showed in Figure 5, and the modularity of our community structure is greater than the other algorithms.

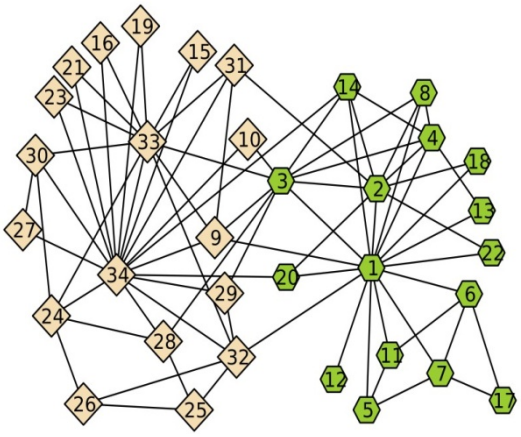

Figure 5. True Community Structure of Zachary's Karate Club Network

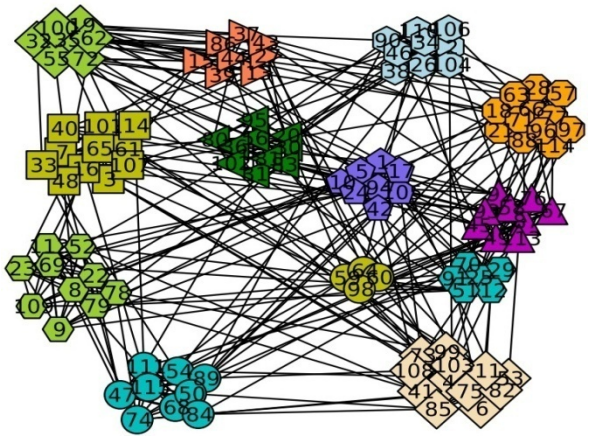

Figure 6. True Community Structure of American College Football Network

To the American College Football network, only 4 nodes ('59', '60', '64', '98', in Figure 6) are misclassified into wrong communities. In Figure 6, these 4 nodes should be the members of the small community which consists of only these 4 nodes, originally; but the links between these 4 nodes in this small community are not more frequent than links out of the small community, so the small community is broken down by other larger communities. After the process of NSA, the node '60' and node '64' are merged into the community at the top right marked as octagon, node '98' was merged into the rhombic community at the bottom right, and node '59' was merged into the circular community at the bottom left.

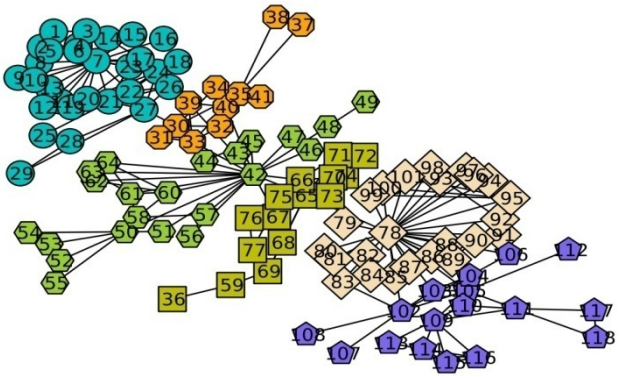

Figure 7. True Community Structure of Santa Fe Institute Collaboration Network

For the case of Santa Fe Institute collaboration network, only one node is assigned into wrong community; this node is marked as '106' in Figure 7, it should be a member of the pentagon community. Since the numbers of edges connected to the node '106' from the rhombic community and from the pentagon community are the same, it is hard to classify the node, and then it is misclassified by NSA into the rhombic community incorrectly.

To validate our proposed algorithm, we have also compared the performance with the classic algorithm FAST*Q* and LPAm. The comparison results are illustrated in the Table I. It is obviously that both the modularity and accuracy of our algorithm is significantly larger than those of other algorithms.

Table 1. Comprisons of Experiment Results for Different Algorithms

	NSA			FASTQ	LPAm
Karate Club	τ =0.3 **Q=0.731 A=0.735**			Q=0.363 A=0.706	Q=0.360 A=0.971
	θ =[1, 3] **Q=0.415** A=0.735		θ =[5, 15] Q=0.372 **A=1**		
	τ =[0.36, 0.37] **Q=0.579** A=0.852		τ =0.41 Q=0.558 **A=0.939**		
College Football	θ =1 Q=0.593 **A=0.870**	θ =[2, 7] **Q=0.600** A=0.852	θ =[1,5] Q=0.581 A=0.965	Q=0.562 A=0.383	Q=0.578 A=0.800
	τ =0.27 **Q=0.706 A=0.754**				
Collaboration	θ =5 **Q=0.747** A=0.941		θ =[6, 10] Q=0.738 **A=0.992**	Q=0.722 A=0.839	Q=0.656 A=0.603

Notes: Q represents modularity of community structure; A represents the accuracy of community structure. θ = [1,3] is equal to θ = {1,2,3}. The values in boldface is the best results

5. Conclusion

In this paper, we proposed a community detection algorithm named NSA based on neighbor similarity. Compared with the FASTQ algorithm, NSA has near linear time complexity; compared with LPAm, NSA is a deterministic algorithm.

In our algorithm, the use of parameter τ is significant. Obviously, the optimal value of τ is dependent on the computation method of similarity measure. Matched with the similarity computation method, we have drawn from the experiments an empirical range, in which the parameter τ should belong in; maybe, further verification and refinement of the range can be done in the future.

References

[1] SH Strogatz. Exploring complex networks. *Nature*. 2001; 268-276.
[2] J Park, MEJ Newman. Statistical mechanics of networks. *Phys. Rev. E.* 2004; 70: 1-13.
[3] SN Dorogovtsev, JFF Mendes. Evolution of networks. *Advances in Physics*. 2002; 51: 1079-1187.
[4] MEJ Newman. The structure and function of complex networks. *SIAM Review.* 2003; 45: 167-256.
[5] Andrea Lancichinetti, Santo Fortunato. Community detection algorithms: A comparative analysis. *Phys. Rev. E*. 2009; 80.
[6] S Fortunato. Community detection in graphs. *Phys. Rep*. 2010; 486(3): 75-174.
[7] RD Alba. A graph-theoretic definition of a sociometric clique. *Mathematical Sociology*. 1973; 3(1): 113-126.
[8] MEJ Newman. Detecting community structure in networks. *Eur. Phys. J. B*. 2004; 38: 321-330.
[9] Mingwei Leng, Jinjin Wang, Pengfei Wang, Xiaoyun Chen. Hierarchical Agglomeration Community Detection Algorithm via Community Similarity Measures. *TELKOMNIKA*. 2012; 10(6): 1510-1518.
[10] Xin Xia, Shu-xin Zhu. A Survey on Weighted Network Measurement and Modeling. *TELKOMNIKA.* 2013; 11(1): 181-186.
[11] MEJ Newman, M Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*. 2004; 69.
[12] MEJ Newman. Fast algorithm for detecting community structure in networks. *Phys. Rev. E.* 2004; 69.
[13] UN Raghavan, R Albert, S Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E.* 2007; 76.
[14] MJ Barber, JW Clark. Detecting network communities by propagating labels under constraints. *Phys. Rev. E.* 2009; 80.
[15] X Liu, T Murata. Advanced modularity-specialized label propagation algorithm for detecting communities in networks. *Physical A*. 2010; 1493-1500.
[16] S Pang, C Chen, T Wei. *A realtime community detection algorithm: Incremental label propagation*. Proceeding of ICFIN conference. Beijing. 2009; 313-317.