Weighted Modularity on *k*-path Graph

YINGHONG MA, WENQIAN WANG
School of Management Science & Engineering
Shandong Normal University, Jinan, China
No.88 Wenhua Road, Jinan, Shandong, 250014
CHINA
yinghongma71@163.com

Abstract: - This Community detection is one of the most interesting problems in the study of social networks. Most of the recent studies focused on how to design algorithms to find the communities without knowing the number of communities in advance. In this paper, we define the k-path graph, and generalize Newman's modularity as weighted modularity. It is also highlight the relationship between eigenvalues and the number of communities of social networks in this paper.

Key-Words: - Social network, k -path graph, modularity, community detection

1 Introduction

The community structure in social networks has been studied for almost one and half century since the "six-degrees of separation" phenomenon was founded [1,2]. Communities in networks usually were defined as a sub-graph in which links are more dense and the rest are comparatively sparse [3,4]. The studies of community detection are potentially useful in real networks because nodes in a community are more likely to have same properties and all these communities may be functional groups. The methods for detecting community are similar to the graph partition problem in graph theory [5,6]. For example, in parallel computing, the pattern of required communications can be represented as a graph or network in which the nodes represent processes and edges are pairs of processes that need to communicate. The problem is to allocate the processes to processors in such a way as roughly to balance the load on each processor, while at the same time minimizing the number of edges that run between processors, so that the amount of inter processor communication is maximized. In general, finding an exact solution to this kind of partition problem is NP-complete, so it is prohibitively difficult to solve it for large graphs, but a wide variety of heuristic algorithms have been developed that give acceptable good solutions in many cases, the best known is perhaps the Kernighan-Lin algorithm which has the complexity $O(n^3)$ on sparse graphs [7]. With the more research efforts, the detection for communities has been extended to many fields such as Internet, biology networks, epidemic theory, social networks, etc.

Many heuristic algorithms on c ommunity detection had been proposed recently. One of a classical partition is by eigenvectors of graph matrix [8]. Newman presented a fast algorithm, which uses maximal modularity Q to determine the communities [9,10]. However the computational complexity of the maximum modularity Q is proved to be NP-complete [11]. They focus on the accuracy of the algorithms without knowing the number of communities in advance which makes the solutions uncertain if we do not know the real number of communities in networks.

In this article, we determine the number of communities based on the eigenvalues of the probability matrix of social networks, and obtain an estimate of this value regardless of binary or not. This paper is formed with the following sections: in Section 1, w e define a k -path graph of a given network, and present a definition to finding k -path matrix. In Section 2, w e explore the relationship between the eigenvalues and modularity, and then calculate the modularity of the social networks with eigenvalues. Finally we give the conclusion and future research problem.

2 k -path weight graphs

We define a matrix consisting of all paths between any two nodes, and then outline an approximation algorithm to determine the number of communities in a social network. We r epresent the agents by nodes in network and the influence between two nodes by a weight on t he link. In the following, a network is denoted by G with n -node set V , and an m -link set E and G is also an undirected

connected graph without loops nor multi-edges. (If G is disconnected, we consider each of connected components. We also view the multi-edges as weights on a single link). The adjacency matrix A of G is a $n\times n$ zero-one matrix denoted by $A=(a_{ij})_{n\times n}$, where $a_{ij}=1$, if there is a link between nodes i and j; $a_{ij}=0$, otherwise. The adjacency matrix of an undirected graph is symmetric. If the network is weighted, we denote the weight of each link by w_{ij} and the weight matrix of G by $W=(w_{ij})_{n\times n}$. Without loss of generality, we write W as the weight matrix regardless G is weighted or not.

For a given positive integer k, denote a path from nodes i to j by a k-path if it is a walk with $k+1$ nodes and without cycle in it. The matrix of k-path graph $S^k=(s_{ij}^k)$ is found as follows:

If $k=0$, $s_{ii}^0=1$ and $s_{ij}=0$ for all $i\neq j$. That is, S^0 is the identity matrix;

If $k=1$, $S^1=W$. That is, S^1 is the weight matrix of G;

For all $k\geq 2$, $s_{ij}{}^k=\frac{1}{k}\sum_s\sum_{l=1}^k w_{i_{l-1}^s i_l^s}$ where s is the number of edge-disjoint k-paths from nodes i to j. The value s_{ij}^k can be viewed as the weight of an edge connecting i and j. Hence, we can define a *k-path weight graph* G_k on G.

Definition of k-path weight graph: For a fixed k, let $w_{ij}(k)=\sum_{l=1}^k s_{ij}^l$, for all l-paths join nodes i and j, where $1\leq l\leq k$. We call $G_k=(V,E_k)$ a *k-path weight graph on G*, where $ij\in E_k$ if there are paths with length no more than k from i to j in G, denote the weight matrix of G_k by $W(k)=(w_{ij}(k))_{n\times n}$. If $k=n-1$, G_{n-1} is a complete graph.

3 Modularity on G_k

The modularity is defined on binary graphs first and it has many generalization (see [3,9,12]). However, the values of all the generalized modularity are obtained by calculating the direct relations of the nodes, that is, they considered the edges weights rather than the indirectly weights. In fact, there is much useful information about the structure of the networks stored in the indirect relations. For example, in many society relationships, such as the economic systems, agents in system influence one to another directly or indirectly: a rush to buy or sell a particular asset can prompt the other to do the same. In most probability, the agents are influenced by their neighbors and the neighbors' neighbors who joint by some relationship. All the buyers and sellers form an inseparable structure, a community, and have very little interactions with the outside. Therefore, we use the modularity to measure the direct and indirect information of path weight graph and to detect the number of communities in the network G.

The modularity Q^* of k-path weight graph is much similar to Newman's, we take the matrix $W(k)$ of k-path weight graph to replace modular matrix in Newman's definition of modularity, because the expected matrix in Newman's is not a fixed matrix in calculating modularity. Here, we define the objective function Q^* which maximize the weight of inter communities of k-path weight graph.

We study k-path weight graph $G_k=(V,E_k)$ with matrix $W(k)$, and $(i,j)\in E_k$, $w_{ij}(k)$ is defined in k-path weight graph in the previous section. Assume that there are $q(q\leq n/2)$ non-overlapping communities in G, denoted by $\mathbb{C}=\{C_1,C_2,\cdots C_q\}$, where $\cup_i C_i=V$.

Case 1. $q=2$

If there are exactly two communities in G, namely V_1 and V_2.

If G is a binary graph and $k=1$, the k-path weight graph G_k is the graph G. So, Q^* is Newman's modularity.

If G is a weighted graph or $k\geq 2$, define the objective function Q^* to be the maximum of the actual weights of inner communities, that is,

$$Q^*=\max_{V_1\cup V_2=V,V_1\cap V_2=\varnothing}\sum w_{ij},\ \text{where}\ (i,j\in V_1)\ \text{or}\ (i,j\in V_2).$$

On the other hand, maximizing the weights of inner communities means to minimize the inter weights of communities since the total actual edges weights of networks $\sum_{(i,j)\in E} w_{ij}$ is a constant. That is why we do not take the expected matrix in Newman's modularity into the objective function.

To calculate Q^*, we define an indictor vector $\mathbf{r}$ on V, $\mathbf{r}=(r_1,r_2,\cdots,r_n)$, where $r_i=1$ if node $i\in V_1$; $r_i=-1$ if $i\in V_2$. Then $r_ir_j=1$ if i,j are in the same group; $r_ir_j=-1$ if they are in different groups. That is, $r_ir_j+1=2$ if i,j are in the same group; $r_ir_j+1=0$ otherwise. Hence, Q^* is

$$\begin{aligned} Q^* &= \frac{1}{2}\max \sum_{(i,j\in V_1)\text{or}(i,j\in V_2)} w_{ij}(r_ir_j+1) \\ &= \max \sum_{(i,j\in V_1)\text{or}(i,j\in V_2)} w_{ij}r_ir_jn \\ &= \max \mathbf{r}^T W(k)\mathbf{r}. \end{aligned} \tag{1}$$

We denote the eigenvector of $W(k)$ corresponding to the eigenvalue β_i by $\mathbf{u_i}$, and $\mathbf{r}=\sum_i b_i\mathbf{u_i}$ as the linear normalization of all the eigenvectors of $W(k)$. Hence $b_i=\mathbf{u_i}^T\mathbf{r}$. Then Eq.(1) is equivalent to

$$Q^* = \max \sum_i b_i^2\beta_i. \tag{2}$$

The optimal value of Q^* in Eq. 2 relies not only on the positive eigenvalues β_i as well as the values of all b_i. Assume the eigenvalues are in decreasing order $\beta_1\geq\beta_2\geq\cdots\geq\beta_n$, then largest eigenvalue β_1 of $W(k)$ is the possible solution such that the Eq. 2 achieves the optimal solution. Let $\mathbf{u_1}=(u_1^{(1)},u_2^{(1)},\cdots,u_n^{(1)})^T$ be the eigenvector corresponding to the largest eigenvalue β_1. Then the indictor vector $\mathbf{r}$ is obtained by $r_i=1$ if $u_i^{(1)}\geq 0$; $r_i=-1$ if $u_i^{(1)}<0$. Hence the value of objective function is $(\sum_{i=1}^n|u_i^{(1)}|)^2\beta_1$, corresponding to the eigenvalue β_1. And the two communities have $|\{i\,|\,u_i^{(1)}\geq 0\}|$ nodes and $n-|\{i\,|\,u_i^{(1)}\geq 0\}|$ nodes respectively.

Case 2. $q>2$

The objective function Q^* for two communities in Case 1 can be naturally generalized to the case of more communities. We assume that there are q groups in the network and define an indictor matrix $\mathbf{R}=(\mathbf{r_1},\cdots,\mathbf{r_q})$ with $\mathbf{r_i}=(r_{i1},r_{i2},\cdots,r_{in})^T$ and $r_{ij}=1$, if node i is in the community j; $r_{ij}=0$, otherwise. Since all the communities are non-overlapping, each pair of columns are mutually orthogonal and total number of nodes is n, hence $Tr(\mathbf{R^TR})=n$. Applying the similar analysis with the equations Eq. 1 and Eq. 2, we have

$$Q^* = \max \sum_{j=1}^{n}\sum_{s=1}^{q}(\mathbf{u_j}^T\mathbf{r_s})^2\beta_j, \tag{3}$$

where $\mathbf{u_j}$ are eigenvectors of matrix $W(k)$ corresponding to eigenvalues β_j ($j=1,2\ldots,q$). Without loss of generality, assume that all the positive eigenvalues are in decreasing order $\beta_1\geq\beta_2\geq\cdots\geq\beta_c$. Clearly, $q\leq c+1$, since the objective is to maximize Q^* in Eq. 3 with the c positive eigenvalues corresponding to c parts from V and the remaining is the $(c+1)$th part. In order to make Q^* as large as possible, we choose the first q largest eigenvalues from all the positive eigenvalues. Then $\sum_{j=1}^{n}\sum_{s=1}^{q}(\mathbf{u_j}^T\mathbf{r_s})^2\beta_j$ is the maximum value in the Eq. 3. However, there are q indictor vectors $\mathbf{r_i}$ ($q>2$), it is not as easy as how we choose the components of indictor vector in Case 1, and may not be able to choose as many ones as in the indictor vectors corresponding to the first q largest eigenvalues and eigenvectors. By the Eq. 2 and Eq. 3, the maximum value of the objective function is closely related to all the positive eigenvalues and the eigenvectors of matrix $W(k)$ of G_k. Because the exact numbers of nodes in each community are unknown, and there doesn't exist a method to choose the indictor vectors in $\mathbf{R}$, we have to rely on the only known information, matrix $W(k)$, to determine the exact number of communities. Therefore, we need other means to estimate the number of communities in G_k.

In the worst case, the computation complexity of the weight modularity is $O(\mathrm{n^2q})$, where n and q are the size of nodes and communities respectively.

4 Conclusion

In this article, we investigate the number of communities in spares social networks. Unlike the other methods focusing on the community detection without knowledge of the number of communities, we study the eigenvalues of the k-path graph matrix,

and reveal the relationship between the eigenvalues and the number of communities in graph.

In this topic, we do not consider the network structure changes with time evolves, the number of communities based on the steady state of the network would bias the real number of communities without information of the evolution of the networks. It will be more interesting to characterize the relationship between eigenvalues and the evolving communities' structure.

Acknowledgement: This work is supported by Natural Science Foundation of China (71471106) and Specialized Research Fund for the Doctoral Program of Higher Education(20133704110003).

References:

[1] S. Milgram, The Small World Problem, Psychol. Today, Vol. 2, pp:60-67, 1967.
[2] J. Guare, Six Degrees of Separation, Vintage, New York, 1990.
[3] M. E. J. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci., Vol. 103(23), pp:8577-8581, 2006.
[4] M. Girvan and M. E. J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. Vol. 99, pp:7821-7826, 2002.
[5] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory ofNP-Completeness, Freeman, San Francisco, 1979.
[6] J. Scott, Social Network Analysis: A Handbook, 2nd Edition, Sage Publications, London, 2000.
[7] B. W. Kernighan and S. Lin, A efficient heuristic procedure for partitioning graphs, Bell System Technical Journal, Vol. 49, pp:291-307, 1970.
[8] A. Pothen, H. Simon and K. P. Liou, Partitioning sparse matrices with eigenvectors of graphs, SIAM J. Matrix Anal. Appl., Vol. 11, pp:430-439, 1990.
[9] M. E. J. Newman. Fast algorithm for detecting community structure in networks, Phys. Rev. E, Vol. 69, 066133, 2004.
[10] F. Radicchi, C. Castellano and F. Cecconi. Defining and identifying communities in networks, Proc. Natl. Acad. Sci., Vol. 101, pp:2658-2663, 2004.
[11] M. Latapy and P. Pons, Computing communities in large networks using random walks, in Proceedings of the 20th International Symposium on Computer and Information Sciences, Lecture Notes in Computer Science, Vol. 3733, pp:284-293, 2005.
[12] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. PHYSICAL REVIEW E.Vol.74, 036104,pp:1-19(2006).