# Community Detection in Social Networks Based on Influential Nodes

Chang Su
Chongqing University of Posts and Telecommunications, Chongqing, China
Email: changsu@cqupt.edu.cn

Yukun Wang
Chongqing University of Posts and Telecommunications, Chongqing, China
Email: airfer@126.com

Lan Zhang
Capital University of Economics and Business, Beijing, China
Email: lan@cueb.edu.cn

***Abstract*—Large-scale social networks emerged rapidly in recent years. Social networks have become complex networks. The structure of social networks is an important research area and has attracted much scientific interest. Community is an important structure in social networks. In this paper, we propose a community detection algorithm based on influential nodes. First, we introduce how to find influential nodes based on random walk. Then we combine the algorithm with order statistics theory to find community structure. We apply our algorithm in three classical data sets and compare to other algorithms. Our community detection algorithm is proved to be effective in the experiments. Our algorithm also has applications in data mining and recommendations.**

***Index Terms*—Influential nodes, Social network, Community structure, Random walk, Order statistics**

## I. Introduction

There are varieties of social networks, such as Facebook, LinkedIn, citation networks, Amazon, the circle of friends in QQ, etc. With the development of the smart phones, more and more people log into their social networks through their smart phones and share text and multimedia information with their friends online. Large-scale social networks have become the complex networks, like www networks [1], metabolic networks [2-4], epidemiology networks [5,6], ecological webs [7], etc.

Social networks are usually modeled as graphs. A social network consists of a set of nodes along with edges connecting the nodes. The nodes represent the objects in the social networks, such as people, papers, commodities, etc. The edges represent the relationships between objects. Taking the circle of friends as an example, the nodes represent an individual, and the edges between the nodes represent the friendship between individuals.

Community is an important structure in social networks. Although there is no accepted definition of a community, it is generally accepted that a community is a set of nodes with higher edge density than the null model. In recent years, how to effectively find influential nodes (key-nodes that have great influence in the network), and how to use the influential nodes for community detection has becoming a hot topic in social network analysis and the field of data mining.

In social network analysis, finding the influential nodes has a wide range of applications. E.g., if we can dig out the most influential customers in marketing, then through the community constructed by the influential nodes, a product brand can be rapidly promoted. This procedure is similar with virus spreading. Moreover, take shopping on Amazon as an example. Assume we want to purchase the book, 'Advanced programming in UNIX environment'. As can be seen the topology constructed by Amazon in Fig.1, the book 'Advanced programming in Unix environment' can be looked as an influential node and plays a very important role in recommendation. The community structure formed by influential nodes can be used for personalized recommendations in some online stores.

In this paper, we use random walk as the basic theory to discover influential nodes, and apply influential nodes for community detection in social networks. The modularity is used as the measure standard to evaluate the accuracy and efficiency of the algorithm. This idea can be used to improve the performance in some traditional community detection algorithms such as K-means [8], or Fuzzy c-means [9,10], which needs to know the number of clusters in advance.

## II. Related Work

Community detection is important in the field of social network analysis. In 2002, Girvan and Newman [11] published an article which explored the community

Manuscript received January 20, 2014; 2011; revised March 12, 2014; accepted March 18, 2014.

doi:10.4304/jsw.9.9.2409-2416

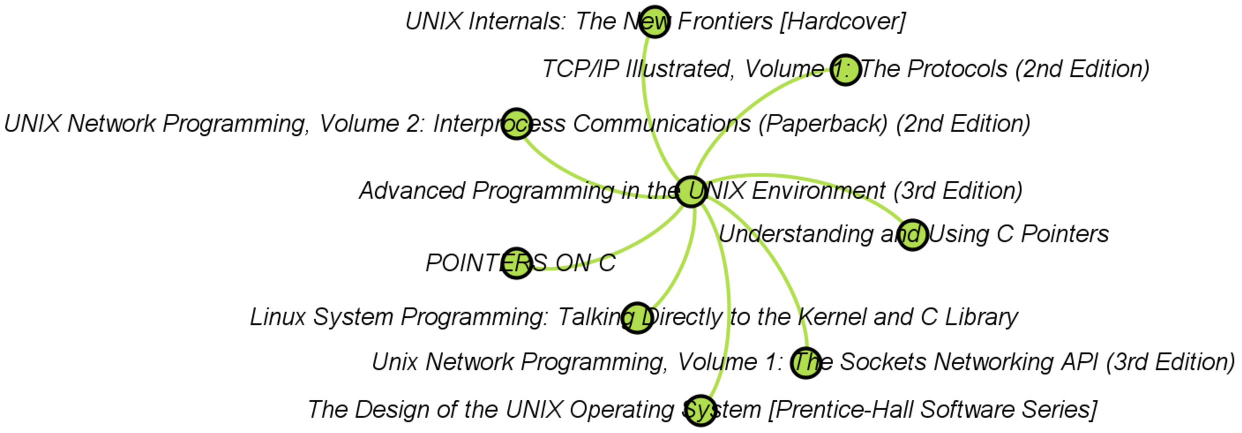

UNIX Internals: The New Frontiers [Hardcover]
TCP/IP Illustrated, Volume 1: The Protocols (2nd Edition)
UNIX Network Programming, Volume 2: Interprocess Communications (Paperback) (2nd Edition)
Advanced Programming in the UNIX Environment (3rd Edition)
Understanding and Using C Pointers
POINTERS ON C
Linux System Programming: Talking Directly to the Kernel and C Library
Unix Network Programming, Volume 1: The Sockets Networking API (3rd Edition)
The Design of the UNIX Operating System [Prentice-Hall Software Series]

Figure1. Books topology in Amazon

structure of social networks and biological networks. This paper started a period of rapid development in finding communities. In 2004, Newman and Girvan [12] proposed the concept of modularity, and introduced the null model into the analysis of the community structure. They also proposed the concept of betweenness, from different aspects, such as the perspective of shortest path, random walk, current. They adopted a divisive hierarchical clustering algorithm for community discovery. In the following years, many community detection algorithms were proposed based on modularity, such as the greedy algorithm [13,14], the simulated annealing algorithm [15] and the ultimate optimization algorithm [16]. These algorithms all use the null model as the reference.

In 2007 Fortunato and Barthelemy published an article in PANS [17], pointing out that the algorithms based on modularity have resolution limits, which make the accuracy of community detection by these algorithms suspect. The modularity and null model proposed by Newman and Girvan needs to be further improved, but it emphasized the idea of using the null model as the reference when compared with real local networks.

Using random walk to find community structure was adopted in early work such as [18-23]. For example, in 2005, Pascal Pons and Matthieu Latapy [23] designed an algorithm based on random walk for community detection. As for the concept of order statistic, it was proposed by Radicchi F, Lancichinetti A, Ramasco [24, 25]. We combine the order statistic theory and the influential nodes to find communities structure in social networks. The algorithm can be extended to detect overlapping community.

In recent years, many new algorithms have been proposed for community detection, such as, in 2012, a cooperative and heuristic community detecting algorithm proposed by Ruixin Ma [26]. In the same year, Shuzhi Li [27] proposed a novel community structure partition algorithm based on multi-gene family. Apart from that, much attention was focused on the analysis of the community structure. For example, Gyeong-Mi Park [28] gave a full description about the structure analysis of social network from the perspective of characters in literature texts.

This paper is consisted of 5 sections. In the first section, we talk about the history of community detection in social networks. In the second section, we explain how to preprocess using random walk. In the third section, how to find influential nodes and form initial division in social networks are introduced. In the fourth part, we talk about how to divide nodes into communities using influential nodes combined with the order statistic model. In the fifth part, we put the algorithm into real network and check out the algorithm efficiency.

## III. Our Algorithm Based on Influential Nodes for Community Detection

### A. Preliminaries for Find Influential Nodes

Many well known classical community detection algorithms use adjacent matrix to represent social networks. In order to obtain more obvious community structure, the original adjacency matrix is preprocessed. During the preprocess procedure, the degree of nodes is modified and weights are assigned to the edges. The new degree and new edges weight are named as $P\text{-}degree$ and $P\text{-}weight$ respectively. Random walk [18,19] can explore the underlying network structure and this concept has already been employed to reveal communities structure in social networks . Our community detection algorithm is also based on the random walk using the original adjacency matrix $A$. A transition matrix $P$ is calculated based on the adjacent matrix $A$ . The elements of the matrix $P$ are $P_{ij} = A_{ij} / k_i$ , where $k_i$ is the degree of the node $i$ . A random walker [18, 19, 23] spends more time in regions of the graph that have higher probability density than in sparser regions.

The theory talked above can be explained by common neighbors theory .We can make a conclusion from the fact that, if two nodes have more common neighbors than others, the two nodes have greater chances to be in the same community. If we put this theory into matrix, the

```
Algorithm : find seed nodes in nodes sets
Input : All nodes in graph
Output : Seed nodes
Sorting all nodes in decreasing order
For all nodes in sorting list
        If ((degree(r) > degree(t)) && 
                        (node r is not in any sub-region))
            Then ( put node r and the nodes
                        connected to r to form a group ) else
            If ( degree(r) > degree(t) &&
                        ( node r already in a sub-region ))
                    Then ( skip node r )
                    r++ else
                If (degree(r) < degree(t))
                    puts(the procedure finish)
                end
            end
        end
end
```

Figure2. Pseudo-source code

corresponding row vectors of the two nodes in matrix P will have much in common. If we use the similarity to represent this, then the similarity of the two vectors will be high. This is the point we emphasized here.

In other words, if a random walker starts walk from a node in a community, after some steps all the other nodes in the community will have similar behavior. In this article we define the similar behavior as the trajectory of the random walk. If two nodes have similar behavior, the trajectories of the two nodes have some overlapping portions. This is the core reason that $P\text{-}degree$ and $P\text{-}weight$ based on random walk can make the community structure more obvious.

Therefore, if two nodes have similar behavior, then with high probability the two nodes are considered to be in the same community. Otherwise, the two nodes belong to different communities. For the behavior of the nodes, the tests have been done and explained in the experimental section. The reason we use behavior of the nodes is to make weight difference between the edges in community and the outside the community more obvious. The algorithm based on random walk adds more weight to inter-edges (the edges between the nodes in the community) and reduce the weight of intra-edges (the edges connecting nodes in different communities). The new values of weight and the degree of nodes are named as $P\text{-}weight$ , $P\text{-}degree$ . After doing these, the degree of the nodes inside community will become higher and the degree of the nodes outside the community will become lower.

As mentioned before, $P$ is the transition matrix corresponding to the real social network, and is also the matrix obtained by the random walk after one step. When the walker walked $t$ steps, the transition matrix can be represented as $P^t$ [24]. The element $P_{ij}^t$ represents the probability that a random walker walk from node $i$ to the node $j$ at the $t^{th}$ step. The behavior of the nodes can be taken as the row vector of the matrix $P_{ij}^t$. According to the size of the social network, we can put different weight to $P_{ij}^t$.

The matrix PF is used to represent the final matrix which will be used for later in the processing.

$$PF = P*\theta_1 + P^2*\theta_2 + P^3*\theta_3 \ldots\ldots + P^t*\theta_t \qquad (1)$$

In Eq.(1), $\theta_1,\theta_2,\theta_2\ldots\ldots\theta_t$, $0 \le \theta_i \le 1$ , $1 \le i \le t$ are positive integers. These parameters represent the weight added to the transition matrix $P^t$ .

From many times experimental testing, it is concluded that if $t$ is between three and five we can get the best result. If $t$ is bigger, then it will be very close to stationary probability. If $t$ is smaller, the probability may not have enough time to distribute to the whole graph. In default case, $t$ is set as 3, and $\theta_1,\theta_2,\theta_2\ldots\ldots\theta_t$ are all set as 1. You can adjust the values according to the different situations. When the transition matrix is $PF$ obtained, we use $cos\text{-}similarity$ [29] to present the weight of the edges.

$$cos\text{-}similarity(v_i,v_j)=(v_i,v_j)/(\sqrt{(v_i,v_i)}*\sqrt{(v_j,v_j)}) \qquad (2)$$

In the Eq.(2), $v_i$ and $v_j$ represent the $i^{th}$ row vector and the $j^{th}$ row vector of the matrix $PF$ respectively. After $\cos\text{–}similarity$ is calculated, we can get $P\text{-}weight(i,j)$.

$$(P\text{-}weight(i,j)) = w*(cos\text{-}similarity) \qquad (3)$$

In the Eq.(3) above, $w$ is another weight parameter, because the value of $cos\text{-}similarity$ is in the range of 0 and 1. $P\text{-}degree(i)$ is calculated as the following.

$$(P\text{-}degree(i)) = PF(i,i) \qquad (4)$$

In the Eq.(4) above, $PF(i,i)$ represents the probability that a random walker starts from node $i$ and then end in node $i$ within $t$ steps.

This is the procedure of preprocessing which assign new weights to the edges and new degrees to the nodes. Through this preprocessing procedure we obtain a new network having more obvious weight differentiation between nodes inside the community and the nodes outside the community. This prepared well for the following work.

### *B. Find Influential Nodes and Initialize the Division of the Network*

After completing the preprocess procedure, new degree and edge weight are obtained. In order to find communities, the social network should be divided into different sub-regions first. The influential nodes can be considered as the most important node in the sub-region, and they carry most energy of the network. Therefore, in order to get the sub-regions, the first thing should be done is to find the influential nodes. Then we can obtain the initial divisions from the sub-graphs of the influential nodes.

The procedure used to find influential nodes can be concluded as following: firstly, we sort all the nodes in decreasing order in the social network based on $P\text{-}degree$ of the nodes. If $a_i$ represents node $i$, the result after sorting can be showed as following:

$$a_i \geq a_j \geq a_k \ldots\ldots \geq a_n \tag{5}$$

From Eq.(5), shows that the $P\text{-}degree$ of node $i$ is larger than the $P\text{-}degree$ of the node $j$, and the $P\text{-}degree$ of the node $j$ is larger than the $P\text{-}degree$ of the node $k$, etc.

Secondly, find the threshold of the influential nodes among all the nodes in the social network. If the social network has $n$ nodes, take the $P\text{-}degree$ of the $(n/4)^{th}$ nodes in the sorting list as the threshold. The $(n/4)^{th}$ node is named as $a_t$, $0 < t < n$. Supposing we have got the node $a_t$, we set $P\text{-}degree(t)$ as the $P\text{-}degree$ of the node $a_t$.

Thirdly, check every node in sorting list which is sorted according to the decreasing value of $P\text{-}degree$. If the $P\text{-}degree$ of a node $r$ is larger than $P\text{-}degree(t)$ and it does not belong to any sub-regions at the same time, then the node $r$ is an influential node. Put the node $r$ and the nodes connected to node $r$ into a group which is a sub-region. If the $P\text{-}degree$ of a node $r$ is larger than $P\text{-}degree(t)$, but node $r$ has belong to some sub-regions, then skip the node $r$ and continue to find the next node in the sorting list which is not belong to any sub-regions. When we find no nodes whose $P\text{-}degree$ is larger than $P\text{-}degree(t)$ and the node does not belong to any sub-regions, the procedure is finished.

In order to make it easier to be understood, the pseudo code is represented in Fig.2. It may be helpful for you to understand the whole procedure. Finally, after finishing the above steps, the nodes in social network are divided into two parts. The first part is the nodes that belong to some sub-regions, and the other includes the nodes do not belong to any sub-regions. Take these nodes as free nodes. As for free nodes, we check whether a free node has connection to the sub-region we found. If a free node has a connection to a sub-region, then we put the free node to the sub-region. If a free node has connections to several sub-regions, put the node to these sub-regions at the same time. In the following chapter how to make a free node belongs to only one sub-region will be explained.

When the initial division of the social network is completed, there is one more step to go. Check whether two different sub-regions can be merged into a larger sub-region. A measure is needed for testing. The measure we use is $similarity$ [29], supposing we have two influential nodes $a_{s1}$ and $a_{s2}$, then the similarity between the two nodes can be calculated as following:

$$Similar(v_{s1},v_{s2})=(v_{s1},v_{s2})/(\sqrt{(v_{s1},v_{s1})}*\sqrt{(v_{s2},v_{s2})}) \tag{6}$$

In the Eq.(6) $v_{s1},v_{s2}$ are the row vectors of the influential nodes $a_{s1}$ and $a_{s2}$ respectively. If the value we get is larger than $P_{th}$, then consider that the two sub-regions that the influential nodes lying in can be merged. Otherwise the two communities can not be merged together.

### C. Order Statistic Model using $P\text{-}degree$

Now we have obtained some sub-regions and a node may be belongs to several sub-regions. To make certain which sub-region that the free nodes should belong to is what should be done. Here we use $P\text{-}weight(i,j)$.

Suppose there is a node $i$ that belongs to several sub-regions at the same time. These sub-regions are named as $C_1, C_2, C_2 \ldots\ldots C_t$, and these sub-regions have $N_1, N_2, N_2 \ldots\ldots N_t$ nodes respectively. Using $\delta$ to represent the sum of the $P\text{-}weight(i,j)$, as for sub-region $C_t$, node $i$ in sub-region $C_t$ has the value as following:

$$\delta_t = \sum_{j=1}^{N_t} (p\text{-}weight(i,j)) \tag{7}$$

For different sub-regions that node $i$ belongs to, calculate the value $\delta$, based on the following formula:

$$\delta = max(\delta_1, \delta_2, \delta_3 \ldots\ldots \delta_t) \tag{8}$$

In the Eq.(8), e.g. if $\delta_k$ is the biggest, then node $i$ belongs to $C_k$.

According to the order statistic model, the sub-regions discovered by our algorithm are considered as communities. But sometimes the communities we found have large difference to the real ones. The number of nodes of the largest community is much bigger than some small ones. Therefore it is needed to analysis these small communities and check whether small communities can further merged into a bigger community. Now we use the Eq.(9) to define the concept of very small communities.

$$min\_length = aver\_length(a_{s1}, a_{s2} \ldots\ldots a_{st})/4 \tag{9}$$

If the number of nodes in a community is less than $min\_length$, then the community need to be merged into a bigger community, $aver\_length()$, represents the average number of nodes in the communities.

We use the Eq. (10) to define the rulers of mergence

$$a_{sk} = max\_link(link(a_{s1},a_{sk}),link(a_{s2},a_{sk})\cdots\cdots link(a_{st},a_{sk})) \tag{10}$$

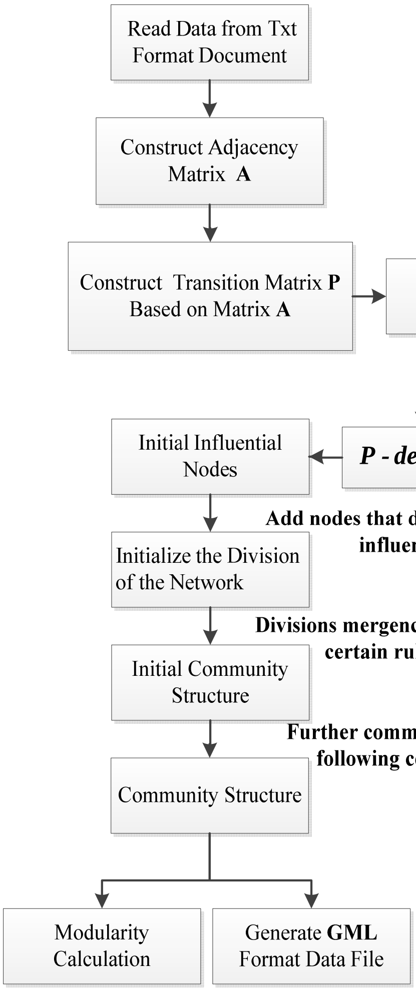

Read Data from Txt Format Document
Construct Adjacency Matrix **A**
Construct Transition Matrix **P** Based on Matrix **A**
Construct $P^t$
Random walk theory
*P - degree(i)*
Initial Influential Nodes
Add nodes that directly connecting to influential nodes
Initialize the Division of the Network
Divisions mergence following certain rulers
Initial Community Structure
Further community mergence following certain rulers
Community Structure
Modularity Calculation
Generate **GML** Format Data File

Figure.3 Software design flowchart

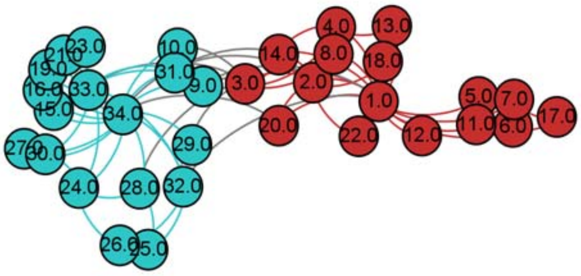


Figure.4 Karate club network topology

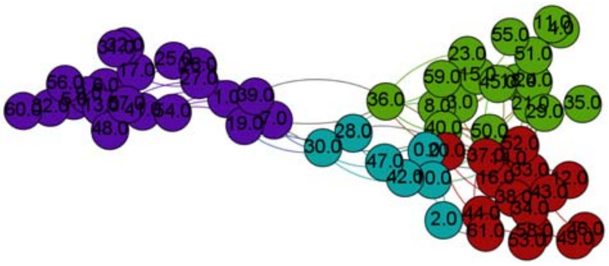


Figure.5 Dolphins network topology

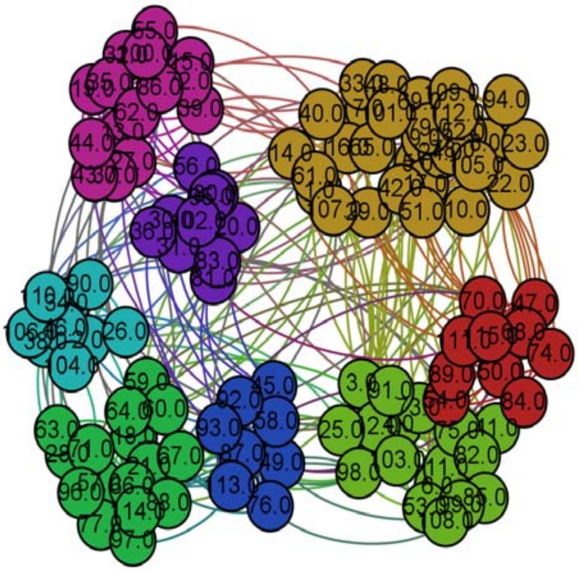


Figure.6 American football network topology

In Eq.(10), $link(a_{st}, a_{sk})$ represents the number of links between $a_{st}, a_{sk}$. If a community $a_{st}$ has the largest number of links to community $a_{sk}$, then we merged $t^{th}$ community to $s^{th}$ community.

```
Algorithm : Calculate Modularity
Input : Adjacent Matrix A
Output : Modularity
Degree(i) represents the degree of the node i
For( j = 1; j <= n; j++)
    Degree(i) += A(i, j);
Sum_degree represents the whole degree of all nodes.
For (i = 1; i <= n; i < n)
    Sum_degree += degree(i);
Calculate modularity
i = 1;
While(i <= n){
    j = 1;
    While( j <= n){
        If(i, j in same cluster)
            Modularity_sum +=
        (A[i][j] - degree(i)(j)) / sum_degree;
    }
}
Modularity = 2* modularity / modulairy_sum;
Cout << " Modularity = " << Modularity << endl;
```

Figure.7 Pseudo-source code for modularity

```
Alogrithm : Generate GML file
Input: Clusters of Nodes
            vector < vector < node > > Clusters
Output : GML format file
Bind output stream outfile to "name" .gml
Outfile << "graph [ ";
Iter_begin = Clusters.begin();
   While(Iter_begin! = Clusters.end())
     Iter_sbegin = Iter_begin- > begin();
     While(Iter_sbegin! = Iter_begin- > end()){
        Outfile << "node[ id int("id")
                       Community int("com") ] ";
        ++Iter_sbegin;
     }
     + + Iter_begin;
   }
Bind input stream in_file to "name" .txt;
In_file.open("name".txt);
While(getline(infile,line)){
   Istringstream stream(line);
   While(stream >> node1 >> node2)
     Outfile << "edge [ source" << node1
                       "target " << node2 << " ] ";
}
Outfile << " ]";
```

Figure.8 Pseudo-source code for GML format file

## IV. Experiments

In this part, we test the effectiveness of the algorithm. The experimental environment is as following: CPU Core: i7 2.4GHZ, Memory: 8G, Operating system: CentOS.

The design flowchart of the software can be seen in Fig.3. The Raw data is stored in the following format: $source \mid target \mid weight$. If the graph is undirected, the value of the weight is always 1. Our software is implemented by C++ programming language, we use $matrix < double > m(index,index)$ that defined in the ublas library which is a part of C++ boost library, *index* is the dimension of the matrix. If *m2* can be defined as $matrix < double > m2(index,index)$, then *prod()* function can be used for matrix multiplication like $m2 = prod(m,m)$. Use function *stable_sort()* to sort nodes according to the value of *P - degree*.

The information of nodes can be stored in data structure *mulitset*, we can use function *set_union()* for set union and function *set_intersection()* for set intersection. At last we calculate the modularity of the community structure we find, and store the result in a GML file. The format of the GML file in a way like *Graph [ id "t" label "lable" community "c" ]* *....edge[source i target j ]....edge[source i target j ] ]*, We use Gephi, an interactive visualization and exploration platform, for further analysis. Pseudo-source code for modularity can be seen in Fig.7, Fig.8 shows pseudo-source code for generating GML format file.

We take three datasets for our experiments, the Zachary karate network [30], the dolphins community network [31,32], the US college football network [11], respectively. About the detail of three networks can be seen in Table I. In Table I, APL represents average path length, ACD, represents average clustering coefficient. We can conclude from the table that with the increasing number of nodes, the average path length remains small, which also verified the Milgram's small world theory [33]. The average clustering coefficient describe the clustering properties of the network, so the larger the average path length, the better of the community detection. Now compare our algorithm with Newman algorithm [12], GN algorithm [11], CNM algorithm [5] and BGLL algorithm [14]. All the algorithms will be applied respectively to these datasets, and to test the effectiveness of our algorithm.

### A. Zachary Karate Club

The Zachary karate club network may be one of the most famous networks in terms of community structure. The club had got the infighting between the administrator and the instructor. As a result, the instructor had left the club and taken about half of the original members with him which provides actual evidence of community structure. Fig.4 shows the community structure detected by our algorithm. The result can be seen in Table II and Fig.10.

### B. Dolphins Network

Lusseau [31] spent seven years of field studies of the dolphins, with ties between dolphin pairs being established by observation of statistically significant frequent association. The network splits naturally into two large groups. Fig.5 shows the community structure detected by our algorithm. The result can be seen in table II and Fig.10.

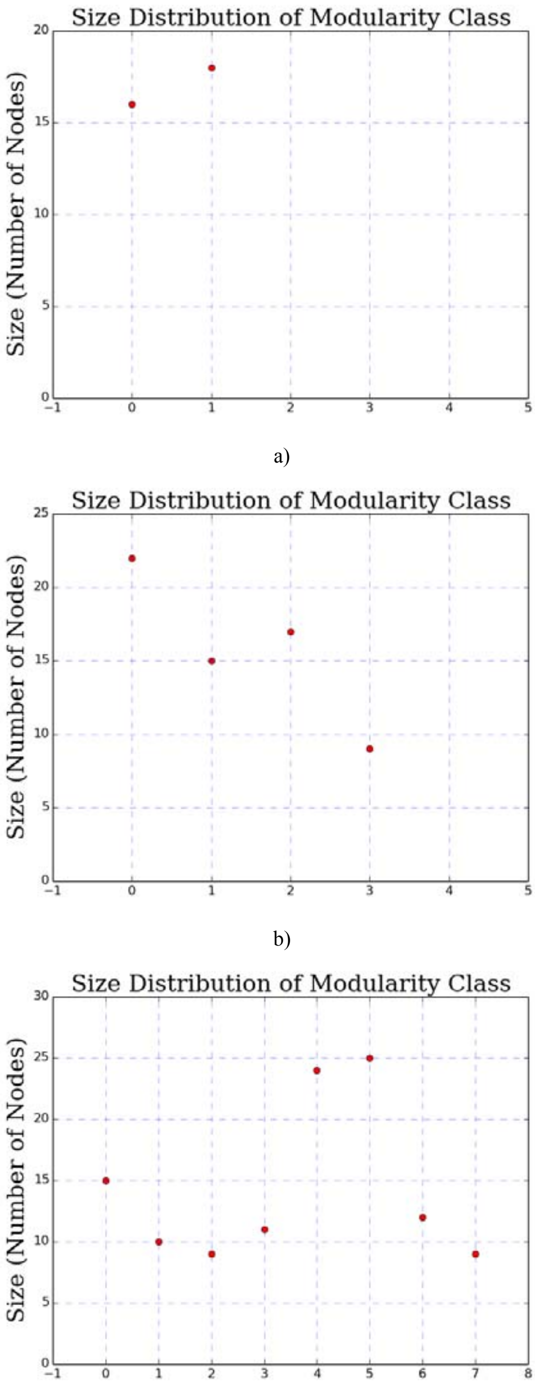
### C. US College Football

The US college football network is a representation of the schedule of Division IA games during regular season Fall 2000. Vertices in the graph represent teams and edges represent regular season games between the two teams they connect. The teams have been divided into 12 conferences, and this provides the actual community structure. Fig.6 shows the community structure detected by our algorithm. The result can be seen in table II and Fig.10.

As we can see the actual number of the communities about dolphins and football network are different from real ones, but it's not mean our algorithm is useless. On the contrary, our algorithm focuses on the influential

TALBEL I
PARAMETERS OF THREE NETWORKS

| Network | Nodes | Edges | APL | ACD |
|---|---|---|---|---|
| Karate | 34 | 78 | 2.408 | 0.588 |
| Dolphins | 62 | 159 | 3.3569 | 0.303 |
| Football | 115 | 616 | 2.508 | 0.403 |



a)

b)

c)

Figure.9: Size-distribution vs Modularity class graph of networks: a) Karate club b) Dolphins c) American football

TABLE II
THE RESULTS OF OUR EXPERIMENTS

| Datasets | Algorithm | Cluster Numbers | Modularity |
|---|---|---|---|
| Karate | Newman | 2 | 0.381 |
|  | Our algorithm | 2 | 0.421 |
|  | GN | 5 | 0.401 |
|  | CNM | 2 | 0.381 |
|  | BGLL | 4 | 0.418 |
| Dolphins | Newman | 2 | 0.465 |
|  | Our algorithm | 4 | 0.520 |
|  | GN | 5 | 0.519 |
|  | CNM | 4 | 0.495 |
|  | BGLL | 4 | 0.526 |
| Football | Newman | 6 | 0.524 |
|  | Our algorithm | 8 | 0.565 |
|  | GN | 11 | 0.599 |
|  | CNM | 10 | 0.577 |
|  | BGLL | 7 | 0.604 |

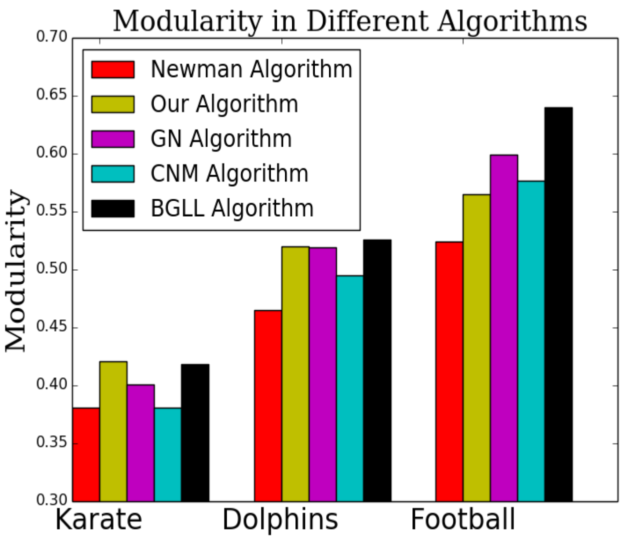


Figure.10 Modularity in different algorithms

nodes, and can quickly find the most influential nodes which plays an important role in the network, and also give a better description of the clustering effects.

It can be concluded from table II that our algorithm has good performance in modularity compared with other classical algorithms. Now we can obtain the size distribution of the nodes with the modularity as Fig. 9.

## V. CONCLUSION

This paper is focuses on how to use random walk to find influential node, and then combine the influential nodes with order statistic theory to find community structure. We apply our algorithm and other classical algorithms to three datasets, karate club network, dolphins network and American football network respectively. We found that our algorithm have a good performance in modularity compared with other classical algorithms.

These experiments proved our algorithm is effective in community detection. In later research, we will extend our algorithm to the detection for fuzzy community detection.

## ACKNOWLEDGMENT

We would like to show our deepest gratitude to Professor John E. Hopcroft, who has provided us with valuable guidance. This work was supported by the National Nature Science Foundation of China under contract number 61271259, the Chongqing Nature Science Foundation under contract number CTSC2011jjA40006, the Research Project of Chongqing Education Commission under contract number KJ120501.

## REFERENCES

[1] Faloutsos M, Faloutsos P, Faloutsos C. On power-law relationships of the internet topology[C]//ACM SIGCOMM Computer Communication Review. ACM, 1999, 29(4): 251-262.

[2] Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N.& Barabási, A.-L. (2000) *Nature* 407**,** 651–654.

[3] Wagner A, Fell D A. The small world inside large metabolic networks[J]. Proceedings of the Royal Society of London. Series B: Biological Sciences, 2001, 268(1478): 1803-1810.

[4] Ravasz E, Somera A L, Mongru D A, et al. Hierarchical organization of modularity in metabolic networks[J]. science, 2002, 297(5586): 1551-1555.

[5] Moore, C. & Newman, M. E. J. (2000) Phys. Rev. E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top. 61**,** 5678–5682.

[6] Pastor-Satorras R, Vespignani A. Epidemic spreading in scale-free networks[J]. Physical review letters, 2001, 86(14): 3200.

[7] Williams, R. J. & Martinez, N. D. (2000) Nature (London) 404, 180–183.

[8] J. MacQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of the 5th Berkeley symposium on mathematical statistics and probability, volume 1, pages 281–297. California, USA, 1967.

[9] Bezdek J C. Pattern recognition with fuzzy objective function algorithms[M]. Kluwer Academic Publishers, 1981.

[10] Dunn J C. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters[J]. 1973.

[11] Girvan M, Newman M E J. Community structure in social and biological networks[J]. Proceedings of the National Academy of Sciences, 2002, 99(12): 7821-7826.

[12] Newman M E J, Girvan M. Finding and evaluating community structure in networks[J]. Physical review E, 2004, 69(2): 026113.

[13] Newman M E J. Fast algorithm for detecting community structure in networks[J]. Physical review E, 2004, 69(6): 066133.

[14] Blondel V D, Guillaume J L, Lambiotte R, et al. Fast unfolding of communities in large networks[J]. Journal of Statistical Mechanics: Theory and Experiment, 2008, 2008(10): P10008.

[15] Guimera R, Amaral L A N. Functional cartography of complex metabolic networks[J]. Nature, 2005, 433(7028): 895-900.

[16] Duch J, Arenas A. Community detection in complex networks using extremal optimization[J]. Physical review E, 2005, 72(2): 027104.

[17] Fortunato S, Barthelemy M. Resolution limit in community detection[J]. Proceedings of the National Academy of Sciences, 2007, 104(1): 36-41.

[18] Lambiotte R, Delvenne J C, Barahona M. Laplacian dynamics and multiscale modular structure in networks[J]. arXiv preprint arXiv:0812.1770, 2008.

[19] Evans T S, Lambiotte R. Line graphs, link partitions, and overlapping communities[J]. Physical Review E, 2009, 80(1): 016105.

[20] Zhou H. Network landscape from a Brownian particle's perspective[J]. Physical Review E, 2003, 67(4): 041908..

[21] Zhou H. Distance, dissimilarity index, and network community structure[J]. Physical review e, 2003, 67(6): 061901.

[22] D. Lai, H. Lu, and C. Nardini, Physica A: Statistical Mechanics and its Applications 389, 2443 (2010).

[23] Pons P, Latapy M. Computing communities in large networks using random walks[M]//Computer and Information Sciences-ISCIS 2005. Springer Berlin Heidelberg, 2005: 284-293.

[24] Lancichinetti A, Radicchi F, Ramasco J J. Statistical significance of communities in networks[J]. Physical Review E, 2010, 81(4): 046110.

[25] Radicchi F, Lancichinetti A, Ramasco J J. Combinatorial approach to modularity[J]. Physical Review E, 2010, 82(2): 026102.

[26] Ma R, Deng G, Wang X. A Cooperative and Heuristic Community Detecting Algorithm[J]. Journal of Computers, 2012, 7(1).

[27] Li S, Wang X. Complex Networks Community Structure Division Algorithm Based on Multi-gene Families Encoding[J]. Journal of Computers, 2013, 8(12).

[28] Park G M, Kim S H, Cho H G. Structural Analysis on Social Network Constructed from Characters in Literature Texts[J]. Journal of Computers, 2013, 8(9).

[29] Zachary W W. An information flow model for conflict and fission in small groups[J]. Journal of anthropological research, 1977: 452-473.

[30] L. Donetti and M. A. Munoz, Journal of Statistical Mechanics: Theory and Experiment, P10012(2004).

[31] Lusseau D. The emergent properties of a dolphin social network[J]. Proceedings of the Royal Society of London. Series B: Biological Sciences, 2003, 270(Suppl 2): S186-S188.

[32] Lusseau D, Schneider K, Boisseau O J, et al. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations[J]. Behavioral Ecology and Sociobiology, 2003, 54(4): 396-405.

[33] Milgram S. The small world problem[J]. Psychology today, 1967, 2(1): 60-67.