International Journal of Smart Home
Vol. 10, No. 9 (2016), pp. 83-92
http://dx.doi.org/10.14257/ijsh.2016.10.9.08

The Community Discovery Algorithm Based on Label Cohesion

Gui Qiong [1, 2], Zhu Dejun [2] and Cheng Xiaohui[*2]

[1] *School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China*
[2] *College of Information Science and Engineering, Guilin University of Technology, Guilin 541004, China*
guilucky@163.com

Abstract

Label Propagation Algorithm is a kind of community discovery method. This algorithm contains large numbers of random selections, which made the result uncertain and reduced the stability of the algorithm. In order to solve these problems, this paper proposed Label Cohesion Algorithm (LCA). In LCA algorithm the label propagation process is divided into two steps. The first step is taking pretreatments on the original labels. The second step is label updating. In the first step we change node label though node centripetal. In the second step the paper use label Cohesion as the judgement to choose the new label. Finally the experimental result shows that the accuracy of the algorithm has been improved.

Keywords*: Community discovery; Label Cohesion; Node centripetal; Node attraction.*

1. Introduction

Community division method can help us resolve complex network structure and reduce the difficulty of looking for effective information in the network. The concept of community discovery is first proposed by Girvan and Newman in 2002. They put forward GN algorithm which based on edge threshold method [1]. After then, in 2004, Newman proposed the concept of modularity Q. The calculation formula of Q is

$$Q = \sum_{i=1}^{k}\left(\frac{l_i}{|E|} - \left(\frac{d_i}{2|E|}\right)^2\right) \quad (1)$$

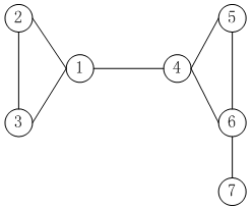
In this formula l_i means the number of edges in i community, d_i means the addition of all node degrees in the same community. E is all edges in the original community. Newman put forward FN [2] algorithm. This algorithm divide community by in seeking the maximum Q in each step. Then Bu and his teammates put forward a community discovery method based on modular fast parallel optimization algorithm (FPMQA) [3]. GN algorithm, FN algorithm, and some of their improvement algorithm have the same problem. The time complexity of these method is high. These algorithms are not suitable for processing large networks. To solve these problems, Raghavan [4] gives out the idea of label propagation [5]. As it is the first time use the label propagation method in community discovery. So most researchers often call this algorithm directly the Label Propagation Algorithm. The time complexity of label propagation algorithm is close to the linear time complexity.

The thought of label propagation algorithm is: Assign to each node a unique label, then through label update to make division of the communities. Rules for node label update is randomly select a not visited node. Then select an adjacent label as the new label of the

[*] Corresponding Author

ISSN: 1975-4094 IJSH

node, the adjacent label we choose should appeared most frequently on the neighbor nodes. If there are several labels appear same time choose one of them randomly. After all the node label do not change any more, having the same tag nodes in the same community. Label propagation algorithm solves the high time complexity problem which exist in community division algorithm before label propagation algorithm. But the label propagation algorithm brings some new problems too. It select nodes randomly may lead to label flow phenomenon. Sometimes the edge node label affect core node label. Like Figure 1 and Figure 2 shows



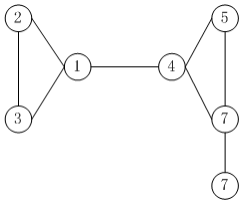


Figure 1. Initial Node Label Figure **Figure 2. Label Flow Phenomenon**

From above picture we can know that when we choose the sixth node as the original node, we may choose the label seven as the new label, we called this phenomenon label flow. Because it well lead to the increase of the number of iterations. When use label propagation algorithm we may face to pick a label form several similar nodes. Most time we made a random choice, which may reduce the correctness of the algorithm. Such as Figure3 shows

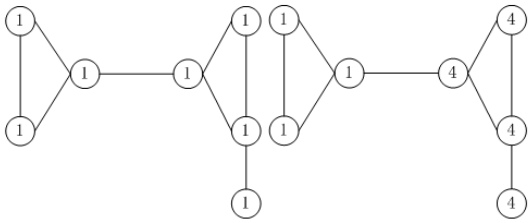


Figure 3. Result of Update Label with Random Choice

From above pictures we can know when we choose different label as the new label of the node, it may bring different community discover result. To solve this problem, Barber and his cooperates proposed a label propagation algorithm based on module degrees optimization [6]. Liu and his teammates after in-depth research find that the LPAm algorithm may trap in a loop when they try to find local optimal answer. To solve this problem they proposed an improved algorithm called LPAm+ algorithm [7] which based on multistep greedy algorithm. Through several experiment Leung [8] and his cooperate find after five iterations about ninety-five percent nodes have been correct assembled. And iteration after that can only help update nodes in communities. So we can say that the first iteration play important role in community discovery. Zhao [9] based on label influence put forward label influence based algorithm, the label influence based algorithm try to find some seed nodes and give these seed nodes initial labels, then spread out the labels from these seed nodes. Ma [10] and his group improved LPA algorithm by find some core figures which plays the same role like seed nodes. Although the literature [6][7] have already improved the stability of LPA algorithm, it also improved the time complexity. The literature [9][10] have some problems in choose initial seed nodes.

In order to solve these problems, we proposed Label Cohesion Algorithm. LCA algorithm is a kind of community discovery algorithm which based on Label Cohesion.

2. Basic Concept

In LCA algorithm we divided the label propagation process into two steps. Based on the theory in literature [11] [12], we comprehensive consider about the influence of node degree and node attraction. In this paper we proposed a new measurement standard based on node attraction. We use undirected graph G (V, E) on behalf of the network structure. We use V as all nodes and E as edges in graph G.

Definition 1: (Node centripetal): Node centripetal means the node is subject to other node.

$$S_i = \frac{\sum_j^J n_j}{d_i} \tag{2}$$

$$f(i) = \begin{cases} 1, S_i = 1 \\ 0, S_i \neq 1 \end{cases} \tag{3}$$

Definition 2: (Node attraction): Node attraction means the ability of aggregate other nodes. The node attraction formula is

$$D_i = \sum_j^J f(i) \tag{4}$$

Definition 3: (Label cohesion): Label cohesion is decided by occurrences of label, and the numbers of node covered by the label. The label cohesion formula is

$$T(l) = \sum N_l \tag{5}$$

$$C_i(l) = T(l) + \frac{\sum_l^L D_l}{d_i} \tag{6}$$

In above formulas j is an adjacent node of i; J is the set of j. n_j is the number of nodes who have the same adjacent node with node i in J, d_i means degree of node i, f(i) is the value of node centripetal. D_i means the node attraction of node i. And l means the value of node label, N_l means the number of those nodes who have the label value l, T(l) means the amounts of labels, who have the same value, appears at the neighbor nodes of node i. When there are several label appeared the maximum times at the adjunct node of node i, we evaluating the $C_i(l)$ of those labels, and then select the maximum value label as the updated label. $C_i(l)$ means the cohesion of label.

3. Label Cohesion Algorithm

3.1. The Thought of LCA Algorithm

The first step of LCA algorithm is taking pretreatments on the original label of all nodes. Though this treatment we can get a better initial label scheme. Literature [13] is proved the triangle go through an edge, the more important the edge will be. So the two nodes connected by the edge are likely to be divided into the same community. We choose node attraction as the judgement in this paper. It can help avoid label reflux phenomenon by make the label propagate from higher cohesion node to lower cohesion node. After the pretreatment we can make nodes who have closer connection in the same original community. From literature [14], we can find that better initial label assigned can help with reduce the number of iterations.

The second step is set label cohesion as the judgement to choose the new label of all nodes. The literature proved that if community has a closer internal structure, this community will have a greater attractive to those external community node. Use label

cohesion as a judgement can reduce the number of randomly chosen. It can improve accuracy and stability of the algorithm too.

3.2. The Description of LCA Algorithm

The first step is computing centripetal of all nodes, and find out those nodes whose centripetal value is zero. We give different labels in order to those nodes, and update the label of all nodes in G. The rule for label updating is making those nodes who do not have label get label form their adjacent node who has label. At last give different labels to nodes without a label. Then we order all nodes by their degree and put the result in set D.

The second step is label updating. The updated label of a node is the label who appeared most times at the adjacent nodes of node i. If there are several label appeared the maximum times at the adjunct node of node i, we evaluating the Ci(l) of those labels, and then select the maximum value label as the updated label. If they have the same Ci(l) we choose the label with the largest node degree as the new label. Then we update node labels according to this algorithm until all the labels do not change any more.

Let us suppose that the number of nodes in G is n and the number of edges is m. So time complexity for determine degrees of all nodes is O (m). The time complexity for node ranking is O (d (radix + n)), d means all the nodes ranked d times, radix means the lists used to collect nodes. Generally $d \leq 5$, radix=10. So the time complexity of high priority based radix sort algorithm is O (n). And the time complexity for count Di of all nodes is O (m). So the time complexity of the first step is O (n). The pseudocode for Label Cohesion Algorithm is shown as bellow

Input: G (V, E); i is original node serial number; original label Li;

```
// Label pretreatment
for each i ∈ V do
  compute f(i);
  compute d_i and order the nodes by the value of d_i;
  for each node f(i) == 0 do
    assign these nodes different labels in order;
    put node i adjacent nodes in set J;
    compute S_j;
      for each node in set J && S_j == 1
        assign the label of node i to these nodes;
      end for
    end for
    assign different labels in order to the remaining nodes;
 end for
// label update
for node i in D
  compute x;
  //x is the number of labels who appeared most times at the adjacent nodes of i;
  if (x == 1)
    assign the maximum appeared label to node i;
  end if
  else if (x == 0)
    the node label stays the same;
    else
      compute the Ci(l) and find the label who has the max Ci(l);
      if the max Ci(l) label is not unique
        using the new label instead of the original label;
      end if
      else
```


30. Select the one who has the largest degree as the new label;
31. end else
32. end else
33. end if
34. end for
35. find the resulting communities;
36. **Output:** Communities;

In the second step we only deal with nodes who have several label appeared the maximum times at the adjunct nodes, so it almost have no effects on the time complexity of the algorithm. So the time complexity of LCA algorithm is close to linear time complexity.

4. Experiments and Analysis

4.1. Karate Club Network Analysis

Zachary karate club membership network one of the most common small test network data set used in the field of complex networks and sociological analysis for community discovery. It takes Wayne Zachary three years to observe the relationship between the karate club members. And then he make out this data set. In the process of investigation, the club divided into two steps because of the problem whether to raise the club fees or not. Zachary karate club membership network has 34 nodes and 78 edges. Each node replace a member in the club and each edge means a connection between two members. The two algorithm based on Karate club network as shown in the figure below

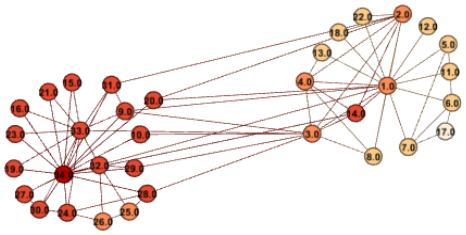


Figure 4. Karate Club Network Partitioning Visual Result

```
label: 0 size: 13 [1, 2, 3, 4, 8, 9, 12, 13, 14, 18, 20, 22, 31]
label: 2 size: 16 [10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 34]
label: 6 size: 5 [5, 6, 7, 11, 17]
0.034093106295
>>> ================================ RESTART ================================
>>>
label: 10 size: 5 [5, 6, 7, 11, 17]
label: 13 size: 12 [1, 2, 3, 4, 8, 10, 12, 13, 14, 18, 20, 22]
label: 29 size: 17 [9, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34]
0.0981388347119
>>> ================================ RESTART ================================
>>>
label: 31 size: 4 [25, 26, 29, 32]
label: 5 size: 10 [1, 5, 6, 7, 11, 12, 13, 17, 20, 22]
label: 14 size: 14 [9, 10, 15, 16, 19, 21, 23, 24, 27, 28, 30, 31, 33, 34]
label: 13 size: 6 [2, 3, 4, 8, 14, 18]
0.0477161448915
>>> ================================ RESTART ================================
>>>
label: 0 size: 11 [1, 2, 3, 4, 8, 12, 13, 14, 18, 20, 22]
label: 33 size: 18 [9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34]
label: 10 size: 2 [5, 11]
label: 5 size: 3 [6, 7, 17]
0.0260050813913
```

Figure 5. Four Times Result of LPA based on Karate Club Network

```
label: 21 size: 16 [1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 17, 18, 20, 22]
label: 31 size: 18 [9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34]
0.043369991219
>>> ========== RESTART ==========
>>>
label: 21 size: 16 [1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 17, 18, 20, 22]
label: 31 size: 18 [9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34]
0.0434976623055
>>> ========== RESTART ==========
>>>
label: 21 size: 16 [1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 17, 18, 20, 22]
label: 31 size: 18 [9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34]
0.0876209539863
>>> ========== RESTART ==========
>>>
label: 21 size: 16 [1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 17, 18, 20, 22]
label: 31 size: 18 [9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34]
0.018896141829
>>> ========== RESTART ==========
```

Figure 6. Four Times Result of LCA based on Karate Club Network

From above pictures we can find LCA algorithm divided the club into two parts, one choose node 1 as the core, the other choose node 34 as the core, the result is more accord with the facts. LPA algorithm divided the club into three parts or more, the result of LPA algorithm are different in each time. As LCA algorithm have the same result in four times test, we can say LCA algorithm have better accuracy than LPA algorithm.

4.2. American College Football Network Analysis

American college football network data set, is based on the college football data in 2000. American college football network has 105 nodes and 613 edges. In this network each node means a team in the union, an edge means there have been a game between two teams. Generally teams in same union have more frequently games than those in different unions. According to the reality the 115 football teams are from 12 different unions. The two algorithm based on American college football network as shown in the figure below

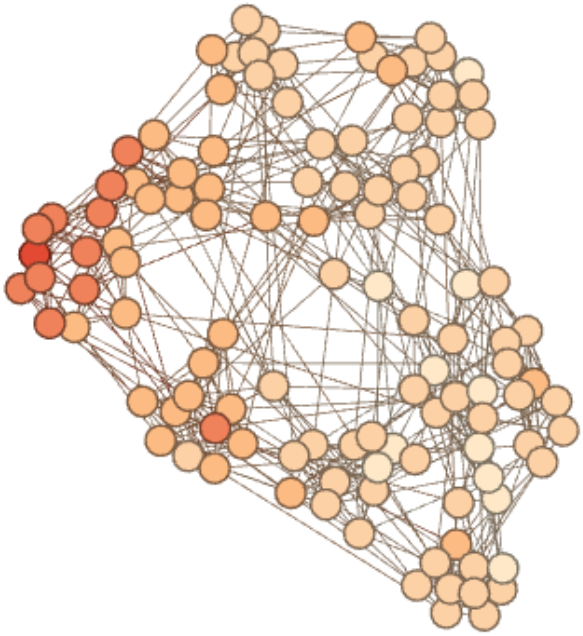


Figure 7. American College Football Network Partitioning Visual Result

```
label: 1 size: 9 [2, 26, 34, 38, 46, 90, 104, 106, 110]
label: 75 size: 9 [45, 49, 58, 67, 76, 87, 92, 93, 113]
label: 69 size: 6 [12, 25, 29, 51, 70, 91]
label: 49 size: 9 [47, 50, 54, 68, 74, 84, 89, 111, 115]
label: 107 size: 12 [4, 6, 11, 41, 53, 73, 75, 82, 85, 99, 103, 108]
label: 111 size: 18 [1, 5, 8, 9, 10, 17, 22, 23, 24, 42, 52, 69, 78, 79, 94, 105
, 109, 112]
label: 113 size: 16 [18, 21, 28, 57, 59, 60, 63, 64, 66, 71, 77, 88, 96, 97, 98,
 114]
label: 101 size: 10 [20, 30, 31, 36, 56, 80, 81, 83, 95, 102]
label: 31 size: 26 [3, 7, 13, 14, 15, 16, 19, 27, 32, 33, 35, 37, 39, 40, 43, 44
, 48, 55, 61, 62, 65, 72, 86, 100, 101, 107]
0.209424995084
>>> ================================ RESTART ================================
>>>
label: 64 size: 11 [3, 7, 14, 16, 33, 40, 48, 61, 65, 101, 107]
label: 49 size: 9 [47, 50, 54, 68, 74, 84, 89, 111, 115]
label: 103 size: 9 [2, 26, 34, 38, 46, 90, 104, 106, 110]
label: 74 size: 12 [4, 6, 11, 41, 53, 73, 75, 82, 85, 99, 103, 108]
label: 77 size: 10 [8, 9, 22, 23, 52, 69, 78, 79, 109, 112]
label: 79 size: 10 [20, 30, 31, 36, 56, 80, 81, 83, 95, 102]
label: 48 size: 9 [45, 49, 58, 67, 76, 87, 92, 93, 113]
label: 17 size: 18 [12, 18, 21, 25, 28, 29, 51, 57, 63, 66, 70, 71, 77, 88, 91,
96, 97, 114]
label: 23 size: 8 [1, 5, 10, 17, 24, 42, 94, 105]
label: 58 size: 4 [59, 60, 64, 98]
label: 61 size: 15 [13, 15, 19, 27, 32, 35, 37, 39, 43, 44, 55, 62, 72, 86, 100]
0.217365298714
>>>
```

Figure 8. Two Times Result of LPA based on American College Football Network

```
label: 97 size: 9 [12, 25, 29, 51, 60, 64, 70, 91, 98]
label: 99 size: 7 [19, 32, 35, 55, 62, 72, 100]
label: 101 size: 10 [20, 30, 31, 36, 56, 80, 81, 83, 95, 102]
label: 103 size: 15 [2, 26, 28, 34, 38, 46, 57, 63, 71, 77, 90, 96, 104, 106,
0]
label: 40 size: 12 [4, 6, 11, 41, 53, 73, 75, 82, 85, 99, 103, 108]
label: 9 size: 10 [8, 9, 22, 23, 52, 69, 78, 79, 109, 112]
label: 16 size: 8 [1, 5, 10, 17, 24, 42, 94, 105]
label: 114 size: 10 [47, 50, 54, 59, 68, 74, 84, 89, 111, 115]
label: 85 size: 8 [13, 15, 27, 37, 39, 43, 44, 86]
label: 57 size: 9 [45, 49, 58, 67, 76, 87, 92, 93, 113]
label: 60 size: 11 [3, 7, 14, 16, 33, 40, 48, 61, 65, 101, 107]
label: 95 size: 6 [18, 21, 66, 88, 97, 114]
0.231419367578
>>> ================================ RESTART ================================
>>>
label: 97 size: 9 [12, 25, 29, 51, 60, 64, 70, 91, 98]
label: 99 size: 7 [19, 32, 35, 55, 62, 72, 100]
label: 101 size: 10 [20, 30, 31, 36, 56, 80, 81, 83, 95, 102]
label: 103 size: 15 [2, 26, 28, 34, 38, 46, 57, 63, 71, 77, 90, 96, 104, 106,
0]
label: 40 size: 12 [4, 6, 11, 41, 53, 73, 75, 82, 85, 99, 103, 108]
label: 9 size: 10 [8, 9, 22, 23, 52, 69, 78, 79, 109, 112]
label: 16 size: 8 [1, 5, 10, 17, 24, 42, 94, 105]
label: 114 size: 10 [47, 50, 54, 59, 68, 74, 84, 89, 111, 115]
label: 85 size: 8 [13, 15, 27, 37, 39, 43, 44, 86]
label: 57 size: 9 [45, 49, 58, 67, 76, 87, 92, 93, 113]
label: 60 size: 11 [3, 7, 14, 16, 33, 40, 48, 61, 65, 101, 107]
label: 95 size: 6 [18, 21, 66, 88, 97, 114]
0.227845383522
```

Figure 9. Two Times Result of LCA based on American College Football Network

From above pictures, we can find the result of LPA algorithm is not the same and the LCA algorithm can give out a fixed result. So we can say LCA algorithm have better accuracy than LPA algorithm. Two algorithms running time compare as shown in the figure below

Figure 10. Two Algorithms Running Time Comparison

The left picture is running time of program based on karate club dataset. The right picture is running time of program based on American college football network data set. From above picture we can find the two algorithms have similar average running time, but LCA algorithm running time is more gently than LPA algorithm. So it proves LCA algorithm succeed in promoting the correctness of algorithm while maintaining the low time complexity.

5. Conclusions

Compared with LPA algorithm, LCA algorithm made label propagation process divided into two steps. In the first step we optimized the distribution of the original label. It helps reduce the iteration times, and solve the label reflux phenomenon problem. In the second step we take full consideration of the influence of node attraction and optimized the method of label updating. Though this way we can the key node more accurately. Finally the experimental results shows that the accuracy of the algorithm has been improved.

Acknowledgements

This research was supported by National Natural Science Foundation of China (No: 61262075), Scientific Research Project of Guangxi Education Department (200911LX119) and major scientific research project of Guangxi higher education (No: 201201ZD012).

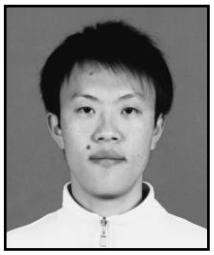
References

[1] M. Girvan, MEJ. “Newman Community Structure in Social and Biological networks [J]”, P Natl Acad Sci USA, **(2002)**, vol. 99, no. 12, pp. 7821-7826.
[2] M E J. Newman, “Fast Algorithm for Detecting Community structure in networks [J]”, Phys Rev E, **(2004)**, vol. 69, no. 6, 066133.
[3] Z. Bu, C. Zhang, Z. Xia, *et al*. “A fast parallel modularity optimization algorithm (FPMQA) for community detection in online social network [J]”, Knowledge-Based Systems, **(2013)**, vol. 50, no. 3, pp. 246–259.
[4] X. Zhu, Z. Ghanramani, “Learning from labeled and unlabeled data with label propagation[R]”, Pittsburghers: Carnegie Mellon University, **(2002)**.
[5] UN Raghavan, R. Albert, S. Kumara, “Near linear time algorithm to detect community structures in large-scale networks [J]. Physical Review E, **(2007)**, vol. 76, no. 3, 036106.

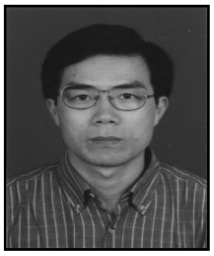
[6] M J Barber, J W Clark, "Detecting network communities by propagating labels under constraints [J]", Physical Review E, **(2009)**, vol. 80, no. 2, pp. 283-289.
[7] X. Liu, T. Murata, "Advanced modularity-specialized label propagation algorithm for detecting communities in networks [J]", Physica A Statistical Mechanics & Its Applications, **(2010)**, vol. 389, no. 7, pp. 1493-1500.
[8] I X Y Leung, P. Hui, P. Liò, *et al*. "Towards real - time community detection in large networks［J］.Phys rev, **(2009)**, vol. 79, no. 6, 066107.066107.
[9] Z. Zhao, Y. Wang, J. Tian, *Etc*. "A novel algorithm for community in social networks based on label propagation [J]", Research and development of the computer, **(2011)**, S3.
[10] J. Ma, L. han, Z. Pan, *Etc*. "Label propagation algorithm based on community core for community detection [J]", Computer Science, 2015, the first phase, vol. 1, pp. 119-121. DOI:doi:10.11896/j.issn.1002-137X.**(2015)**.1.028.
[11] S. Fortunato, A. Lancichinetti, "Community detection algorithms: A comparative analysis [J]", Physical Review E, **(2009)**, vol. 80, no. 5, pp. 733-737.
[12] X. Wu, X. Zhu, G Q Wu, *et al*. "Data Mining with Big Data[J]", IEEE Transactions on Knowledge & Data Engineering, **(2014)**, vol. 26, no. 1, pp. 97-107.
[13] X W Zhao, X J Liu, M Y Yin. "Data Mining Clustering Algorithm [J]", Computer Knowledge & Technology, **(2014)**.
[14] G. Isaak, D B. West, "The Edge-Count Criterion for Graphic Lists [J]", Electronic Journal of Combinatorics, **(2010)**, vol. 17, no. 1, pp. 1945-1957.

Authors

Gui Qiong Associate Professor, Master, College of Information Science and Engineering, Guilin University of Technology, Postcode: 541004, e-mail:guilucky@163.com.



Zhu Dejun Bachelor, College of Information Science and Engineering, Guilin University of Technology, Postcode: 541004, e-mail:zlr0913@163.com.



Cheng Xiaohui Professor, College of Information Science and Engineering, Guilin University of Technology, Postcode: 541004,e-mail:cxiaohui@glut.edu.cn

