Hindawi Publishing Corporation
The Scientific World Journal
Volume 2013, Article ID 368568, 9 pages
http://dx.doi.org/10.1155/2013/368568

Research Article

"Follow the Leader": A Centrality Guided Clustering and Its Application to Social Network Analysis

Qin Wu,[1,2] Xingqin Qi,[2,3] Eddie Fuller,[2] and Cun-Quan Zhang[2]

[1] *Department of Computer Science, Jiangnan University, Wuxi, Jiangsu 214122, China*
[2] *Department of Mathematics, West Virginia University, Morgantown, WV 26505, USA*
[3] *School of Mathematics and Statistics, Shandong University at Weihai, Weihai 264209, China*

Correspondence should be addressed to Qin Wu; qinwu@math.wvu.edu

Received 8 August 2013; Accepted 10 September 2013

Academic Editors: T. C. Chan and Y. Wei

Within graph theory and network analysis, centrality of a vertex measures the relative importance of a vertex within a graph. The centrality plays key role in network analysis and has been widely studied using different methods. Inspired by the idea of vertex centrality, a novel centrality guided clustering (CGC) is proposed in this paper. Different from traditional clustering methods which usually choose the initial center of a cluster randomly, the CGC clustering algorithm starts from a "LEADER"—a vertex with the highest centrality score—and a new "member" is added into the same cluster as the "LEADER" when some criterion is satisfied. The CGC algorithm also supports overlapping membership. Experiments on three benchmark social network data sets are presented and the results indicate that the proposed CGC algorithm works well in social network clustering.

1. Introduction

Clustering is a process of partitioning a set of data into meaningful subsets so that all data in the same group are similar and the data in different groups are dissimilar in some sense. It is a method of data exploration and a way of looking for patterns or structure in the data that are of interest. Clustering has wide applications in social science, biology, chemistry, and information sciences. A general review of cluster analysis can be found in many references such as [1–4].

The commonly used clustering methods are partitional clustering and hierarchical clustering. Partitional algorithms typically determine all clusters at once. K-means [5] clustering algorithm is a typical partitional clustering. Given the number of clusters (say k), the procedure of K-means clustering is as follows. (i) Randomly generate k points as cluster centers and assign each point to the nearest cluster center. (ii) Recompute the new cluster centers. (iii) Repeat the two previous steps until some convergence criterion is met. The main advantages of the K-means algorithm are its simplicity and speed which allows it to run on large datasets. However, it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments. And the number of clusters has to be predefined.

The hierarchical clustering is either agglomerative or divisive. Agglomerative algorithms begin with each element as a separate cluster and two clusters separated by the shortest distance are merged successively. Most hierarchical clustering algorithms are agglomerative, such as SLINK [6] for sing linkage and CLINK [7] for complete linkage. Divisive starts with one big cluster and splits are performed recursively as one moves down the hierarchy. The hierarchical clustering builds a hierarchy tree of clusters, which is called dendrogram. The way in which elements are clustered is clearly shown in the dendrogram.

In recent years, social network analysis has gained much attention. Social network analysis is the study of social relations in terms of networks. A social network is usually modeled as a directed graph or undirected graph. The set of nodes in the graph represent individual members. The set of edges in the graph represent relationship between the individuals, such as friendship, coauthorship, and so forth. A fundamental problem related to social networks is the discovery of clusters or communities. Porter et al. [8] summarized

different clustering methods for social network clustering. Wu and Huberman [9] proposed to find communities based on notions of voltage drops across networks. Girvan and Newman [10] proposed to discover community structure based on edge betweenness. Newman [11] proposed to find community structure based on the eigenvectors of matrices. Clauset et al. [12] proposed a modularity-based method for finding community structure in very large networks.

In this work, a novel hierarchical clustering algorithm is proposed for social network clustering. Traditional clustering methods, such as K-means, usually choose clustering centers randomly, and the hierarchical clustering algorithms usually start from two elements with shortest distance. Different from these methods, this work chooses the vertex with highest centrality score as the starting point. If one does some analysis on social network datasets, one may notice that in each community, there is usually some member (or leader) who plays a key role in that community. In fact, centrality is an important concept [13] within social network analysis. High centrality scores identify members with the greatest structural importance in a network and these members are expected to play key roles in the network. Based on this observation, this work proposes to start clustering from the member with highest centrality score. That is, a group is formed starting from its "leader," and a new "member" is added into an existing group based on its total relation with the group. The main procedure is as follows. Choose the vertex with the highest centrality score which is not included in any existing group yet and call this vertex a "LEADER." A new group is created with this "LEADER." Repeatedly add one vertex to an existing group if the following criterion is satisfied: the density of the newly extended group is above a given threshold.

The paper is organized as follows. Different centrality measurements are discussed in Section 2. The proposed clustering algorithm is described in Section 3. In Section 4, test results of the new algorithm on some social network benchmark datasets are compared with ground truth and some traditional methods. Conclusions are made in Section 5.

2. Measures of Centrality

Centrality is one of the most widely studied concepts in social network analysis. Within graph theory and network analysis, centrality of a vertex measures the relative importance of a vertex within the graph. For example, how important a person is within a social network or how well used a road is within an urban network. During past years, various measures of the centrality of a vertex have been proposed. Centrality measurement, such as degree centrality, betweenness, and eigenvector centrality, are among the most popular ones.

Degree centrality is the simplest centrality measurement. Given a graph G, denote the set of vertices of G as $V(G)$, and then the degree centrality for any $v \in V(G)$ is defined as

$$C_D(v) = \frac{d(v)}{|V(G)| - 1}, \quad (1)$$

where $d(v)$ is the degree of v and $|V(G)|$ is the number of vertices in G.

Degree centrality considers only the local topology of the network. It can be interpreted as a measure of immediate influence, as opposed to global effect in the network [14].

The betweenness centrality for any $v \in V(G)$ is defined as

$$C_B(v) = \frac{2}{(|V(G)| - 1)(|V(G)| - 2)} \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}, \quad (2)$$

where $s, v, t \in V(G)$, σ_{st} is the number of shortest paths from s to t, and $\sigma_{st}(v)$ is the number of shortest paths from s to t that pass through the vertex v.

Betweenness centrality is one of the most popular centrality measures which consider the global structure of the network. It characterizes how influential a vertex is in communicating between vertex pairs [15].

The eigenvector centrality score of the ith vertex in the network is defined as the ith component of the eigenvector corresponding to the greatest eigenvalue of the following characteristic equation:

$$Ax = \lambda x, \quad (3)$$

where A is the adjacency matrix of the network, λ is the largest eigenvalue of A, and x is the corresponding eigenvector. It simulates a mechanism in which each vertex affects all of its neighbors simultaneously [16].

Eigenvector centrality is a sort of extended degree centrality which is proportional to the sum of the centralities of the vertex's neighbors. A vertex has large value of eigenvector centrality score either if it is connected to many other vertices or if it is connected to others that themselves have high eigenvector centrality [17].

Due to the fact that different centrality measures are based on different aspect of a network, the final centrality scores and ranking of the nodes in the network may be different. The difference will be discussed in Section 4.

3. Centrality Guided Clustering

In this section, some notation and terminology are introduced and the centrality guided clustering (CGC) algorithm is presented.

Given an input dataset, the dataset is modeled as a weighted graph $G = (V, E, w)$. V is the vertex set. Each vertex in V represents an element in the dataset. $|V(G)|$ represents the number of vertices in G (or elements in the dataset). E is the edge set. Each edge represents a relationship between a pair of elements. w is the edge weight function. $w(u, v)$ and $w(e)$ denote the weight of the edge e between two vertices u and v. If there is no edge between two vertices u and v, then $w(u, v) = 0$. If the graph is an unweighted graph, then

$$w(uv) = \begin{cases} 1, & \text{if } uv \in E(G), \\ 0, & \text{if } uv \notin E(G). \end{cases} \quad (4)$$

Input: a weighted graph G.
Output: clustering dendrogram of the graph G.
(Initialization) $l = 1$, $G_l = G$,
while $|V(G_l)| > 1$
(GROUPING) Cluster the vertices in G_l into different groups.
(MERGING) Merge those groups with large percentage of overlap.
(CONTRACTION) Contract those vertices in the same groups to a new vertex, calculate the edge weights in the contracted graph.
Denote the contracted graph as G_{l+1}, $l = l + 1$.

Algorithm 1: CGC algorithm.

Let C be a subgraph of G, the density of the subgraph C is defined as

$$\text{density}(C) = \frac{2\sum_{e\in E(C)} w(e)}{|V(C)|(|V(C)|-1)}, \quad \text{if } |V(C)| > 1. \tag{5}$$

The density of the subgraph C could be looked as the intracluster similarity. Good clustering should have high intracluster similarity and low intercluster similarity. If all nodes in C belong to the same cluster, then density(C) should be large.

As discussed in Section 2, the centrality of a vertex measures the relative importance of the vertex within the network. One would expect that after clustering, each group has a center and the center has relative high centrality score. On the other side, if a clustering algorithm starts from the vertex (called it a "LEADER") with high centrality score, one would expect those vertices with tight connection with the LEADER to be grouped together. The clustering result will have high intrasimilarity and low intersimilarity. This is the motivation of the CGC algorithm. Denote the centrality score of the vertex v in the graph G as score(v). For any set S, denote the number of elements in S as $|S|$.

For any vertex $v \notin V(C)$, the contribution of v to C is defined as

$$\text{contribution}(v, C) = \frac{\sum_{u\in V(C)} w(uv)}{|V(C)|}. \tag{6}$$

A vertex $v \notin V(C)$ is called a neighbor of C if there is a vertex $u \in C$ such that $uv \in E(G)$. The vertex v is called a *candidate neighbor* of C if v satisfies the following three conditions:

(a) v is a neighbor of the subgraph C;

(b) there exists a vertex $u \in V(C)$, such that

$$\begin{aligned} &w(u, v) \geq \alpha * \max\{w(e) \mid e \in E(G)\}, \quad \text{if } |V(C)| = 1, \\ &\text{contribution}(v, C) > \beta * \text{density}(C), \quad \text{if } |V(C)| > 1; \end{aligned} \tag{7}$$

(c) score(v) < max{score(u) | $u \in C$}.

The set of all candidate neighbors of the subgraph C is denoted as $N(C)$.

Condition (a) says that a vertex must be a neighbor of the subgraph C in order to be considered to be clustered into the current group C. Condition (b) is to control the density of the subgraph C such that the density will not decrease too much after the candidate neighbor is added into the subgraph C. Condition (c) says that only those vertices with centrality score lower than the centrality score of *some* vertex in C are considered. That is, if a vertex $v \in N(C)$ has higher centrality score than any vertices in C, then the vertex v must have already been clustered into another group, so v will not be grouped into the group C. α and β are used to control the clustering so that the density of the new subgraph will not decrease too much after a *candidate neighbor* is added into the subgraph C. In another paper [18], we proved that if $\alpha = 0.8$ and $\beta = 1 - (1/(2 * (|V(C)| + 1)))$, then the density of the new subgraph with a set of candidate neighbors added to the subgraph C will not decrease over 1/3.

The overall structure of the CGC algorithm is shown in Algorithm 1. The three main steps are GROUPING, MERGING, and CONTRACTION.

The details of the GROUPING step is shown in Algorithm 2. The GROUPING algorithm creates a new group from the vertex with the highest centrality score which has not been clustered into any group yet. And this vertex is called the center (or leader) of the new group. Denote this vertex as the center of current group C_i. Then the next vertex is chosen from the candidate neighbor set $N(C_i)$ with the largest contribution to C_i.

In the CGC algorithm, every vertex is allowed to be belonged to more than one group. So after the GROUPING step, different groups may have some overlapping elements. If the number of overlapping elements in two groups exceeds some threshold, it is better to merge all vertices in the two groups into a new larger group. The following criterion is applied to determine whether two groups should be merged. Given any two groups, say C_i and C_j, if C_i and C_j satisfy the following criterion, then C_i and C_j are merged into one group:

$$\frac{|C_i \cap C_j|}{\min\{|C_i|, |C_j|\}} \geq \frac{1}{2}. \tag{8}$$

That is, if the size of overlapping of two groups is greater than half of the size of the smaller one of the two groups, the two groups are merged into one group. The MERGING algorithm

Input: a weighted graph G_l.
Output: each vertex is assigned to a group.
Calculate the centrality score of each vertex $v \in G_l$,
Order $v \in V(G_l)$ via their centrality scores, such
that $Q = (v_1, v_2, \ldots, v_n)$ with score $(v_1) \geq$ score $(v_2) \geq \cdots \geq$ score (v_n).
$i = 0$
while $Q \neq \emptyset$
  $i = i + 1$
  Create a new group $C_i = \{v_{i_1}\}$, where v_{i_1} is the first vertex in the vertex queue Q.
  New_Q = $Q - \{v_{i_1}\}$
  while |New_Q| < |Q|
    Q = New_Q
    Find the **candidate neighbor set** $N(C_i)$ of C_i.
    Calculate the contribution to C_i of each vertex in $N(C_i)$.
    Sort $v \in N(C_i)$ in descending order by their contribution to C_i, that is,
    $Q_N = (v_{n_1}, v_{n_2}, \ldots, v_{n_k})$, where $v_{n_i} \in N(C_i)$ and
    contribution $(v_{n_1}, C_i) \geq$ contribution $(v_{n_2}, C_i) \geq \cdots \geq$ contribution (v_{n_k}, C_i).
    if $N(C_i) = \emptyset$, break.
    else
      $C_i = C_i \cup \{v_{n_1}\}$
      New_Q = $Q - \{v_{n_1}\}$, $Q_N = Q_N - \{v_{n_1}\}$

Algorithm 2: GROUPING algorithm.

Input: a weighted graph G_l which has already been
clustered into s groups as in the GROUPING algorithm.
Output: each vertex is assigned to a new group.
List all groups of G_l as $L = \{C_1, C_2, \ldots, C_s\}$ such
that $|V(C_1)| \geq |V(C_2)| \geq \cdots \geq |V(C_s)|$
  $h = 2$, $j = 1$
  while $j < h$
    if $|C_j \cap C_h| \geq \frac{1}{2} * \min(|C_j|, |C_h|)$,
      $L = L \cup \{C_j \cup C_h\} - \{C_j\} - \{C_h\}$
      $s = s - 1$, $h = \max\{h - 2, 1\}$,
      $h = h + 1$, $j = 1$
    else
      $j = j + 1$

Algorithm 3: MERGING algorithm.

(see Algorithm 3) describes the details about how to merge two groups.

After the MERGING step, each group C_i is contracted into a new vertex v_i. If there is an edge between two groups C_i and C_j, then there will be an edge $v_i v_j$ in the contracted graph. The weight of the edge, $w(v_i, v_j)$, is calculated as follows:

$$w(v_i, v_j) = \frac{\sum_{e \in E(C_i, C_j)} w(e)}{|V(C_i)| * |V(C_j)|}, \quad (9)$$

where $E(C_i, C_j)$ is the set of crossing edges, $E(C_i, C_j) = \{xy : x \in V(C_i), y \in V(C_j), x \neq y\}$. The details are presented in the CONTRACTION algorithm (see Algorithm 4).

4. Results and Discussion

To evaluate the effectiveness of the CGC algorithm, three benchmark datasets on social network analysis are tested. The three benchmark datasets and the clustering results are described in Sections 4.1, 4.2, and 4.3. The betweenness centrality is used to calculate centrality scores in the CGC algorithm. The results of the CGC algorithm are compared with the ground truth and the results of the Girvan-Newman algorithm [10]. The Girvan-Newman algorithm is one of the most popular algorithms for detecting communities in complex systems. The communities are detected by calculating the edge betweenness centralities of all edges and removing the edge with the highest betweenness value recursively.

To test whether the centrality measures will influence the results, different centrality measures are applied to the CGC

Input: a weighted graph G_l which has already been merged into s groups as in the MERGING algorithm,
Output: a contracted graph with edge weights.
List all groups of G_l after the MERGING step as $L = \{C_1, C_2, \ldots, C_s\}$
Generate a new vertex v_p to represent the group C_p.
for $i = 1$ to s
  for $j = 1$ to i

$$w\left(v_i, v_j\right) = \frac{\sum_{e \in E(C_i, C_j)} w(e)}{|V(C_i)| * |V(C_j)|}$$

$$w\left(v_j, v_i\right) = w\left(v_i, v_j\right)$$

Algorithm 4: CONTRACTION algorithm.

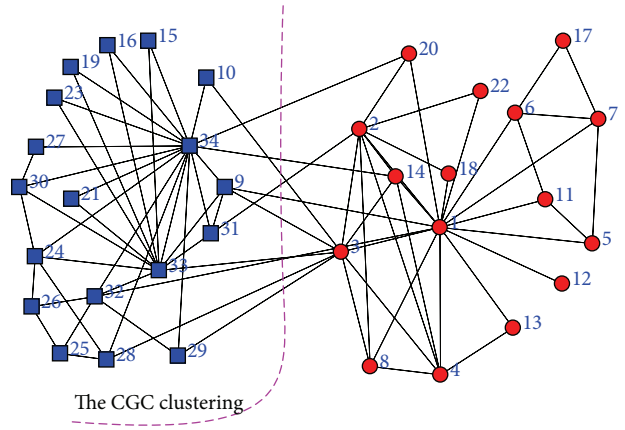

Figure 1: The social network of Zachary's karate club. Red dots denote the supporters of instructor and blue squares denote the supporters of the president. The dashed curve is the partition by the CGC algorithm.

algorithm independently and the clustering results are compared in Section 4.4. All of the three datasets could be downloaded from Newman's website [19].

4.1. Zachary's Karate Club. Zachary's karate club dataset is a typical dataset which is used to test the clustering algorithm in social network analysis. It is a social network of friendships between 34 members of a karate club at a US university [20]. Zachary recorded the interaction of the karate club in the university for three years. The social network of relationships in Zachary's karate club is shown in Figure 1. Node 1 represents the instructor of the club and node 34 represents the president of the club. During the observation, there was an incipient conflict between the instructor and the president. And the conflict subsequently led to a formal separation of the club into two organizations: one group is the supporters of the instructor and the other group is the supporters of the president. The ground truth groups are denoted as red dots and blue squares in Figure 1. The red dots denote the supporters of instructor and the blue squares denote the supporters of the president.

When the Girvan-Newman algorithm is applied to this dataset, node 3 is misclassified. The partition by the CGC algorithm is shown as the dashed curve in Figure 1, which is exactly the same as the ground truth. Figure 2 is the dendrogram corresponding to the result of the CGC algorithm. Another important observation is that when the betweenness centrality is used, the node with the highest betweenness centrality scores is node 1 and the second highest is node 34, which are the instructor and the president, the true leaders of the two groups.

4.2. Dolphin Social Network. The dolphin social network dataset is another representative dataset to test the accuracy of clustering algorithms. It is a social network of frequent associations between dolphins in a community in Doubtful Sound, New Zealand [21]. The social network of the dolphins is presented in Figure 3. There are 62 vertices and 159 edges in the network. The vertices represent the bottlenose dolphins, and the edges between the vertices represent associations between dolphin pairs occurring more often than expected by chance. During the course of the study, the dolphins split into two groups following the departure of a key member (represented as the yellow triangle in the Figure 3) of the population.

The ground truth groups are represented by the shapes of the vertices in Figure 3. The vertices represented as squares are in one group and the vertices represented as dots and triangle are in the other group. The dashed curve represents

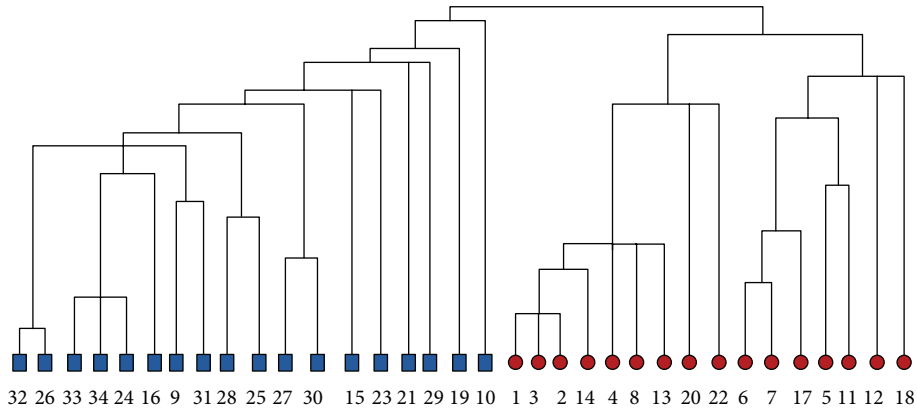

Figure 2: The dendrogram of the karate club dataset by the CGC algorithm.

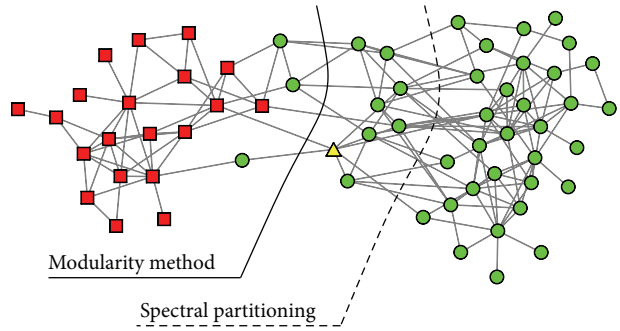

Figure 3: The social network of the dolphins. The dashed curve denotes the division of the network into two equal-size groups found by the standard spectral partitioning method, and the solid curve represents the division found by the modularity-based method by Newman [11].

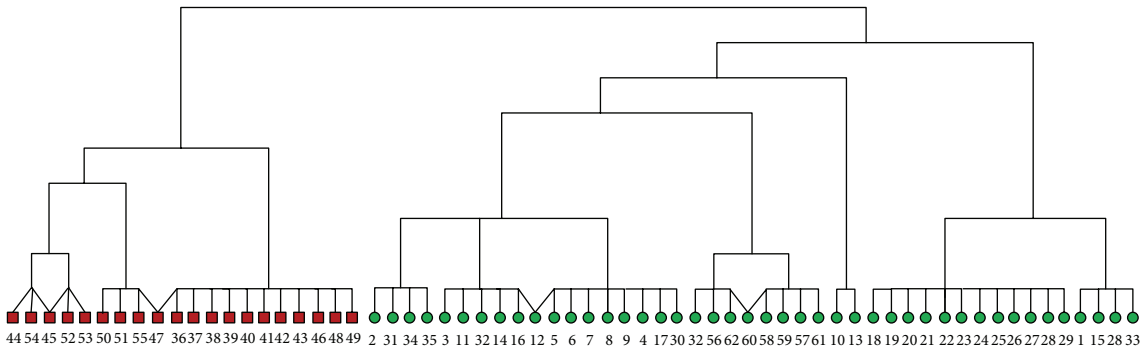

Figure 4: The dendrogram of the dolphin dataset by the CGC algorithm.

the division of the network into two equal-size groups found by the standard spectral partitioning method proposed by Newman [11]; 11 out of 62 dolphins are misclassified. The solid curve represents the division found by the modularity-based method by Newman [11]; 3 out of 62 dolphins are misclassified. When the Girvan-Newman algorithm is applied to this dataset, 2 out of 62 dolphins are misclassified. When the CGC algorithm is applied to the dolphin social network, it divides the dolphins into two groups, which is exactly the same as the ground truth. The corresponding dendrogram produced by the CGC algorithm is shown in Figure 4.

4.3. Social Network of Political Books. The third example is a social network map of political books based on purchase patterns from the online book seller Amazon.com. This dataset is provided by Krebs [22]. And the groups of different books are shown in Figure 5. The 105 nodes represent 105 books about US politics. Each book is manually labeled as liberal, neutral, or conservative. Correspondingly, the three types of books are illustrated using three different shapes: triangles for neutral books, dots for conservative books, and squares for liberal books, as in Figure 5. For simplicity, the 105 books are denoted as 1 to 105 instead of book names.

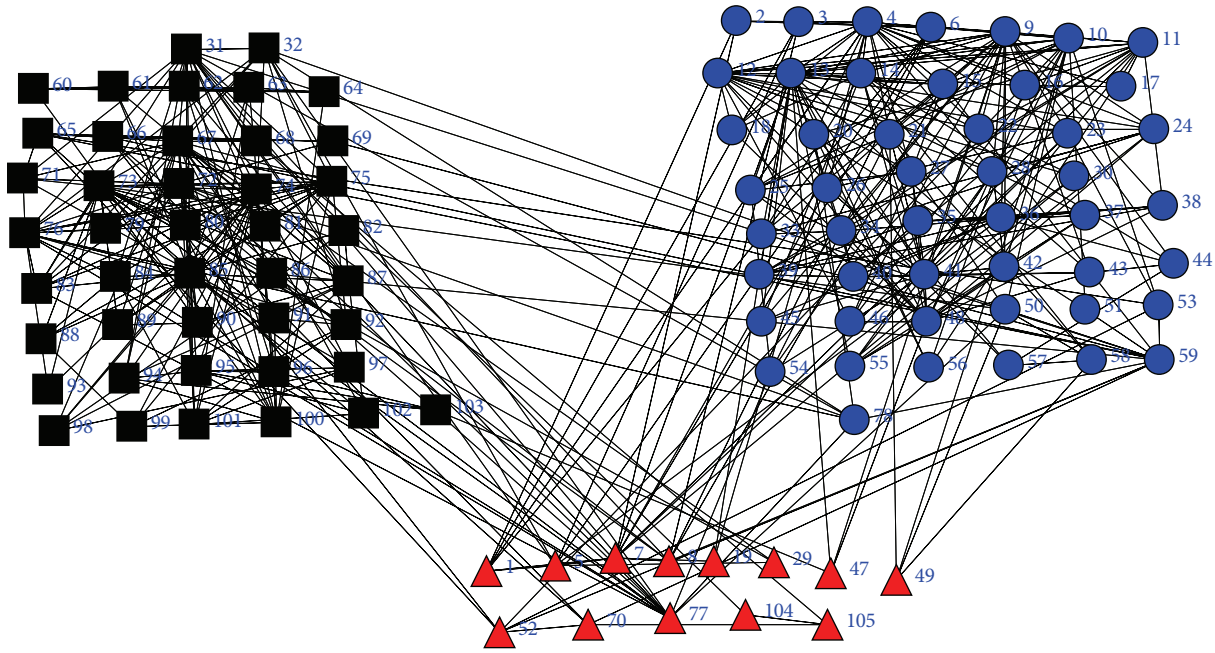

Figure 5: The ground truth partition of the political books. Triangles for neutral books, dots for conservative books, and squares for liberal books.

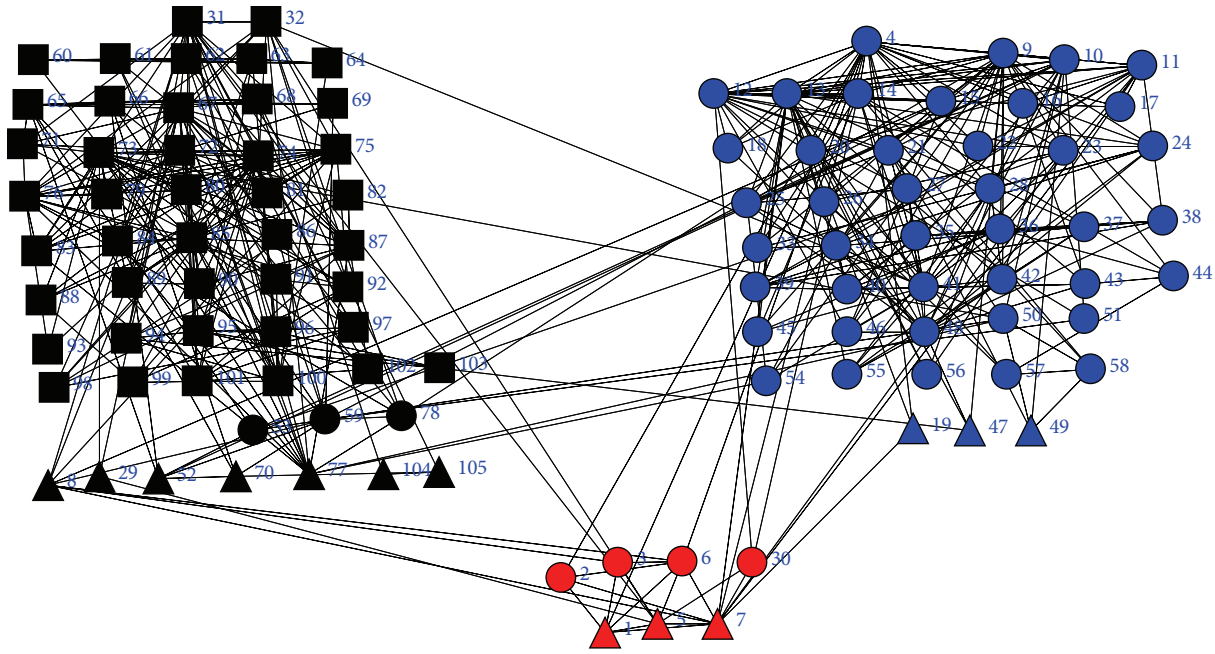

Figure 6: The clustering result of the political books by the Girvan-Newman algorithm. Red for neutral books, blue for conservative books, and black for liberal books.

Two books are linked in the social network if they were frequently copurchased by the same customer. Figure 5 shows the ground truth classification for the 105 books.

In order to see the clustering results based on the book copurchase information, the Girvan-Newman algorithm [10] and the CGC algorithm are applied independently to the adjacency matrix of the political books. When the Girvan-Newman algorithm is applied to the adjacency matrix of the social network, 17 books (2, 3, 6, 8, 19, 29, 30, 47, 49, 52, 53, 59, 70, 77, 78, 104, and 105) are misclassified. The clustering result of the Girvan-Newman algorithm is shown in Figure 6. When betweenness centrality is used and the CGC algorithm

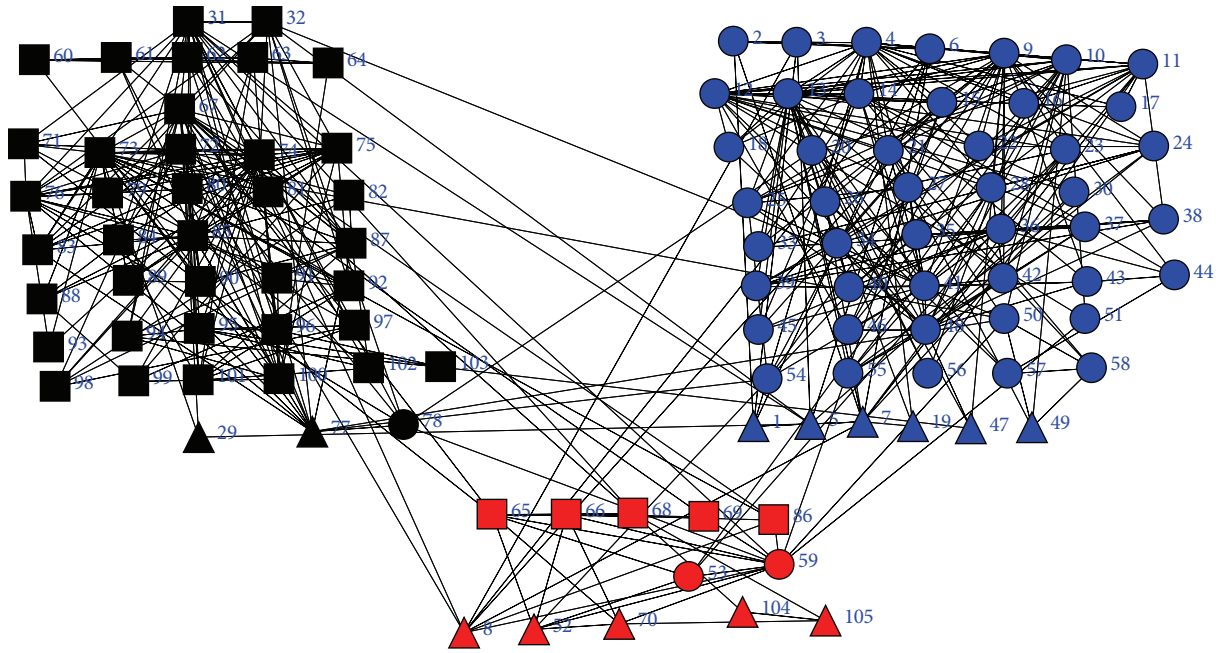

Figure 7: The clustering result of the political books by the CGC algorithm. Red for neutral books, blue for conservative books, and black for liberal books.

Table 1: The number of misclassified members by the CGC algorithm based on different centrality measures.

	Karate club	Dolphin	Political books
Degree	0	1	17
Eigenvalue	0	2	16
Betweenness	0	0	16

is applied to the same dataset, only 16 books (1, 5, 7, 19, 29, 47, 49, 53, 59, 65, 66, 68, 69, 77, 78, and 86) are misclassified. The clustering result of the CGC algorithm is shown in Figure 7.

4.4. Clustering with Different Centrality Measures. As mentioned in previous sections, the centrality score of a node in a network could be looked as how important a node is in the network. And the importance of the nodes could be sorted by their centrality scores from large to small. When different centrality measures are applied to the same dataset, the ordering of nodes may be different.

The purpose of this subsection is to test whether the starting clustering node will influence the final clustering result and to compare the effectiveness of different centrality measure when combined with the CGC algorithm. In this subsection, degree centrality, eigenvalue centrality, and betweenness centrality are independently applied to the CGC algorithm. And the same three datasets as in Sections 4.1, 4.2, and 4.3 are used in the experiments.

Table 1 lists the number of misclassified nodes when different centrality measurements are applied to the CGC algorithm. From the table, one could observe that the initial starting node do influence the final results. For the Zachary's karate club dataset, the three centrality measures all produce perfect results. The degree centrality works better than eigenvalue centrality on the dolphin dataset. But on the political book dataset, the degree centrality is worse than the eigenvalue centrality. Overall, the betweenness centrality measure works best with the CGC algorithm.

5. Conclusions

In this work, the importance of the centrality score of vertices in a network is discussed and a centrality guided clustering method is proposed. The CGC algorithm initiates the clustering process at a vertex with highest centrality score, which is a potential leader of a community. The CGC algorithm is applied to several benchmark social network datasets. Experimental results show that CGC algorithm works well on social network clustering.

Centrality measurements may influence the results of the CGC algorithm. The degree criterion serves as a very local measurement for centrality, while betweenness centrality and eigenvalue centrality search for global "leaders" of the entire networks. The experiment results show that the betweenness centrality works better than the other two centrality measures for the CGC algorithm.

One may notice that in Figure 4, one single node, such as nodes 45, 47, 12, and 60 in the lowest level, is clustered into two different groups. In fact, it is reasonable for some individual to belong to two different groups. Say for example, some people may be affiliated with two or more organizations. In fact, allowing one object to be clustered into two or more groups is one of the properties of the CGC algorithm, which makes the CGC algorithm different from other clustering algorithms.

The CGC algorithm is a hierarchical clustering algorithm. One direction for future research would be to apply

the centrality score guided idea to other clustering methods such as K-means clustering. Hopefully, it will also produce promising clustering results.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

Acknowledgments

The authors would like to thank the anonymous reviewers for suggesting many ways to improve the paper. The work is partially supported by the National Natural Science Foundation of China (no. 61202312); the NSA Grant H98230-12-1-0233, the NSF Grant DMS-1264800; the Fundamental Research Funds for the Central Universities, China (no. JUSRP11231); and the Shandong Province Natural Science Foundation of China (no. ZR2010AQ018).

References

[1] G. Milligan, *Encyclopedia of Statistical Sciences*, Wiley, New York, NY, USA, 1998.

[2] W. K. Härdle and L. Simar, *Applied Multivariate Statistical Analysis*, Springer, Berlin, Germany, 2003.

[3] P. Hansen and B. Jaumard, "Cluster analysis and mathematical programming," *Mathematical Programming B*, vol. 79, no. 1–3, pp. 191–215, 1997.

[4] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[5] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, University of California Press, 1967.

[6] R. Sibson, "Slink: an optimally efficient algorithm for the single-link cluster method," *Computer Journal*, vol. 16, no. 1, pp. 30–34, 1973.

[7] D. Defays, "An efficient algorithm for a complete link method," *Computer Journal*, vol. 20, no. 4, pp. 364–366, 1977.

[8] M. A. Porter, J.-P. Onnela, and P. J. Mucha, "Communities in networks," *Notices of the American Mathematical Society*, vol. 56, no. 9, pp. 1082–1097, 2009.

[9] F. Wu and B. A. Huberman, "Finding communities in linear time: a physics approach," *European Physical Journal B*, vol. 38, no. 2, pp. 331–338, 2004.

[10] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.

[11] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, Article ID 036104, 2006.

[12] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, no. 6, Article ID 066111, 6 pages, 2004.

[13] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, vol. 8 of *Structural Analysis in the Social Sciences*, Cambridge University Press, New York, NY, USA, 1st edition, 1994.

[14] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978.

[15] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977.

[16] P. Bonacich, "Power and centrality: a family of measures," *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, 1987.

[17] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, Article ID 026113, 2004.

[18] Y. Ou and C.-Q. Zhang, "A new multimembership clustering method," *Journal of Industrial and Management Optimization*, vol. 3, no. 4, pp. 619–624, 2007.

[19] http://www-personal.umich.edu/~mejn/netdata/.

[20] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, pp. 452–473, 1977.

[21] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations: can geographic isolation explain this unique trait?" *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, 2003.

[22] V. Krebs, http://www.orgnet.com/.