Community Detection by Affinity Propagation

Zhiyuan Liu, Peng Li, Yabin Zheng, and Maosong Sun

Department of Computer Science and Technology,
State Key Lab on Intelligent Technology and Systems,
National Lab for Information Science and Technology,
Tsinghua University, Beijing, China, 100084
{liuliudong,pengli09,yabin.zheng}@gmail.com, sms@tsinghua.edu.cn
http://nlp.csai.tsinghua.edu.cn

Abstract. Community structure in networks indicates groups of vertices within which are dense connections and between which are sparse connections. Community detection, an important topic in data mining and social network analysis, has attracted considerable research interests in recent years. Motivated by the idea that community detection is in fact a clustering problem on graphs, we propose several similarity metrics of vertex to transform a community detection problem into a clustering problem, and further adopt a recently-proposed clustering method, namely 'Affinity Propagation', to extract communities from graphs. We demonstrate that the method achieves significant quality in detecting community structures in both computer-generated and real-world network data in near-linear time. Furthermore, the method could automatically determine the number of communities.

Key words: community detection, vertex similarity, affinity propagation, social network, clustering

1 Introduction

Vertices of many real-world networks, such as Internet, World Wide Web, biological and social networks, are often organized into communities or groups with dense connections within groups and sparse connections between groups. The vertices within a community are likely to share common properties and play similar roles within the graph. For instance, communities in a social network might correspond to groups of people with similar hobbies, while communities in a web graph might correspond to groups of websites with related topics, and communities in a scientific collaboration graph might correspond to groups of researchers sharing close research interests. The community structure is valuable and useful information for both commercial and scientific purposes. Hence, an efficient and cost-effective algorithm to detect and analyze community structures may be of considerable use in many fields.

In recent years, community detection has received an enormous amount of attention from the scientific communities, such as data mining, information retrieval and social network analysis. Various methods have been proposed for detecting communities. However, most proposed methods perform poorly in scalability. In other words, previous methods are not appropriate for analyzing large-scale networks either in space cost or time cost, while moderate-to-large networks are ubiquitous in real world.

Motivated by the idea that community detection is in nature a clustering problem on graphs, we investigate several metrics for measuring vertex similarities and transform a community detection problem into a clustering problem. We further adopt an recently-proposed effective and scalable clustering method, 'Affinity Propagation' (AP) [1, 2], for detecting community structures in networks. Affinity Propagation has been widely used ever since reported in *Science* Magazine in 2007. We demonstrate that the method is highly effective and efficient of detecting community structures in both computer-generated and real-world network data.

The contribution of this paper is that: (1) a novel method is proposed to transform a community detection problem to a clustering problem based on several similarity metrics of vertex; and (2) an appropriate, effective and scalable clustering method is selected and presented for detecting communities, which could automatically determine the number of communities and achieves significant performance.

2 Related Work

Traditional methods of detecting communities were mainly from Computer Science and Social Network Analysis. In Computer Science, the community detection task is usually called Graph Partitioning Problem. Two well-known algorithms are Kernighan-Lin Algorithm [3] and Spectral Bisection Method [4, 5]. These methods perform well when the number of communities in networks is given. However, in real-world networks, it is usually not possible for people to know the number of communities in advance.

In study of social networks, sociologists developed *hierarchical clustering* methods [6] for detecting communities in networks. Firstly, one should develop a measure of vertices similarity $s(i, j)$ between vertex pairs (i, j) based on the network structure. The method can be split into two strategies: agglomerative and divisive. For the first strategy, starting from an empty network of n vertices but no edges, edges are added between pairs of vertices in decreasing order of similarity. For the latter strategy, starting from the complete network, edges are removed in increasing order of similarity, until no edges remain. And the process could be terminated at any step and gets a correspond community structure of networks.

Traditional methods are somehow not ideal for general network analysis, many algorithms have been proposed in recent years. One of the most popular methods was a divisive method of hierarchical clustering, proposed by Girvan and Newman [7]. In this method, they proposed a new similarity measure called *edge betweenness*. The betweenness of an edge is defined as the number of shortest paths between vertex pairs that run along the edge in question, summed over all vertex pairs. Generally, the edge betweenness of vertex pairs within communities is less than the one between communities. For a network with n vertices and m edges, the betweenness for all edges can be calculated in time that goes as $O(mn)$. This calculation has to be repeated once for each edge when it is removed, therefore the entire Girvan-Newman method runs in worst-case time $O(m^2n)$, and for a sparse network it should approximately be $O(n^3)$.

Girvan-Newman method itself does not provide a measure to determine the best split of communities in a network. A most widely accepted measure, *modularity* [8–10], was proposed to measure the quality of a division of a network into groups or communities. Suppose we are given a candidate division with g communities, we define a $g \times g$ matrix e whose component e_{ij} is the fraction of edges in the original network that connect vertices in community i to those in community j. The sum of ith row $a_i = \Sigma_j e_{ij}$ indicates the fraction of edges connected to community i. The modularity Q is defined as [8]

$$Q = \Sigma_i (e_{ii} - a_i^2) = Tr(e) - \|e^2\| \qquad (1)$$

where $\|x\|$ indicates the sum of the elements of the matrix x. The modularity measures the fraction of the edges that fall within the given groups minus the expected such fraction if edges were distributed at random. Hence, we will get $Q = 0$ if the number of within-community edges is no more than random. And Q could reach the maximum value 1 when networks are of strong community structure. It is reported that the modularity values for real-world networks typically fall in the range from about 0.3 to 0.7 [8]. Based on this measure, a fast algorithm was proposed by greedily optimize the modularity values of agglomerative hieratical clustering [9] which takes worst-case running time $O((m+n)n)$ or $O(n^2)$ on a sparse network. A detailed review on community detection could be found in [11].

3 Community Detection by Affinity Propagation

3.1 Affinity Propagation

Affinity propagation(AP) is a new clustering method proposed by Frey, et al [1, 2] in *Science* Magazine in 2007. It takes *negative* real-valued similarities between pairs of data points as input and simultaneously considers all data points as potential exemplars. Two types of messages are exchanged between data points and a set of exemplars and corresponding clusters will gradually emerges. There are many advantages to use AP for community detections:

1. AP could find clusters with much lower error than other clustering methods, such as k-means method [12] and vertex substitution heuristic method [13].
2. The running time of AP scales linearly with the number of similarities, which is one-hundred less the amount of time compared to other popular clustering methods, especially for moderate-to-large problems [1, 2, 14, 15] and is also much less than most popular community detection methods such as Girvan-Newman method ($O(n^3)$) [7] and fast algorithm ($O(n^2)$) [9].

3. Rather than requiring pre-specified number of communities or borrowing external measures such as *modularity* to determine when to terminate, AP takes as input a real number, called *preference*, for each data point to quantify the likelihood it is chosen as exemplars. The number of identified exemplars would emerge from the message-passing procedure although influenced by the values of the input preferences.
4. AP identifies exemplars for each cluster or group which is relatively straightforward to the key player problem in social networks. This is an important byproduct of community detection by AP.
5. The method could be applied to problems where the similarities are not symmetric [i.e., $s(i,k) \neq s(k,i)$] and to problems where the similarities violate the triangle inequality [i.e., $s(i,k) < s(i,j) + s(j,k)$]. This corresponds with the common phenomenon in social network analysis, where an individual A knows B well does not indicates B knows A well, moreover, A knows B well and B knows C well does not indicate how much A knows C.

As indicated above, it is desirable to use AP for community detection which could find an appropriate split of communities in high efficiency. However, we could not adopt AP directly on graph, because the data only records the link information between vertices and does not supply vertex similarities. An appropriate similarity metric of vertex is crucial for transforming community detection to a clustering problem. In the following subsection we will elaborate how to measure the vertex similarities of a network and transform the problem.

3.2 Similarity Metrics of Vertex

Vertex similarity is an important network concept in social network analysis and data mining. The problem of quantifying similarity between vertices in a network has a long history and there are many perspectives to answer whether two vertices are similar. In a general community detection problem, we measure vertex similarities solely based on the structure of a network given only the pattern of edges between vertices in a network, which is usually called *structural similarity* [16].

In social network analysis, it is reasonable to consider that two individuals in a social network have something in common if they share many same friends. Hence, the most common approach in previous work focuses on *structural equivalence* [17], that is two vertices are considered structurally equivalent if they share many same network neighbors. Denoting N_i as the neighborhood of vertex i in a network, namely the vertices directly connected to i, the number of common friends of i and j is

$$s_{naive}(i,j) = |Ni \bigcap N_j| \tag{2}$$

where $|x|$ indicates the number of elements in set x. However, this similarity function is not satisfactory for it tends to take large values for vertices with high degree even if they have only a small fraction of neighbors. There are at least three previously proposed metrics eliminating the bias successfully [16, 18–20],

$$s_{Jaccard}(i,j) = \frac{|Ni \bigcap N_j|}{|N_i \bigcup N_j|} \tag{3}$$

$$s_{cosine}(i,j) = \frac{|Ni \bigcap N_j|}{\sqrt{|N_i||N_j|}} \tag{4}$$

$$s_{min}(i,j) = \frac{|Ni \bigcap N_j|}{min(|N_i|, |N_j|)} \tag{5}$$

All these three metrics could normalize the similarity value and get the maximum value 1 only when $N_i = N_j$. Notably, these similarity metrics of two vertices does not take the connectivity condition themselves into account, although it is apparent that two connected vertices are more similar than those do not connected directly no matter how much fraction of their neighbors are the same. In this paper we will alter these three metrics in consideration of connectivity condition. Take Jaccard metric for example, considering a *negative* real-valued similarities compulsorily required by AP, the altered version should be

$$s_{New\_Jaccard}(i,j) = \begin{cases} s_{Jaccard}(i,j) - 2 & \text{if } e(i,j) = 0 \\ s_{Jaccard}(i,j) - 1 & \text{if } e(i,j) = 1 \end{cases} \tag{6}$$

where $e(i,j) = 0$ indicates there is no edge between i and j and vice versa. Both 'cosine' and 'min' metrics are similar as stated above.

Besides based on common neighbors, we could measure vertex similarity in other perspective. Three reasonable metrics are defined as follows,

1. An immediate way is considering two vertices with edges are more similar there two with no edges. Therefore, the similarity of two vertices could be assigned a greater negative constant number if there is an edge between them and assigned a less negative constant number if there is no edge. That is,

$$s_{constant}(i,j) = \begin{cases} -w & \text{if } e(i,j) = 0 \\ -v & \text{if } e(i,j) = 1 \end{cases} \tag{7}$$

   where $0 < v < w$.
2. It is reasonable to consider two vertices are similar if there are short paths between them. Therefore, the similarity of two vertices is assigned as the negative shortest path length between them. That is

$$s_{path}(i,j) = -p(i,j) \tag{8}$$

   where $p(i,j)$ is the shortest path length between i and j in the network.
3. Edge betweenness measures the tendency of an edge to be within a community or between communities in a network. Therefore, the similarity of two vertices could be measured in terms of the edge betweenness between them. That is,

$$s_{betweenness}(i,j) = -b(i,j) \tag{9}$$

   where $b(i,j)$ is the betweenness value of the edge between i and j.

In this paper, we will use the above six vertex similarity metrics and try to figure out the most appropriate metric for the transformation process from community detection to clustering.

4 Experiment and Evaluation

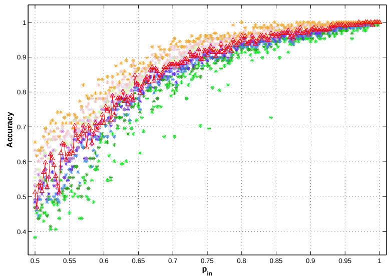
We present various tests of AP for community detection both on computer-generated graphs and real world social networks. The results of each test show that AP could reliably find community structures under an appropriate vertex similarity metric. In the following tests, AP is configured by setting the maximum number of iterations as $maxits = 2000$, the damping fact as $dampfact = 0.9$ which may be needed if oscillations occur, and if the estimated exemplars stay fixed for $convits = 200$ iterations AP terminates. As a prior, all vertices are equally suitable as exemplars and the preferences are set to a common value.

4.1 Computer-Generated Graphs

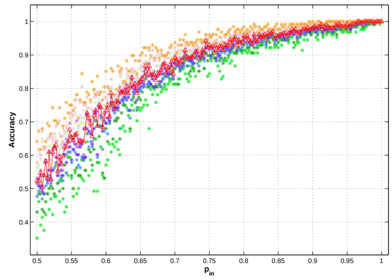
Firstly, we apply the method to a set of artificial, computer-generated graphs. Each graph is constructed with 128 vertices divided into four communities of 32 vertices each, and the expected average degree of each vertex equal to 16. Each edge is assigned between vertices independently at random, with probability p_{in} for vertices within the same community and p_{out} for vertices in different communities. We should set $p_{out} < p_{in}$ to ensure the community structure in the generated network.

Using AP for community detection of these generated graphs, we measure the fraction of vertices that were classified into their correct communities, as a function of the probability p_{in}. We switch the probability p_{in} from 0.5 to 1.0 stepped by 0.05. In each step, 10 networks are generated separately and fed to AP for community detection. In Fig. 1(a) - 1(e) we show the accuracies under five different vertex similarity metrics. For more discriminative display, we introduce various colors to identify different accuracy values of ten tests in each step, with 'orange' for maximum, 'green' for minimum and 'red' for medians of the ten values. And in Fig. 1(f) we compare the medians of accuracies of different metrics. The result of metric based on 'edge betweenness' is not plotted here because AP under this metric suffers severe numerical oscillation and is not able to terminate.

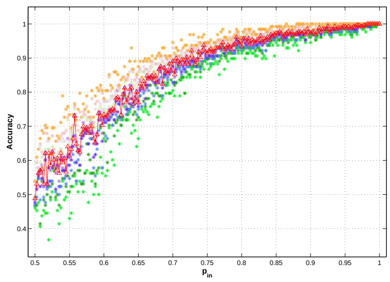
As Fig. 1(a) - 1(c) show, when $p_{in} > 0.7$, AP under the metrics of 'Jaccard', 'cosine' and 'min' performs nearly perfect and classify 90% or more of the vertices correctly. And even for the situation when each vertex has as many iter-community edges as intra-community ones, the method performs well and get the accuracy more than 50%. From Fig. 1(d) and 1(e), we found the metrics of 'constant' and 'shortest path length' are not satisfactory measures for similarities of vertex. However, they could still get high performance when p_{in} is high enough and the community structure is strong, and it is considerably appropriate to use them for community detection of large-scale sparse networks because vertices similarities under these metrics can be computed in limited time.



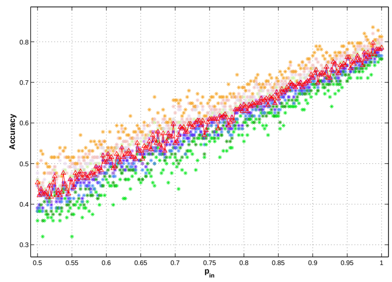
(a) Metric: Jaccard



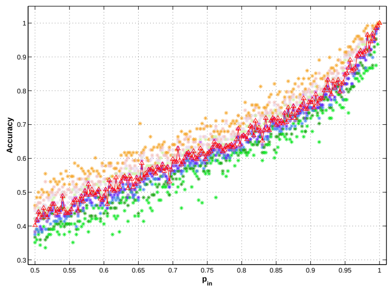
(b) Metric: cosine



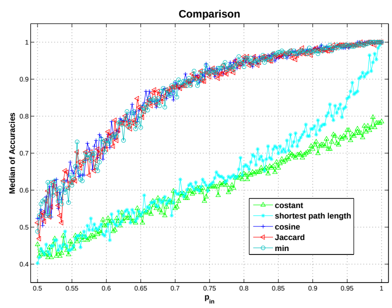
(c) Metric: min



(d) Metric: constant



(e) Metric: shortest path length



(f) Comparison

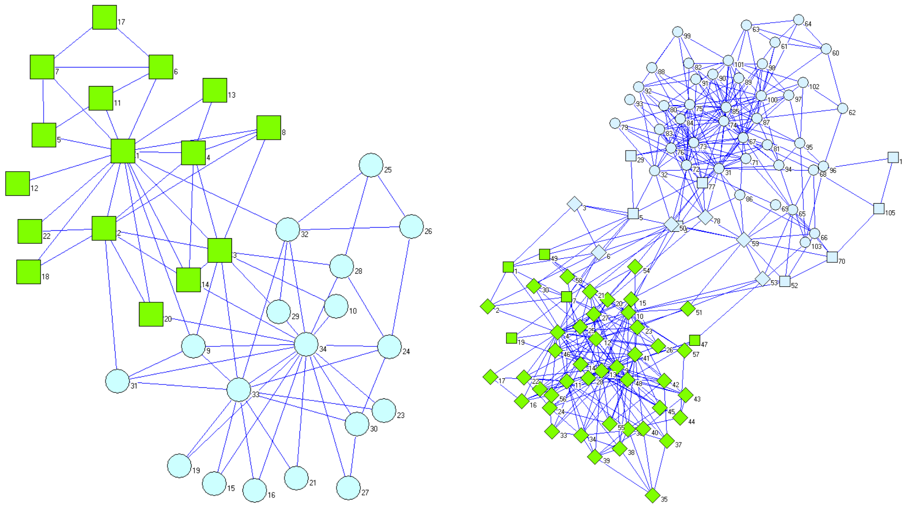
Fig. 1. Accuracies of AP for community detection of computer-generated networks under five vertex similarity metrics.

4.2 Real world social networks

Besides the test on computer-generated networks, it is also necessary to test the method on data from real-world networks as well. In this section we introduce some most popular real-world social network data sets to test and verify the performance of our method. People could access these network data via `http://www-personal.umich.edu/~mejn/netdata/`.

Zachary's Karate Club. The first of real world networks is from the well known karate club study of Zachary [21]. Zachary observed 34 members of a karate club over two years. During the period, a disagreement developed between the administrator of the club and the club's instructor. This resulted in the instructor leaving and starting a new club, taking about a half of the original club's members with him. Zachary constructed a network of friendships between members of the club. The network is shown in Fig. 2(a), and the shapes 'square' and 'circle' of vertices indicate two divisions, led separately by the instructor and the administrator represented by vertices 1 and 34. Our method under the metric 'min' as well as 'Jaccard' and 'cosine' splits the network into two communities with modularity value 0.3715, and the vertices of different communities are marked with different colors as shown in Fig. 2(a), which is perfectly matching with the real division the club members after the break-up and leaving. While the Girvan-Newman method [7] and fast algorithm [9] could also split the network in a right way, but they both classifies one vertex wrongly (vertex 10 and 3 separately). Moreover, the method finds the exemplars of each communities, 1 and 34, which are exactly the instructor and the administrator. In addition, our method could get a larger modularity 0.39 splitting the network into 4 communities. This indicates the modularity provides a useful quantitive measure of success for community detection, although it does not always completely fit the human observations on communities of network.

Books on US Politics. This is a network of 105 books on recent US politics sold by Amazon.com. Edges between books represent frequent co-purchasing of books by the same buyers, indicated by the 'customers who bought this book also bought these other books' feature on Amazon. The network data was compiled by V. Krebs which is still unpublished. This network is shown in Fig. 2(b), and the shapes 'circle', 'square' and 'diamond' of vertices indicate whether they are 'liberal', 'neutral' or 'conservative'. These alignments were assigned by Mark Newman based on a reading of the descriptions and reviews of the books posted on Amazon. Our method splitted the network into two communities with modularity value 0.44. These books of different communities are marked with different colors as shown in Fig. 2(b) which is almost perfectly identical with the actual division of them according to political orientations. The exemplars found by the method are 13 and 31. The first book is '*Off with Their Heads: Traitors, Crooks & Obstructionists in American Politics, Media & Business*', written by Dick Morris, an American political author who became an adviser to the Bill Clinton administration after Clinton was elected president in 1992. And the latter is '*The Price of Loyalty: George W. Bush, the White House, and the Education of Paul OŃeill*', written by Ron Suskind, an American investigative journalist and author, who won the Pulitzer Prize for Feature Writing in 1995. Both of them are best sellers in Amazon.com.



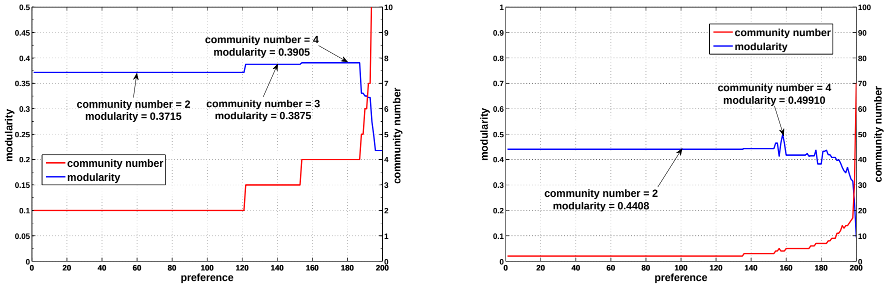
(a) Zachary's Karate Club (b) Books on US politics

Fig. 2. Communities of networks.

In Fig. 3 we demonstrate the influence of input preferences on the number of found communities. Firstly, we traversal the vertex similarities to determine the minimum and maximum values of preferences and run AP under 200 different preferences uniformly distributed within the boundary. Although it could not always find communities to achieve maximum modularity, we could get an approximately optimal result in a wide range of preference values.

Two more networks are listed as follows. **The Bottlenose Dolphin Social Network** is consist of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand [22]. Our method could also find the natural split of two communities with modularity 0.38, but get a more large modularity value 0.50 when split into 5 sub-groups. **The Network of American College Football Teams** represents the schedule of Division I Games for the 2000 season [7]. 115 vertices in the network represents football teams and 616 edges represent games between the two teams they link together. These teams are usually divided into conferences containing around 8-12 teams each and games are more frequent between members of the same conference. On average, teams played about 7 intra-conference games and 4 inter-conference games and the community structure is strong. Our method could find a split of 10 communities with modularity 0.54, which is near the real number 12 of conferences.

The vertex preferences were set to a common value in the above experiments. In fact we could speed up the convergence by setting the preferences to different

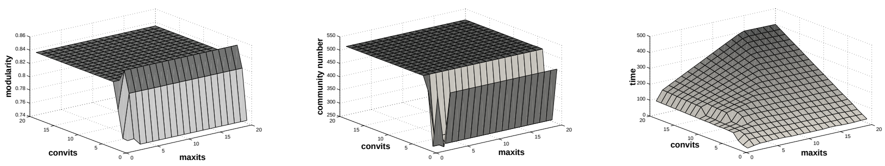


(a) Zachary's Karate Club (b) Books on US Politics

Fig. 3. Community detection of networks.

values according to some prior knowledge such as the degrees of vertices. A common sense is that the vertices with more degrees are more likely to be exemplars. Take *Zachary's Karate Club* for example, we set the preference value of the vertex with maximum degree to p_{max}, the one with minimum degree to p_{min}, and the preferences of other vertices are assigned to a real number in (p_{min}, p_{max}) based on linear interpolation. Then AP could get the convergence result only after 3 iterations, which is much faster than previous preference setting method with 15 iterations. For *Books on US Politics*, the results are 43 vs. 73. The new preference setting method is still too naive to achieve optimal modularity values, which could only get modularity 0.372 and 0.369 for the two networks, slightly worse than optimal result. Whereas it might be a practical choice for finding communities of a large-scale network.

From the above four real-world networks, it is clear that AP is quite capable of extracting community structures from networks under appropriate vertex similarity metrics. Furthermore, we investigate the speed and efficiency of AP on a moderate network, the **Co-authorship Network of scientists on Network Theory**. The network records the collaborations of $1,589$ scientists working on network theory, compiled by Newman mainly from the bibliographies of two review articles on networks [23, 24]. We ran a group of experiments based on *Jaccard* metric, where the parameter $maxits$ is varied within $[100, 2000]$ stepped by 100 and given $maxits$ the parameter $convits$ is varied within $(0, maxits]$ stepped by $maxits/20$, on a PC with $1.5G$ memory and $2.66GHz$ CPU, and the modularity value, community number and elapsed time were separately plotted in Fig. 4. The plot illustrates that previous configuration of AP is actually over-conservative, and we got the maximum modularity 0.8522 by extracting 280 communities from the network under the parameters $maxits = 300$ and $convits = 30$ within 64 seconds, which is fast enough considering the fact that AP was implemented by Matlab in a naive way with no optimization. In fact, we took only 8 seconds to get an acceptable modularity 0.7536 with 451 communities when we set $maxits = 100$ and $convits = 5$, which is relatively fast for solving practical problems.



(a) Modularity (b) Community Number (c) Time

Fig. 4. Community Detections on Co-authorship Network.

5 Conclusions and Future Work

In this paper we have described a method combined vertex similarity metrics with Affinity Propagation for detecting communities from networks, which is considerable efficient over previous popular methods. Moreover, in contrast to previous methods which need pre-specified communities number or external measures as stop criterion, the method could automatically determine the number of communities and select exemplars or leaders for each community.

Some possible future work may includes: (1) Optimize the computation process of vertex similarities and further improve the efficiency; and (2) find more practical method for setting preference values to speed up convergence of AP with no influence on performance; and (3) extend the method to more complicated networks with weights, directions and nested or overlapped community structures.

6 Acknowledgements

References

1. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science **315**(5814) (2007) 972–976
2. Dueck, D., Frey, B.J.: Non-metric affinity propagation for unsupervised image categorization. In: ICCV 2007. ICCV 2007 (2007) 1–8
3. Kernighan, B.W., Lin, S.: An efficient heuristic procedure for partitioning graphs. Bell System Technical Journal **49**(2) (1970) 291–307
4. Fiedler, M.: A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. Czechoslovak Mathematical Journal **23**(298) (1973) 619–633
5. Pothen, A., Simon, H.D., Liou, K.P.: Partitioning sparse matrices with eigenvectors of graphs. Siam Journal On Matrix Analysis And Applications **11**(3) (1990) 430–452
6. Scott, J.: Social Network Analysis: A Handbook. Sage Publications (2000)
7. Girvan, M., Newman, M.E.: Community structure in social and biological networks. Proceedings Of The National Academy Of Sciences Of The United States Of America **99**(12) (2002) 7821–7826

8. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E **69**(2) (2004)
9. Newman, M.E.: Fast algorithm for detecting community structure in networks. Physical Review E **69**(6) (2004) 066133
10. Newman, M.E.: Modularity and community structure in networks. Proceedings Of The National Academy Of Sciences Of The United States Of America **103**(23) (2006) 8577–8582
11. Newman, M.E.: Detecting community structure in networks. European Physical Journal B **38**(2) (2004) 321–330
12. Macqueen, J.B.: Some methods of classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability. (1967) 281–297
13. Teitz, M.B., Bart, P.: Heuristic methods for estimating the generalized vertex median of a weighted graph. Operations Research **16**(5) (1968) 955–961
14. Brusco, M.J., Kohn, H.F.: Comment on "clustering by passing messages between data points". Science **319**(5864) (2008) 726c
15. Frey, B.J., Dueck, D.: Response to comment on "clustering by passing messages between data points". Science **319**(5864) (2008) 726d
16. Leicht, E.A., Holme, P., Newman, M.E.: Vertex similarity in networks. Physical Review E **73**(2) (2006) 026120
17. Lorrain, F., White, H.: Structural equivalence of individuals in social networks. Journal of Mathematical Sociology **1** (1971) 49–80
18. Jaccard, P.: Distribution de la flore alpine dans le bassin des dranses et dans quelques rgions voisines. Bulletin del la Socit Vaudoise des Sciences Naturelles **37** (1901) 241–272
19. Salton, G.: Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley, Reading, MA (1989)
20. Ravasz, E., Somera, A.L., Mongru, D.A., Oltvai, Z.N., Barabasi, A.L.: Hierarchical Organization of Modularity in Metabolic Networks. Science **297**(5586) (2002) 1551–1555
21. Zachary, W.W.: An information flow model for conflict and fission in small groups. Journal of Anthropological Research **33**(4) (1977) 452–473
22. Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology **54** (2003) 396–405
23. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: Structure and dynamics. Physics Reports **424**(4-5) (2006) 175–308
24. Newman, M.E.: The structure and function of complex networks. SIAM Review **45**(2) (2003) 167–256