1-1-2009

Identifying Cohesive Local Community Structures in Networks

Jennifer Xu
Bentley University, jxu@bentley.edu

IDENTIFYING COHESIVE LOCAL COMMUNITY STRUCTURE IN NETWORKS

Research-in-Progress

Jennifer Xu
Bentley University
Waltham, MA, U.S.A.
jxu@bentley.edu

Abstract

Identifying community structure in networks is an important topic in data mining research. One of the challenges is to find local communities without requiring the global knowledge of the entire network. Exiting techniques have several limitations. First, there is no widely accepted definition for community. Second, these algorithms either lack good stopping criteria or depend on predefined threshold parameters. In this research I propose a local cohesion based algorithm to identify local communities in networks. This algorithm is grounded on the widely accepted group cohesion definition in social network analysis research. The algorithm is self-contained and does not depend on predefined threshold parameter to terminate the identification process. The evaluation results show that the proposed algorithm is more effective than the benchmark algorithm and can identify meaningful local communities in very large networks such as the Amazon co-purchasing network.

Keywords: Network, community identification, local community, cohesion

Introduction

Networks are prevalent in nature and society. Examples include social networks, the Internet (Faloutsos et al. 1999), the World Wide Web (Albert et al. 1999), citation networks (Hajra and Sen 2005), electric power grid (Watts and Strogatz 1998), and biological networks (Dorogovtsev and Mendes 2003). Regardless of its context, a network is often modeled as a graph with a set of nodes (entities) connected by links (relationships). There has been a growing interest in the study of networks partially due to the Web and, more recently, the increasing popularity of online social networking sites such as *Facebook* and *MySpace*. As a result, network mining has attracted great attention in the data mining research area (Backstrom et al. 2006; Cook and Holder 2000; Goldberg et al. 2008).

Community structure identification is one of the widely studied topics of network mining research. In general, a community is a tightly-knit sub-graph in a network in which the within-group links are stronger or denser than between-group links (Wasserman and Faust 1994). Finding community structure in networks has important empirical implications. For example, identifying Web communities can be very helpful for designing focused crawlers, developing Web portals, and improving search engine performance (Flake et al. 2002; Kumar et al. 1999). In business, companies can take the viral marketing strategy to introduce their new products or services to customers and their friend circles (Porter and Golan 2006). Online stores may find groups of products that are often purchased together and use the information to design product bundles or make better purchase recommendations to customers (Schafer et al. 1999).

A number of techniques have been proposed to identify communities in networks (Clauset et al. 2004; Girvan and Newman 2002; Newman and Girvan 2004; Xu et al. 2007). However, most of these techniques are global approaches in that they require the complete knowledge of the entire network structure. For networks such as the Web, which is extremely large and fast evolving and whose complete structure is almost impossible to obtain, global approaches will have much difficulty to partition the networks into communities. To address this problem, researchers have proposed several methods to detect communities based only on the local structure of a network (Bagrow 2008; Clauset 2005; Luo et al. 2006). However, these methods also have some problems. First of all, there is no commonly accepted definition for community (Luo et al. 2006; Palla et al. 2005). As a result, it is difficult to assess and compare the quality of the communities identified by different algorithms. Moreover, these algorithms either lack good stopping criteria or must depend on a predefined threshold to terminate the identification process (Bagrow 2008). It is thus hard to tell when the local community has been found. In this research I propose a local community identification algorithm based on the *cohesion* measure, which is a widely accepted community definition and quality metric in social network analysis research. This method is self-contained and the stopping criterion does not depend on any predefined threshold.

The remainder of this paper is organized as follows. The second section reviews existing local community identification methods. The third section presents my local cohesion measure and the algorithm. Section 4 reports results from the evaluations, in which I test my algorithm's effectiveness using computer generated networks and the Amazon co-purchasing network that contains more than 500,000 nodes and three million links. The last section concludes the paper and lays out my plans for future research.

Literature Review

In this section I review state-of-the-art techniques for identifying local communities in networks. I focus my attention on the community definition and the stopping criteria of the detection algorithms.

Definition of Community

A network is usually represented by a graph $G = \{V, E\}$, in which a set of n nodes (vertexes) are connected by m links (edges), where $n = |V|$ and $m = |E|$. I here focus on undirected, unweighted graphs. Generally speaking, the local community identification problem is to find a sub-graph $C \in G$ for a starting node such that nodes in the community have more links with nodes inside the community than with nodes outside the community.

Different community structure measures have been proposed. Newman (2004) proposes a measure called network *modularity*, $Q = \frac{1}{2m}\sum_{i,j}(A_{ij} - \frac{k_i k_j}{2m})\delta(C_i, C_j)$, where A is the adjacency matrix representing the graph with a_{ij} indicating if the node v_i and node v_j are connected; k_i and k_j are v_i's and v_j's degrees, respectively (the degree of a node is defined as the number of links it has); the function $\delta(C_i,C_j)$ = 1 if v_i and v_j are in the same community and 0 otherwise. Network modularity and its variation (Muff et al. 2005) are used to measure if a network has a significant community structure. It has been found that when the value of Q is greater than 0.3 for a network, significant communities exist in the network (Clauset et al. 2004; Newman 2004). Modularity-based algorithms have been used to partition networks into meaningful communities in various applications. However, this modularity measure is not a definition for community *per se*. Moreover, it is a global metric because it requires the adjacency matrix, A, and the total number of links in the entire network, m.

To define community, Flake *et al.* (2000) propose that a sub-graph C is a community if each node in C connects with more nodes from within C than with nodes from outside C. That is, for any node $v_i \in C, k_i^{in}(C) > k_i^{out}(C)$, where k_i^{in} and k_i^{out} are the node v_i's internal degree and external degree, respectively. This definition has been used to define and identify Web communities that consist of Web pages sharing similar topics (Flake et al. 2000; Flake et al. 2002). The problem with this definition is that it is too strong and strict. In many situations a tightly-knit sub-graph that exhibits a clear community characteristic may not be qualified as a community. In Figure 1(a), for example, sub-graphs II and III are not communities by definition because each has at least one node that does not have a larger internal degree than external degree. Only sub-graph I can be counted as a community although it actually is a rather loosely-knit one.

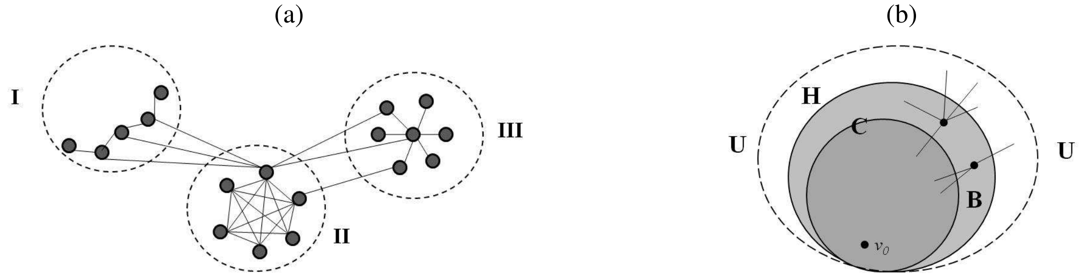

Figure 1. (a) Illustrations of communities. (b) The illustration of a local community and its external area.

Treating the definition by Flake *et al.* (2000) as *strong community*, Radicchi *et al.* (2004) propose a *weak community* definition. In this definition, a sub-graph is a community if its total internal degree is greater than its total external degree: $\sum k_i^{in}(C) > \sum k_i^{out}(C), \forall v_i \in C$. However, this definition counts each internal link twice because $\sum k_i^{in}(C) = 2m^{in}(C)$, where m^{in} is the total number of internal links within C. As a result, even a very loosely-knit sub-graph may be qualified as a community. Recognizing this drawback, Luo *et al.* (2006) revise the definition by requiring the total number of internal links, m^{in}, be greater than the total external degree. By this definition, all the three sub-graphs in Figure 1(a) are weak communities.

Both the strong and weak community definitions have limitations. They sometimes identify poorly-knit sub-graphs that can hardly be considered as communities. The sub-graph I in Figure 1(a), for example, is not a community by common sense as it is a very loose structure with each node connecting with only one or two other nodes. Such a structure, however, is not only a weak but also a strong community by definition. This problem needs to be addressed by a better definition of community.

Local Community Identification Algorithms

Many global community detection methods exist for partitioning a network into communities including the G-N algorithm (Girvan and Newman 2002), the global modularity based methods (Clauset et al. 2004; Newman 2004), and the SCAN algorithm (Xu et al. 2007), among many others. Some of these algorithms are fairly effective and efficient and can partition very large networks into meaningful clusters. However, in the situation where the global structure of a network is unknown, these methods would not be applicable.

Researchers thus have started working on methods that can identify communities based only on local structural information. Most of these algorithms work in a greedy, one-node-at-a-time manner. As illustrated in Figure 1(b), from the perspective of the starting node v_0, a graph is divided into several regions: the community C, the neighborhood H, and the unknown part U. Initially, C contains only v_0 and H is empty. The algorithm then explores U by visiting v_0's immediate neighbors and adds them to H. The algorithm selects one of the neighbors in H and merges it into the community. The neighborhood H is updated by adding the selected node's neighbors. At the second step another node is selected from H to be merged into C. This process iterates until certain stopping criteria are satisfied and the local community surrounding the starting node is considered to be found. Such methods are very similar to the breadth-first-search graph traversal algorithm that is often used in Web crawlers or spiders to fetch Web pages by following hyperlinks (Bagrow and Bollt 2005; Clauset 2005).

These local algorithms differ in the measures used to select the node for merging at each step and in the stopping criteria. Clauset (2005) proposes a local modularity-based algorithm which considers a boundary set B within the community C (Figure 2(b)). Each node in the boundary has at least one neighbor in H. The *local modularity R* is defined as $R = \frac{m_B^{in}}{m_B}$, where $m_B{}^{in}$ is the number of internal links between boundary nodes and their internal neighbors in C, and m_B is the total number of links of boundary nodes including those links that connect boundary nodes with their external neighbors in H. The R actually measures the "sharpness" of the community boundary. At each step, the algorithm selects the node that causes the largest increase or least decrease in the value of R and the process stops when a predefined number of nodes have been visited.

The local modularity measure proposed by Luo *et al.* (2006) is directly derived from their community definition in which $M = \frac{\frac{1}{2}\Sigma k_i^{in}(C)}{\Sigma k_i^{out}(C)} = \frac{m^{in}(C)}{m^{out}(C)}$. A sub-graph is a community if $M > 1$. Similar to the R algorithm, the M algorithm chooses a node that can provide the largest increase in M at each step and it terminates when M ceases to increase. The stopping criterion of this method does not depend on a predefined threshold value. However, the sub-graphs identified are all weak communities by definition and may not be meaningful.

Bagrow and Bollt (2005) propose an *l*-shell algorithm which uses the emerging degree and total emerging degree to select a node at each step. The problem with this algorithm is that its stopping criterion depends on an arbitrary parameter for the emerging degree. Bagrow (2008) later proposes a node *outwardness* measure $\Omega_i = \frac{1}{k_i}(k_i^{out} - k_i^{in})$ that directly compares the number of internal and external links a node has and selects the node with the least outwardness to merge into the community. This method is simple and straightforward. However, it lacks a community quality measure and does not provide a stopping criterion. To determine when to stop for a meaningful community, one has to apply a Trailing Least-Square approach to fit a polynomial function to the plotted m^{out} to identify the cusp points that indicate the border of a community.

In summary, existing local community identification methods can detect interesting sub-graphs in networks. However, there is not a widely accepted community definition. In addition, the algorithms either lack a good stopping criterion or have to depend on a predefined threshold value. In the next section I present the local cohesion based algorithm that is grounded on a commonly accepted community definition and does not depend on a threshold parameter.

Local Cohesion Based Community Identification Algorithm

I propose a local cohesion based algorithm to identify local community structure in a network. Both the strong and weak community definitions have problems because they only compare the internal degree and external degree for a community. However, what really matters is not the difference between degrees but that between the internal and external densities of a community. In social network analysis, the *density* of a graph is defined as $d = \frac{m}{\frac{1}{2}n(n-1)}$, where ½$n$($n$-1) is the number of possible links in a graph of size n, and m is the number of actual links (Wasserman and Faust 1994). The value of density ranges from 0 to 1. When the density is 1 the graph becomes a clique in which every node is connected with everyone else.

In social network analysis a measure called *group cohesion* has long been used to determine if a sub-graph can be considered as a community or not. This cohesion measure for a sub-graph C is defined as (Wasserman and Faust 1994):

$$Cohesion(C) = \frac{d^{in}(C)}{d^{out}(C)} = \frac{\dfrac{2m^{in}(C)}{n_c(n_c-1)}}{\dfrac{m^{out}(C)}{n_c(n-n_c)}} = 2\frac{(n-n_c)m^{in}(C)}{(n_c-1)m^{out}(C)} \qquad (1),$$

where n_c = |*C*|, and $n_c(n\text{-}n_c)$ is the number of possible links between the community members and the rest of the network. A sub-graph is a cohesive community if its internal link density is greater than its external link density, or *cohesion* > 1. Note that for a totally isolated community with no external links ($m^{out} = 0$), the cohesion value is undefined. This definition provides a natural way to measure the quality of the identified community. The higher the cohesion value, the tighter the community is inside than outside. A cohesive community can be found by maximizing the cohesion value, thereby maximizing the internal density and minimizing the external density at the same time.

However, this measure cannot be used directly to identify local communities because of two reasons. First, it requires the global knowledge of the network size, *n*, which is often unknown. Second, for very large networks such as the Web, *n* usually is extremely large. As a result, any sub-graph will be considered as a cohesive community because of its highly inflated cohesion value.

I thus propose a *local cohesion* measure that does not need the value of *n*. I replace $(n\text{-}n_c)$ with n_h in equation (1), where n_h = |*H*|. In other words, the possible number of external links for *C* is calculated based on the number of links between *C* and the neighborhood *H* outside the community. Intuitively, from the perspective of a community such as a person's friend circle, what are relevant are just a limited number of outside friends but not the rest of the whole world. This local cohesion measure no longer needs the global knowledge of the size of the entire network but only depends on the nodes in the neighborhood *H* at each step. In Figure 1(a), the local cohesion values for the three sub-graphs are 2/3, 84/15, and 4/3, respectively. Therefore, sub-graph I is not a community by this definition. Sub-graphs II and III are cohesive communities and community II is more cohesive than III. This cohesion measure can be easily applied to weighted networks as well.

Based on this definition, a local community can be found by merging the node that provides the largest increase in the cohesion value at each step. The process terminates when the cohesion value starts to decrease. No predefined threshold value is needed.

This local cohesion based method has its own limitations, however. I have found that like other local algorithms, my method is sensitive to the selection of the starting node (Bagrow 2008; Clauset 2005; Luo et al. 2006). When the starting node is deep in the middle of the community, the resulting community often is well defined and separated from the rest of the graph. However, when the starting node sits in the boundary and also connects with nodes from a different community, the algorithm may accidentally identify the other community as the one surrounding the starting node. To address this problem I add a "pruning" phase to the algorithm after the merging phase. In the pruning phase, each node is temporarily removed from the community. If the removal causes the cohesion value to increase it means that this node should not be a member of the community, and thus is permanently removed. After finishing examining all the nodes, the algorithm checks the community to see if the starting node is still contained in the community. If not, it is determined that there is no local community for the starting node. The pruning phase may remove those low degree leaf nodes or outliers (Bagrow 2008; Xu et al. 2007), whose removal will cause the internal density, and cohesion accordingly, to increase. In the situation where the starting node is a boundary node, the pruning phase will detect that the identified sub-graph is not the community for the starting node because it connects with only a few nodes in the sub-graph and appears as a leaf node. A similar node deletion phase is also used in the M algorithm (Luo et al. 2006).

The pseudo code of the algorithm is presented in Figure 2. The algorithm takes a node v_0 as the starting point and outputs the community *C* for v_0 or no community. The new cohesion for a candidate node v_i with the internal degree of k_i^{in} and external degree of k_i^{out} can be quickly calculated by:

$$Cohesion(C)_i = 2\frac{n_h{}' [m^{in}(C) + k_i^{in}]}{n_c [m^{out}(C) - k_i^{in} + k_i^{out}]} \qquad (2),$$

where n_h' is the size of the new neighborhood if merging v_i into *C*. Note the internal and external degrees of v_i and its neighbors must be updated after each merge.

The running time for the algorithm is $O(\bar{k}_c^2 n_c)$, where $\bar{k}_c$ is the average degree of nodes in *C*. As noted in (Clauset 2005), for applications such as local Web community identification, most of the running time will be spent on the retrieval of Web pages because the crawler has to connect to the remote servers. Omitting this retrieval requirement, the algorithm is linear in the size of the community $O(n_c)$.

```
C = {v0}
```

```
add all neighbors of $v_0$ to H

/*The merging phase*/
while H is not empty
  for each $v_i$ in H do
    calculate $\Delta$Cohesion if $v_i$ is merged into C
  end for

  find $v_j$ that provides the maximal positive $\Delta$Cohesion
  if no such $v_j$ is found
    exit the while loop
  else
    add $v_j$ to C
    add all external neighbors of $v_j$ to H
    update Cohesion
    update the internal and external degrees of $v_j$ and its neighbors
end while

/*The pruning phase*/
for each $v_i$ in C do
  calculate $\Delta$Cohesion if $v_i$ is removed from C
  if $\Delta$Cohesion > 0
    remove $v_i$ from C and update Cohesion
    if $v_i$ = $v_0$
      print "No local community found for $v_0$."
      exit the for loop
end for
```

Figure 2. The pseudo code for the local cohesion based algorithm.

Evaluation Results

To evaluate the performance of my algorithm I used two different datasets. The first was a set of synthetic networks with known community structure. The second was the Amazon co-purchasing network.

Synthetic Networks

It has become a standard practice to test community identification techniques using synthetic networks, which can be manipulated to control how well separated communities are (Clauset 2005; Girvan and Newman 2002; Newman 2004; Radicchi et al. 2004). These computer-generated random networks typically have 128 nodes that are divided into four groups of equal sizes. The average degree is set to be 16. One parameter p^{out} (or p^{in}) can be used to control the community structure, where p^{out} is the probability that a community member is connected with an external node and p^{in} is the probability of internal links. Alternatively, the external degree k^{out} can also be used as $k^{out} = p^{out}(n\text{-}n_c)$. When $p^{out} = 0$, the graph consists of four isolated communities without any external links. As p^{out} increases, external links between communities start to appear and increase in number, and the boundaries of communities become weaker. When $p^{out} = p^{in} = 0.125$ ($k^{out} = 12$, $k^{in} = 4$), the graph becomes totally random and no community exists. Figure 3 gives three example networks with different p^{out} values.

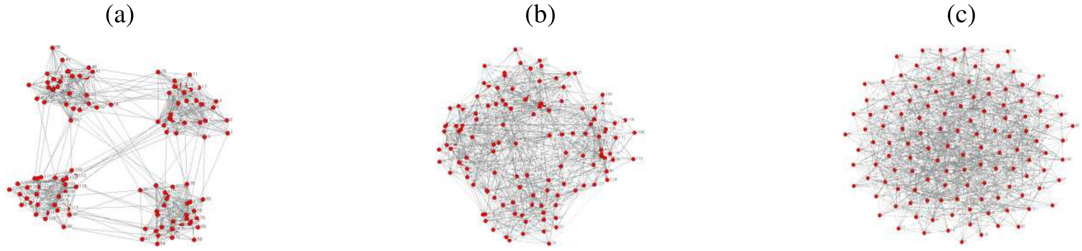

Figure 3. Illustrations of three computer generated networks with different p^{out} values: (a) $p^{out} = 0.01$ (k^{out} = 1); (b) $p^{out} = 0.05$ ($k^{out} = 5$); (c) $p^{out} = 0.125$ ($k^{out} = 12$).

Since the community structure of a synthetic network is known, the effectiveness of a community identification algorithm can be objectively measured by *precision*, *recall*, and *F*. These three metrics have been used for evaluating clustering effectiveness (Roussinov and Chen 1999). Given a starting node v_i, the true community that v_i belongs to is denoted as C_i^{true} and the community identified by the algorithm is denoted as C_i^{alg}. The metrics then are defined as:

$$Precision_i = \frac{|C_i^{true} \cap C_i^{alg}|}{|C_i^{alg}|} \quad (3.1), \qquad Recall_i = \frac{|C_i^{true} \cap C_i^{alg}|}{|C_i^{true}|} \quad (3.2), \qquad F_i = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3.3).$$

For each p^{out} value I generated 30 random graphs. For each graph, I selected each of the 128 nodes as the starting node. The *precision*, *recall*, and *F* values were then averaged over all the nodes for all the 30 graphs. I plotted these averages against k^{out} in Figure 4. It can be seen that as the number of external links increases, it becomes harder to identify the community structure. When the boundary is rather weak, say $k^{out} = 8$, the precision drops to around 0.35 and the recall is only about 0.2.

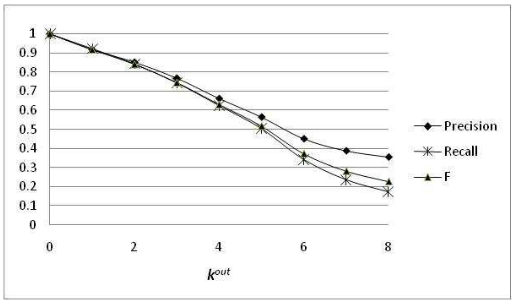

Figure 4. The effectiveness of the local cohesion based algorithm.

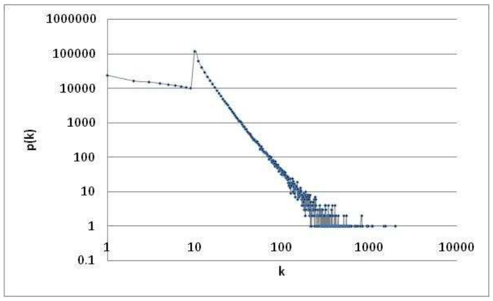

Figure 5. The log-log plot of the degree distribution of the Amazon co-purchasing network.

I also compared the performance with Clauset's R algorithm (Clauset 2005) as the benchmark. This R algorithm needs a predefined number of nodes *g* to terminate the detection process. Since the size of the communities was supposed to be unknown to the algorithm, I set *g* to be the size of the graph. It turned out that the R algorithm always output the entire graph, causing a perfect recall of 1.0 and a poor precision of 0.25 ($F = 0.40$). My algorithm, in contrast, did not require a predefined threshold value and could determine when to stop by checking the value of the cohesion. As a result, its effectiveness was better than the R algorithm.

Amazon Co-Purchasing Network

I applied my algorithm to a real dataset about the recommender network from Amazon.com. The data were collected in 2006 and was used in the study by Luo *et al.* (2006). A similar but older dataset was used in the evaluation by Clauset (2005). Nodes in the network are items like books and digital media, and links exist between items that are frequently purchased together by customers. Amazon.com uses such co-purchasing data to recommend products to customers by two functions on its Web site: "Customers Who Bought This Item Also Bought" and "Frequently Bought Together." The network has n = 585,283 nodes and m = 3,448,747 links. The average degree is 11.78 and the highest degree is 1989. The overall density is very low with $d = 2.01e^{-5}$. However, local densities can be much higher indicating significant local community structures. Figure 5 presents the log-log scatter plot of the network's degree distribution. The degree distribution $p(k)$ of a network is defined as the probability that an arbitrary node has exactly k links. This co-purchasing network has an interesting two-regime power-law distribution (Barabási et al. 2002), which is worth further studying.

I chose three qualitatively different digital media as the starting nodes: the compact disc *Alegria* by Cirque de Soleil, the DVD *Cirque Reinvente* by Cirque de Soleil, and the compact disc *Sunday School Songs* by Cedarmont Kids. The degrees of these three items are 11, 15, and 17, respectively. These items were used in the evaluation in ref. (Luo et al. 2006) and the first item was also used in ref. (Clauset 2005).

I found local communities for all the three items. Their sizes are 24, 24, and 21; the densities are 0.384, 0.384, and 0.595; and the corresponding cohesion values are 8.31, 9.66, and 8.78, respectively. I compared my results with those in ref. (Luo et al. 2006). I did not compare the first community with that from ref. (Clauset 2005) as the items were not labeled in that paper. I generated visualizations for these communities using a network drawing tool called *NetDraw* (Analytic Technologies 2009). Due to the space limit, I present only two communities in Figure 6, in which nodes represent the items and the links are co-purchasing relationships.

Among the 24 items in the community for *Algeria*, 23 were also found in its community identified by Luo *et al.* (2006). I found that all these 23 items were by Cirque du Soleil. The two items that were missed from my community were *Xotika* by Rene Dupere and *Cirque Du Soleil* by Cirque du Soleil. Clearly, *Xotika* should not be in the community as it had only one link with the rest of the community and was mistakenly included by Luo's M algorithm. The other one, however, should be a member of the community. My algorithm missed it resulting in a false negative error. The extra item that was identified as a member by my algorithm is *The Best of Meco* by Meco (#B000001EZW). It seemed to be an outlier to the community since it was not by Cirque du Soleil. It was a false positive error of my algorithm.

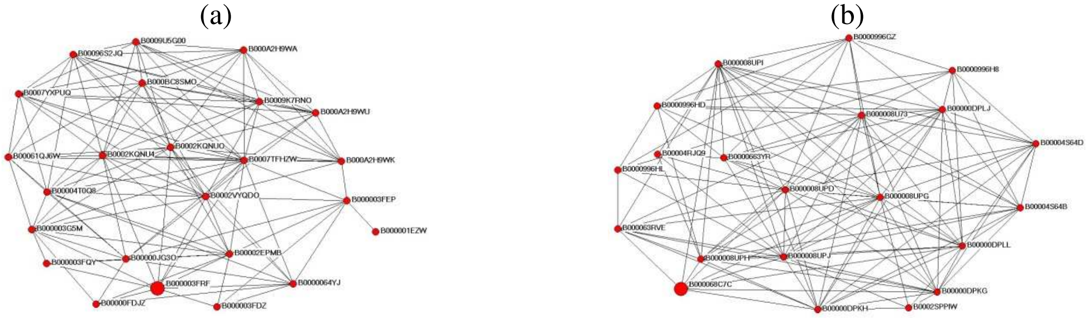

Figure 6. Visualizations of two local communities in the Amazon co-purchasing network. The starting nodes are highlighted by their larger sizes. (a) The community for *Alegria* by Cirque de Soleil. (b) The community for *Sunday School Songs* by Cedarmont Kids.

The community of *Cirque Reinvente* contained all the 18 DVDs by Cirque Du Soleil and two books about Cirque Du Soleil, which were also shown in the community in ref. (Luo et al. 2006). The four items that were in ours but not in Luo's were DVDs or compact discs by Yanni *et al.* and Kenny G. They were outliers resulted from false positive errors of my algorithm. My algorithm, however, correctly excluded two other items that resulted from false positive errors by the M algorithm.

Because the items were not labeled in ref. (Luo et al. 2006) for the community for the compact disc *Sunday School Songs*, I could not compare my result with theirs. I thus manually checked the 21 items in the identified community and found that they all were compact discs of children's songs by Cedarmont Kids or various artists.

Conclusion

Community structure has important implications for many network applications. The problem of community identification faces several challenges, one of which is to find local communities without requiring the global knowledge of the network. Exiting local community identification algorithms have limitations. For example, there is no widely accepted definition for community. Moreover, these algorithms either lack a good stopping criterion or depend on predefined threshold parameters. In this research I propose a local cohesion based, self-contained algorithm to identify local community structure in networks. This algorithm is grounded on the widely accepted cohesion definition in sociology and can automatically determine when to terminate without depending on a threshold parameter. The evaluation results show that the proposed algorithm is more effective than the benchmark algorithm and can identify meaningful local communities in very large networks such as product co-purchasing networks. In addition, the proposed algorithm is efficient as it is linear in the size of the local community.

As a research-in-progress study my research has several limitations, however. Foremost, although the algorithm is effective for well-separated communities, the effectiveness drops as the community boundary becomes more blurred. In addition, I compared my algorithm's performance with only one benchmark algorithm and evaluated it using only one real dataset with a few starting nodes. The method needs to be tested more thoroughly with more real datasets against different benchmark algorithms.

My future research will focus on the improvement of the effectiveness of the algorithm. I will seek ways to enhance the precision, recall, and accuracy for communities that are not well separated. In addition, I will also improve the algorithm's robustness to outliers and compare my algorithms to more benchmark methods.

Acknowledgements

I thank Eric Promislow, Feng Luo, and James Wang for providing the Amazon co-purchasing data.

References

Albert, R., Jeong, H., and Barabási, A.-L. "Diameter of the World-Wide Web," *Nature* (401), 1999, pp. 130-131.

Analytic Technologies, "NetDraw: Network visualization," http://www.analytictech.com/netdraw.htm, accessed on May 2, 2009.

Backstrom, L., Huttenlocher, D., Kleinberg, J., and Lan, X. "Group formation in large social networks: Membership, Growth, and Evolution," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Philadelphia, PA, 2006.

Bagrow, J.P. "Evaluating local community methods in networks," *Journal of Statistical Mechanics: Theory and Experiment* (May), 2008, p. P05001.

Bagrow, J.P., and Bollt, E.M. "Local method for detecting communities," *Physical Review E* (72), 2005, p. 046108.

Barabási, A.-L., Jeong, H., Zéda, Z., Ravasz, E., Schubert, A., and Vicsek, T. "Evolution of the social network of scientific collaborations," *Physica A* (311:3-4), 2002, pp. 590-614.

Clauset, A. "Finding local community structure in networks," *Physical Review E* (72), 2005, p. 026132.

Clauset, A., Newman, M.E.J., and Moore, C. "Finding community structure in very large networks," *Physical Review E* (70), 2004, p. 066111.

Cook, D.J., and Holder, L.B. "Graph-based data mining," *IEEE Intelligent Systems* (15), 2000, pp. 32-41.

Dorogovtsev, S.N., and Mendes, J.F.F. *Evolution of Networks: From Biological Nets to the Internet and WWW* Oxford University Press, New York, NY, 2003.

Faloutsos, M., Faloutsos, P., and Faloutsos, C. "On power-law relationships of the Internet topology," in *Proceedings of Annual Conference of the Special Interest Group on Data Communication (SIGCOMM '99)*, Cambridge, MA, 1999, pp. 251-262.

Flake, G.W., Lawrence, S., and Giles, C.L. "Efficient identification of web communities," in *Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD 2000)*, ACM, Boston, MA, 2000, pp. 150-160.

Flake, G.W., Lawrence, S., Giles, C.L., and Coetzee, F.M. "Self-organization and identification of web communities," *IEEE Computer* (35:3), 2002, pp. 66-71.

Girvan, M., and Newman, M.E.J. "Community structure in social and biological networks," *Proceedings of the National Academy of Science of the United States of America* (99), 2002, pp. 7821-7826.

Goldberg, M., Kelley, S., Magdon-Ismail, M., Mertsalov, K., and Wallace, W. "Communication dynamics of blog networks," in *Proceedings of the 2nd ACM SIGKDD Workshop on Social Network Mining and Analysis*, Las Vegas, NV, 2008.

Hajra, K.B., and Sen, P. "Aging in citation networks," *Physica A* (346), 2005, pp. 44-48.

Kumar, S.R., Raghavan, P., Rajagopalan, S., and Tomkins, A. "Trawling the web for emerging cyber-communities," *Computer Networks* (31:11-16), 1999, pp. 1481-1493.

Luo, F., Wang, J.Z., and Promislow, E. "Exploring local community structure in large networks," in *Proceedings of 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI'06)*, Hong Kong, China, 2006.

Muff, S., Rao, F., and Caflisch, A. "local modularity measure for network clusterizations," *Physical Review E* (72), 2005, p. 056107.

Newman, M.E.J. "Fast algorithm for detecting community structure in networks," *Physical Review E* (69:6), 2004, p. 066133.

Newman, M.E.J., and Girvan, M. "Finding and evaluating community structure in networks," *Physical Review E* (69:2), 2004, p. 026113.

Palla, G., Derenyi, I., Farkas, I., and Vicsek, T. "Uncovering the overlapping community structure of complex networks in nature and society," *Nature* (435:9), 2005, pp. 814-818.

Porter, L., and Golan, G. "From subservient chickens to brawny men: A comparison of viral advertising to television advertising," *Journal of Interactive Advertising* (6:2), 2006, pp. 30-38.

Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., and Parisi, D. "Defining and identifying communities in networks," *Proceedings of the National Academy of Science of the United States of America* (101), 2004, pp. 2658-2663.

Roussinov, D.G., and Chen, H. "Document clustering for electronic meetings: An experimental comparison of two techniques," *Decision Support Systems* (27), 1999, pp. 67-79.

Schafer, J.B., Konstan, J., and Riedi, J. "Recommender systems in e-commerce," in *Proceedings of the 1st ACM conference on Electronic commerce* Denver, Colorado, 1999, pp. 158-166.

Wasserman, S., and Faust, K. *Social Network Analysis: Methods and Applications* Cambridge University Press, Cambridge, UK, 1994.

Watts, D.J., and Strogatz, S.H. "Collective dynamics of "small-world" networks," *Nature* (393:6684), 1998, pp. 440-442.

Xu, X., Yuruk, N., Feng, Z., and Schweiger, T.A.J. "SCAN: A structural clustering algorithm for networks," in *Proceedings of the 13th International Conference on Knowledge Discovery and Data Mining*, San Jose, CA, 2007.