A Fast Algorithm in Finding Communities of Book Network

Shijun Wang, Changshui Zhang
State Key Laboratory of Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing 100084, China
wsj02@mails.tsinghua.edu.cn, zcs@mail.tsinghua.edu.cn

Abstract

In this paper, we present a new fast algorithm named NDBC in finding communities within large network systems. The algorithm is developed based on DBSCAN and makes use of the structural information of complex networks. We test NDBC on the book network which is constructed by the readers' borrowing behaviors. The experimental result shows that it can quickly find communities of a big network which contains thousands upon thousands vertices.

Keywords: complex network, community, book network

1. Introduction

Many systems in the real world can be abstracted as complex networks. The examples of complex network include the Internet, science collaboration network, neural networks, metabolic networks, food web, etc. [1]. Recent research on complex networks has revealed a number of distinctive statistical properties that most networks seem to share [1] [2] [3], such as power law distribution of degree, high clustering coefficient and short average path length.

"It is widely assumed that most social networks show "community structure", i.e., groups of vertices that have a high density of edges within them, with a lower density of edges between groups." [4] How to discover communities in large network systems within a short time is an interesting problem.

A recent influential algorithm that has been used is based on the idea of betweenness, proposed by Girvan and Newman [5] [6]. The betweenness of an edge is defined as the number of shortest paths that traverse it. Their method iteratively removes edges which lie between two clusters and has highest betweenness from the network. In this method, the time involved to discover the community structure of the graph scales as $O(n^3)$, with n the number of vertices in the network.

In the computer science literature, there are a number of fast heuristics, such as "FM-Mincut". Flake et al. [7] use a maximum flow/minimal cut algorithm to define the edges and vertices that act as boundary between communities.

Wu et al. [8] present a method which is based on notions of voltage drops across networks that are both intuitive and easy to solve. However, their algorithm has to specify the number of communities beforehand.

Zhong Su et al. [9] present a recursive density-based clustering algorithm for web document clustering based on DBSCAN [10]. In order to cut off the bridge between two clusters, their algorithm varies *Eps* and *MinPts* whenever necessary.

In this paper we present a new method named NDBC that can discover communities within networks of arbitrary size in a very short time. The key idea of our method is that we combine the clustering algorithm DBSCAN with the structural information of complex networks in finding communities. We apply our method to finding communities of book network. The result shows that NDBC is very efficient because we just need scan the vertices of the data set once. Moreover, it does not require a predetermined cluster number to operate. Our method achieves the same goal as Zhong Su [9] but need only one constant predefined in the algorithm.

The outline of this paper is as follows. In Sec. 2 we introduce the book network. In Sec. 3 we show the key idea of DBSCAN and its drawback in finding communities. In Sec. 4 our algorithm NDBC is described. The dataset and experimental results are listed in Sec. 5. In Sec. 6 we give our conclusions.

2. Book Network

In the book network, the vertices are the books, and two vertices have a common edge if the corresponding books have been borrowed together by the same person. That means if two books occur in someone's book borrowing records, then the two books are associated. If these two books are borrowed together by more than one person, then the weight of link between them are accumulated. So the book network is a weighted and undirected network.

If two books are borrowed together by N persons separately, we define the distance between those two books is $1/N$. Set D as the book set. The *Eps*-

neighborhood of a book p is defined as $N_{Eps}(p) = \{q \in D \mid dist(q, p) \le 1/MinTime\}$ where $Eps = 1/MinTime$. *MinTime* is the least times that two books are borrowed together.

A clustering CL of book set D with respect to *Eps*, *MinPts* is a set of density-connected [10] sets with respect to *Eps*, *MinPts* in D. For any points in a space, where a point corresponds to a book, the more books that co-occur with it in some reader's borrowing records, the higher its density is.

3. DBSCAN

The clustering algorithm DBSCAN [10] based on sample density is designed to discover clusters of arbitrary shape as well as to distinguish noise.

"The key idea of a density-based cluster is that for each point of a cluster it's *Eps*-neighborhood for some given $Eps > 0$ has to contain at least a *MinPts* minimum number of points, i.e. the "density" in the *Eps*-neighborhood of points has to exceed some threshold." [11]

In order to find communities of book network, we first tried DBSCAN to the book network. The result shows that DBSCAN often leads to a single, giant cluster which is not desirable. The reason why all these books are connected together lies in the inherent nature of DBSCAN. As pointed above, DBSCAN is based on sample density. For many popular books and reference books of various categories, the probability that they are borrowed together is very high. For example, book A is a science fiction and book B is a reference book of physics. If A and B are borrowed together many times, then the weight between them is high. So A and B serve as a bridge between science fictions and references of physics and these two clusters are connected together. This is illustrated in Fig. 1.

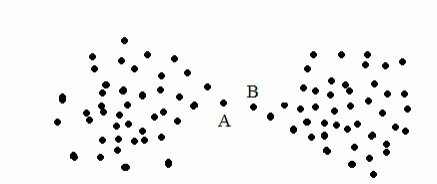

Figure 1. Bridge between two communities

4. NDBC: Neighbor Density Based Clustering algorithm

To solve this problem, we propose a clustering algorithm called NDBC that attempts to overcome the drawback of DBSCAN in finding communities by utilizing the high clustering of complex network.

4.1. Clustering Coefficient

The DBSCAN algorithm just considers the distances of samples from each other, but in a complex network, there is also structural information between vertices besides distance information. The edges between vertices form the topology of a network.

In order to describe the connections in the environment closest to a vertex, we often use the so-called clustering coefficient [1]. "For the network with undirected edges, the number of all possible connections of the nearest neighbors of a vertex μ ($z_1^{(\mu)}$ nearest neighbors) equals $z_1^{(\mu)}(z_1^{(\mu)} - 1)/2$. Let only $y^{(\mu)}$ of them be present. The clustering coefficient of this vertex, $C^{(\mu)} \equiv y^{(\mu)} / \left[z_1^{\mu}(z_1^{\mu} - 1)/2 \right]$, is the fraction of existing connections between nearest neighbors of the vertex." [2] The clustering coefficient C of the network is the average of $C^{(\mu)}$ over all vertices of a network. The clustering coefficient is the probability that two nearest neighbors of a vertex are nearest neighbors also of one another.

In simple terms the clustering coefficient of a book in the book network tells us the likelihood a book's neighbors are borrowed together.

4.2. NDBC

Most complex networks exhibit a large degree of clustering [1] [2] [4]. In a network with high clustering coefficient, though two communities may be connected by a bridge, the vertices on the bridge belonging to different communities have few common neighbors. Based on the phenomena observed above, we can cut off the bridge between these two communities by neighborhood check. From the statistics of clustering coefficient of book network, illustrated in Fig. 4, we can see that the clustering coefficient of book network is very high compared with random network. This means the neighbors of a book are often connected. So based on the high clustering coefficients of most complex networks, we extend DBSCAN by considering the superposition degree of neighbors between a seed vertex and it's subsequent seed vertex in the expansion progress of a cluster.

If vertex A as a seed belongs to community C1 and vertex B as A's neighbor belongs to community C2, suppose that there are enough neighbors of A at given *Eps*, then at the expansion of this cluster from seed A, vertex B's relationship with A's neighbor is checked. If B is connected with most of A's neighbors, then B is assigned the same label with A; otherwise, B is treated as a noise. A constant called *superposition* is defined as a threshold in order to check the superposition degree of these two vertices' neighbors. Because most of A's

neighbors belong to C1 and most of B's neighbors belong to C2, the number of common neighbors of A and B is small. By means of checking two vertices' neighbor superposition degree, the bridge between two different communities is truncated.

The algorithm is listed below:

```
Algorithm NDBC(DB, MinTime, MinBook)
For each v∈DB do
  If v is not yet assigned to a cluster then
     Expand(v, MinTime, MinBook);
     Assign them to a new cluster or noise;
```

```
Expand(v, MinTime, MinBook)
Find v's neighbors NSet with weight greater than
MinTime;
Find vertices whose neighbors have few
superposition with v's neighbors, and delete them
from NSet;
If the size of NSet is greater than MinBook
  Expand the cluster from vertices of Nset;
```

Figure 2. Neighbor Density Based Clustering algorithm

4.3. Analytical Evaluation

The runtime of NDBC is $O(n * \textit{runtime of a neighborhood query})$: n objects are visited and exactly one neighborhood query is performed for each of them [11]. Thus, the overall runtime depends on the performance of the neighborhood query. Fortunately, all the interesting neighborhood predicates are based on adjacency matrix – like distance predicates – which can be efficiently supported by sparse matrix when the network contains many vertices. So the runtime of NDBC is very short.

5. Experiment

5.1. Data Set

```
b1010768x;03-04-08 15:35;p10583956
b10107939;03-04-08 10:18;p10824522
b10126235;03-04-08 09:28;p10809545
b10255977;03-04-08 14:05;p10633716
```

Figure 3. Book borrowing records

We first analyze the data set under consideration. Our experiments draw on data collected from Library of Tsinghua University, China. Fig. 3 is an example of Tsinghua University Library's book borrowing records. The first column is ID of book and the third is ID of reader. It consists of book borrowing records from 15:35:00 March 4, 2003 through 11:21:00 November 12, 2003. In this period there are 573,862 book borrowing records in which 142,081 books appear.

The average clustering coefficient for the book network is 0.6188 with 170 isolated vertices. The distribution of clustering coefficients for the book network is shown in Fig. 4.

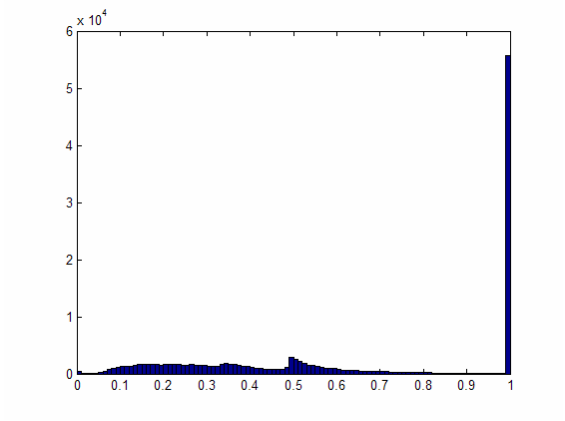

Figure 4. Distribution of Clustering Coefficients

5.2. Experimental Results

To find the communities which DBSCAN can't, we applied our community-finding method to the same book network from Library of Tsinghua University.

In Table 1 we show several communities found by NDBC. By examining the names of books belonging to the same cluster, we can see that the clusters found by NDBC are more reasonable than that found by DBSCAN since similar books are indeed grouped together.

Table 1: Some clutering results using NDBC.

Cluster ID	Book Name
77	Basic algebra Sheaves on manifolds Algebraic geometry Lectures on algebraic topology Geometric integration theory …
71	A physicist's guide to Mathematics Classical mechanics Quantum mechanics Computational fluid dynamics The universe in a nutshell Modern physics …
122	Thinking in C++ C++ Primer The design and evolution of C++ The Annotated STL Sources (Using SGI STL) …
…	…

In Fig. 5 we illustrate 3 clusters of the result from the application of our algorithm to the book network. It shows community structures clearly.

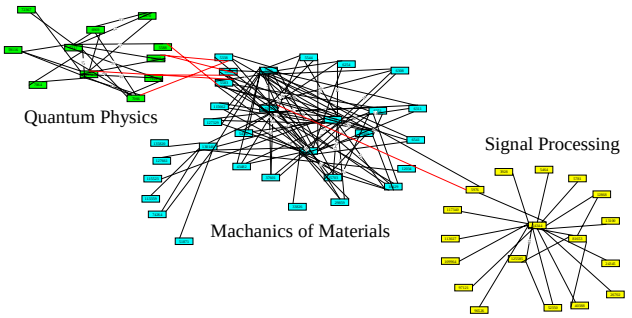

Figure 5. 3 Clusters of Book Network

Table 2 and Figure 6 show the clustering result and efficiency comparison between the two clustering algorithms (Runtime of Table 2 does not include the time reading data from disk). We see that using NDBC, we obtain more clusters for the book network and generate clusters with more even distribution than DBSCAN. In the result of DBSCAN, we can see from Fig. 6 that a cluster contains books far more than that of other clusters. The runtimes of these two algorithms are approximate.

Table 2. Comparison of NDBC and DBSCAN

	NDBC	DBSCAN
Number of Books	142081	142081
Runtime (Sec)	4	3
Eps/*MinPts*	0.125/10	0.125/10
Number of Clusters	142	69

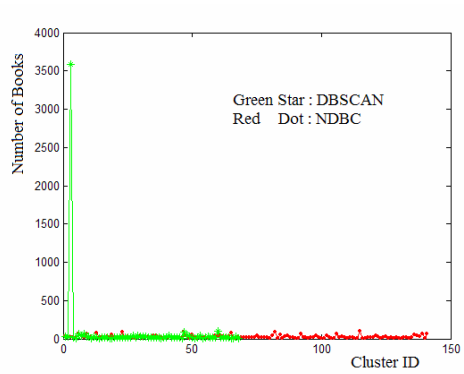

Figure 6: Number of Books in every Cluster

Our algorithm seems to find two types of communities: pure community and complex community. The pure community just contains books belonging to the same subject; the complex community often contains books belonging to several subjects. From the result of clustering, we find that the books relating to computer programming often appear in the same cluster with some books of other subjects. The phenomenon reveals that many students studying in different subjects are interested in programming. It shows that the Computer Science is closely related with other subjects.

6. Conclusions

In this paper we present a new method that allows the quick discovery of communities within a big network. The traditional clustering method DBSCAN fails in the discovery of communities of complex networks. In order to overwhelm the shortcoming of DBSCAN, we introduce structural information of complex networks in finding communities. By utilizing the high clustering coefficients of complex networks, our method succeeds in cutting of the bridge between two communities. Moreover, we do not have to specify the number of communities we wish to divide the network into.

We test the algorithm by applying it to book network. The experimental result shows its validity.

A possible defect of our method is that in order to divide the popular books coming from different categories, we select a high *MinTime*. Unfortunately, the books which are rarely borrowed together with popular books within the same category are treated as noise. In some sense, we just find the cores of the book network communities.

In spite of possible shortcoming we believe that the algorithm we have presented is valid and fast when trying to quickly find communities within large complex networks.

References

[1] R. Albert and A. Barabasi, Statistical mechanics of complex networks, *Reviews of Modern physics*, 2002, pp. 74.47.

[2] S.N. Dorogovtsev and J.F.F. Mendes, Evolution of networks, *Adv. Phys.*, 2002, pp. 51, 1079-1187.

[3] Steven H. Strogatz, Exploring complex networks, *Nature* 410, March 2001.

[4] M. E. J. Newman, The structure and function of complex networks, *SIAM Review*, 2003, pp. 45, 167-256.

[5] M. Girvan and M. E. J. Newman, Community structure in social and biological networks, *Proc. Natl. Acad. Sci.* USA 99, 2002, pp. 8271-8276.

[6] M. E. J. Newman and M. Girvan, Finding and evaluating community structure in network, *Phys. Rev. E*, 2003.

[7] Gary Flake, Steve Lawrence, and C. Lee Giles, Efficient identification of web communities, *In Sixth ACM SIGKDD*, Boston, MA, August 20-23 2000, pp. 150-160.

[8] Fang Wu and Bernardo A. Huberman, Finding communities in linear time: a physics approach, *cond-mat/0310600*, 2003.

[9] Zhong Su, Qiang Yang, Hongjiang Zhang, Xiaowei Xu and Yuhen hu, Correlation-based document clustering using web logs, *34th Annual Hawaii International Conference on System Sciences*, 2001, pp. 5022.

[10] M. Ester, H.P Kriegal, J. Sander, and X. Xu, A density-Based algorithm for discovering clusters in large spatial databases with noise, *KDD 96*, 1996.

[11] M. Ester, H.P Kriegal, J. Sander, and X. Xu. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and its Applications, *Data Mining and Knowledge Discovery* , Kluwer Academic Publishers, Norwell, MA, June 1998, pp. 169-194.