Bibliometric Approach to Community Discovery

Narsingh Deo
School of Computer Science
University of Central Florida
Orlando, Florida 32816-2362
deo@cs.ucf.edu

Hemant Balakrishnan
School of Computer Science
University of Central Florida
Orlando, Florida 32816-2362
hemant@cs.ucf.edu

ABSTRACT

Recent research suggests that most of the real-world random networks organize themselves into communities. *Communities* are formed by subsets of nodes in a graph, which are closely related. Extracting these communities would lead to a better understanding of such networks. In this paper we propose a novel approach to discover communities using bibliographic metrics, and test the proposed algorithm on real-world networks as well as with computer-generated models with known community structure.

Categories and Subject Descriptors

G.2.2 [**Discrete Mathematics**]: Graph Theory—*graph algorithms, network problems.*

General Terms

Algorithms.

Keywords

Community discovery/identification, graph clustering.

1. INTRODUCTION

Study of real world networks has revealed a number of interesting and significant statistical properties; such as degree distribution, average distance between pairs of nodes, and network transitivity. One property of recent interest is community structure. It has been found that nodes in these networks can be grouped such that the nodes within a group are all of similar type. Each of these groups constitutes a community.

Being a qualitative measure the term community has several definitions. Initially *cliques* and *near cliques* were used to represent a community with the idea that nodes that are well connected would be related in some aspect. Kleinberg introduced the concept of "hubs" and "authorities" in web graphs. *Authorities* are web pages which are highly referenced and *hubs* are web pages that reference many authority pages. Later Gibson, Kleinberg and Raghavan define communities in web graph as a core of central, authoritative pages connected together by hub pages [2]. Kumar *et al.* define communities as bipartite cores: a *bipartite core* in a graph G consists of two (not necessarily disjoint) sets of nodes L and R, such that every node in L links to every node in R [4]. Flake, Lawrence and Giles define them as a set of nodes C in a graph G that have more links (in either direction) to members of the community than to non-members [1]. Newman and Girvan define communities as subsets of nodes within which edges are dense, but between which edges are sparse [3]. Community related research has focused on two main problems, community discovery and community identification. From a graph theoretic perspective *community discovery* is the problem of classifying nodes of a graph $G = (V, E)$ into subsets $C_i \subseteq V$, $0 \leq i < k$, such that nodes belonging to a subset C_i are all closely related whereas *community identification* is the problem of identifying the community C_i to which a set of nodes $S \subseteq V$ belong to.

Extracting communities in a graph has number of applications: in social and biological networks we could use communities to study interactions between people or animals, in web graphs to automate the process of creating web directories like `http://directory.google.com` or as a tool for visualizing search results, in *image segmentation* to separate the background of an image from its foreground.

2. EXISTING ALGORITHMS

Community discovery algorithms may be classified broadly into two main types–divisive and agglomerative. The agglomerative algorithms initially consider each node in the graph to belong to an individual community and during the course of the algorithm combine nodes that are closely related to form bigger communities. Divisive algorithms on the other hand initially consider all the nodes in the graph to belong to a single community and during the course of the algorithm remove edges between pairs of nodes that are not well related and thus subdivide the graph into smaller but tighter communities.

Hierarchial Clustering algorithm [5] uses the agglomerative approach. The algorithm first computes the number of node independent or edge independent paths between pairs of nodes. A high value represents better similarity. After computing the similarities between all pairs of nodes we start with n isolated nodes from the input graph and introduce edges between pairs of nodes, starting with the pair of highest similarity and progressing to the weakest. The

43rd ACM Southeast Conference, March 18-20, 2005, Kennesaw, GA, USA.
Copyright 2005 ACM 1-59593-059-0/05/0003 ...$5.00.

hierarchical clustering algorithm fails on some graphs, for example it fails on graphs with pendent nodes. These nodes end up forming a community of their own.

Girvan and Newman [3] came up with a divisive approach that uses uses inverse of edge betweenness as a weight measure of the edges. *Edge betweenness* of an edge is defined as the number of shortest paths between pairs of vertices that pass through the edge. Hence the edge betweenness of intercommunity edges would be high. After computing the edge betweenness of all the edges in the graph, one can remove the edges with low weights and there by expose the underlying community structure. Any bi-partite graph consists of two sets of nodes, each of which would represent a community but removal of edges in any order would not provide us with the two expected communities. Hence all divisive algorithms fail on such graphs.

3. BIBLIOMETRIC APPROACH

The motivation for the current work is from bibliographic metrics which have been used to determine similarity between publications. There are two measures which have widely been used: bibliographic coupling and co-citation coupling. Given two documents, *bibliographic coupling* is defined as number of publications that cite both the given documents and *co-citation coupling* is defined as the number of publications that are cited by both the given documents. Combining the above two measures we obtain a unified metric that can be used to determine similarity between two nodes in a graph. The measure of similarity between two nodes u and v in a graph G is given by: $\frac{\mid N[u] \cap N[v] \mid}{min(d_u,\, d_v) + 1}$, where $N[v]$ refers to the closed neighborhood of node v and d_v refers to its degree. Given a graph G of order n we compute the measure of similarity between every pair of nodes in the graph. This could be done in $O(n\Delta^2)$ time where Δ is the maximum degree of the graph. To obtain the communities we now start with n isolated nodes and introduce edges between pairs of nodes starting with the pair of highest similarity and progressing to the weakest.

One of the main drawbacks of the agglomerative algorithms developed so far is that they classify pendent nodes as separate communities [3]. This is because the similarity metric used is some global property like number of paths or number of node independent paths between node pairs. As a result this value is low for edges connecting pendent nodes to the rest of the graph. This drawback could be overcome by using a local measure of similarity like the one introduce above. And by using an agglomerative approach rather than a divisive one, we would be able to recognize communities in graphs like bi-partite graphs where there are no edges between nodes of the same community.

4. RESULTS AND CONCLUSION

We present our preliminary results on certain graphs which may be considered as benchmarks for community discovery. *Computer-generated networks:* Graphs with known community structure were generated as described by Girvan and Newman in [3]. Each of the generated graphs consist of 128 nodes divided into 4 communities of equal size. Edges were placed uniformly at random, such that each node on average has z_{in} neighbors in the same community and z_{out} neighbors outside. The average degree of the graph is kept close to 16. Our algorithm was tested on these graphs and the fraction of nodes that were classified correctly was measured by varying the number of intercommunity edges per vertex from 0 to 16. The algorithm correctly classified up to 90% of the nodes in graphs with $z_{out} \leq 6$ and close to 70% of the nodes in graphs with $6 < z_{out} \leq 8$. For graphs with $z_{out} > 8$ each node on average has more neighbors outside the community than inside and the graphs no longer posses a well defined community structure.

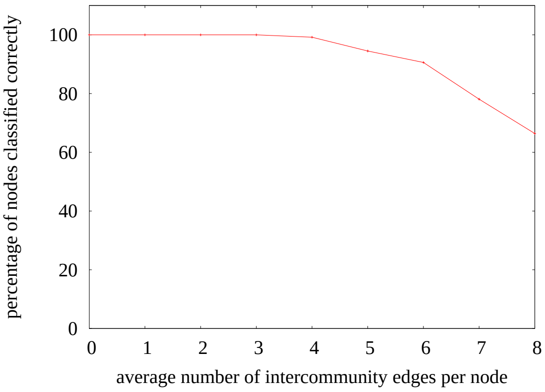

Figure 1: Performance on computer-generated models

Real-world networks: The Zachary Karate Club network is a social network consisting of 34 nodes representing people and edges representing friendships between them. There are two known communities in this network. Our algorithm was able to successfully extract these two communities except for two nodes which were not classified into any community.

The proposed algorithm addresses some of the drawbacks which have been found in the existing approaches to community discovery. In near future we intend to test our approach on large, sparse, random networks like semantic networks, the Internet and the World Wide Web. We would also be employing the similarity measure introduced above to perform community identification.

5. REFERENCES

[1] G. W. Flake, S. Lawrence, and C. L. Giles. Efficient Identification of Web Communities. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 150–160, Boston, MA, Aug 20–23 2000.

[2] D. Gibson, J. M. Kleinberg, and P. Raghavan. Inferring Web Communities from Link Topology. In *Proceedings of the 9th ACM Conference on Hypertext and Hypermedia*, pages 225–234, Pittsburgh, PA, Jun 20-24 1998.

[3] M. Girvan and M. E. J. Newman. Community Structure in Social and Biological Networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[4] S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Extracting Large-Scale Knowledge Bases from the Web. In *Proceedings of 25th International Conference on Very Large Data Bases*, pages 639–650, Edinburgh, Scotland, Sep 7-10 1999.

[5] J. Scott. *Social Network Analysis: A Handbook.* Sage Publications, London, 2nd edition, 2000.