What Do Large Networks Look Like?

Hao Wang, Jianguo Lu
School of Computer Science, University of Windsor
401 Sunset Avenue, Windsor, Ontario N9B 3P4. Canada
Email: {wang115o, jlu}@uwindsor.ca

ABSTRACT

What do large networks look like? Can we visually tell the topological difference between networks such as the Web graph and Facebook network? Due to the huge size of the network, the overall structure will not be discernible if all the nodes and edges are plotted regardless of the graph layout. We reduce the number of nodes and edges by producing a representative subgraph. The nodes are sampled with probability proportional to their degrees, so that large nodes with more connections have a higher probability of being sampled. The edges are reduced further using uniform random spanning tree. The efficacy of the method is demonstrated to preserve the community structure that is characterized by the Network Community Profile (NCP). The result is supported by six real-world large networks, and demonstrated on Twitter user network which contains 4.1×10^7 nodes.

Keywords

Network visualization, Random spanning tree, Big data, Random walk, Web graph, Online Social Network.

1. INTRODUCTION

The topology of large network is hard to visualize, yet it is crucial for data mining applications. If we plot a network with millions of nodes, not to mention hundreds of millions of them, it will be hard to discern the community structure no matter what graph layout is used, and how powerful the computer is. To reveal the visual cues to the structure of the network, we need to reduce the number of nodes and edges by producing a representative subgraph.

Network visualization has been widely studied [3]. Most approaches can only handle graphs of size up to hundreds of nodes and thousands of edges [9]. Beyond this limit, it will be hard to discern the nodes from edges, preventing the discovery of patterns in the graph. Since the tree layout algorithm has the simplest complexity to implement, it is a common practice to reduce the number of edges by turning the graph into a tree, especially a spanning tree representation [2]. The crucial issue is which spanning tree is more representative of the original graph. A spanning tree obtained by breadth-first search will distort the structure of the original network. Numerous efforts have been devoted in adding weights and finding the minimal spanning tree. When the network is very large, computationally it may not be feasible to compute the minimal spanning tree.

Instead of artificially tweaking the parameters for a better spanning tree, we argue that a uniform random spanning tree should be a more natural choice. A typical algorithm to find the uniform spanning tree borrows the idea from random walk [1], therefore, the complexity of the algorithm is the same as the random walk cover time. Although for uniform random graphs the cover time is in the order of $O(NlogN)$, where N is the number of nodes in the network, real-world networks are often scale-free and clustered. Thus we need to prepare for the worst case complexity which is $O(N^3)$ [7]. Obviously, the cost is too high for large networks if we use that algorithm directly.

We observe that it is not necessary to keep all the nodes to reveal the topological structure of large networks. The number of nodes also need to be cut down for very large networks even when tree representation is adopted. We can imagine that a large network has many layers of meshes lying in stack. When all the layers are plotted, the nodes and the structure are obscured by the meshes. If we plot only one random mesh, the crucial nodes and the structure are revealed.

Such random mesh can be obtained by casting the edges uniform randomly–each edge has the same probability of being selected. When an edge is casted, two nodes incident to the edge are collected. In this way, a node will be selected with probability proportional to its degree size. Since random edge selection is not supported in many online social networks, we use simple random walk to approximate the process, considering that the node selection probability is the same asymptotically as random edge sampling [7]. Based on this random walk we simultaneously generate the corresponding random spanning tree. Thereby we reduce the number of nodes and edges at the same time efficiently.

The evaluation of the visualization also imposes a challenge. Since the entire network can not be effectively plotted, the visual comparison between the sub-graph and the original graph is impossible. In particular we would like to see whether the community structure can be visualized. For this purpose we use NCP (network community profile) [6] to evaluate the visualization. As a result, we find that our visualization corresponds to NCP very well.

Contributions 1) We propose an efficient algorithm to visualize large networks. It can scale to very large networks when they are scale-free and crucial nodes and subsequent

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright 2008 ACM 0001-0782/08/0X00 ...$5.00.

Algorithm 1: Random Spanning Tree

```
Data: Graph G;
Result: Random spanning tree $T$ of size $n$.
Let $n_0$ be a uniform random node from G;
mark $n_0$;
while $i<n$ do
    neighbours($n_{i-1}$)= all the neighbours of $n_{i-1}$;
    $n_i$ is a random node of neighbours($n_{i-1}$) ;
    if $n_i$ is not marked then
        i++;
        mark $n_i$;
        add edge $(n_{i-1}, n_i)$ to $T$ ;
    end
end
```

structure can be surfaced quickly using random walk; 2) We demonstrate that the visualization can preserve the community structure by comparison to the NCP; 3) The random spanning tree algorithm is adapted into our random walk node sampling process, reducing the potential high complexity ($O(N^3)$) to a linear algorithm.

2. OUR METHOD

There are at least two ways to select the representative nodes in a graph: by selecting the nodes uniform randomly, or selecting the nodes with probability proportional to their sizes (PPS). When uniform random node selection is applied, most of the nodes will be small nodes with low degrees due to the scale-free nature of the network. The large node with many connections most probably will not be sampled and omitted in the subgraph. Thus we use PPS sampling to obtain the representative nodes, where large nodes have higher probability of being selected. Simple random walk is an efficient sampling method that is supported by many real online social networks, and node sampling probability is proportional to its size asymptotically. Since our random walk is rather long (6×10^4 distinct nodes in our experiments) and well exceeds the mixing time of the graph, the sampling probability can approximate PPS sampling.

Even when the number of nodes are reduced, the network structure is still being obscured by excessive number of edges. Various methods have been proposed to reduce edge size, such as turning the graph into a spanning tree [3, 2]. We propose to use random spanning tree, which can be generated using random walk as illustrated in Algorithm 1. It was originally given by [1], and can be explained as follows: we perform the simple random walk as usual, but add an edge to the tree only when the edge does not form a loop. According to [1], we have the following surprising result:

Theorem 1. *Among all the spanning trees of graph G, T is one of the uniform random sample.*

It may take very long random walk to cover all the nodes of a graph, especially when the graph is scale-free and clustered. The worst case complexity is in the order of $O(N^3)$. Since node selection also uses random walk, we combine the two random walks together to trim nodes and edges simultaneously, avoiding the need to cover all the nodes.

When the random spanning tree is plotted using two-dimensional layouts such as the well-known spring model, the structure is still cluttered for trees containing tens of thousands nodes. We use 3D hyperbolic layout [9] to ameliorate the problem.

Table 1: Statistics of the six networks, each has a citation indicating where the data is from. $\langle d \rangle$ is the average degree, CV stands for coefficient of variation.

Network	# Nodes	CV	$\langle d \rangle$	Max degree
Flickr [5]	105,936	2.65	43.43	5,425
NotreDame[5]	325,729	6.40	5.25	10,721
Stanford[5]	281,903	11.79	14.14	38,625
Amazon[5]	410,236	1.27	11.89	2,760
Facebook [11]	63,731	1.56	25.64	1,098
Youtube[8]	1,138,499	9.65	5.25	28,754

3. COMMUNITY STRUCTURE

We demonstrate our method on the discovery of community structure. The community structure is measured using NCP (network community profile) plot proposed in [6]. We refer to Figure 5 in [6] for a good explanation of NCP, where complete small network visualizations are compared side-by-side with NCP. That figure explains that NCP corresponds well to network visualization in small size (∼ 100 nodes), while we show that the profile is also reflected in our visualization for large networks consisting of millions of nodes.

In network studies, one important measurement for network structure is its conductance, which can be used to characterize the spectral gap and random walk mixing time [10]. The conductance is defined as follows: Let V be the set of nodes of a graph. The conductance of a subset of nodes S of V is

$$\Phi(S) = \frac{\sum_{i \in S, j \in V \backslash S} A_{ij}}{min(A(S), A(V \backslash S))} \qquad (1)$$

where A is the adjacency matrix of the graph, and $A(S) = \sum_{i \in S, j \in V} A_{ij}$. The conductance of the graph is $\Phi = min_S \Phi(S)$. NCP not only looks at the minimal graph conductance, but also the component conductance over the component size.

We conducted experiments on dozens of large networks we can find. Most of them are from Stanford SNAP graph collection [5]. Due to space limitation, we only report the comparison with NCP on six networks[1]. Their statistics are summarized in Table 1. To demonstrate the scalability of our method, we plot a subgraph obtained from the complete Twitter user network that contains 4.1×10^7 nodes and 1.4×10^9 edges [4] in Figure 1. The overall structure clearly differs from other networks plotted in Figure 3. In contrast to the well enmeshed Facebook network, Twitter has a string of super large nodes(bloggers) stacking on each other. Each super node has its own circle of fans with little interaction between them. The veracity of such topology is not easy to verify using NCP, because NCP can not be calculated due to the huge size. However, we can gain some confidence from other relatively smaller networks where NCP can be computed as shown in Figures 2 and 3.

[1] Complete data description and programs can be found at http://cs.uwindsor.ca/∼jlu/visualization.

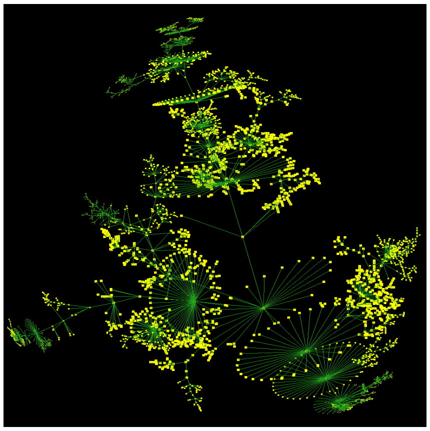


Figure 1: Visualization of Twitter user network.

Figure 2 shows the NCP plots, the conductances over the size of the subcomponents for the original six networks. They are plotted using SNAP API [6]. The insets (in red colour) are the NCP plots obtained from the corresponding subgraphs. We can see that the NCP from subgraph resembles the shape of NCP from the original graph.

Our visualizations of these networks are plotted in Figure 3. The colour of the nodes represents the node degree in the original network. Among the six networks, three of them (Flickr, NotreDame, and Stanford) have low graph conductance, while three others (Facebook, Amazon and Youtube) have high conductance as comparison.

Overall, each visualization corresponds well to its NCP plot of the original network. Several networks are remarkably different from others. Take the first network, Flickr, for example. The NCP plot of the original network in Figure 2 shows a sharp dip ($\sim 10^{-3.5}$) around the component size 10^4, indicating that there is a large component separated from the remaining part. Our visualization in Figure 3 reflects this dumbbell structure clearly. There is a long link connecting these two components, the nodes along the link are mostly of blue and green colour, indicating that the passage between those two components is narrow in the original network. These two components are well enmeshed, coinciding with the NCP plot showing that for most component sizes the conductance is rather large (above 10^{-2}).

NotreDame and Stanford web graphs exhibit a different pattern in both visualization and NCP plots. In their NCP plots, there is a low conductance when the component size is commensurate to the total size. Correspondingly, in the visualization there are clusters of similar sizes. In NCP plots, there are many low conductances when the component size is small. Correspondingly, in the visualization there are many small clusters that is obviously different from the Flickr network.

Amazon, Facebook, and Youtube networks have high conductances as shown in their NCP plots. Correspondingly their visualizations show well enmeshed networks. Note that although the minimal conductance of Amazon network is rather small, the cut happens when the component size is around 100, well below the total size. Therefore, its visualization does not show large clusters.

4. CONCLUSIONS

We demonstrate a practical method to visualize the structure of large networks. The method reduces both the number of nodes and edges of the network dramatically, yet it retains the global topology of the networks. More importantly, our method is very efficient, and works even when the data in its entirety is not available as long as simple random walk is supported.

This is the first attempt to use random spanning tree to reduce the size of the graph for visualization purpose. Direct application of the random spanning tree algorithm does not scale. By combining the random spanning tree algorithm with PPS node sampling, we propose a very efficient algorithm to reduce both the number of nodes and the number of edges leveraging the scale-free nature of the networks. 3D layout is also essential to capture the overall structure.

The calculation of NCP requires the access of the entire data, and may not be feasible for very large networks. As a companion to NCP (network community profile), our fast visualization method sheds a light for the prediction of NCP using only a small sample of the data.

5. REFERENCES

[1] D. J. Aldous. The random walk construction of uniform spanning trees and uniform labelled trees. *SIAM Journal on Discrete Mathematics*, 3(4):450–465, 1990.

[2] C. Chen and S. Morris. Visualizing evolving networks: Minimum spanning trees versus pathfinder networks. In *Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on*, pages 67–74. IEEE, 2003.

[3] I. Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *Visualization and Computer Graphics, IEEE Transactions on*, 6(1):24–43, 2000.

[4] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *WWW*, pages 591–600. ACM, 2010.

[5] J. Leskovec and C. Faloutsos. Sampling from large graphs. In *SIGKDD*, pages 631–636. ACM, 2006.

[6] J. Leskovec, K. Lang, A. Dasgupta, and M. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[7] L. Lovász. Random walks on graphs: A survey. *Combinatorics, Paul Erdos is Eighty*, 2(1):1–46, 1993.

[8] A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *SIGCOMM*, pages 29–42. ACM, 2007.

[9] T. Munzner. Drawing large graphs with h3viewer and site manager. In *Graph Drawing*, pages 384–393. Springer, 1998.

[10] A. Sinclair and M. Jerrum. Conductance and the rapid mixing property for markov chains: the approximation of the permanent resolved. In *Proc. 20th ACM STOC*, pages 235–244, 1988.

[11] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. On the evolution of user interaction in facebook. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN'09)*, August 2009.

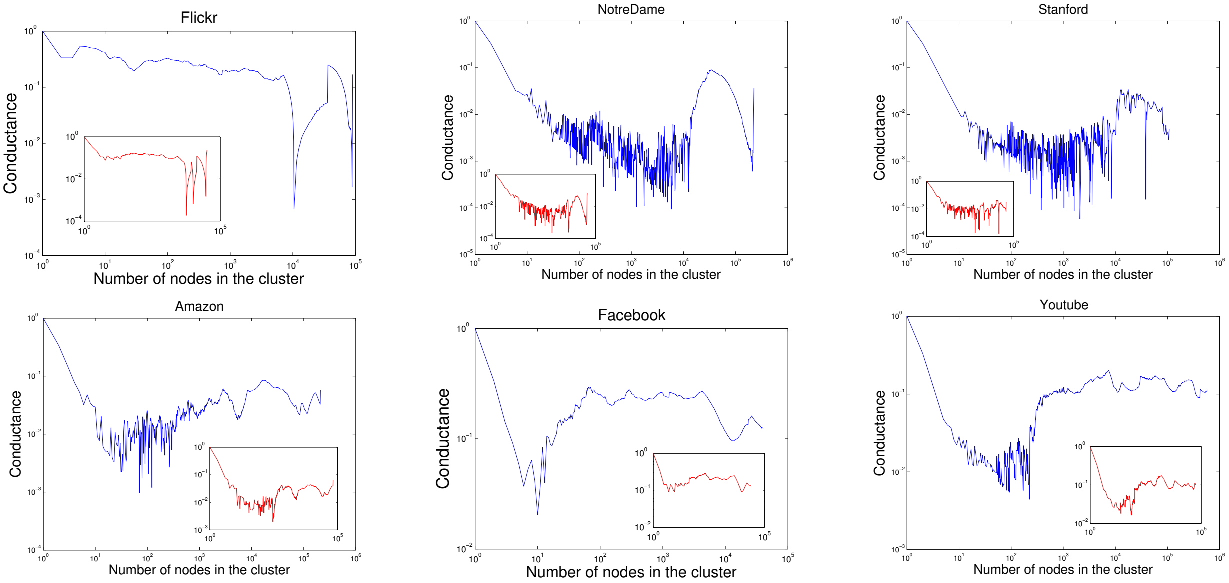
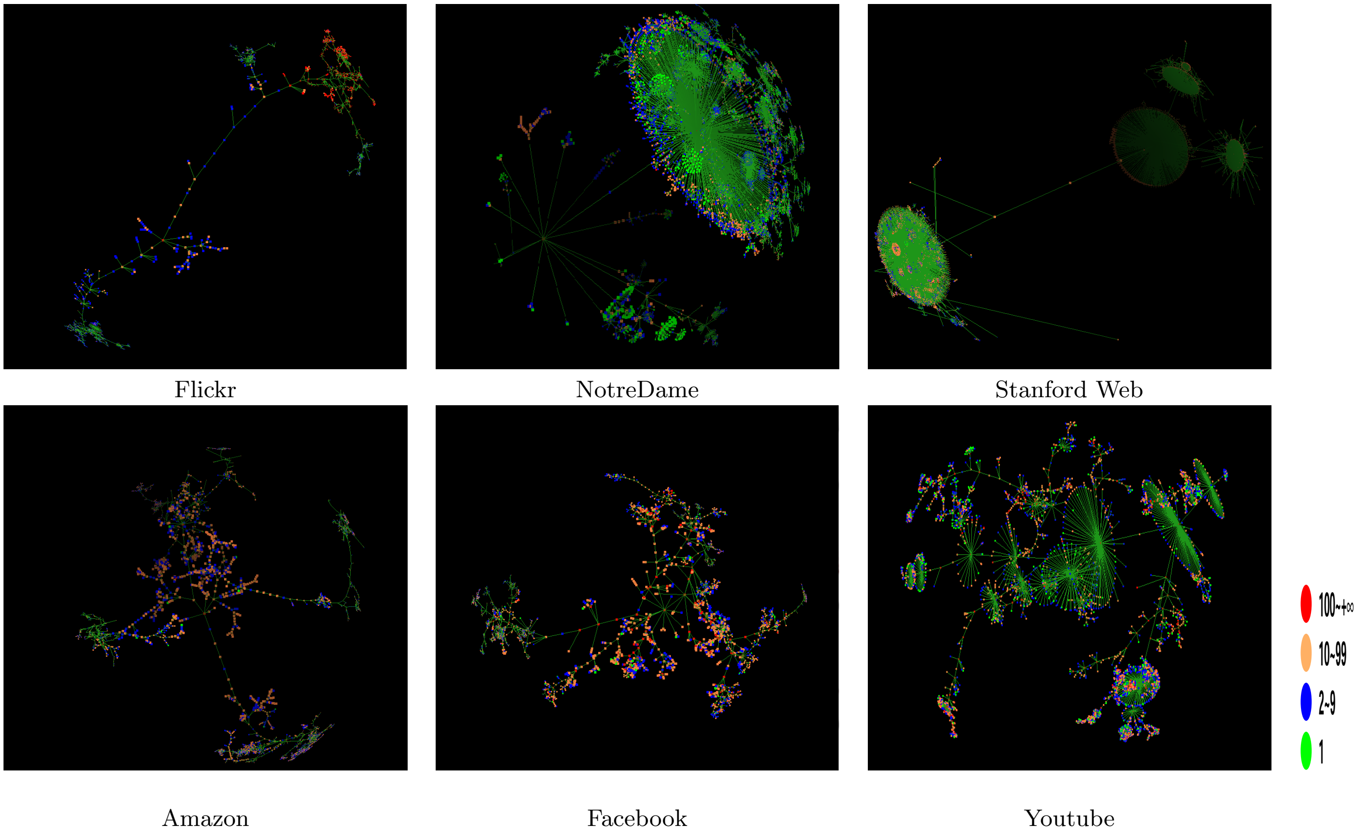


Figure 2: Conductance $\Phi(S)$ over $|S|$, the size of the the components, for six networks. Insets: The corresponding NCP plots obtained from the subgraphs.


Flickr

NotreDame

Stanford Web

Amazon

Facebook

Youtube

Figure 3: (Best viewed in colour) Visualization of six networks. The networks in the first row (Flickr, NotreDame, and Stanford) are clustered, while the networks in the second row (Amazon, Facebook and Youtube) are well enmeshed. Node colour indicates the node degree in the original network.