CERIAS
The Center for Education and Research in Information Assurance and Security

ANALYSIS OF COMMUNITY DETECTION ALGORITHMS FOR LARGE SCALE CYBER NETWORKS

Prachita Mane, Sunanda Shanbhag, Tanmayee Kamath
Department of Computer and Information Technology , Purdue University
Technical Director: Patrick Mackey, Pacific Northwest National Laboratory

ABSTRACT

The recent boom in the usage of Internet has led to systems being more prone to cyber attacks. Considering such a huge amount of traffic over IP networks, it is very difficult to visualize and detect attacks occurring over the network. Community detection can be useful to view and analyze the network from different levels of granularity. This study will carry out a comparative analysis of various community detection algorithms and their performance on IP networks of varying sizes.

RESEARCH QUESTION

Which methods for graph partitioning and community detection perform best in terms of running time for the purpose of grouping IP networks into clusters (supernodes)?

MOTIVATION

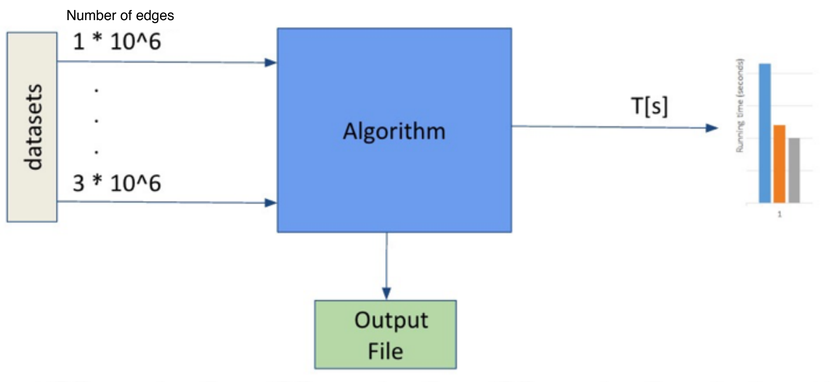
Growth in the Internet → Increasing complexity of structure → More efforts in monitoring and management

- One of the difficulties faced in the detection of cyber attacks in large networks is the huge number of packets to be analyzed, which makes it very slow.
- As the network sizes increase, it becomes important to come up with ways to make the analysis easier and more efficient
- Massive amount of communication happens across IP networks. Thus a cyber attack can occur at any point of time and the time required for it to cause a significant damage to the concerned entity is nominal.
- In order to detect an attack, the IP network needs to be partitioned into supernodes in a nominal amount of time.

SCOPE

The scope of this study is to carry out a comparative analysis of different community detection algorithms on large scale IP networks for cybersecurity purposes.
The comparison will be based on the running time of the algorithms.
We will be looking at how different algorithms work for different sizes, in terms of number of nodes and number of edges.

BLOCK DIAGRAM



DATASETS

- CAIDA (3016379 nodes)
- University of Victoria, ISOT lab botnet dataset

Feature extraction

- Frequency
- Popularity
- Volume of data transferred

ALGORITHMS

INFOMAP

Metric used: Mapequation

Mapequation addresses two problems of large networks: compression of data and identifying modular structures in the network.

Infomap is an agglomerative algorithm that makes use of map equation to compress data and detect community structure in large networks.

GRAPPOLO

Metric used: Modularity

Modularity measures the density of links inside communities as compared to links between communities.

Grappolo is a parallel version of the Louvain method, which is one of the most commonly used community detection algorithms. It uses graph coloring and labeling for parallel heuristics and implements them in OpenMP.

BCALS

Metric used: Betweenness centrality

It is called BCALS in short.

BCALS selects at first, leaders according to their measure of betweenness centrality, then it selects randomly a node and calculates its local function for all communities and assigns it to the community that optimizes its local function

RESULTS

No of Edges (in millions)	Running time (in seconds)	
	InfoMap	Grappolo
0.5	29	1.42
1	54	1.73
1.5	100	2.17
2	262	2.73
2.5	272	3.38
3	310	4.63

Visualization:

- The communities will be visualized using Gephi.
- Each community is represented by a different color.
- C++ code to calculate community score. The community score represents the number of nodes in each community.