Chapter 1
A Survey of Parallel Community Detection Algorithms

Sobin C. C.
IIT Roorkee, India

Vaskar Raychoudhury
IIT Roorkee, India

Snehanshu Saha
PESIT-BSC, India

ABSTRACT

The amount of data generated by online social networks such as Facebook, Twitter, etc., has recently experienced an enormous growth. Extracting useful information such as community structure, from such large networks is very important in many applications. Community is a collection of nodes, having dense internal connections and sparse external connections. Community detection algorithms aim to group nodes into different communities by extracting similarities and social relations between nodes. Although, many community detection algorithms in literature, they are not scalable enough to handle large volumes of data generated by many of the today's big data applications. So, researchers are focusing on developing parallel community detection algorithms, which can handle networks consisting of millions of edges and vertices. In this article, we present a comprehensive survey of parallel community detection algorithms, which is the first ever survey in this domain, although, multiple papers exist in literature related to sequential community detection algorithms.

INTRODUCTION

Today, many applications generate enormous amounts of data. As per the latest statistics (Statista, 2016), Facebook has 1.35 billion daily active users and they share nearly 2.5 million pieces of content in every minute. Similarly, Twitter has 271 million active users, with 300,000 tweets in every minute (ACI, 2016). Analysis of such massive volumes of data helps to uncover much useful information, which can be used for many applications for better decision making. Examples of such applications include, finding similarity in

DOI: 10.4018/978-1-5225-2498-4.ch001

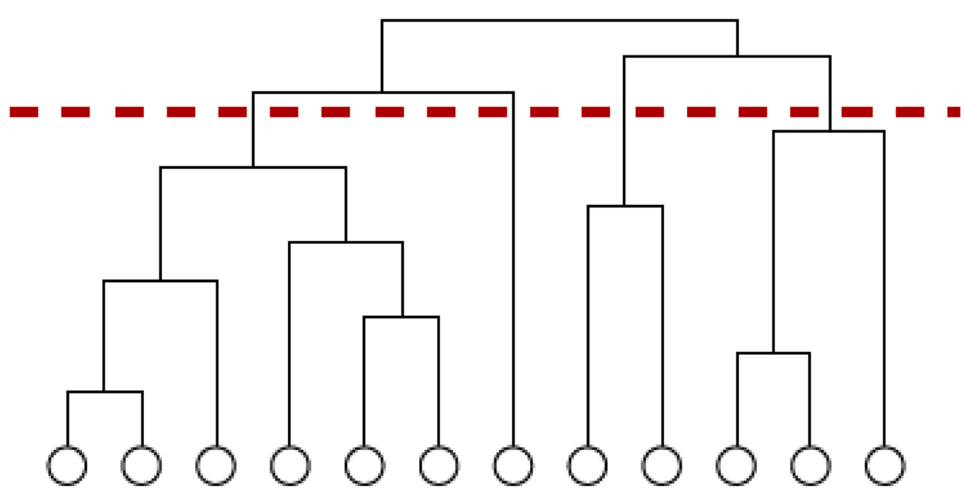
authorship networks, relay selection in delay tolerant networks, identifying users for targeted marketing, etc. Communities can be defined as, densely connected groups of vertices, which share common properties and are sparsely connected to the rest of the nodes in a graph, (Newman, 2003). Community can be used to uncover structural and functional properties of a network. Community detection is a process of identifying communities, (both overlapping and disjoint), in a network. Detecting community structure in large networks is an NP hard problem (Fortunato, 2010). So approximation algorithms are applied to the task of community detection. However, the existing community detection algorithms are sequential in nature and cannot scale to a large extent. Such inefficiency of existing sequential community detection algorithms necessitated the need of developing parallel community detection algorithms, which is the focus of this book chapter.

Community structure exhibits hierarchical properties and hence to represent them dendrograms (Figure 1) are used. Detecting community structure of large network is difficult, not only because of the huge size of the network alone, but also the structure of the large network. For example, Scale-free networks, are networks, in which the degree distribution of the nodes follows a powerlaw distribution, i.e. a few hub nodes have a high degree of connectivity and many of the nodes have a low degree of connectivity. Mathematically, the fraction of the nodes, *P(k)*, in a network, with a degree of *k*, follows a *power law distribution* as follows

$$P(k) \sim k^{-\gamma} \tag{1}$$

Where, $2 < \gamma < 3$. Many networks like collaboration networks, Internet, Airline networks, etc., are *scale-free networks,* which follow a power law distribution.All those networks share a common property that, few nodes have large interconnections to other nodes, where most of the nodes have few interconnections. Although, scale-free networks exhibit community structure naturally, detection procedure is a difficult task. A detailed description of the community and community detection algorithms can be found in the survey by Fortunato (Fortunato, 2010).

Figure 1. Dendrogram representing community structure



Related Content