Big Data Analytics and Mining for Knowledge Discovery

11

Carson K. Leung
https://orcid.org/0000-0002-7541-9127
University of Manitoba, Canada

INTRODUCTION

Progresses in information science and technology have enabled the collection and generation of huge volumes of valuable data—such as streams of banking, financial, marketing, organizational, and transactional data—at a high velocity from a wide variety of rich data source in various real-life business, engineering, education, healthcare, hospitality and tourism, scientific, as well as social applications and services in government, organizations and society. These *big data* (Madden, 2012; Leung, 2015; Bellatreche et al., 2019) may be of different levels of veracities (e.g., precise data, imprecise and uncertain data) and/or of a variety of types or formats (e.g., structured data in relational databases; semi-structured data in extensible markup language (XML) or JavaScript object notion (JSON) format stored in document-oriented or graph databases; unstructured data in images, audios and videos). Embedded in the big data is implicit, previously unknown, and potentially useful information and knowledge. However, the big data come with volumes beyond the ability of commonly-used software to capture, manage, and process within a tolerable elapsed time. Hence, new forms of information science and technology—such as *big data analytics and mining for knowledge discovery*—are needed to process and analyze the big data so to as enable the enhanced decision making, insight, and process optimization. For instance, the discovery of organizational knowledge (e.g., common customer complaints, main causes of employee turnover, sets of popular merchandise items in shopping carts)—via techniques like big data analysis, statistics, and business analytics—helps reveal important patterns about an organization. This organizational knowledge helps executive and management teams of the organization to get a better understanding of the organization so that they could make better use of human resources and technology, focus more on education and growth, keep customers top of mind, and further improve quality of services and products. To a further extent, the discovery of organizational knowledge and its subsequent actions help the organization to meet goals, gain competitive advantage, and ultimately ensure sustainability, organizational growth and development.

Over the past two decades, algorithms have been proposed for various big data analytics, mining and knowledge discovery—including clustering (which groups similar data together), classification (which categorizes groups of similar data), outlier detection (which identifies anomalies), and frequent pattern mining (which discovers interesting knowledge in the forms of frequently occurring sets of merchandise items or events). Many of these algorithms use the *MapReduce* model—which mines the search space with distributed or parallel computing (Shim, 2012). Among different big data analytics and mining tasks, this chapter focuses on applying the MapReduce model to big (organizational) data for the discovery of frequent patterns.

DOI: 10.4018/978-1-7998-3473-1.ch125

BACKGROUND

Since the introduction of the research problem of *frequent pattern mining* (Agrawal, Imieliński, & Swami, 1993), numerous algorithms have been proposed (Hipp, Güntzer, & Nakhaeizadeh, 2000; Ullman, 2000; Ceglar & Roddick, 2006; Aggarwal, Bhuiyan, & Al Hasan, 2014; Leung et al., 2017c). Notable ones include the classical Apriori algorithm (Agrawal & Srikant, 1994) and its variants such as the Partition algorithm (Savasere, Omiecinski, & Navathe, 1995). The Apriori algorithm uses a level-wise breadth-first bottom-up approach with a candidate generate-and-test paradigm to mine frequent patterns from transactional databases of precise data. The Partition algorithm divides the databases into several partitions and applies the Apriori algorithm to each partition to obtain patterns that are locally frequent in the partition. As being locally frequent is a necessary condition for a pattern to be globally frequent, these locally frequent patterns are tested to see if they are globally frequent in the databases. To avoid the candidate generate-and-test paradigm, the tree-based FP-growth algorithm (Han, Pei, & Yin, 2000) was proposed. It uses a depth-first pattern-growth (i.e., divide-and-conquer) approach to mine frequent patterns using a tree structure that captures the contents of the databases. Specifically, the algorithm recursively extracts appropriate tree paths to form projected databases containing relevant transactions and to discover frequent patterns from these projected databases.

In various real-life business, engineering, healthcare, scientific, and social applications and services in modern organizations and society, the available data are not necessarily *precise* but *imprecise or uncertain* (Leung, 2014; Leung, MacKinnon, & Tanbeer, 2014; Cheng et al., 2019; Rahman, Ahmed, & Leung, 2019; Titarenko et al., 2019). Examples include sensor data and privacy-preserving data (Leung et al., 2018; Chen et al., 2019; Leung, Braun, & Cuzzocrea, 2019; Li & Xu, 2019). Over the past decade, several algorithms have been proposed to mine and analyze these uncertain data. The tree-based UF-growth algorithm (Leung, Mateo, & Brajczuk, 2008) is an example.

With huge volumes of big data, it is not unusual for users to have some phenomenon in mind. For example, a manager in an organization is interested in some promotional items. Hence, it would be more desirable if data mining algorithms return only those patterns containing the promotional items rather than returning all frequent patterns, out of which many may be uninteresting to the manager. It leads to *constrained mining*, in which users can express their interests by specifying constraints and the mining algorithm can reduce the computational effort by focusing on mining those patterns that are interesting to the users.

Besides the aforementioned algorithms discover frequent patterns *in serial*, there are also *parallel and distributed* frequent pattern mining algorithms (Zaki, 1999). For example, the Count Distribution algorithm (Agrawal & Shafer, 1996) is a parallelization of the Apriori algorithm. It divides transactional databases of precise data and assigns them to parallel processors. Each processor counts the frequency of patterns assigned to it and exchanges this frequency information with other processors. This counting and information exchange process is repeated for each pass/database scan.

As we are moving into the new era of big data, more efficient mining algorithms are needed because these data are wide varieties of valuable data of different veracities with volumes beyond the ability of commonly-used algorithms for mining and analyzing within a tolerable elapsed time. To handle big data, researchers proposed the use of the *MapReduce programming model*.