International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 18 (2017) pp. 7355-7359

Algorithms for Frequent Pattern Mining of Big Data

Syed Zubair Ahmad Shah[1], Mohammad Amjad[1], Ahmad Ali Habeeb[2], Mohd Huzaifa Faruqui[1] and Mudasir Shafi[3]

[1]Department of Computer Engineering, Jamia Millia Islamia, New Delhi, India.
[2]Department of Information Technology & Cloud, Ericsson India Global Services, Gurgaon, Haryana, India.
[3]Department of Electronics and Communication Engineering, Maharshi Dayanand University, Rohtak, Haryana, India.

[1]Orcid ID: 0000-0003-1847-8216

Abstract

Frequent pattern mining is a field of data mining aimed at unsheathing frequent patterns in data in order to deduce knowledge that may help in decision making. Numerous algorithms for frequent pattern mining have been developed during the last two decades most of which have been found to be non-scalable for Big Data. For Big Data some scalable distributed algorithms have also been proposed. In this paper, we analyze such algorithms in detail. We start with age old algorithms like Count Distribution and move on to traverse other algorithms one by one and finally to algorithms based on MapReduce programming model.

Keywords: Frequent Pattern Mining, Distributed Systems, Big Data Analytics

INTRODUCTION

Data mining is the task of scanning large datasets with the aim to generate new information or with the aim of knowledge discovery. Broadly speaking, data mining includes fields like clustering, frequent pattern mining (FPM), classification and outlier detection [1, 2, 3]. FPM is a popular data mining approach [4] which is used to find attributes that occur together frequently. Some of its important applications are market basket analysis, share market analysis, drug discovery, etc. [5]. Several FPM algorithms have been proposed in the last few decades such as Apriori [6], Eclat [7], FP-Growth [8], etc. [9, 10]. Identifying all frequent patterns is a complex task and is also computationally costly due to a large number of candidate patterns and so most algorithms become inefficient when it comes to the kind of data we deal with in this age, the Big Data. Social networking places like Facebook produce over 300TB of data every single day [11]. Amazon, Walmart and other giants register billions of transactions every year. Gene sequencing platforms can generate terabytes of data. To cope up with the problem of huge volumes of data, researchers proposed several parallel algorithms for the FPM problem.

This paper reviews some of the classical as well as recent parallel FPM algorithms. Various parallel FPM algorithms have been discussed and analyzed in detail starting with the classical algorithms and then moving onto the state of the art algorithms.

ALGORITHMS FOR PERFORMING FREQUENT PATTERN MINING ON A DISTRIBUTED ENVIRONMENT

A number of algorithms have been proposed for efficient frequent pattern mining on a distributed system which include Count Distribution, Data Distribution and Candidate Distribution algorithms by Agrawal and Shafer [12], Parallel Eclat by Zaki el. al. [13], Parallel FP-growth algorithm by Li el. al. [14], Single Pass Counting, Fixed Passes Combined-counting and Dynamic Passes Combined-counting algorithms by Lin et. al. [15] and Distributed Eclat algorithm by Moens et. al. [16]. We discuss a few of these herein.

Count Distribution Algorithm

Notations:

k-itemset	Itemset with k items
$Freq_k$	Set of frequent k-itemsets
$Cand_k$	Candidate itemset of size k
$Proc^i$	Processor i
DS^i	Dataset local to $Proc^i$
DSR^i	Dataset local to $Proc^i$ after re-partition
$Cand_k^i$	Candidate set of $Proc^i$ during pass k

The Count Distribution (CD) algorithm tries to minimize interaction. It partitions the dataset in horizontal manner into equal parts for the various processors. Each processor autonomously runs the Apriori algorithm and creates the hash-tree on the locally stored transactions. After each iteration, global count of candidates is calculated by adding all counts