Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 112 (2017) 417–426

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France

Vertical Pattern Mining Algorithm for Multiple Support Thresholds

Sadeq Darrab[a], Belgin Ergenc[a]*

[a]*Izmir Institute of Technology, Computer Engineering Department, Izmir, 35447, Turkey*

Abstract

Frequent pattern mining is an important task in discovering hidden items that co-occur (itemset) more than a predefined threshold in a database. Mining frequent itemsets has drawn attention although rarely occurring ones might have more interesting insights. In existing studies, to find these interesting patterns (rare itemsets), user defined single threshold should be set low enough but this results in generation of huge amount of redundant itemsets. We present Multiple Item Support-eclat; *MIS-eclat* algorithm, to mine frequent patterns including rare itemsets under multiple support thresholds (MIS) by utilizing a vertical representation of data. We compare *MIS-eclat* to our previous tree based algorithm, MISFP-growth[28] and another recent algorithm, CFP-growth++[22] in terms of execution time, memory usage and scalability on both sparse and dense databases. Experimental results reveal that *MIS-eclat* and MISFP-growth outperform CFP-growth++ in terms of execution time, memory usage and scalability.

Peer-review under responsibility of KES International

Keywords: frequent pattern mining, multiple support thresholds, vertical mining, pattern growth tree

1. Introduction

Association rule mining has drawn a big attention since it was first proposed in[1] due to its applicability in various domains as medical studies, telecommunication data, market basket analysis, etc. Association rule mining focuses on finding co-occurrence of items in databases where the relationships between these items are expressed as association rules. The overall goal of this process is to discover all interesting rules from a database that have the

* Corresponding author. Tel.: +90-532-226-4125; fax: +90-232-750-7862
E-mail address: belginergenc@iyte.edu.tr

1877-0509
Peer-review under responsibility of KES International
10.1016/j.procs.2017.08.051

form: $X \rightarrow Y \mid X \cap Y = \emptyset$ where X and Y are the set of items in the database. An interesting rule should satisfy two statistical measures defined by the user and known as minimum support threshold denoted as *minsup* and minimum confidence threshold denoted as *minconf. Minsup* refers to the percentage of transactions in the database that contain $X \cup Y$ whereas *minconf* refers to the conditional probability of finding $X \cup Y$ given the transactions that contain X. An itemset is frequent if its support exceeds *minsup,* an association rule is frequent if its confidence exceeds *minconf.*

Frequent association rules can be found in two consecutive steps; 1. Generation of all frequent patterns that satisfy a given *minsup*, 2. Generation of association rules which satisfy both of *minsup* and *minconf,* from frequent patterns found in step 1. Since the first step is computationally expensive almost all research on association rule mining focuses on generating frequent patterns that frequently co-occur (itemset) in a database. For the same reason, association rule mining, frequent pattern mining and frequent itemset mining are used interchangeably.

Since the introduction of frequent itemset mining[1], most of the well-studied algorithms, such as Join-based algorithms[1,2,3,4,5,6,7,8], Tree-based algorithms[9,10,11,12] or Vertical Mining [13,14] employ the uniform *minsup* at all levels. Using single *minsup* allows the utilization of downward closure property which says "any subset of frequent itemset should be frequent" and reduces the search space and computation cost considerably. Thus, these algorithms avoid a huge amount of infrequent itemsets from being processed. However these algorithms assume that all items in the database have the same nature and similar frequencies but this assumption is not true for real-life applications. In many applications, some items appear very frequently in the database whereas others hardly ever appear. Users require finding not only frequent itemsets but rare items as well. To identify the frequent and rare itemsets, single *minsup* is not adequate since when *minsup* is set low, the number of frequent itemsets goes up dramatically, and the performance of these algorithms degenerates quickly. If *minsup* is set too high, interesting rare patterns may be missed. This problem is called rare item problem[15]. Rare item problem is also studied in tasks like classification[30] or periodic pattern mining[31], in this paper we only consider mining both frequent patterns and rare ones by utilizing Multiple Item Support (MIS) thresholds instead of single support threshold, in association rule mining process.

Several algorithms are studied to reduce search space and execution time while generating frequent patterns under MIS[15,16,17,18,19,20,21,22,23,24,25,28]. These algorithms overcome the rare item problem by discovering the complete set of frequent patterns including rarely occurring ones since they apply different support threshold to each item. These algorithms can be classified in terms of search strategies as breadth-first search algorithms[15,16,17,18,19] and depth-first search algorithms[20, 21, 22, 23,24,25,28]. Breadth-first search algorithms which are based on Apriori algorithm scan the databases many times with their candidate generation-and-test approach. In order to overcome this weakness, depth-first search algorithms based on FP-Tree structure are proposed. These algorithms require database scan at most twice since FP-Tree holds all necessary information that is needed in mining process. However, these algorithms are still far from being efficient since they require huge amount of memory due to the management of irrelevant nodes and high execution time for tree operations in pre-mining phase as pruning and reconstruction. There is no algorithm that is devised for multiple item support thresholds that uses vertical representation of data and discarding property.

In this paper; we present a new itemset mining under MIS algorithm; *MIS-eclat* and compare it to our previous algorithm MISFP-growth[28] and another recent algorithm, CFP-growth++[22]. *MIS-eclat* 1) utilizes the vertical representation of data to find the support of itemsets, and 2) constructs AdjacencyMIS structure with useful items that have support greater than the least MIS in order to increase efficiency in terms of execution time and memory usage. In order to be self-contained, we revisit our previous algorithm, MISFP-growth[28] that is pattern growth tree based. To assess the performance of these algorithms, execution time, memory usage and scalability experiments are conducted on both sparse and dense databases in comparison to a recent tree based algorithm; CFP-growth++[22]. The experimental results show the superiority of *MIS-eclat* and MISFP-growth algorithms in comparison to CFP-growth++, in terms of runtime and memory usage. The results also show that *MIS-eclat* and MISFP-growth scale up much better than CFP-growth++ as the size of database increases except with dense databases.

The organization of the paper is as follows: in section 2, we give preliminaries of the challenge. In section 3, *MIS-eclat* is presented and MISFP-growth is revisited to be self-contained. Experimental results are shown in section 4 while the related work is discussed in section 5. Conclusion remarks are given in section 6.