Efficient Implementations for UWEP Incremental Frequent Itemset Mining Algorithm

Mehmet Bicer, Graduate Center, City University of New York, USA

Daniel Indictor, Columbia University, USA

Ryan Yang, Massachusetts Institute of Technology, USA

Xiaowen Zhang, College of Staten Island, City University of New York, USA

ABSTRACT

Association rule mining is a common technique used in discovering interesting frequent patterns in data acquired in various application domains. The search space combinatorically explodes as the size of the data increases. Furthermore, the introduction of new data can invalidate old frequent patterns and introduce new ones. Hence, while finding the association rules efficiently is an important problem, maintaining and updating them is also crucial. Several algorithms have been introduced to find the association rules efficiently. One of them is Apriori. There are also algorithms written to update or maintain the existing association rules. Update with early pruning (UWEP) is one such algorithm. In this paper, the authors propose that in certain conditions it is preferable to use an incremental algorithm as opposed to the classic Apriori algorithm. They also propose new implementation techniques and improvements to the original UWEP paper in an algorithm we call UWEP2. These include the use of memorization and lazy evaluation to reduce scans of the dataset.

KEYWORDS

Association Rule, Confidence, Frequent (Large) Itemset, Itemset, K-Itemset, Support

1. INTRODUCTION

Association rule mining (ARM) is a common technique used in discovering interesting frequent patterns in data acquired in various application domains. There are many algorithms (Agrawal et al., 1993; Agrawal & Srikant, 1994; Borgelt, 2003) proposed to deal with this problem. One of the classic algorithms is Apriori (Agrawal & Srikant, 1994), which is an iterative, breadth-first, level-wise algorithm with join steps and candidate generation. Apriori requires a full scan of the database at each level. Given that the set of *k*-itemsets can be grouped into *k* levels, we would need to scan the database *k* times.

In ARM, the search space combinatorically explodes as the size of the data increases. If the data is static, that is if it never gets updated, running one of the classical algorithms, such as Apriori, will suffice to discover the association rules. But in the real world, new transactions are added to existing

DOI: 10.4018/IJAL.2021010102

Table 1. An example database

TID	Items_purchased
001	1, 3, 5
002	2, 3, 5
003	1, 4, 5
004	1, 3
005	2, 3, 4, 5

databases every day. This addition may invalidate old association rules and create new ones. While finding the association rules on a fresh dataset efficiently is an important problem, maintaining and updating them is also crucial, as it allows the researchers to avoid rerunning the Apriori for the entire dataset to maintain the association rules each time the database gets updated.

Incremental algorithms were created to deal with periodic or continuous data added to the transaction databases (Cheung et al., 1996; Cheung et al., 1997; Omiecinski & Savasere, 1998; Sarda & Srinivas, 1998). The advantage of incremental algorithms is that they allow the user to avoid reading and analyzing the same dataset multiple times by saving select characteristics of it for future runs. Conventional algorithms like Apriori require the entire dataset to be reevaluated when it is updated. One incremental algorithm is Update with Early Pruning (UWEP) (Ayan et al., 1999). As the database scan is one of the major costly operations in finding frequent itemsets, decreasing the amount of reading can substantially improve runtime performance. UWEP reads the existing database at most once and the new database exactly once, while creating and counting a minimal number of candidate itemsets (Ayan et al., 1999). In addition, the UWEP algorithm has mechanisms to prune infrequent itemsets as soon as they are identified. Apriori, by contrast, waits for the appropriate level to prune infrequent itemsets.

The rest of the article is organized as follows. In Section 2, we will describe the association rule mining and incremental association rule mining . In Section 3 we will describe Apriori and Apriori-based incremental algorithms FUP and FUP2. In Section problem 4 we will explain UWEP algorithm in detail. In Section 5 we describe our algorithm, UWEP2. In Section 6 we will explain the experimental results and the datasets used. We conclude the article in Section 7.

Table 2. Support count and relative support (support) of 1-itemsets

Itemset	Support_count *σ(itemset)*	Support % *σ(X)/N*
1	3	60
2	2	40
3	4	80
4	2	40
5	4	80