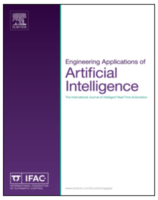
Engineering Applications of Artificial Intelligence 52 (2016) 213–234

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Erasable itemset mining over incremental databases with weight conditions

Gangin Lee, Unil Yun*, Heungmo Ryang, Donggyu Kim

Department of Computer Engineering, Sejong University, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 17 April 2015
Received in revised form
5 October 2015
Accepted 13 March 2016

Keywords:
Erasable itemset
Incremental database
Weight constraint
Itemset pruning
Data mining

ABSTRACT

Erasable itemset mining is an approach for mining itemsets with low profits from large-scale product databases in order to solve financial crises of plants in manufacturing industries. Previous erasable itemset mining methods deal with static product databases only, and ignore any characteristics such as items' own values when they extract the erasable itemsets. Therefore, such approaches may fail to solve financial crises of plants because they have to iterate a significant number of mining processes in order to deal with real-time product data accumulated from plants in the real world. In this paper, we propose a new tree-based erasable itemset mining algorithm for dynamic databases, which finds erasable itemsets considering the weight conditions from incremental databases. The proposed algorithm uses new tree and list data structures for performing its mining operations more efficiently. Furthermore, the proposed algorithm is capable of reducing the number of mined erasable itemsets by considering the different weight information of items within product databases. We also compare the proposed approach with other tree-based state-of-the-art methods. By performing runtime, memory, pattern quality, and scalability comparisons with respect to various real and synthetic incremental datasets, we show that the proposed algorithm is outstanding in comparison to other previous methods.

1. Introduction

With the increase of database usage, data mining has attracted the attention of researchers and has been considered an important issue for decades. Data mining is a strong data analysis method that can discover interesting knowledge by constructing various models or finding meaningful results automatically from databases. As one of the major data mining areas, itemset mining refers to a series of processes for extracting useful itemset information from databases. The concept of itemset mining was first proposed in *Apriori* (Agrawal and Srikant, 1994), which is an algorithm that finds frequent itemsets with supports higher than or equal to a given threshold. Frequent itemset mining has been researched with various real world applications such as bio research (Chen and Liu, 2013), social network analysis (Nohuddin et al., 2012), electronic commerce management (Chang and Chen, 2012), and multi-dimensional network analysis (Berlingerio et al., 2013). In addition, this area also includes a variety of variations such as top-k pattern mining (Pyun and Yun, 2014), utility pattern mining (Ryang and Yun, 2016; Ryang et al., 2014; Yun et al., 2014), and representative pattern mining (Yun et al., 2015; Yun and Ryu, 2013).

Since such traditional approaches focus on extracting meaningful information from large databases considering minimum support thresholds, they may have difficulty in dealing with empirical applications such as discovering relatively meaningless information in specific environments. Let us consider a manufacturing plant's situation with a financial crisis as the most common scenario. If the plant wants to survive from this financial crisis, it is then necessary to remove components with low profits from production lines by comparing and analyzing both profits and losses of the products' parts. To perform such an analysis process, the concept of erasable itemset mining was proposed in *META* (Deng et al., 2009). This method finds a set of itemsets with smaller profits than a given threshold (called erasable itemsets) from product databases. After *META*, various approaches (Deng, 2013; Lee et al., 2014; Nguyen et al., 2014) have been proposed to mine erasable itemsets more efficiently. In particular, most of these erasable itemset mining methods such as *MERIT* (Deng and Xu, 2012) and *dMERIT+* (Le et al., 2013) employ tree-based mining processes like previous traditional frequent pattern mining approaches (Han et al., 2004; Pyun et al., 2014).

Meanwhile, these previous traditional itemset mining methods have several problems in terms of data accumulation and characteristics of items because of the following issues. First, such approaches suppose that databases processed by them are static;

* Corresponding author.
E-mail addresses: ganginlee@sju.ac.kr (G. Lee), yunei@sejong.ac.kr (U. Yun), ryang@sju.ac.kr (H. Ryang), donggyukim@sju.ac.kr (D. Kim).

http://dx.doi.org/10.1016/j.engappai.2016.03.003
0952-1976/© 2016 Elsevier Ltd. All rights reserved.

however, product data in the real world becomes gradually larger according to the activation and growth of the plants. In this case, huge amounts of data can accumulate in product databases and lead to various changes; therefore, if new data are inputted in the databases, erasable itemsets previously mined from them are no longer considered to be of interest. For this reason, previous methods cannot efficiently deal with such incremental data. Moreover, traditional approaches assume that all of the components composing products have the same importance or value regardless of component types when they mine erasable itemsets. In the real world, each product is composed of numerous components, where they have values different from one another because of their various characteristics such as price, rarity, and other additional features. Such characteristics of the components are important factors that can affect the profits of products. Hence, previous erasable mining methods that ignore them cannot mine meaningful erasable itemsets reflecting the features of the real world.

In order to solve such problems, we propose a new algorithm for discovering erasable itemsets considering weight conditions in incremental databases, called Incremental mining of Weighted Erasable Itemsets (*IWEI*). The main contributions of our algorithm are as follows.

1. The proposed algorithm can store product information more efficiently than previous ones by employing an improved tree structure, and it also efficiently performs incremental erasable itemset mining operations on the basis of tree restructuring and updating processes.
2. We propose an efficient algorithm that extracts erasable itemsets reflecting information constantly accumulated from the real world. An additional list structure is used to improve resource efficiency of the algorithm by reducing duplicated, unnecessary information in the tree structure.
3. We suggest a technique that utilizes weight conditions considering the characteristics of erasable itemset mining. This technique uses weight information to deal with different values of items in the real world; thereby, the proposed algorithm effectively reduces the search space needed for mining patterns as well as mines weighted erasable itemsets that are more useful than previous erasable patterns.
4. Comparisons among our method and tree-based state-of-the-art erasable itemset mining methods are conducted through multifarious analyses including performance evaluations of various real and synthetic datasets in multiple aspects. The results show that the proposed algorithm has better performance than that of the previous ones in terms of runtime, memory usage, pattern quality, and scalability.

This paper is organized as follows. We describe related work and compare the proposed method with previous approaches in Section 2. We introduce preliminaries for the related work and our method in Section 3 and explain details of the proposed algorithm in Section 4. After that, we show the experimental results of our method in Section 5 and finally discuss and conclude this paper in Sections 6 and 7, respectively.

2. Related work

2.1. Mining erasable itemsets

The concept of erasable itemset mining was proposed to solve financial crises that may occur in manufacturing plants. Given a product database, each transaction signifies a production line and items of the transaction become components composing the corresponding product. Then, an erasable itemset means a set of components to be excluded from previous production lines, where each erasable itemset has a total profit lower than or equal to a user-specified threshold. Various algorithms have been proposed (Deng, 2013; Nguyen et al., 2014) to mine such erasable itemsets. In an initial erasable pattern mining approach based on a level-wise method, *META* (Deng et al., 2009), the concept of erasable itemsets has been proposed and the *anti-monotone property* used in frequent itemset mining has been adopted in its erasable itemset mining process. *META* first finds candidate erasable itemsets scanning a product database multiple times and then extracts valid erasable itemsets from the candidates.

Data structures employed in traditional erasable itemset mining are mainly categorized as two types: list and tree structures. List-based approaches such as *VME* (Deng and Xu, 2010) and *MEI* (Le and Vo, 2014) store product information including IDs and profits of products into list structures and then utilize this information during their own mining processes. As an improved version of *META*, *VME* uses its own list data structure, called *PID-list*. If an item, i, is contained in several products of a given database, $\{P_{i1}, P_{i2}, ..., P_{in}\}$, *PID-list* of i is composed as follows: $\{\langle P_{i1}.PID, P_{i1}.Profit\rangle, \langle P_{i2}.PID, P_{i2}.Profit\rangle, ..., \langle P_{in}.PID, P_{in}.Profit\rangle\}$. While *META* conducts a numerous number of database scanning operations in order to calculate profits of erasable itemsets, *VME* requires two database scans to generate *PID-list* and then performs its mining works using the list structure without any additional database scan. *MEI* is another list-based algorithm that has improved the resource efficiency problem of the previous approaches. The algorithm employs *dPidset*, which is an advanced list structure of *PID-list*. Basically, these two list structures have the same form. However, in *dPidset*, there is no data duplication that usually occurs in *PID-list* when performing itemset expanding processes. Moreover, since the itemset expanding process of *MEI* is performed by comparing two different *dPidsets*, it uses resources more efficiently than *VME*.

There are other erasable itemset mining approaches based on tree structures (Deng and Xu, 2012; Le et al., 2013). Unlike list-based approaches, tree-based methods construct tree structures from product databases and then conduct their mining works using the constructed trees without any further database scan. *MERIT* (Deng and Xu, 2012) is a representative tree-based algorithm and uses its own tree structure, *WPPC-tree*, where the tree consists of multiple nodes and each node has a *WPPC-code*. Given a node, N, its *WPPC-code* is denoted as $\langle(N.\text{pre-order}, N.\text{post-order}):N.\text{weight}\rangle$. In addition, as in the case of list-based methods that perform erasable itemset expanding operations comparing information of lists such as *PID-list* and *dPidset*, *MERIT* also similarly conducts such works through comparison of *NC-set*, a set of *WPPC-codes*. Although *MERIT* is faster than *META*, its mining process causes information losses of erasable itemsets because of its performance improving technique. To overcome this issue, *MERIT*+ (Le et al., 2013) has been devised. *dMERIT*+ (Le et al., 2013) is an advanced version of *MERIT*+ that can mine erasable itemsets more efficiently by removing redundant information of *MERIT*+. By using its own list structure, *dNC-set*, *dMERIT*+ mines erasable itemsets with smaller runtime and memory resources than those of *MERIT*+.

Although previous algorithms discover erasable itemsets, their coverages are limited to static product databases only. Therefore, in order to process dynamic databases, they have to conduct mining operations from scratch whenever the databases are updated. In particular, previous tree-based algorithms construct their own tree structures by scanning databases several times. Hence, if such database scans are iterated many times by frequent database updates, enormous overheads can occur. Moreover, it is hard for them to immediately reflect real time information of databases in this case. Meanwhile, such a problem would not occur in the algorithm proposed in this paper because it is equipped with strategies and techniques for efficiently dealing with incremental product databases.