Engineering Applications of Artificial Intelligence 60 (2017) 83–96

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Mining of frequent patterns with multiple minimum supports

Wensheng Gan[a], Jerry Chun-Wei Lin[a,*], Philippe Fournier-Viger[b], Han-Chieh Chao[a,c], Justin Zhan[d]

[a] School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), China
[b] School of Natural Sciences and Humanities, Harbin Institute of Technology (Shenzhen), China
[c] Department of Computer Science and Information Engineering National Dong Hwa University, Hualien County, Taiwan
[d] Department of Computer Science, University of Nevada, Las Vegas, USA

ARTICLE INFO

Keywords:
Frequent patterns
Multiple minimum supports
Sorted downward closure property
Set-enumeration-tree
DiffSet

ABSTRACT

Frequent pattern mining (FPM) is an important topic in data mining for discovering the implicit but useful information. Many algorithms have been proposed for this task but most of them suffer from an important limitation, which relies on a single uniform minimum support threshold as the sole criterion to identify frequent patterns (FPs). Using a single threshold value to assess the usefulness of all items in a database is inadequate and unfair in real-life applications since each item is different and not all items should be treated as the same. Several algorithms have been developed for mining FPs with multiple minimum supports but most of them suffer from the time-consuming problem and require a large amount of memory. In this paper, we address this issue by introducing the novel approach named **F**requent **P**attern mining with **M**ultiple minimum supports from the **E**numeration-tree (FP-ME). In the developed Set-**E**numeration-**tree** with **M**ultiple minimum supports (ME-tree) structure, a new *sorted downward closure* (*SDC*) property of FPs and the least minimum support (*LMS*) concept with multiple minimum supports are used to effectively prune the search space. The proposed FP-ME algorithm can directly discover FPs from the ME-tree without candidate generation. Moreover, an improved algorithm, named FP-ME$_{\text{DiffSet}}$, is also developed based on the *DiffSet* concept, to further increase mining performance. Substantial experiments on both real-life and synthetic datasets show that the proposed algorithms can not only avoid the "*rare item problem*", but also efficiently and effectively discover the complete set of FPs in transactional databases while considering multiple minimum supports and outperform the state-of-the-art CFP-growth++ algorithm in terms of execution time, memory usage and scalability.

1. Introduction

With the rapid development of sensor technology, knowledge discovery in database (KDD) has become a powerful tool for finding meaningful and valuable information from the amounts of mass data. In the field of data mining, frequent pattern mining (FPM) and association rule mining (ARM) (Chen et al., 1996; Han et al., 2004; Lin et al., 2009) are the fundamental task in data mining and have numerous real-world applications. Most studies in FPM have been extensively studied, such as incremental mining of FPs (Hong et al., 2008; Lin et al., 2009), constrain-based FPM (Hong et al., 2009; Pei and Han, 2002; Zaki and Hsiao, 2002), weighted-based frequent pattern mining (Gan et al., 2016; Lin et al., 2015a, 2015d; Vo et al., 2013), and interesting FPM (Geng and Hamilton, 2006; Lin et al., 2015c, 2016), among others (Grahne and Zhu, 2005; Han et al., 2004; Pei et al., 2001; Schlegel et al., 2011). In general, most of them focus on developing efficient algorithms to mine FPs in transactional databases (Chen et al., 1996; Han et al., 2004).

The above approaches suffers, however, from an important limitation, which has to utilize a single minimum support threshold as the measure to identify the set of FPs. Using a single support threshold value to assess the occur frequency of all items in a database is inadequate since each item is different and they should not be treated the same. The reasons are described as follows. In retail business, customers may buy some items with a high frequency but buy other items very rarely. In general, the necessary, consumable and low-price products are frequently bought, while the luxury goods, electric appliances and high-price products are rarely bought. For the above situations, if the *minsup* is set too high, all the discovered patterns are concerned with those low-price products, which only contribute a small

* Corresponding author.
E-mail addresses: wsgan001@gmail.com (W. Gan), jerrylin@ieee.org (J.C.-W. Lin), philfv@hitsz.edu.cn (P. Fournier-Viger), hcc@ndhu.edu.tw (H.-C. Chao), justin.zhan@unlv.edu (J. Zhan).

http://dx.doi.org/10.1016/j.engappai.2017.01.009
Received 16 June 2016; Received in revised form 13 December 2016; Accepted 12 January 2017

portion of the profit to the business. Otherwise, if the *minsup* is set too low, it generates too many meaningless FPs and the decision makers may be confused and misled to make the wrong decisions. Thus, a traditional FPM algorithm may discover many itemsets that are frequent but generate a low profit and fail to discover itemsets that are rarer but generate high profit. For example, clothes i.e., {*shirt*, *tie, trousers*, *suits*} occurs much more frequent than {*diamond*} in a supermarket but having positive contribution to increase the profit amount. If the value of *minsup* is set too high, though the rule {*shirt, tie* → *trousers*} can be found, we would never find the rule {*shirt, tie* → *diamond*}. To find the second rule, the *minsup* is necessary to set very low. However, this will cause lots of meaningless rules to be found at the same time.

To address the "*rare item problem*" in FPM (Liu et al., 1999; Lee et al., 2005), the problem of mining frequent patterns with multiple minimum supports (FP-MMS) has been studied. Liu et al. (1999) first introduced the problem of FPM with multiple minimum supports, and also proposed the MSApriori algorithm by extending the level-wise Apriori algorithm. The goal of FP-MMS is to discover the valuable set of patterns that are "frequent" for the users, i.e., frequent patterns (FPs), it allows the users to freely set multiple minimum supports instead of an uniform minimum support to reflect different natures and frequencies of all items. Up to now, several approaches have been designed for the mining task of FP-MMS, such as MSApriori (Liu et al., 1999), MMS_Cumulate and MMS_Stratify (Tseng and Lin, 2007), CFP-growth (Hu and Chen, 2006), CFP-growth++ (Kiran and Reddy, 2011), and so on. As the enhanced algorithm of CFP-growth, the state-of-the-art CFP-growth++ was proposed by extending the well-known FP-growth approach to mine FPs from a condensed CFP-tree structure (Kiran and Reddy, 2011). However, the mining efficiency of them is still a major problem. For example, the FP-MMS still suffers from the time-consuming and memory usage problems. It is thus quite challenge and critically important to design an efficient algorithm to solve this problem.

In this paper, we propose a novel mining model named mining FPs from the Set-enumeration-tree with multiple minimum supports (FP-ME) is designed to address this important research gap. In the designed FP-ME model, each item has its own unique minimum support threshold instead of a single uniform minimum support threshold for all items. This increases the applicability of FPM in real-life situations, which allows the user to specify multiple minimum supports and reflect different nature and frequency of items. The key contributions of this paper are summarized as follows:

1. In contrast to the Apriori-like and FP-growth-based approaches, we propose a novel **F**requent **P**atterns with **M**ultiple minimum supports from the Set-**E**numeration-tree (abbreviated as FP-ME) algorithm to directly extract FPs. It allows mining FPs by considering different minimum supports for each item instead of using a single minimum support threshold.
2. Based on the proposed compact tree structure named Set-**E**numeration-**tree** with **M**ultiple minimum supports (ME-tree), a new *sorted downward closure* (*SDC*) property of FPs w.r.t. the conditional anti-monotonicity of FPs, and the least minimum support (*LMS*) concept w.r.t. the global anti-monotonicity of FPs with multiple minimum supports, can guarantee the correctness and completeness of derived results. The FP-ME algorithm can directly discover FPs by spanning the ME-tree without candidate generation-and-test approach and multiple time-consuming database scans, which can greatly reduce the running time and memory consumption.
3. The *DiffSet* concept is further extended to early prune the huge amount of unpromising patterns, thus speeding up the process for mining FPs. The improved FP-ME$_{\text{DiffSet}}$ algorithm can discover the complete set of FPs with only two database scans, which greatly decreases the execution time and memory consumption.
4. Extensive experiments were conducted on both real-life and synthetic datasets to evaluate the performance of the proposed algorithms. Results showed that the proposed algorithms can efficiently identify all FPs from a transactional database while considering multiple minimum supports and can avoid the "*rare item problem*". Both the proposed two algorithms significantly outperform the state-of-the-art CFP-growth++ algorithm in terms of execution time, memory usage and scalability. Besides, the improved algorithm considerably outperforms the baseline algorithm.

The rest of this paper is organized as follows. Related work is briefly reviewed in Section 2. Preliminaries and the problem statement of frequent pattern mining with multiple minimum supports (FP-MMS) are presented in Section 3. The proposed baseline FP-ME algorithm and the improved FP-ME$_{\text{DiffSet}}$ algorithm are presented in Section 4. An experimental evaluation comparing the performance of the proposed approaches is provided in Section 5. Finally, conclusions are drawn in Section 6.

2. Related work

To confront the "*rare item problem*" which has been presented above, the problem of FPM involving rare items using multiple minimum support thresholds has been studied. Up to now, several algorithms such as MSApriori (Liu et al., 1999), MMS_Cumulate and MMS_Stratify (Tseng and Lin, 2007), CFP-growth (Hu and Chen, 2006), CFP-growth++ (Kiran and Reddy, 2011), REMMAR (Liu et al., 2011) and FQSP-MMS (Huang, 2013), etc. have been proposed. Characteristics of the related algorithms are shown in Table 1. Those algorithms allow the user to specify multiple minimum supports (MMS) instead of a single minimum support to reflect the nature of the items and their varied frequency in the database.

MSApriori (Liu et al., 1999) is the first framework to address the FP-MMS problem. In MSApriori, each item is associated with a specific *minimum item support* (*MIS*) value, and each pattern satisfy a *minsup* depending upon the *MIS* value of the items within it. The MSApriori extends the well-known Apriori algorithm to mine FPs or ARs by considering multiple minimum supports, rare ARs can be discovered without generating a large number of meaningless rules (Liu et al., 1999). MSApriori uses *sorted closure* property to reduce the search space, but it may easily suffer from the combinatorial explosion. Then, an improved tree-based algorithm named Conditional Frequent Pattern-growth (CFP-growth) was proposed (Hu and Chen, 2006). CFP-growth mines FPs with multiple minimum supports using the pattern growth method based on a new MIS-Tree structure. CFP-growth recursively creates a series of conditional trees to generate all desired FPs. Then, Tseng et al. proposed two algorithms, MMS_Cumulate (Tseng and Lin, 2007) and MMS_Stratify (Tseng and Lin, 2007), to mine ARs in the presence of taxonomies, which allows any form of user-specified multiple minimum supports. In addition, mining ARs with multiple minimum supports using maximum constraints was introduced by using an Apriori-like approach, and it showed that the number of the derived FPs and ARs using maximum constraints is less than those using the minimum constraints (Lee et al., 2005). An enhanced CFP-growth++(Kiran and Reddy, 2011) was then proposed, which employs *LMS* (least minimum support) instead of *MIN* and three improved strategies to reduce the search space and improve performance. The CFP-growth and CFP-growth++ algorithms are, however, needed to perform an exhaustive search on the constructed conditional trees to discover the complete set of FPs, which causing performance problem. A key drawback of the two pattern-growth approaches is how to reduce the traversal and construction cost of a series of conditional sub-trees, and reduce the total number of conditional sub-trees which are needed to be constructed for deriving FPs. They are, however, always hard to both reduce the traversal and the construction cost at the same time. Thus, it is quite