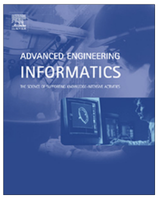
Advanced Engineering Informatics xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Advanced Engineering Informatics

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



A fast updated algorithm to maintain the discovered high-utility itemsets for transaction modification[☆]

Jerry Chun-Wei Lin[a,b,*], Wensheng Gan[a], Tzung-Pei Hong[c,d]

[a] *Innovative Information Industry Research Center (IIIRC), School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, HIT Campus, Shenzhen University Town, Xili, Shenzhen, PR China*
[b] *Shenzhen Key Laboratory of Internet Information Collaboration, School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, HIT Campus, Shenzhen University Town, Xili, Shenzhen, PR China*
[c] *Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung, Taiwan, ROC*
[d] *Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, ROC*

ARTICLE INFO

Article history:
Received 10 December 2014
Received in revised form 11 May 2015
Accepted 12 May 2015
Available online xxxx

Keywords:
Utility mining
Pre-large concept
Transaction modification
Two-Phase procedure
Dynamic databases

ABSTRACT

High-utility itemsets mining (HUIM) is a critical issue which concerns not only the occurrence frequencies of itemsets in association-rule mining (ARM), but also the factors of quantity and profit in real-life applications. Many algorithms have been developed to efficiently mine high-utility itemsets (HUIs) from a static database. Discovered HUIs may become invalid or new HUIs may arise when transactions are inserted, deleted or modified. Existing approaches are required to re-process the updated database and re-mine HUIs each time, as previously discovered HUIs are not maintained. Previously, a pre-large concept was proposed to efficiently maintain and update the discovered information in ARM, which cannot be directly applied into HUIM. In this paper, a maintenance (PRE-HUI-MOD) algorithm with transaction modification based on a new pre-large strategy is presented to efficiently maintain and update the discovered HUIs. When the transactions are consequentially modified from the original database, the discovered information is divided into three parts with nine cases. A specific procedure is then performed to maintain and update the discovered information for each case. Based on the designed PRE-HUI-MOD algorithm, it is unnecessary to rescan original database until the accumulative total utility of the modified transactions achieves the designed safety bound, which can greatly reduce the computations of multiple database scans when compared to the batch-mode approaches.

1. Introduction

Data mining is used to reveal meaningful or useful information from an extensive database. Discovered information or knowledge can be used to aid mangers or retailers in making efficient decisions or strategies. Association-rule mining (ARM) [2,3,7,22] is a common way to present the binary relationships between the purchased products in market basket analysis. Agarwal et al. designed the Apriori algorithm [3] to first mine frequent itemsets (FIs) in a level-wise way based on the minimum support threshold, and then combine the discovered FIs to generate association rules (ARs) based on the minimum confidence threshold. It is insufficient to mine high profitable itemsets by only considering occurrence frequency in ARM. High-utility itemsets mining (HUIM) was proposed to find rare itemsets with high profits by considering both profit and quantity factors [6,25,26]. An item/itemset is considered as a high-utility itemset (HUI) if its utility is no less than the minimum utility threshold. Liu et al. presented a Two-Phase model to maintain the transaction-weighted downward closure (TWDC) property of the designed high transaction-weighted utilization itemsets (HTWUIs) [19]. An additional database scan is required to determine the actual utilities of the remaining HTWUIs. Many algorithms have been proposed to efficiently mine HUIs from a static database in batch mode [5,10,15,23]. When the transactions are frequently changed through insertion [8], deletion [9] or modification [13], most approaches discard the previously discovered information and then perform a conventional scan on the updated database to re-mine the required information, which is not practical in real-life situations.

[☆] Handled by C.-H. Chen.

[*] Corresponding author at: Innovative Information Industry Research Center (IIIRC), School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, HIT Campus, Shenzhen University Town, Xili, Shenzhen, PR China.

E-mail addresses: jerrylin@ieee.org (J.C.-W. Lin), wsgan001@gmail.com (W. Gan), tphong@nuk.edu.tw (T.-P. Hong).

http://dx.doi.org/10.1016/j.aei.2015.05.003
1474-0346/© 2015 Elsevier Ltd. All rights reserved.

Table 1
A quantitative database.

TID	Transaction
1	$(A{:}2)$, $(B{:}3)$, $(D{:}1)$
2	$(A{:}2)$, $(B{:}2)$, $(D{:}1)$, $(E{:}1)$
3	$(C{:}1)$, $(D{:}2)$, $(G{:}1)$
4	$(A{:}3)$, $(B{:}2)$, $(D{:}1)$, $(F{:}1)$
5	$(A{:}1)$, $(B{:}1)$, $(C{:}1)$, $(D{:}1)$
6	$(A{:}1)$, $(D{:}1)$, $(E{:}1)$
7	$(B{:}2)$, $(C{:}1)$
8	$(A{:}1)$, $(B{:}2)$, $(D{:}1)$, $(G{:}1)$

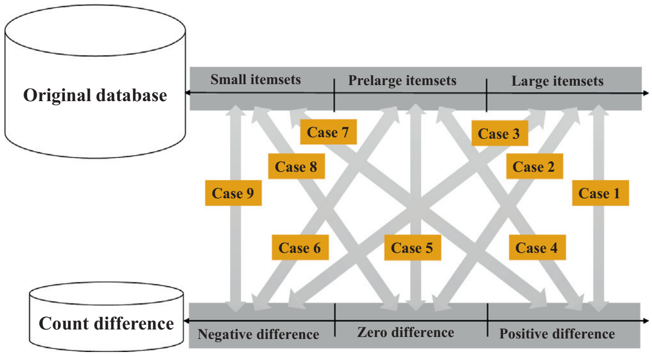


Fig. 1. Nine cases are created due to transaction modification.

Since transactions in a real-life environment may change dynamically, it is a critical issue to efficiently maintain the discovered knowledge without rescanning the original database each time. In the past, Fast UPdated (FUP) [8], FUP2 [9], and pre-large [11,12] concepts were proposed to maintain and update the discovered information for transaction insertion and transaction deletion, respectively. For HUIM, it is not an easy task to maintain the discovered HUIs compared to traditional ARM since quantity and profit factors are both concerned in HUIM. Fewer maintenance algorithms for HUIM were proposed to efficiently maintain and update the discovered HUIs in a dynamic environment with transaction insertion [5,16] and transaction deletion [14]. As one of the three common operations (transaction insertion, deletion, and modification) in databases, transaction modification is also commonly seen in real-life situations since many typos or errors may occur when the collected data from periodic transactions is inputted into a computer using a keyboard. For instance, the example database as shown in Table 1, when the transaction $\{6, \langle(A{:}1), (D{:}1), (E{:}1)\rangle\}$ in the database needs to be modified as $\{6, \langle(A{:}2), (B{:}1), (F{:}1), (G{:}9)\rangle\}$ due to the previous typos, the final HUIs of the updated database will be changed. Thus, some information may become invalid or new information may arise. Although the maintenance process of transaction modification can be done by two procedures which can be performed in either order: first, delete the incorrect transactions, and second, insert the correct transactions. It requires twice computations, which is very time-consuming and impractical. Thus, an efficient maintenance process for transaction modification is a critical issue and necessary in the dynamic environment. However, the issue of HUIM with transaction modification has not been proposed, to the best of our knowledge.

In this paper, a novel PRE-HUI-MOD algorithm is presented to maintain and update the discovered HTWUIs and HUIs with transaction modification. The proposed PRE-HUI-MOD algorithm adopts a Two-Phase model [19] to set the effective upper bound for discovering HTWUIs and HUIs from the original databases. The discovered HTWUIs and all transaction-weighted utilization itemsets in the modified transactions are then divided into three parts with nine cases. Each case is handled by the designed procedure to determine whether the discovered HTWUI will still remain as HTWUI or become non-HTWUI in the updated database. An additional database scan is required to determine the actual HUIs of the remaining HTWUIs. Based on the designed framework with transaction modification, the number of computations can be greatly reduced until the accumulative total utility in the modified transactions achieves the designed safety bound. In addition, previously discovered HTWUIs can be used to help the maintenance process, thus speeding up computations. Major contributions of the proposed approach are listed below.

1. We have designed a maintenance algorithm to efficiently update the discovered HTWUIs for producing HUIs with transaction modification compared to the state-of-the-art algorithms running in batch mode.
2. We extended the pre-large concept in ARM to maintain and update the discovered HUIs and HTWUIs without multiple database scans in HUIM.
3. Based on the designed safety bound mechanism, it is unnecessary to rescan the updated database each time, unlike batch-mode algorithms, until the accumulative total utility achieves the safety bound, which can greatly reduce the database scan computations.
4. Experiments are conducted to show that the proposed maintenance algorithm can efficiently handle the dynamic database with transaction modification of HUIM, and generally has better performance compared to the state-of-the-art batch-mode HUIM algorithms.

2. Related work

In this section, the high-utility itemset mining (HUIM) and the pre-large concept of ARM are briefly reviewed.

2.1. High-utility itemset mining

High-utility mining is an extension of frequent-itemset mining [5,6,10,19,25,26]. It considers both quantity and profit factors to produce more useful and profitable itemsets. An itemset is concerned as a HUI if its utility is no less than the minimum utility threshold. In the past, Chan et al. proposed a top-k mechanism to mine both positive and negative high-utility closed patterns [6]. Highly statistical patterns with a new developed pruning strategy can be discovered efficiently in a level-wise way. Yao et al. applied mathematical properties to mine HUIs based on designed utility constraints [23]. Two pruning methods were proposed to reduce the search space by the utility upper bounds and the expected utility upper bounds. Liu et al. first extended the downward closure (DC) property of ARM into the transaction-weighted downward closure (TWDC) property, and presented a Two-Phase model for mining HUIs [19]. The Two-Phase model is thus designed to efficiently speed up the computations for discovering high transaction-weighted utilization itemsets (HTWUIs) in a level-wise way. An additional database scan is then required to mine actual HUIs from the remaining HTWUIs. The Two-Phase model can also be modified as a parallelized algorithm using a Common Count Partitioned Database (CCPD) strategy on shared memory multi-process architecture [20].

Table 2
Profit table.

Item	*A*	*B*	*C*	*D*	*E*	*F*	*G*
Profit	10	3	5	25	2	3	5

Please cite this article in press as: J.-C.W. Lin et al., A fast updated algorithm to maintain the discovered high-utility itemsets for transaction modification, Adv. Eng. Informat. (2015), http://dx.doi.org/10.1016/j.aei.2015.05.003