Advanced Engineering Informatics 30 (2016) 109–126

Contents lists available at ScienceDirect

Advanced Engineering Informatics

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

Fast algorithms for mining high-utility itemsets with various discount strategies

Jerry Chun-Wei Lin [a,*], Wensheng Gan [a], Philippe Fournier-Viger [b], Tzung-Pei Hong [c,d], Vincent S. Tseng [e]

[a] School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, 518055, China
[b] School of Natural Sciences and Humanities, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, 518055, China
[c] Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung, 811, Taiwan
[d] Department of Computer Science and Engineering, National Sun Yat-sen University, 804, Kaohsiung, Taiwan
[e] Department of Computer Science, National Chiao Tung University, Hsinchu, 300, Taiwan

ARTICLE INFO

Article history:
Received 19 May 2015
Received in revised form 8 February 2016
Accepted 11 February 2016
Available online 2 March 2016

Keywords:
High-utility itemsets
Discount strategies
Downward closure property
Pruning strategies
PNU-list

ABSTRACT

In recent years, mining high-utility itemsets (HUIs) has emerged as a key topic in data mining. It consists of discovering sets of items generating a high profit in a transactional database by considering both purchase quantities and unit profits of items. Many algorithms have been proposed for this task. However, most of them assume the unrealistic assumption that unit profits of items remain unchanged over time. But in real-life, the profit of an item or itemset varies as a function of cost prices, sales prices and sale strategies. Recently, a three-phase algorithm has been proposed to mine HUIs, while considering that each item may have different discount strategies. However, the complete set of HUIs cannot be retrieved based on the traditional TWU model with its defined discount strategies. Moreover, it suffers from the well-known drawbacks of Apriori-based algorithms such as maintaining a huge amount of candidates in memory and repeatedly performing time-consuming database scans. In this paper, a HUI-DTP algorithm for mining HUIs when considering discount strategies of items is introduced. The HUI-DTP is designed as a two-phase algorithm to mine the complete set of HUIs based on a novel downward closure property and a vertical TID-list structure. Furthermore, the HUI-DMiner is an algorithm relying on a compact data structure (Positive-and-Negative Utility-list, PNU-list) and properties of two new pruning strategies to efficiently discover HUIs without candidate generation, while considerably reducing the size of the search space. Moreover, a strategy named Estimated Utility Co-occurrence Strategy which stores the relationships between 2-itemsets is also applied in the improved HUI-DEMiner algorithm to speed up computation. An extensive experimental study carried on several real-life datasets shows that the proposed algorithms outperform the previous best algorithm in terms of runtime, memory consumption and scalability.

1. Introduction

Frequent itemset mining (FIM) or association rule mining (ARM) [2,3,8,14] is a fundamental data mining task, which consists of discovering relationships between items in a transactional database. Many algorithms have also been proposed to efficiently mine frequent itemsets and association rules, usually adopting a level-wise approach [3] or pattern-growth approach [14]. An unrealistic assumption of FIM or ARM is that it only considers binary quantities of items in transactions (a customer may buy one or zero unit of an item). But in real-life, a customer may buy several units of the same item. Furthermore, FIM or ARM does not consider other important factors such as the unit profit of items (all items are considered as having the same unit profit). As a result, many patterns found may be frequent but may generate a very low profit, and thus may be uninteresting for the user. For example, diamond sales may be much less frequent than clothing sales in a department store, but the former has a much larger profit margin than the latter, and may thus be more relevant. Finding associations between items solely based on their occurrence frequency as in traditional FIM or ARM is not suitable to identify these highly profitable itemsets. Factors such as price, quantity and cost should also be considered to analyze and predict customer purchase behavior.

To address some of these limitations, high-utility itemset mining (HUIM) [7,27] was proposed. Under HUIM framework, the

* Corresponding author.
E-mail addresses: jerrylin@ieee.org (J. Chun-Wei Lin), wsgan001@gmail.com (W. Gan), philfv@hitsz.edu.cn (P. Fournier-Viger), tphong@nuk.edu.tw (T.-P. Hong), vtseng@cs.nctu.edu.tw (V.S. Tseng).

http://dx.doi.org/10.1016/j.aei.2016.02.003
1474-0346/

"utility" of an itemset can be measured in terms of quantity and profit according to users' preferences. For example, a user may be interested in finding itemsets yielding a high profit, while another user may focus on discovering itemsets causing low pollution during the manufacturing process. When the utility of an itemset is no less than a minimum utility threshold, it is considered as a high-utility itemset (HUI); otherwise, it is a low-utility itemset. The discovered information of HUIs can be generally used in various applications, such as decision support systems [5,7,27], as well as a framework of data mining based analysis [9], to aid managers or retailers for take efficient decisions or choose the most profitable business strategies for their companies.

In real-life retail stores, the profit earned from the sale of items depends on the cost prices, tag prices and discount strategies. Different stores may use different discount strategies to sell the same products. Furthermore, the use of discount strategies may vary based on time periods. Traditional way of measuring the utility of itemsets in the presence of different discount strategies used for each item is to consider not only the positive profit generated by the sale of items but also the negative profit. For example, a popular discount strategy is to sell some items at high discount or even to give them away, such as mouse and keyboard are always sold at a high discount or even free with laptop. This may yield negative profit for the vendor. In such scenario, however, these products are often cross-promoted with others having a positive profit margin, thus leading to an overall positive profit. A cross promotion is a common phenomenon in marketing management that targets buyers of a product with an offer to purchase the related products, and the appropriate discount strategies may affect users' shopping behavior.

Discovering HUIs with the constraint that each item may be associated with its own discount strategy is highly desirable since discounting is done in most real-life stores. However, most algorithms for HUIM are not designed to handle items with profit values that can vary under different discount strategies, which thus limits their usefulness in real-life. Mining HUIs while considering the discount strategy associated to each item is more computationally expensive than traditional HUIM. Furthermore, another challenge is that the pruning strategies or properties of traditional HUIM cannot be directly adopted to solve this problem. Chu et al. developed the HUINIV-Mine algorithm to discover HUIs while considering items with negative profits (HUINIV) [10]. It is a two-phase algorithm, that is, it overestimates the utility of itemsets to prune the search space, and then scans the database again to calculate the exact utility of itemsets. It suffers from well-known drawbacks, which lead to high execution time and memory consumption. Fournier-Viger adopted the utility-list structure and EUCP strategy in the FHN algorithm for mining HUIs with negative profit more efficiently [17]. Li et al. proposed a three-phase algorithm to discover HUIs under four discount strategies [19]. Because this three-phase algorithm also overestimates the utilities of itemsets as in the Two-Phase model [24], it also suffers from the same drawbacks. Moreover, this algorithm is not designed to handle negative unit profit. If negative unit profit is introduced, the algorithm fails to mine the complete set of HUIs. It is thus a challenge to design an efficient algorithm to mine the whole set of HUIs when both positive and negative items are used, and each item has its own discount strategy.

In this paper, we address this issue by proposing algorithms to mine HUIs when each item has its own discount strategy. The proposed algorithms are quite different from the previous HUINIV-Mine algorithm. The reasons are shown as follows: (1) the approach for calculating profit of all items in the transaction database is not the same as previous works; (2) both positive (includes zero) and negative items profits are taken into account; (3) the used data structure and proposed approaches are different from HUINIV-Mine; (4) since discount strategy affects users' shopping behavior by evaluating the maximal total profit which bring from derived HUIs and those common HUIs, the managers or retailers can find the optimal discount strategies and decisions to get the total revenue maximization. Hence, HUIM with discount strategies is a more general problem than previous ones. The proposed algorithms can be used as efficient tools for various applications, including decision support systems (DSS) [5,7,27], for managers or retailers to discover more useful and meaningful information in many real-life applications. Moreover, based on the various definitions of "utility" (profit, benefit, weight, risk, etc.), the proposed technologies can be applied to other various domains, such as multi-dimensional data analysis, benefits evaluation, and possibilities and risk assessment in engineering informatics [6]. The contributions of this paper are as follows.

1. Several algorithms of high-utility itemset mining with various discount strategies, are designed to reveal more useful and meaningful HUIs when considering items having various discount strategies. The proposed algorithms are more adapted to real-life situations than traditional HUIM.
2. To the best of our knowledge, this is the first paper successfully solves the addressed mining problem. Two algorithms are developed to mine HUIs when considering items having various discount strategies without losing any HUIs.
3. The HUI-DTP algorithm is proposed as a baseline algorithm. It relies on a level-wise search to mine HUIs and integrates a new downward closure property to prune unpromising candidates without losing any HUIs.
4. The HUI-DMiner algorithm is proposed as an efficient algorithm. It relies on a vertical structure, called Positive-and-Negative Utility-list (PNU-list), to mine HUIs without generating and maintaining candidates in memory. Two efficient pruning strategies are further proposed to reduce the search space, and thus speed up the discovery of HUIs. Furthermore, a structure called the Estimated Utility Co-occurrence Structure (EUCS) is also used to prune the search space.
5. Based on the designed algorithms, the complete set of HUIs can be efficiently discovered. An extensive experimental study carried on several real-life datasets shows that the proposed algorithms largely outperform the previous best algorithm in terms of runtime, memory consumption and scalability.

The remaining of this paper is organized as follows. Related work in high-utility itemset mining and comparative analysis are reviewed in Section 2. The proposed work of high-utility itemset mining with various discount strategies is described in Section 3. The proposed HUI-DTP and HUI-DMiner algorithms are respectively presented in Sections 4 and 5. Experiments are conducted in Section 6. Finally, Section 7 draws conclusions and provides a discussion.

2. Related work

This section briefly reviews related studies on high-utility itemset mining and presents the difference and relationship between this paper and previous works.

2.1. High-utility itemset mining

Traditional ARM has several important limitations. First, it can only discover relationships between items in a binary database, i.e. where items may appear not more than once in each