Information Sciences 179 (2009) 899–924

Contents lists available at ScienceDirect

Information Sciences

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

Top-down mining of frequent closed patterns from very high dimensional data ☆

Hongyan Liu [a,*], Xiaoyu Wang [a], Jun He [b], Jiawei Han [c], Dong Xin [c], Zheng Shao [c]

[a] *Department of Management Science and Engineering, Tsinghua University, Beijing 100084, China*
[b] *Department of Computer Science, Renmin University of China, Beijing 100872, China*
[c] *Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA*

ARTICLE INFO

Article history:
Received 15 February 2008
Received in revised form 25 November 2008
Accepted 26 November 2008

Keywords:
Data mining
Association rules
Frequent patterns
High dimensional data

ABSTRACT

Frequent pattern mining is an essential theme in data mining. Existing algorithms usually use a bottom-up search strategy. However, for very high dimensional data, this strategy cannot fully utilize the minimum support constraint to prune the rowset search space. In this paper, we propose a new method called top-down mining together with a novel row enumeration tree to make full use of the pruning power of the minimum support constraint. Furthermore, to efficiently check if a rowset is closed, we develop a method called the *trace-based* method. Based on these methods, an algorithm called *TD-Close* is designed for mining a complete set of frequent closed patterns. To enhance its performance further, we improve it by using new pruning strategies and new data structures that lead to a new algorithm *TTD-Close*. Our performance study shows that the top-down strategy is effective in cutting down search space and saving memory space, while the trace-based method facilitates the closeness-checking. As a result, the algorithm *TTD-Close* outperforms the bottom-up search algorithms such as *Carpenter* and *FPclose* in most cases. It also runs faster than *TD-Close*.

1. Introduction

Frequent pattern mining is an essential theme in data mining, with broad applications [9,13,14]. There are many frequent pattern mining algorithms that have been proposed, but most of them [1,10–12,21,22,26,27] deal with transactional databases that usually contain a huge number of rows (or tuples) but only a small number of dimensions (or attributes, columns). These algorithms usually search the ***itemset*** (*i.e.*, set of items) space; therefore, they are called **column enumeration-based algorithms**. However, many applications may involve another kind of database, which is characterized by a small number of rows but a large number of dimensions. We call this kind of data *very high dimensional data*. Gene expression matrices analysis in bioinformatics [2,3,5,15,18,25] and text processing [23] are examples of this kind of application.

Due to the exponential number of combinations of dimensions, very high dimensional data poses great challenges on efficient frequent pattern mining as to most other data mining algorithms. To solve the problem, a new approach is proposed to

☆ A preliminary version of this work has been published in the Proceedings of 2006 SIAM Data Mining Conference (SIDM 2006). Substantial new technical materials have been added to this journal submission. This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 70871068, 70890083 and 70621061, and by the US National Science Foundation NSF IIS-03-08215 and IIS-05-13857.

* Corresponding author. Tel.: +86 10 6278 9961; fax: +86 10 6278 4555.

E-mail addresses: hyliu@tsinghua.edu.cn (H. Liu), wangxy3@sem.tsinghua.edu.cn (X. Wang), hejun@ruc.edu.cn (J. He), hanj@uiuc.edu (J. Han), dongxin@uiuc.edu (D. Xin).

0020-0255/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2008.11.033

search the ***rowset*** (*i.e.,* set of rows) space rather than the itemset space for frequent pattern mining [19,24]. We call these kinds of algorithms **row enumeration-based algorithms**. Based on this work, several other algorithms can also be developed for frequent pattern mining or interesting rule group mining [3,4,20].

The two kinds of algorithms mentioned above adopt a bottom-up search strategy that searches from small itemsets to large itemsets. For column enumeration-based algorithms, bottom-up search can use the anti-monotone property of minimum support threshold [1] to prune the search space, which means that if an itemset of size k is not frequent, any of its super sets of size $(k+1)$ is not frequent either. However, for row enumeration-based algorithms, bottom-up search may suffer from the following two inefficiencies:

(1) The potential search space is large. This is due to the fact that bottom-up search in the rowset space cannot utilize the anti-monotone property of minimum support threshold to prune the space. As a result, many unnecessary searches need to be done.
(2) The data set needs to be checked repeatedly to see if an itemset is closed.

In this paper, we propose a top-down search strategy for row enumeration-based algorithms to overcome these drawbacks. To do that, we develop and integrate the following four techniques:

(1) A novel row enumeration tree is constructed. By traversing this tree, a top-down search of the rowset space is conducted, that is, the rowsets are examined from the large ones to the small ones. This style of search order can take advantage of the anti-monotone heuristic to prune the search space. For example, rowsets smaller than k can be pruned if the minimum support threshold is set at k.
(2) A divide-and-conquer top-down search technique is proposed to mine frequent closed itemsets efficiently. Using this technique, the search space can be partitioned into separate subspaces and handled recursively by the top-down search technique.
(3) A new *trace-based* ***closeness-checking*** method is developed to check whether an itemset is closed. Further, a *output-based method* is designed based on a traditional closeness-checking method and is further improved according to the characteristics of the new top-down search order. Our experiments show that the trace-based *closeness-checking* method performs more efficiently than the output-based method. Unlike other existing closeness-checking methods, the trace-based *closeness-checking* method does not need to scan the data set or the result set of itemsets repeatedly. Moreover, this method can be easily integrated with the top-down search process.
(4) Several other pruning strategies are also developed. These strategies can reduce the search space further.

Integrating the above four techniques, an algorithm called *TD-Close* was designed and implemented by an array-based data structure. To improve its performance further, another algorithm called *TTD-Close* was then developed. It uses more effective pruning strategies aside from those used in *TD-Close*, and it is implemented with a tree-based data structure. A comprehensive experimental study was conducted on both synthetic data sets and real world data sets to compare the performance of these two algorithms as well as with two other frequent pattern mining algorithms: (1) *FPclose* [7], a representative column enumeration-based algorithm, and (2) *Carpenter* [19], a representative row enumeration-based algorithm. Our experimental results confirm the effectiveness of the new trace-based *closeness-checking method* and the improvement of *TTD-Close* over *TD-Close*. Furthermore, both analysis and experimental results indicate that using the top-down search method, *TD-Close* and *TTD-Close* can cut down more search space and use less memory space than *Carpenter* especially when the minimum support is high. It can also achieve better performance than *FPclose* when the minimum support is low.

The remainder of the paper is organized as follows: Section 2 introduces the transposition method. Section 3 describes the proposed top-down search technique. Section 4 presents the new *closeness-checking* approach. Section 5 develops the two new frequent closed pattern mining algorithms based on the top-down search technique and trace-based *closeness-checking* approach. Section 6 presents our performance study, and Section 7 describes the related work. Finally, Section 8 concludes our study and discusses some future research issues.

2. Frequent pattern mining by transposition

2.1. Definition

Let T be a discretized data table (or data set) composed of a set of n rows $\mathscr{S} = \{r_1, r_2, \ldots, r_n\}$, where r_j $(j = 1,\ldots, n)$ is a row ID or *rid* for short. Each row corresponds to a sample consisting of a set of discrete values or intervals, and $\mathscr{V}$ is the complete set of these values or intervals $\mathscr{V} = \{i_1, i_2, \ldots, i_m\}$. For simplicity, we call each i_k $(k = 1,\ldots,m)$ an *item* and a set of items $I \subseteq \mathscr{V}$ an **itemset**. An itemset with k items is called a ***k-itemset***.

Example 1 (*Table T*). Table 1 shows an example of table T with four dimensions (columns): A, B, C, and D. For simplicity, we use number j $(j = 1,2,\ldots,n)$ instead of r_j to represent each *rid*. This table contains five rows and nine items, so $\mathscr{S} = \{1,2,3,4,5\}$ and $\mathscr{V} = \{a_1, a_2, b_1, b_2, c_1, c_2, d_1, d_2, d_3\}$.