A Scalable Algorithm for Constructing Frequent Pattern Tree

Zailani Abdullah, Department of Computer Science, Universiti Malaysia, Terengganu, Malaysia

Tutut Herawan, Department of Mathematics Education, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

A. Noraziah, Computer System and Software Engineering, Universiti Malaysia, Kuantan, Malaysia

Mustafa Mat Deris, Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia

ABSTRACT

Frequent Pattern Tree (FP-Tree) is a compact data structure of representing frequent itemsets. The construction of FP-Tree is very important prior to frequent patterns mining. However, there have been too limited efforts specifically focused on constructing FP-Tree data structure beyond from its original database. In typical FP-Tree construction, besides the prior knowledge on support threshold, it also requires two database scans; first to build and sort the frequent patterns and second to build its prefix paths. Thus, twice database scanning is a key and major limitation in completing the construction of FP-Tree. Therefore, this paper suggests scalable Trie Transformation Technique Algorithm (T3A) to convert our predefined tree data structure, Disorder Support Trie Itemset (DOSTrieIT) into FP-Tree. Experiment results through two UCI benchmark datasets show that the proposed T3A generates FP-Tree up to 3 magnitudes faster than that the benchmarked FP-Growth.

Keywords: Association Rules, Data Mining, Frequent patterns, Frequent Pattern Tree (FP-Tree), Scalable Algorithm

INTRODUCTION

Mining frequent pattern has attracted much research interests (Han, et al., 2000; Han & Pei, 2004; Zaki et al., 1997) for a past decade because of its broad domain applications. The focal point of mining frequent pattern is to find the interesting relationships among set of items in the data repositories. Firstly, it was proposed by Agrawal et al. (1993) and still played important roles in knowledge discovery communities. Recently, more than hundreds of research papers have been published including new or modification of algorithms in order to resolve the problems of mining frequent pattern. Association rule (Agrawal, et al., 1993) is a probabilistic statement about the co-occurrence of the data according to the if-then statement. One of the important components that are typically used in deriving the association rule is frequent patterns. Since both of them are closely interrelated; hence quite a number of studies

DOI: 10.4018/ijiit.2014010103
have been performed heretofore (Abdullah et al., 2013; Herawan et al., 2013a; Herawan et al. 2013b; Abdullah et al., 2012a; Abdullah et al. 2012b; Herawan et al. 2012a; Herawan et al., 2012b; Herawan and Abdullah, 2012; Abdullah et al., 2011a; Abdullah et al., 2011b; Abdullah et al., 2011c; Veeramalai & Kannan, 2011; Ashrafi et al., 2005). In data mining, a set of items in pattern is also defined as an itemset. The itemset is said to be frequent, if it appears equal or exceed to the predefined minimum supports. The item (or itemset) support is defined as a probability of item (or itemset) occurs in the transaction. Besides that, confidence is another alternative measurement used in pair with support. The confidence is defined as a probability of the rule’s consequent that also contain the antecedent in the transaction. Association rules are said to be strong if it meets the minimum confidence.

The main problem in frequent pattern mining is to achieve an efficiency of managing the huge amount of data in computer’s memory. As a result, frequent pattern tree (FP-Tree) (Han, et al., 2000) has become one of the alternative data structures to represent the vast transactional of database in a compressed manner. Several variations of constructing or updating the FP-Tree have been proposed and it can be categorized into multiple and single database scans. For the first category and including FP-Tree (Han, et al., 2000), the related studies are Adjusting FP-Tree for Incremental Mining (AFPIM) (Koh & Shieh, 2004) and Extending FP-tree for Incremental Mining (EPFIM) (Li, et al., 2006). The related researches in the second category are Compressed and Arranged Transaction Sequence (CATS) tree (Cheung & Zaïane, 2003), Branch Sorting Method (BSM) (Tanbeer, et al., 2009) and Batch Incremented Tree (BIT) (Totad, et al., 2010).

However, there are two major drawbacks encountered from the aforesaid studies. Firstly, the construction of FP-Tree is still based on extracting the patterns that fulfills the support threshold from the semi structured of its original databases. Secondly, if the existing databases are updated, the current FP-Tree must be rebuilt from the beginning because of the changes in items and patterns supports. In some research extensions, the structure of FP-Tree will be reorganized with extensive modification operations (deletion and insertion) due to additional transactions in databases. In fact, computational extensive in constructing FP-Tree is still an obstacle or outstanding issues in efficiently mining frequent patterns.

Therefore, in order to overcome the mentioned above issues, there three objectives have been derived. Firstly, to study the limitations and strengths of the existing tree data structure as a basis to propose a better solution. Secondly, to design a new tree data structure and algorithm that can facilitate the transformation process of the data into FP-Tree. Finally, to evaluate the performance of the proposed solutions in term of data structure and algorithm using the benchmarked UCI data sets.

In summary, three main contributions of this work are described as follow:

1. In order to improve the limitation of the previous algorithms, a prefix-tree called Disorder Support of Trie Itemset (DOSTrieIT) is proposed and experimented with two UCI dataset (www.ics.uci.edu/~mlearn). DOSTrieIT is constructed in online manner rather than offline (batch) upon receiving every new line of transactions.
2. A new Transformation Trie Technique Algorithm (T3A) is proposed to carry out a transformation from DOSTrieIT to FP-Tree. T3A transforms all prefix paths in DOSTrieIT into prefix path in FP-Tree and not from the original database.
3. The proposed algorithm is very efficient in memory consumption while generating a complete set of FP-Tree. The construction of FP-Tree is not relied on semi-structured of original databases anymore but it is based on the compressed data structure (DOSTrieIT). Moreover, T3A provides a good mechanism to smoothly transform from the proposed data structure into FP-Tree.
Agile Workflow Technology and Case-Based Change Reuse for Long-Term Processes
Mirjam Minor, Alexander Tartakovski and Daniel Schmalen (2008). *International Journal of Intelligent Information Technologies (pp. 80-98).*
www.igi-global.com/article/agile-workflow-technology-case-based/2431?camid=4v1a