International Journal of Information and Computation Technology.
ISSN 0974-2239 Volume 4, Number 5 (2014), pp. 475-478

http://www. irphouse.com /ijict.htm

Efficient Data Mining Algorithm for Reducing High Toll Transactions

Vinay Singh, [2]S. Gavasker and [3]Nitin Kumar

M.TECH(CSE) Final Year, Galgotias University, Greater Noida, U.P, India.
[2]Galgotias University, Greater Noida.
[3]M.TECH(CSE) Final Year, Galgotias University, Greater Noida, U.P, India.

Abstract

Mining Cost Efficient item-sets from a transactional database refers to the discovery of transaction sets with Cost Efficient characteristics improving overall profits. Although a number of relevant algorithms have been proposed in recent years, they incur the problem of producing a large number of candidate item-sets for Cost Efficient item-sets. Such a large number of candidate item-sets degrade the mining performance in terms of execution time and space requirement.
The situation may become worse when the database contains lots of long transactions or long Cost Efficient item-sets. In this paper, we propose algorithm, for mining Cost Efficient item-sets with a set of effective strategies for pruning candidate item-sets. The information of Cost Efficient item-sets is maintained in a tree-based data structure such that candidate item-sets can be generated efficiently with only two scans of database.
The performance of these algorithms is compared with the state-of-the-art algorithms on many types of both real and synthetic data sets. Experimental results show that the proposed algorithms, especially Improved utility pattern, not only reduce the number of candidates effectively but also outperform other algorithms substantially in terms of runtime, especially when databases contain lots of long transactions.

Keywords: Data mining, transactional data base, pruning, utility pattern.

1. Introduction and Problem Related to Algorithms

Experimental results in this section show that the proposed methods outperform the state- of the art algorithms almost in all cases on both real and synthetic data sets. The reasons are described as follows. First, node utilities in the nodes of global utility Tree are much less than TWUs in the nodes of Tree since DGU and DGN effectively decrease overestimated utilities during the construction of a global Utility tree. Second, utility pattern growth and improved utility pattern generate much fewer candidates than FP-growth since DLU, DLN,DNU, and DNN are applied during the construction of local UP Trees. By the proposed algorithms with the strategies, generations of candidates in phase I can be more efficient since lots of candidates to be pruned with the help of any effective algorithm for reduction of any toll .

2. Proposed Mining Methodology

Utility tree, a basic method for generating PHUIs is to mine Utility tree by FP-Growth. However too many candidates will be generated. Thus, we propose an algorithm Utility pattern growth by pushing two more strategies into the framework of FP-Growth. By the strategies, overestimated utilities of item sets can be decreased and thus the number of PHUIs can be further reduced.

To address this issue, we propose two novel algorithms as well as a compact data structure for efficiently discovering Cost Efficient item sets from transactional databases. Major contributions of this work are summarized as follows:

1. Two algorithms, named utility pattern growth (UP Growth) and Improved utility pattern, and a compact tree structure, called utility pattern tree (Utility tree), for discovering Cost Efficient item sets and maintaining important information related to utility patterns within databases are proposed. High-utility item sets can be generated from Utility tree efficiently with only two scans of original databases.
2. Several strategies are proposed for facilitating the mining processes of Utility pattern growth and Utility pattern growth by maintaining only essential information in Utility tree. By these strategies, overestimated utilities of candidates can be well reduced by discarding utilities of the items that cannot be Cost Efficient or are not involved in the search space. The proposed strategies can not only decrease the overestimated utilities of PHUIs but also greatly reduce the number of candidates.
3. Different types of both real and synthetic data sets are used in a series of experiments to compare the performance of the proposed algorithms with the state-of-the-art utility mining algorithms. Experimental results show that Utility pattern growth and Improved utility pattern outperform other algorithms substantially in terms of execution time, especially when databases contain lots of long transactions or low transcations.

3. IUPG-Algorithm (Utility Pattern Growth Algorithm)

Input: Transaction database D, user specified threshold.
Output: high utility itemsets.
Begin
Steps under this algorithm are mentioned to be as follows:

1. Scan database of transactions Td ϵ D
2. Determine transaction utility of Td in D and TWU of itemset (X)
3. Compute min_sup (MTWU * user specified threshold)
4. If (TWU(X) ≤ min_sup) then Remove Items from transaction database
5. Else insert into header table H and to keep the items in he descending order.
6. Repeat step 4 & 5 until end of the D.
7. Insert Td into global UP-Tree
8. Apply DGU and DGN strategies on global UP- tree
9. Re-construct the UP-Tree
10. For each item ai in H do
11. Generate a PHUI Y= X U ai
12. Estimate utility of Y is set as ai's utility value in H
13. Put local promising items in Y-CPB into H
14. Apply strategy DLU to reduce path utilities of the paths
15. Apply strategy DLN and insert paths into T_d

4. Conclusion

Utility pattern growth achieves better performance than FP-Growth by using DLU and DLN to decrease overestimated utilities of item sets. However, the overestimated utilities can be closer to their actual utilities by eliminating the estimated utilities that are closer to actual utilities of unpromising items and descendant nodes. In this section, we propose an improved method, named Improved utility pattern, for reducing overestimated utilities more effectively.

In Utility pattern growth, minimum item utility table is used to reduce the overestimated utilities. In Improved utility pattern, minimal node utilities in each path are used to make the estimated pruning values closer to real utility values of the pruned items in database.

References

[1] Adinarayanareddy B, O Srinivasa Rao, MHM Krishna Prasad " An Improved UP-Growth High Utility Itemset Mining International Journal of Computer Applications (0975 – 8887) Volume 58– No.2, November 2012

[2] Ms. Ruchi Patel, Assistant Professor, Department of Information Technology Gyan Ganga Institute of Technology and Sciences, Jabalpur "A Parallel Approach For High Utility Patterns Mining From Distributed Databases International Journal of Engineering Research & Technology (IJERT)Vol. 1 Issue 8, October – 2012 ISSN: 2278-0181.