Global Journal of Computational Intelligence Research.
ISSN 2249-0000 Volume 1, Number 1 (2011), pp. 49-56

http://www.ripublication.com/gjcir.htm

Improved Index based GenMax Algorithm for Efficient Pruning of Frequent Item Set Mining

C. Sathya[1] and C. Chandrasekaran[2]

[1]Assistant Professor in Computer Applications,
Vellalar College for Women,
Erode, TamilNadu, India.
[2]Reader in Computer Science, Periyar University,
Salem, TamilNadu, India.
E-mail: [1]sathyavenkateswaran@gmail.com, [2]ccsekar@gmail.com

Abstract

Mining frequent item sets is a fundamental and essential problem in many data mining applications such as the discovery of association rules, strong rules, correlations, multidimensional patterns, and many other important discovery tasks. Fast implementations and efficient memory utilization for solving the problems of frequent item sets are highly required in transactional databases. Methods for mining frequent item sets have been implemented using a prefix-tree structure, known as an FP-tree, for storing compressed information about frequent item sets which is too large to fit in memory. GenMax, a search based algorithm is used for mining maximal frequent item sets. GenMax uses a number of optimizations to prune the search space. It uses a technique called progressive focusing to perform maximal checking, and differential set propagation to perform fast frequency computation. Genmax algorithm was not implemented for closed frequent item set

The proposal in this paper present an improved index based enhancement on Genmax algorithm for effective fast and less memory utilized pruning of maximal frequent item and closed frequent item sets. The extension induces a search tree on the set of frequent closed item sets thereby we can completely enumerate closed item sets without duplications. The memory use of mining the maximal frequent item set does not depend on the number of frequent closed item sets, even if there are many frequent closed item sets. The proposed model reduce the number of disk I/Os and make frequent item set mining scale to large transactional databases. Experimental results shows a comparison of improved index based GenMax and existing GenMax for

efficient pruning of maximal frequent and closed frequent item sets in terms of item precision and fastness

Keywords: Item Mining, Index structure, Genmax, Maximal item sets, Closed Item Sets.

Introduction

Mining of association rules from large data sets has been a focused topic in recent data mining research [1]. Many applications at mining associations require that mining be performed at multiple levels of abstraction. For example, besides finding 80 percent of customers that purchase milk may also purchase- bread, it is interesting to allow users to drill-down and show that 75 percent of people buy wheat bread if they buy 2 percent milk. The association relationship in the latter statement is expressed at a lower level of abstraction but carries more specific and concrete information than that in the former. Therefore, a data mining system should provide efficient methods for mining multiple-level association rules.

To explore multiple-level association rule mining,[5] one needs to provide i.e., data at multiple levels of abstraction, and efficient methods for multiple-level rule mining. The first requirement can be satisfied by providing concept taxonomies from the primitive level concepts to higher levels. In many applications, the taxonomy information is either stored implicitly in the database, such as, Milka Wonder wheat bread is a wheat bread which is in turn a bread, or provided by experts or users, such as, Freshman is an undergraduate student, or computed by applying some clustering analysis methods.

With the recent development of data warehousing and OLAP technology, arranging data at multiple levels of abstraction has been a common practice [6]. Therefore, in this paper, we assume such concept taxonomies exist, and our analysis is focused at the second requirement, the efficient methods for multiple-level rule mining. There are several possible directions to explore efficient mining of multiple-level association rules. One choice is the direct application of the existing single-level association rule mining methods to multiple-level association mining. For example, one may apply the Apriori algorithm [3], [7] to examine data items at multiple levels of abstraction under the same minimum support and minimum confidence thresholds.

The fundamental problem in item set mining is formulated as follows: Given a large data base of set of items transactions, find all frequent itemsets, where a frequent itemset is one that occurs in at least a user-specified percentage of the data base. Many of the proposed itemset mining algorithms are a variant of Apriori [12], which employs a bottom-up, breadth-first search that enumerates every single frequent itemset. Inmany applications (especially in dense data) with long frequent patterns enumerating all possible 2m-2 subsets of a *m* length pattern *(m* can easily be 30 or 40 or longer) is computationally unfeasible. Thus, there has been recent interest in mining *maximal* frequent patterns in these “hard” dense databases. Another recent promising direction is to mine only closed sets [11], a set is closed if it has no superset

with the same frequency. Nevertheless, for some of the dense datasets we consider in this paper, even the set of all closed patterns would grow to be too large. The only recourse is to mine the maximal patterns in such domains.

In this paper, indexed based progressive deepening method is developed by extension of the GenMax algorithm[2] for mining maximal frequent item set. The method first finds frequent data items at the top most level and then progressively deepens the mining process into their frequent descendants at lower concept levels. One important assumption that we have made in this study is to explore only the descendants of the frequent items, since we consider if an item occurs rarely, its descendants will occur even less frequently and, thus, are uninteresting. Efficient level-shared mining can be explored based on this assumption. One may wonder whether this may miss the associations containing the items which are frequent according to the reduced minimum support threshold at low level but whose ancestors are infrequent.[4]

The necessity for mining multiple-level association rules or using taxonomy information at mining association rules has also been observed by other researchers, e.g., [10]. A major difference between our paper and theirs is that they use the same support threshold across all the levels [9], whereas we use different support thresholds for different levels of abstraction. As discussed above, using a single support threshold will allow many uninteresting rules to be generated together with the interesting ones if the threshold is rather low, but will disallow many interesting rules to be generated at low levels if the threshold is rather high. Therefore, in their study, substantial efforts have been made on how to identify and remove the redundant rules across different levels. In this paper, besides the investigation of several optimization techniques by exploring level-shared mining, two interestingness measures for filtering uninteresting rules are also analyzed.

Indexed GenMax for Efficient Item Pruning

The proposed Index based GenMax for efficient item pruning model is depicted in Fig 1. The Indexed GenMax consists of indexed structure for tid set, Genmax maximal frequent item sets generator, differential sets, and resultant closed (minimal) frequent items with performance factors such as memory utilization and pruning time. A frequent item set is called maximal if it is not a subset of any other frequent item set. A frequent item set is called closed if there exists no proper superset.

Genmax for Mining Maximal Frequent Item sets

The main goal in mining maximal frequent item sets is to avoid traversing all the branches in the search tree, since for a maximal pattern length, one would have to examine all branches. Furthermore, knowledge of previously found maximal patterns should be utilized to avoid checking branches that cannot possibly lead to maximal patterns, i.e., these branches are entirely subsumed by one of the maximal patterns. GenMax uses backtracking search technique, combined with the ability to utilize previously discovered patterns to compute the set of maximal patterns efficiently.

Differential sets
Each class is totally independent, in the sense that it has a list of all possible item sets, and their transaction id sets (tidsets), that can be combined with each other to produce all frequent patterns sharing a class prefix, our goal is to leverage this property in an efficient manner. The solution is to avoid storing the entire tidset of each member of a class. Instead we will keep track of only the differences in the tids between each class member and the class prefix itemset. These differences in tids are stored in differential set, which is a difference of two tidsets (namely, the prefix tidset and a class member's tidset). Then, these differences are propagated all the way from a node to its children starting from the root. The root node's members can themselves use full tidsets or differences from the empty prefix.

Indexed Genmax for Mining Frequent Closed Itemsets
Indexed Genmax performs a novel search for closed sets using subset properties of differential sets. The initial invocation is with a indexed class at a give tree node. All differences for pairs of elements are computed. However in addition to checking for frequency, indexed Genmax eliminates branches and grows item sets using subset relationships among differential sets.

Improved Index based Genmax Algorithm
The improved indexed based Genmax Algorithm as designed for the implementation process is presented below
Step 1: Extract Data from DS

Step 2: Find CONCEPT_CNT from DS for indexing hierarchy

Step 3: Form TSS from DS to evaluate the possible transaction sets

Step 4: Find ITEMS for each Indexed Concept as per the Tid sets

Step 5: Determine possible frequent items along with combination of rules from TSS

Step 6: Identify ITEMSETS from TSS

Step 7: For each ITEM in ITEMS
Find the support of ITEM
SELECT ITEMs which has support >= SUPP

Step 8: Form ITEMSETS with selected ITEMs
For each ITEMSET in ITEMSETS
If support of each ITEM in ITEMSET is >= SUPP then
Find confidence of ITEMs in ITEMSET
if confidence >= CONF
Generate the rule and show
Otherwise

Continue with Next ITEMSET
End

Implementation of the Indexed Genmax
The implementation of Indexed Genmax for mining multiple-level item sets i.e., maximal and closed, which uses a transaction id based information encoded indexed table instead of the original transaction table. This is because a data mining query is usually in relevance to only a portion of the transaction database, such as food shown in the figure 2 of indexed hierarchy structure, instead of all the items. It is beneficial to first collect the relevant set of data and then work repeatedly on the task-relevant set. Encoding can be performed during the collection of task-relevant data and, thus, there is no extra encoding pass required. Besides, an encoded string, which represents a position in a indexed table, requires fewer bits than the corresponding object-identifier or bar-code. Therefore, it is often beneficial to use an encoded indexed table, although our method does not rely on the derivation of such an encoded table because the encoding can always be performed on the pruning process itself. The steps of implementation are briefed as below

Step 1: Calculate frequent item sets at each concept level, until no more frequent Item sets can be found

Step 2: Indexing of Frequent Item(FI) set

Step 3: Pruning the Index of FI for Closed and Maximal Frequent Item Set

Step 4: Progressive Focusing approach on indexed structure

Step 4.1: Fast retrieval of Frequent Item sets

Step 4.2: Identification of Precise Frequent Item sets

Uniform Support is the same minimum support for all levels i.e., one minimum support threshold. It is not needed to examine item sets containing any item whose ancestors do not have minimum support in the indexed Genmax model. If support threshold is too high, it miss low level associations. If the support threshold is too low, it generates too many high level associations. Reduced Support is the reduced minimum support at lower levels to evaluate the closed frequent items with efficient index structure for different possible strategies to prune the item sets.

Experimental Performance Evaluation

To analyze the performance of the indexed Genmax item pruning on provisionary data set involving food items, we used Pentium Dual core processor 2.5 GHz with 2 GB of main memory running under Windows XP. The training sample consists of

transactions in car databases taken from UCI repository. The total number of items, being 100, the number of potentially frequent item sets to be 30, and the total number of transactions were 3000. The scale-up tests on total number of items, average size of transactions, and total number of tuples, also performed to verify the efficiency of the indexed Genmax item pruning model which showed satisfactory results briefed in below sections.

Mining Frequent Item Set

The transaction database is converted into an indexed encoded transaction table, according to the information above the generalized items in the item description. The maximal level of the concept hierarchy in the item table is set to 4. The number of the top level nodes keeps increasing until the total number of items reaches. The fan-outs at the lower levels are selected based on the normal distribution, and with a variance of 2.0. Not every strong rule so discovered (i.e., passing the minimum support and minimum confidence thresholds) is interesting enough to be presented to users. Two interestingness measures are proposed to filter redundant rules and unnecessary rules. The indexed based Genmax prune more precise frequent item set compared to that of existing Genmax algorithm.(shown in Fig 1)

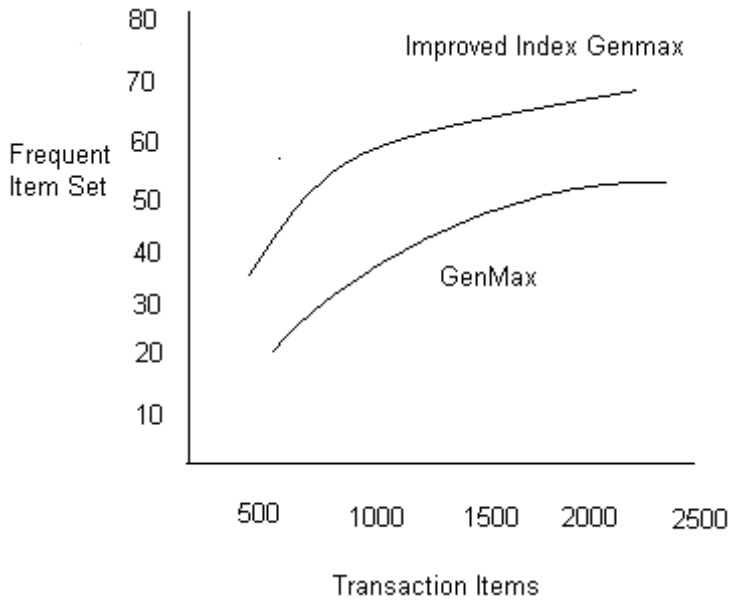

Figure 1: More Number of Frequent Item set pruned by Improved GenMax.

Pruning Time for Indexed based Genmax

The pruning time for indexed based Genmax is low for mining maximal and closed frequent item set compared to that of existing Genmax (shown in Fig 2). The reduction of time in pruning for indexed Genmax is achieved due to the segregation of multilevel hierarchies of the item sets and mainly due to the indexing of transaction id

based on the complete item set. However in case of Genmax the pruning is done for the frequent pattern on all levels of the item set. In case of large transaction, without indexing sequence, exiting Genmax have to prune the frequent patterns from top to bottom for any nearest neighbor pattern also which consumes more time nearly 18% than the indexed Genmax as depicted in Fig 2..

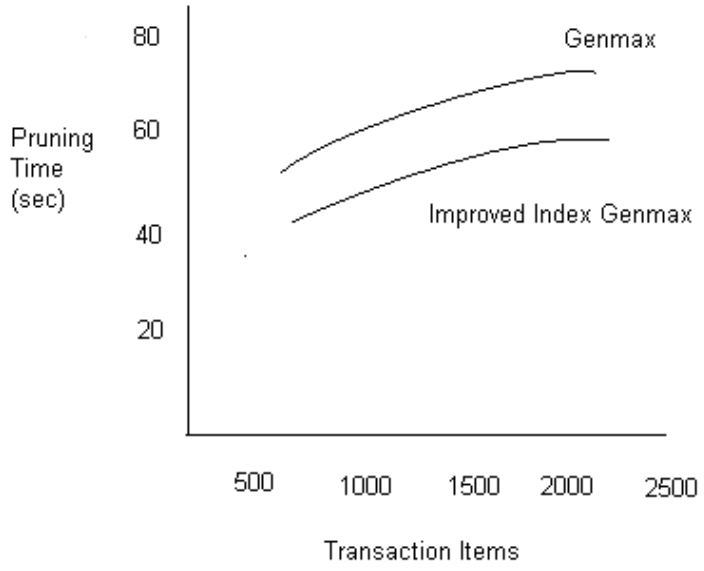

Figure 2: Pruning time for mining closed and maximal frequent item.

Conclusion

The work presented, extended the scope of the analysis of mining association rules from single level to multiple concept, indexed by transaction id. It presents the mechanism for efficiently pruning closed and maximal frequent items in terms of memory utilization and the time required to prune strong and weak items from large transaction databases. Mining multiple-level association rules may lead to progressive mining of refined knowledge from data and have interesting applications for knowledge discovery in transaction-based, as well as other business or engineering, databases. Multi-level association rules are used to compute preferences for items that are not covered by the association rules between items due to the data sparseness.

Through the experiments it is shown that applying multi-level indexed association rules increases recommendation accuracy compared with applying single-level association rules only. The suggested algorithm also shows better performance compared with the basic collaborative filtering as the number of recommendation increases in a sparse environment. The progressive focusing approach using indexing structure is developed for mining maximal and minimal frequent items with more precise item mining and in quicker time. Our performance study shows that the improved indexed GenMax have better performance than the existing GenMax for different distributions of transactional databases.

References

[1] Go'sta Grahne, and Jianfei Zhu, " Fast Algorithms for Frequent Itemset Mining Using FP-Trees " IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 17, NO. 10, OCTOBER 2005 1347

[2] Karam Gouda and Mohammed j. zaki, "GenMax: An Efficient Algorithm for Mining Maximal Frequent Itemsets" , In IEEE Intl. Conf. Data Mining and Knowledge Discovery, 11, 1–20, 2005.

[3] Bayardo, R.J. 1998,. Efficiently mining long patterns from databases, In ACM IGMOD Conf. on Management of Data, pp. 85–93

[4] Burdick, D., Calimlim,M., and Gehrke, J. 2001. MAFIA: A maximal frequent itemset algorithm for transactional databases, In IEEE Intl. Conf. on Data Engineering, pp. 443–452.

[5] G. Grahne and J. Zhu. High performance mining of maxi-mal frequent itemsets. In SIAM'03 Workshop on High Performance Data Mining: Pervasive and Data Stream Mining, May 2003.

[6] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In Proceedings of ACM SIGMOD'00, pages 1–12, May 2000.

[7] J. Han, J. Wang, Y. Lu, and P. Tzvetkov. Mining top-k frequent closed patterns without minimum support. In Proceedings of ICDM'02, pages 211–218, Dec. 2002.

[8] J. Pei, J. Han, and R. Mao. CLOSET: An efficient algorithm for mining frequent closed itemsets. In ACM SIGMOD'00 Workshop on Research Issues in Data Mining and Knowledge Discovery, pages 21–30, 2000.

[9] J. Wang, J. Han, and J. Pei. Closet+: Searching for the best strategies for mining frequent closed itemsets. ACM SIGKDD'03, Washington, DC, 2003.

[10] M. Zaki and K. Gouda. Fast vertical mining using diffsets. ACM SIGKDD'03, Washington, DC, Aug. 2003.

[11] M. Zaki and C. Hsiao. Charm: An efficient algorithm for closed itemset mining. IAM'02, Arlington, Apr. 2002.

[12] Agrawal, R., Aggarwal, C., and Prasad, V. 2000. Depth first generation of long patterns. In 7th Int'l Conference on Knowledge Discovery and Data Mining, pp. 108 118.