Hardware-Enhanced Association Rule Mining with Hashing and Pipelining

Mr. Praveen S Patil
Asst. Professor , Dept of MCA
The Oxford College of Engineering
Bommanhalli, Hosur Road, Bangalore -68
Email : sit.praveen@gmail.com

Abstract: Knowledge mining means extracting the knowledge from the database. Software algorithms are implemented for extracting the knowledge from database. As the volume of the data size is increasing faster than cpu execution speed, software algorithms performance has degraded. Hence to improve the efficiency, in this paper, we propose a new hardware architecture .In hardware architecture, we use the hash table filters to reduce the number of candidates item sets. Another hardware module used is trimming filter. Trimming filter is used to reduce the items from each transaction. The items which are having the minimum support count are trimmed from the transaction .Therefore in this way we reduce the size of the database and extract the knowledge by using another hardware module "systolic array".

1 .Introduction

Data mining technology is used in various fields like bioinformatics, multimedia databases etc .Data mining can be used to provide useful information from the database .As the size of the software algorithms has degraded. Hence there is a need to implement new technology .A new technology proposed is with hardware modules. The hardware architecture contains three modules as shown in figure1.

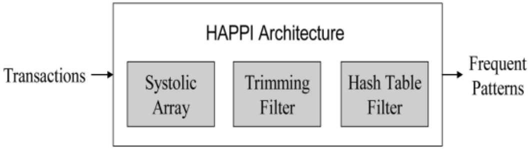


fig.1 System architecture

The systolic array module is used to compare the candidate itemsets with the database and the support count values are calculated. The candidate itemsets having more than minimum support count value are considered as frequent itemsets. The hashtable filters are used to reduce the candidates itemsets. The trimming filters are used to reduce the items from each transaction. In this way we reduce the size of the database and provides useful information for data mining.

In this paper, section 2 describes the related work, section 3 describes the proposed work, section 4 describes the system Architecture, finally we present the conclusion.

2. Related works

In this section, we discuss with two related works. The systolic process Array architecture is proposed in[8] for K-means clustering. A centric of the cluster is calculated and stores the centurion of the cluster in local memory. Each cell computes the distance between the centurion and object input data. The cell updates the minimum distance and the closest centurion of the data object. The system can obtain the closest centurion of each object. The cells are updated by recommitting the centurion.

The systolic array contains number of hardware cells to increase the performance of Apriority algorithm proposed in[9]. As cell contains an ALU ,it performs the comparison operations.

It compares the incoming item with items in memory cell. As each cell contains ALU, performs the comparisions operations and updates the count value in each cell simultaneously.

3. Hardware Architecture

With apriori-based hardware schemes we need to load the candidate item sets into the hardware frequently. Performance decreases with too many candidates item sets .To decrease the number of candidate item sets and size of the database we propose the new architecture. In section 4 we introduce our system architecture, section 4.1 presents transaction trimming scheme, section 4.2 presents the hardware design of hash table filters.

4. System architecture

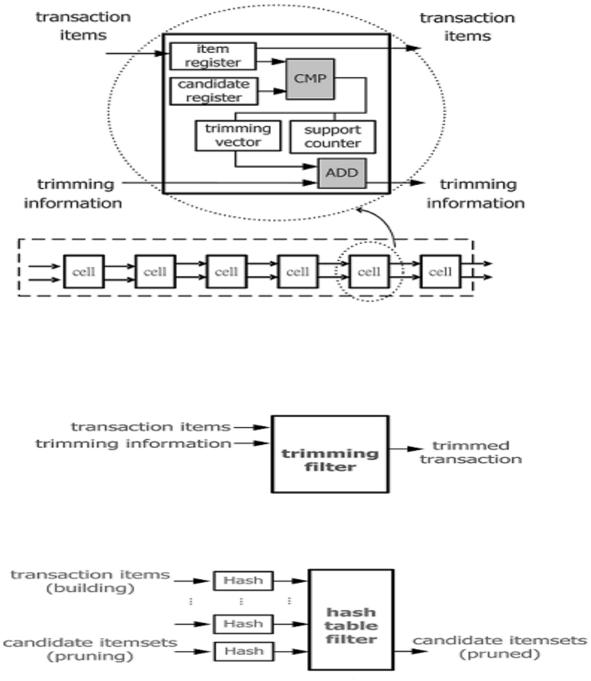


Fig.2 The HAPPI architecture

As shown in fig.2 the architecture consists of a systolic array , a trimming filter and hash table filter. A systolic array consists several hardware cells. As each cell consists ALU, it performs the comparison operation. The cells update the support counters of candidate item sets by comparison . A trimming filter removes the infrequent items in the transactions with available trimming information.

To find the frequent item sets ,we implement five procedures in this architecture using systolic array ,the trimming filter and hash table filter. The procedures are support counting, transaction trimming, hash table building, candidates generation and candidate pruning. The support count procedure finds the frequent item sets by comparing the item sets in the transaction database. The candidate item sets and the database is fed into the systolic array, the frequencies of the candidate item sets are determined .The infrequent items in the database are eliminated by the trimming filter, this operation is called transaction trimming. Then the hash table building procedure generates item sets from the trimmed transactions. Theses item are hashed into the hash table for computing. The candidate generation procedure is executed by the systolic array. The candidate pruning procedure uses the hash table for filtering the candidate item sets that are infrequent. These five procedures are repeated until all frequent items have been found.

4.1 Transaction trimming:

The transaction database doesn’t contain all the useful information .Hence it is required to remove the infrequent items from database.

Extracting the knowledge from the database becomes difficult if the database size is large .Every transaction in a database is not useful for generating frequent item sets .Hence we trim the items from the transaction having less frequent count. When the entire database is fed into the systolic array, the trimming vector is embedded in each hardware cell of the systolic array

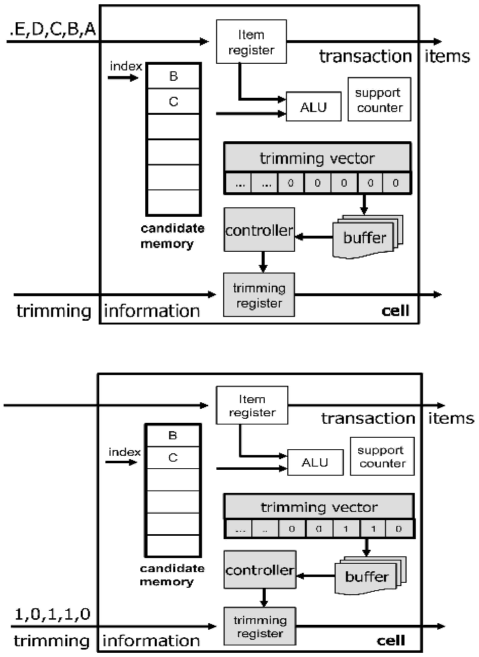
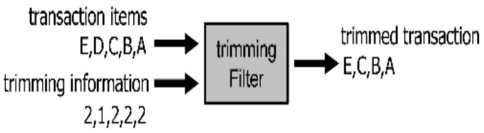


fig.3 An example of streaming of transaction and the corresponding trimming information into the cell

The i th flag of the trimming vector is set to one if the i th item in the transaction matches the candidate item set. Next the transaction items and the trimming information are passed to the trimming filter . The filter trims the items having less frequency. In this way we reduce the size of the database.



4.2Hash table filtering

We use hash table generator and hash table updating modules to built hash table filter. The hash value generator comprises state machine, hash function transaction memory and an index array .the transaction items are feed into the transaction memory and a state machine controller generates the control signals of different lengths these control signals are fed into the index array the i th item is selected by the i th entry of the index array then each item set generated is passed to the hash function and generates the hash value.

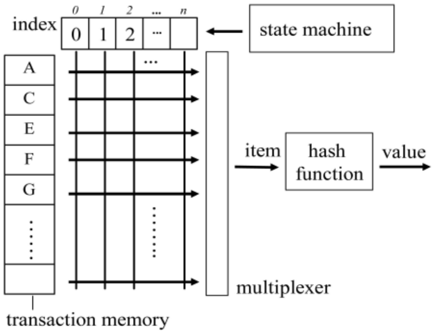
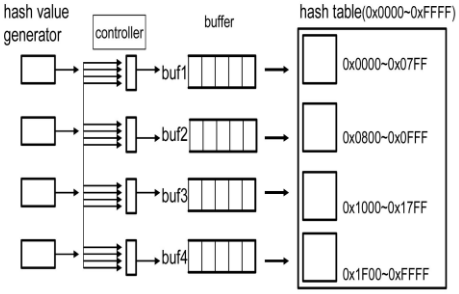


Fig4 Hash table generator

The hash table updating module takes the generated hash values as input and these hash values are passed to the corresponding bugger through the controller .the hash table is divided into several parts to increase the throughput of hash table building these hash values are taken as indexes of the hash table to store the values in the table .

After all the candidate k-item sets are generated , they are pruned by the hash table filter. when the candidates item sets fall into some bucket, we count the number of items in each bucket . If the item sets count value is less than the minimum support count value , the item sets fallen into that bucket are pruned . Therefore with the help of hash table filter we reduce the candidate item sets.



Conclusion

In the apriori –based hardware schemes we need to load the candidate item sets frequently. Hence we need to reduce the size of the database and the number of candidate item sets. To solve this problem we proposed a new hardware architecture . In this architecture , we find the infrequent candidate item sets and items. The infrequent item sets are pruned by the hash table filters . the infrequent items are removed from the transaction database by trimming filter .

References

[1] R.agarwal C. , and V. prasad , ”A tree projection algorithm for generation of frequent item sets, ”parallel and distributed computing 2000.

[2] R.agarwal and R.srikant,”fast algorithms for mining association rules <” proc.20th int’l conf. very large databases

[3] H.hung and C.leiserson, ”systolic Arrays for VLSL”, proc. sparse Matrix,1976.

[4] N.Ling and M. bayoumi, Specification and verification of Systolic ARRAYS. World Scientific Publishing,1999.

[5] H.Toivonen,” Sampling large database for Association Rules” , proc.22nd int 1996.

[6] .J.Han and M. Kamber, data mining: concepts and Techniques .Morgan Kaufmann,2001.

[7] .S.M.Chung and C.Luo,”Parallel Mining of Maximal Frequent itemsets from database.”proc.15th IEEE Int’l Conf.tools with Aritifical Intelligence(ICTAI),2003

[8] M. Gokhale, J. Frigo, K. McCabe, J. Theiler, C. Wolinski, and D.
Lavenier, “Experience with a Hybrid Processor: K-Means Clustering,”
J. Supercomputing, pp. 131-148, 2003.

[9] Z.K. Baker and V.K. Prasanna,
“Efficient Hardware Data Mining
with the Apriori Algorithm on FPGAS,”
Proc. 13th Ann. IEEE
Symp. Field-Programmable Custom
Computing Machines (FCCM),
2005.