ISSN No. 2278-3091

Volume 1, No.3, July – August 2012
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://warse.org/pdfs/ijatcse04132012.pdf

Image Content with Double Hashing Techniques

[1] Mrs.P.Senthil Vadivu, [2] R. Divya
[1]HOD, Department of Computer Applications, Hindusthan College of Arts and Science
Coimbatore – 28.
[2] Research Scholar, Hindusthan College of Arts and Science
Coimbatore – 28
r.divyarun@gmail.com

ABSTRACT

Image mining deals with knowledge discovery in image data bases. In Existing system they represent a data mining techniques to discover significant patterns and they introduce association mining algorithm for discovery of frequent item sets and the generation of association rules. But in the new association rule mining algorithm the completion time of the process is increased. so the proposed work in this paper is the double hashing method for generation of the frequent item sets. Double hashing is another alternative to predict the frequent item sets from tremendous amount of data sets. Double hashing is another method of generating a probing sequence.

Keywords: Apriori, New Association Rule Algorithm, Double Hashing, Quadratic probing.

1. INTRODUCTION

In detailed image databases, there is an advance in image acquisition and storage technology has developed very well. A large amount of images such as medical images and digital photographs are evaluated every day.[1] To discover the relation between variables in large database association rule learning is used which is an popular and well researched methods in data mining. The new association rule algorithm consist of four phases as follows: Transforming the transaction database into the Boolean matrix. Generating the set of 1-itemsets L1,Pruning the Boolean matrix, Generating the set of frequent k-item sets LK(K>1).Based on the concept of rules[1],the regularities between products in large scale transaction data are recorded by point-of-scale(pos) systems.

User-specified minimum support and user specified minimum confidence are satisfied in the process of association rules. In association rule, minimum support is applied first to find all the frequent item sets and minimum confidence constraints are used to form rules. In existing system, the completion time of the process is high. So the efficiency of the process is less. And it stores all data in binary form, so user has some risk to predict the frequent set data.

In this paper, we propose a double hashing method for generation of the frequent item set. Double hashing is another alternative to predict the frequent item set from tremendous amount of data sets.

2. REVIEW OF LITERATURE

In this paper we referred survey papers as follows:

2.1 Mining Frequent Patterns without candidate Generation: A Frequent-Pattern Tree Approach

In this paper [2], the frequent patterns are represented in a tree structure which are extended to an prefix-tree structure for storing quantitative information .FP tree based pattern fragment growth mining method is developed from a frequent pattern(as an initial suffix pattern)and examines its conditional pattern, constructs its FP-tree. The searching technique is used in divide and conquers method rather than Apriori-like level-wise generation of the combinations of frequent item sets. Third, the search technique employed in mining is a partitioning-based, divide-and conquers method rather than Apriori-like level-wise generation of the combinations of frequent item sets.

2.2 Measuring the accuracy and interest of association rules

In this paper[3], they introduce a new framework to assess association rules in order to avoid obtaining misleading rules. A common principle in association rule mining is “the greater the support, the better the item set”, but they think this is only true to some extent. Indeed, item sets with very high support are a source of misleading rules because they appear in most of the transactions, and hence any item set (despite its meaning) seems to be a good predictor of the presence of the high-support item set. To assess the accuracy of association rules they use Shortleaf and Buchanan’s certainty factors instead of confidence.

One of the advantages of their new framework is that it is easy to incorporate it into existing algorithms. Most of them work in two steps:

Step 1.

Find the item sets whose support is greater than minsupp (called frequent item sets). This step is the most computationally expensive.

Step 2.

Obtain rules with accuracy greater than a given threshold from the frequent item sets obtained. To illustrate the problems they have discussed, and to show the performance of their proposals, they have performed some experiments with the CENSUS database. The database they have worked with was extracted using the Data Extraction System from the census bureau database. Specifically, they have worked with a test database containing 99762 instances, obtained from the original database by using MinSet's MIndUtil mineset-to-mlc utility.

2.3 A fast APRIORI implementation

A central data structure of the algorithm is trie or hash tree. Concerning speed, memory need and sensitivity of parameters, tries were proven to outperform hash-trees. In this paper [4],they will show a version of trie that gives the best result in frequent item set mining. In addition to description, theoretical and experimental analysis, they provide implementation details as well. In their approach, tries storing not only candidates, but frequent item sets as well.

2.4 Defining Interestingness for Association Rules

In this paper [5],they will provide an overview of most of the well-known objective interestingness measures, together with their advantages or disadvantages. Furthermore all measures are symmetric measures, so the direction of the rule ($X \Rightarrow Y$ or $Y \Rightarrow X$) is not taken into account. The reason why they do not discuss a-symmetric measures is that, to their opinion, in retail market basket analysis it does not make sense to account for the direction of a rule since the concept of direction in association rules is meaningless in the context of causality. The interested reader is referred to Tan et al. [2001] for an overview of interestingness measures (both symmetric and a-symmetric) and their properties

.

2.5 An Analysis of Co-Occurrence Texture Statistics as A Function Of Grey Level Quantization

In this paper[6] advances the research field by considering the ability of co-occurrence statistics to classify across the full range of available grey level quantization's. This is important, since users usually set the image's grey level quantization arbitrarily without considering that a different quantization might produce improved results. In fact, a popular commercial remote sensing image analysis package determines grey level quantization, preventing the user from providing a more sound choice to potentially improve their results. By investigating the behavior of the co-occurrence statistics across the full range of grey level quantizations, a choice of grey level quantization and co-occurrence statistics can be made. The author is not aware of any other published research that examines co-occurrence probabilities in this manner.

2.6 Optimization of Association Rule Mining Apriori Algorithm Using ACO

ACO has been applied to a broad range of hard combinatorial problems. Problems are defined in terms of components and states, which are sequences of components. Ant Colony Optimization incrementally generates solutions paths in the space of such components, adding new components to a state.

The Optimization and Improvement of the Apriori Algorithm", through the study of Apriori algorithm they discover two aspects that affect the efficiency of the algorithm. One is the frequent scanning database; the other is large scale of the candidate item sets. Therefore, Apriori algorithm is proposed that can reduce the times of scanning database, optimize the join procedure of frequent item sets generated in order to reduce the size of the candidate item sets. In this paper It not only decrease the times of scanning database but also optimize the process that generates candidate item sets.

This work presents an ACO algorithm for the specific problem of minimizing the number of association rules. Apriori algorithm uses transaction data set and uses a user interested support and confidence value then produces the association rule set. These association rule set are discrete and continues therefore weak rule set are required to prune. Optimization of result is needed.

They have proposed in this paper an ACO algorithm for optimization association rule generated using Apriori algorithm. This work describes a method for the problem of association rule mining. An ant colony optimization (ACO) algorithm is proposed in order to minimize number of association rules.

3. EXISTING SYSTEM

In existing system, they introduce association mining algorithm for discover of frequent item sets and the generation of association rules. In general, the new association rule algorithm matrix: The mined transaction database is D, with D having m transactions and n items.

Let T={T1,T2,…,Tm} be the set of transactions and I={I1,I2,…,In} be the set of items. The set up a Boolean matrix Am*n, which has m rows and n columns. Scanning the transaction database D, they use a binning procedure to convert each real valued feature into a set of binary features. The 0 to 1 range for each feature is uniformly divided into k bins, and each of k binary features record whether the feature lies within corresponding range. The Boolean matrix Am*n is scanned and support numbers of all items are computed. The Support number Ij.supth of item Ij is the number of '1s' in the jth column of the Boolean matrix Am*N. If Ij.supth is smaller than the minimum support number, item set {Ij} is not a frequent 1=item set and the jth column of the Boolean matrix Am*n will be deleted from Am*n. Otherwise item set {Ij} is the frequent 1-itemset and is added to the set of frequent 1-itemset L1. The sum of the element values of each row is recomputed, and the rows whose sum of element values is smaller than 2 are deleted from this matrix. Pruning the Boolean matrix means deleting some rows and columns from it. First, the column of the Boolean matrix is pruned Let I. be the set of all items in the frequent set LK-1, where k>2. Compute all |LK-1(j)| where j belongs to I2, and delete the column of correspondence item j if |LK-1(j)| is smaller than k-1. Second, they recomputed the sum of the element values in each row in the Boolean matrix. The rows of the Boolean matrix whose sum of element values is smaller than k are deleted from this matrix. Finally frequent k-item sets are discovered only by "and" relational calculus, which is carried out for the k-vectors combination. If the Boolean matrix Ap*q has q columns where 2<q<=n and minsupth <= p <= m,k q c, combinations of k-vectors will be produced. The 'and' relational calculus is for each combination of k-vectors. If the sum of element values in the "and" calculation result is not smaller than the minimum support number minsupth, the k-item sets corresponding to the combination of k vectors are the frequent k-item sets and are added to the set of frequent k-item set Lk.

4. DOUBLE HASHING

Our proposed system introduces the double hashing method for generation of the frequent item sets. Double hashing is another alternative to predict the frequent items set from tremendous amounts of data sets. While quadratic probing does indeed eliminate the primary clustering problem, it places a restriction on the number of items that can be put in the table—the table must be less than half full. Double hashing is yet another method of generating a probing sequence. It requires two distinct hash functions,

$$h : K \mapsto \{0, 1, \ldots, M-1\},$$
$$h' : K \mapsto \{1, 2, \ldots, M-1\}.$$

The probing sequence is then computed as follows

$$h_i(x) = (h(x) + ih'(x)) \bmod M.$$

That is, the scatter tables are searched as follows:

$$\begin{aligned} h_0 &= (h(x) + 0 \times h'(x)) \bmod M \\ h_1 &= (h(x) + 1 \times h'(x)) \bmod M \\ h_2 &= (h(x) + 2 \times h'(x)) \bmod M \\ h_3 &= (h(x) + 3 \times h'(x)) \bmod M \\ &\vdots \end{aligned}$$

Clearly since c(0)=0, the double hashing method satisfies property 1. Furthermore, property 2 is satisfied as long as h'(x) and M are relatively prime. Since h'(x) can take on any value between 1 and M-1, M must be a prime number.

But what is a suitable choice for the function h'? Recall that h is defined as the composition of two functions, where $h = g \circ f$ where $g(x) = x \bmod M$. We can define h' as the composition $g' \circ f$, where

$$g'(x) = 1 + (x \bmod (M-1)).$$

Double hashing reduces the occurrence of primary clustering since it only does a linear search if h'(x) hashes to the value 1. For a good hash function, this should only happen with probability 1/(M-1). However, for double hashing to work at all, the size of the scatter table, M, Must be a prime number.

5.EXPERIMENTAL RESULTS

Figure1, shows that the results of memory comparison between the Apriori algorithm and the Quadrative probing algorithm. The result shows that memory taken on Y-axis and the Apriori hashing algorithm on X-axis. The stacked column of the red line indicates the memory taken on Quadrative probing and stacked column of the blue line indicates the memory taken on apriori algorithm. These lines indicate the usage of the memory space is less in the quadrative probing algorithm compared to the apriori algorithm.

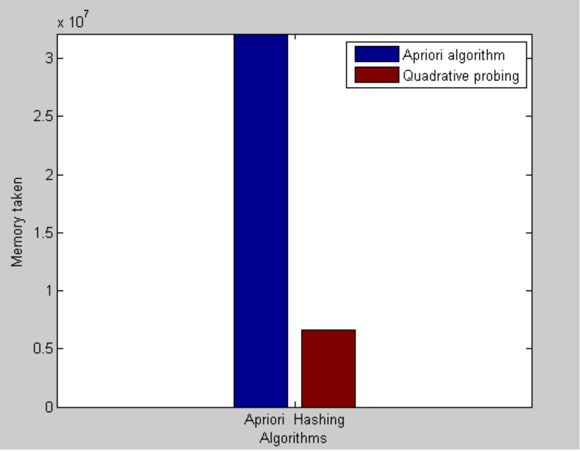

Figure 1: Memory Comparison between Apriori and Quadrative probing algorithms

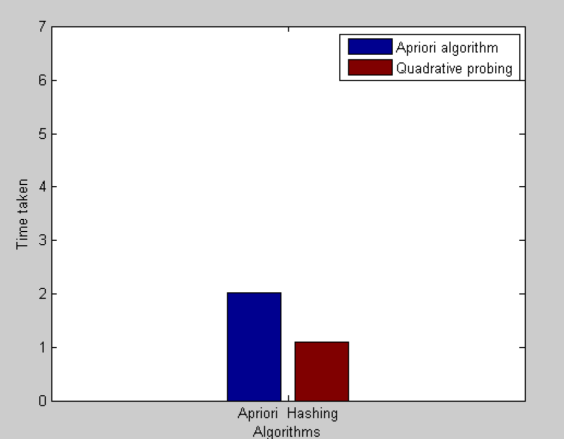

Figure 2: Time Comparison between Aprioir and Quadrative probing algorithms

Figure 2, shows that the results of time comparison between the Apriori algorithm and the Quadrative probing algorithm. The result shows that the time taken on Y-axis and the Apriori hashing algorithm on X-axis. The stacked column of the red line indicates the memory taken on Quadrative probing and stacked column of the blue line indicates the memory taken on Apriori algorithm. These lines indicate the time taken is less in the quadrative probing algorithm compared to the apriori algorithm.

6. CONCLUSION

Conclusion of this work focuses on implement the new method for finding frequent patterns to generate the rules. Our proposed system introduces the double hashing method for generation of the frequent item sets. Double hashing is another alternative to predict the frequent items sets from tremendous amounts of data sets. Basically Double Hashing is hashing on already hashed key. So the computation time of the system is decreased. The experimental results evaluate and show that the proposed method having the minimum support than the existing system.

REFERENCES

1. R. Agrawal, T. Imielinski, and A. Swami. **Mining Association Rules between Sets of Items in Large Databases**, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207–216, Washington, DC, May 26-28 1993.

2. Han, J., Pei, J., and Yin, Y. **Mining Frequent Patterns Candidate Generation,** Proc. 2000 ACM-SIGMOD Int. Management of Data (SIGMOD'00), Dallas, TX.

3. Berzal, F., Blanco, I., Sánchez, D. and Vila, M.A. **Measuring the Accuracy and Importance of Association Rules: A New Framework,** Intelligent Data Analysis, 6:221- 235, 2002.

4**.** Bodon, F**. A Fast Apriori Implementation,** Proc. IEEE ICDM Workshop on Frequent Item set Mining Implementations, 2003.

5.Brijis, T. Vanhoof,K. and Wets, G. **Defining Interestingness for Association rules,** Int.Journal of Information Theories and Applications,10:4,2003.

6. David A. and Clausi, Quan. **An Analysis of Co-occurrence Texture Statistics as a Function of Gray Level Quantization**, Can. J. Remote Sensing, 28, No. 1, pp. 45-62, 2002

7. Xu, Z. and Zhang, S. **An Optimization Algorithm Base on Apriori for Association Rules**, Computer Engineering, 29(19), pp. 83-84.

8. F. Berzal, M. Delgado, D. S´anchez and M.A. Vila. **Measuring the accuracy and importance of association rules,** Technical Report CCIA-00-01-16, Department of Computer Science and Artificial Intelligence, University of Granada, 2000.

9. S. Brin, R. Motwani, J.D. Ullman and S. Tsur. **Dynamic item set counting and implication rules for market basket data**, SIGMOD Record 26(2) (1997), pp.255–264.