COMPARATIVE STUDY OF FREQUENT ITEM SET IN DATA MINING

Brijendra Dhar Dubey,Mayank Sharma,Ritesh Shah

S.I.M.S,Indore ,India

ABSTRACT

In this paper, we are an overview of already presents frequent item set mining algorithms. In these days frequent item set mining algorithm is very popular but in the frequent item set mining computationally expensive task. Here we described different process which use for item set mining, We also compare different concept and algorithm which used for generation of frequent item set mining From the all the types of frequent item set mining algorithms that have been developed we will compare important ones. We will compare the algorithms and analyze their run time performance.

Keywords

Data mining, KDD, Frequent Item Set.

1. Introduction

At present time most of organization use computer system for store data and information. Organization also saves all transaction details. In hospital organization save details about patients like, patients name age disease etc. geological department store data and information regarding earth like mineral data, fossils information and meteorites which is used for analysis work .Banking. Insurance, retails transaction information are also saved by respective organization.

All transaction which saved generally follow concept of database management system .Because using DBMS concept some advantage of DBMS like less redundancy produced. When redundancy are low in database then data inconsistency are go to low.

In previous day after completion of transaction the use of database are used like past transaction details. But know day database show some meaning full hidden information which use full for analysis purpose.

In this information and communication technology era large database and data warehouse are created by the organizations like credit cards, retail, banking ,dicision tree, support ort vector machine and many others availability of less cost storage and evolution of data capturing methods is also increasing and extracting the useful data, explore the database and data warehouse completely and efficiently.

DOI : 10.5121/ijpla.2015.5101

Data mining process used to extract the useful information and pattern from the large data set. KDD is a knowledge Discovery in Databases process in which the data mining is the one of the step of KDD process. The purposes of data mining process are analysis on stored data and find valuable information from huge dataset. This valuable information useful for future use in application of pattern matching and others.

Data mining is a technique that helps us to extract important data from a large database. Data mining is becoming an increasingly important tool to transform this data into information. Extraction of novel and useful knowledge from data in Data Mining has become an effective and analysis decision method in the corporation. Data mining includes the following results, they are as follows: -

- Future Forecasting
- Patterns recognizing
- On the basis of their attributes clustering of people or things into groups
- Prediction that what events are likely to occur

Knowledge Discovery in Databases or KDD [4] refers to the process of finding out knowledge in data, and mainly emphasizes on the "high-level" application of data mining technique. It all interest to researchers in pattern recognition, databases, knowledge acquisition statistics, artificial intelligence, machine learning, for expert systems, and for data visualization. The main goal of the KDD [5] process is to extract knowledge from data in the context of huge databases.

Association rules [10][11][12] are the use for discover useful elements that frequently occur in our database consisting of various independent selection of (such as purchasing transactions), and to discover rules according to it .For example if a customer purchases product X, how likely is he or she to purchase product Y?" and "What products will a customer buy if he or she buys products Z and W?" are answered by association-rule(Market basket analysis).

Association rule are the statements which are used to find out the relationship between any data set in any database. Association rule[6][7] consists of two parts “Antecedent” and “Consequent”. Example, like {egg} => {milk}. Here egg is an antecedent and the milk is the consequent. Antecedent is the element that found in database, and consequent is the element that found in combination with the first. Association rules are generated during searching for frequent patterns [8] [9] [13].

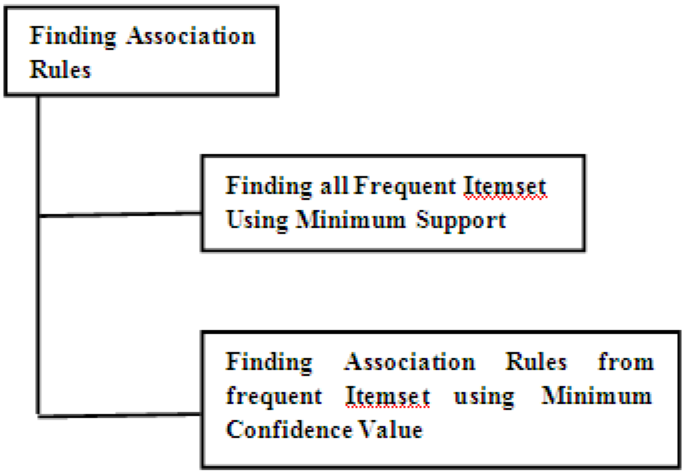
Finding Association Rules
Finding all Frequent Itemset Using Minimum Support
Finding Association Rules from frequent Itemset using Minimum Confidence Value

Figure 1.3 Pattern Generation in Association rules

In association rule are uses the two important criteria are “support ort” and “Confidence”. These two are explained below.

Support ort

The support ort **support (*A*)** of an itemset *X* is defined as the section of transactions in the data set which contain the itemset.

support (A)= no. of transactions which contain the itemset A / total no. of transactions

In the example database, the itemset {item1, item2, item3} has a support out of 6 /20 = 0.30 since it occurs in 30% of all transactions. To be even more explicit we can point out that 6 is the number of transactions from the database which contain the itemset {item1,item2,item3} while 20 represents the total number of transactions.

Confidence

The ***confidence*** of a rule is defined:

$$\textbf{confidence}\ (A \rightarrow B) = \textit{support}\ (A\ U\ B) / \textit{support}\ (A)$$

For the rule {item1, item2}=>{item3} we have the following confidence:

support ({item1,item2,item3}) / support ({item1, item2}) = 0.30/ 0.6 = 0.5

This means that for 50% of the transactions containing item1 and item2 the rule is accurate.

Lift

The rule of ***Lift*** is defined

$$\textbf{confidence}\ (A \rightarrow B) = \frac{\textit{support}\ (A\ UB)}{\textit{support}\ (B) * \textit{support}\ (A)}$$

For the rule { item1, item2}=>{item3} has the following lift:

support ({item1,item2,item3}) / support ({item3}) x support ({item1, item2})= 0.30/0.5 x 0.6= 1

Conviction

The ***Conviction*** of a rule is defined as:

The rule { item1,item2}=>{item3} has the following conviction:
1 – support ({item3})/ 1- conf({item1,item2}=>{item3}) = 1-0.5/1-0.5 = 1.

2. LITERETURE SURVEY

Data Mining frequent item sets is an main problems in data mining and frequent item sets is the first step of deriving association rules [2]. Hence several well-organized item set mining algorithms (e.g., Apriori [2] and FP-growth [10]) have been proposed. The main aim of this algorithm was to eliminate the problem of the Apriori-algorithm in generated and test candidate set. The difficulty of this apriori algorithm was dealt with, by introducing a new data structure, called frequent pattern tree, or FP-tree then based on this structure an FP-tree-based pattern fragment expansion method was developed. FP-growth uses a combination of the vertical and horizontal database outline to store the database in main memory as a substitution for storing and bind for every item in the, it stores the real transactions from the database in a tree structure and every item has a linked list going through all transactions that include that item. This type of structure of data is called FP tree [23].

The SaM (Split and Merge) algorithm establishes by [26] is a simplification of the already simple RElim Recursive Elimination algorithm. While RElim represents a database by storing one transaction list for each item partially vertical representation, the splitting and mergeing algorithm employs only a single transaction list, stored as an array.

Eclat [24] algorithm is basically used depth-first search algorithm. It uses a vertical database layout i.e. instead of clearly listing all transactions; each item is stored together with its wrap (also called tidlist) and uses the intersection based mehtod to compute the support ort of an itemset.

In this approach, the support ort of an itemset A can be easily computed by simply intersecting the covers of any two subsets B, Z □ A, such that B U Z = A. It implice that, when database are stored in the vertical layout, the support ort of a group can be count much easily by simply intersecting the covers of two of its divisions that together give the group itself.

For conditional databases the Aggarwal [1] and Chui [9] developed skilled frequent pattern mining algorithms based on the expected support ort counts of the patterns. However Bernecker et al. [3] Sun [14] and Yiu [16] found that the use of expected support ort may cause to be main patterns absent. Hence they proposed algorithm to compute the possibility that a pattern is frequent and introduced the perception of PFI. In work done in [3] the dynamic programming based solutions were developed to regain PFIs from attribute provisional databases. However their algorithms calculate exact probabilities and confirm that an item set is a PFI in O(n2) time. These proposed model-based methods avoid the use of dynamic programming and are able to verify a PFI much faster. In [16] the estimated algorithms for deriving threshold-based PFIs from tuple-tentative data streams were developed. . The Zhang et al. [16] only well thought-out the mining of sets of single items our solution discovers patterns with more than one item. Recently Sun [14] developed an accurate threshold based PFI mining algorithm. However it does not support ort attribute-tentative data considered in this paper. In a beginning version of this paper [15] we examined a model-based approach for mining PFIs. We study how this algorithm can be extensive to support ort the mining of developing data.

The developments of computed technology in last few years are used to handle large scale data that includes large transaction data, bulletins, emails, retailer etc. Hence information has become a power that made possible for user to voice their opinions and interact. As a result revolves around the practice, data mining [17] come into sites. That time privacy preserving data mining came into the picture. As the database is distributed, different users can access it without interfering with one another. In distributed environment, database is divided into disjoint fragments and each site consists of one fragment.

Data are divided in three different ways that is horizontally partitioned data, vertically partitioned data and mixed partitioned data.

Horizontal partitioning: - The data can be separated horizontally where each fragment consists of a sub division of the records of relation R. Horizontal partitioning [20] [22] [23] [24] divides a table into several tables. The tables have been divided in such a way that query references are done by using small number of tables else excessive UNION queries are used to add the tables sensibly at query time that can be affect the performance and efficiency.

Vertical partitioning: - the data are used firstly separated into the a set of small physical data files each file having sub division of original relation, the relation is the database transaction that normally requires the subsets of the attributes.

Mixed partitioning: - The data is first divdied horizontally and each partitioned fragment is further divided into vertical fragments or the data is first divided vertically and each fragment is further divided into horizontally fragments.

The market basket analysis used association rule mining [20][21] in distributed environment. Association rule mining [18][19][17] is used to find rules that will predict the occurrence of an item to other items in the transaction, search patterns gave association rules where the support ort will be counted as the fraction of transaction that contains an item A and an item B and confidence can be measured in a transaction the item I appear in transaction that also contains an item A

Privacy preserving distributed mining of association rule [21][17] for a horizontally partitioned dataset across multiple sites are computed. The basis of this algorithm [21][17] is the apriori algorithm that uses K-1 frequent sets.

3. PERFORMANCE COMPARISION:-

Frequent itemset mining algorithm was first introduced for mining transaction databases. Let I = {I1, I2, . . . , In} be a set of all items. In which itemset K- itemset belonging to the A from the I itemset. If the transaction occurs in database D does not lower than θ |D| times. Where θ is minimum support and |D| is the total number of transactions in D.

Table 1

Support ort	FP-Growth	Eclat	Relim	SaM
30	0.56	0.54	0.49	0.47
40	0.50	0.50	0.44	0.44
50	0.49	0.45	0.42	0.41
60	0.48	0.44	0.40	0.40
70	0.42	0.40	0.39	0.37

The above table shows the comparison of execution time of the algorithms FP Growth, Eclat, Relim, SaM for different support threshold for adult data set. In this comparison the time of execution is decrease as with the increase support ort threshold.

The graph shows how the time of execution is decreased with the increase support ort threshold.

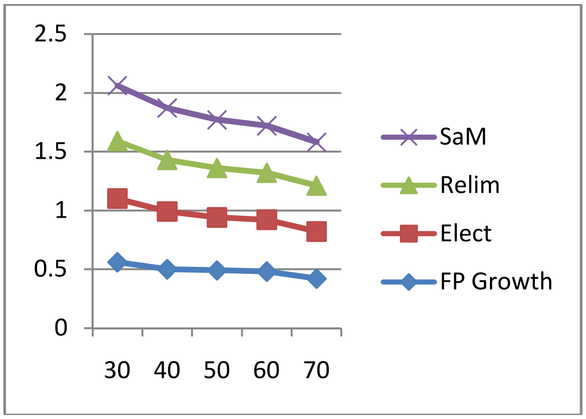

Fig – comparison of algo.

4 CONCLUSION:-

In this paper, we have surveyed presented frequent item set mining techniques and algorithms. We have limited ourselves to the typical frequent item set mining difficulty. Frequent item set mining is the making of all frequent item sets that exists in market basket like data with admiration to minimal thresholds for support ort & confidence.

REFERENCES

[1] C. Aggarwal, Y. Li, J. Wang, and J. Wang, “Frequent Pattern Mining with Uncertain Data,” Proc. 15th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), 2009.

[2] R. Agrawal, T. Imieli_nski, and A. Swami, “Mining Association Rules between Sets of Items in Large Databases,” Proc. ACM SIGMOD Int’l Conf. Management of Data, 1993.

[3] T. Bernecker, H. Kriegel, M. Renz, F. Verhein, and A. Zuefle, “Probabilistic Frequent Itemset Mining in Uncertain Databases,” Proc. 15th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), 2009.

[4] H. Cheng, P. Yu, and J. Han, “Approximate Frequent Itemset Mining in the Presence of Random Noise,” Proc. Soft Computing for Knowledge Discovery and Data Mining, pp. 363-389, 2008.

[5] R. Cheng, D. Kalashnikov, and S. Prabhakar, “Evaluating Probabilistic Queries over Imprecise Data,” Proc. ACM SIGMOD Int’l Conf. Management of Data, 2003.

[6] D. Cheung, J. Han, V. Ng, and C. Wong, “Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique,” Proc. 12th Int’l Conf. Data Eng. (ICDE), 1996.

[7] D. Cheung, S.D. Lee, and B. Kao, “A General Incremental Technique for Maintaining Discovered Association Rules,” Proc. Fifth Int’l Conf. Database Systems for Advanced Applications (DASFAA), 1997.

[8] W. Cheung and O.R. Zaı¨ane, “Incremental Mining of Frequent Patterns without Candidate Generation or Support ort Constraint,” Proc. Seventh Int’l Database Eng. and Applications Symp. (IDEAS), 2003.

[9] C.K. Chui, B. Kao, and E. Hung, “Mining Frequent Itemsets from Uncertain Data,” Proc. 11th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), 2007.

[10] J. Han, J. Pei, and Y. Yin, “Mining Frequent Patterns without Candidate Generation,” Proc. ACM SIGMOD Int’l Conf. Management of Data, 2000.

[11] C. Kuok, A. Fu, and M. Wong, “Mining Fuzzy Association Rules in Databases,” SIGMOD Record, vol. 27, no. 1, pp. 41-46, 1998.

[12] C.K.-S. Leung, Q.I. Khan, and T. Hoque, “Cantree: A Tree Structure for Efficient Incremental Mining of Frequent Patterns,” Proc. IEEE Fifth Int’l Conf. Data Mining (ICDM), 2005.

[13] A. Lu, Y. Ke, J. Cheng, and W. Ng, “Mining Vague Association Rules,” Proc. 12th Int’l Conf. Database Systems for Advanced Applications (DASFAA), 2007.

[14] L. Sun, R. Cheng, D.W. Cheung, and J. Cheng, “Mining Uncertain Data with Probabilistic Guarantees,” Proc. 16th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining, 2010.

[15] L. Wang, R. Cheng, S.D. Lee, and D. Cheung, “Accelerating Probabilistic Frequent Itemset Mining: A Model-Based Approach,” Proc. 19th ACM Int’l Conf. Information and Knowledge Management (CIKM), 2010.

[16] Q. Zhang, F. Li, and K. Yi, “Finding Frequent Items in Probabilistic Data,” Proc. ACM SIGMOD Int’l Conf. Management of Data, 2008.

[17] Agrawal, R., et al “Mining association rules between sets of items in large database”. In: Proc. of ACM SIGMOD’93, D.C, ACM Press, Washington, pp.207-216, 1993.

[18] Agarwal, R., Imielinski, T., Swamy, A. “Mining Association Rules between Sets of Items in Large Databases”, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207-210, 1993.

[19] Srikant, R., Agrawal, R “Mining generalized association rules”, In: VLDB’95, pp.479-488, 1994.

[20] Sugumar, Jayakumar, R., Rengarajan, C “Design a Secure Multi Site Computation System for Privacy Preserving Data Mining”. International Journal of Computer Science and Telecommunications, Vol 3, pp.101-105. 2012.

[21] N V Muthu Lakshmi, Dr. K Sandhya Rani ,“Privacy Preserving Association Rule Mining without Trusted Site for Horizontal Partitioned database”, International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.2, pp.17-29, 2012. [22] J. Han, J. Pei, Y. Yin, and R. Mao. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data

Mining and Knowledge Discovery, 2003[23] C. Borgelt. SaM: Simple Algorithms for Frequent Item Set Mining. IFSA/EUSFLAT 2009 conference- 2009

[24] M. Zaki, S. Parthasarathy, M. Ogihara, and W. Li. NewAlgorithms for Fast Discovery of Association Rules. Proc. 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97), 283–296. AAAI Press, Menlo Park, CA, USA 199