A Survey on Mining Algorithms

Patel Nimisha R., Sheetal Mehta.

Abstract-Data mining is a process that discover the knowledge or hidden pattern from large databases. In the large database using association rules throughfind meaningful relationship between large amount of itemsets and this itemset through create frequent itemset. Association rule mining is the most paramount application in the large database. Most of the Association rule mining algorithm are improved and derivative. The traditional algorithms scan databases many times so, time complexity and space complexity is very high of some of association rule mining . The Latest Researcher are focused on data mining to reduce the scanning time of the large database and increased the mining efficiency. In This paper we are cover the most of the latest algorithm based on association rule mining based on frequent itemsets.

Index Terms- Association rule, Maximal frequent itemsets,Mining algorithm, Data Mining

I. INTRODUCTION

Data mining is known as one of the core processes of KnowledgeDiscovery in Database (KDD). It is the process that results in the discovery of new patterns in large data sets. It is a usefulmethod at the intersection of artificial intelligence, machine learning, statistics, and database systems. It is the principle of picking out relevant information from data. It is usually used by business intelligence organizations, and financial analysts, to extract useful information from large data sets or databases Data mining is use to derive patterns and trends that exist in data. Typically, these patterns cannot be discovered by traditional data exploration. The goal of this technique is to find accurate patterns that were previously not known by us. Organizations like retail stores, hospitals, banks, and insurance companies currently using mining techniques. Data mining is primarily used today by companies with a strong consumer focus - retail, financial, communication, and marketing organizations. It enables these companies to determine relationships among "internal" factors such as price, product positioning, or staff skills, and "external" factors such as economic indicators, competition, and customer demographics.

Manuscript received on January, 2013.

Patel Nimisha, Department of Information Technology Parul Institute of Engg. And Tech Gujarat Technological university Gujarat,India.

Prof. Sheetal Mehta Assistant Professor Department of Computer Science & Engineering Parul Institute of Engg. And Tech Gujarat Technological university Gujarat,India.

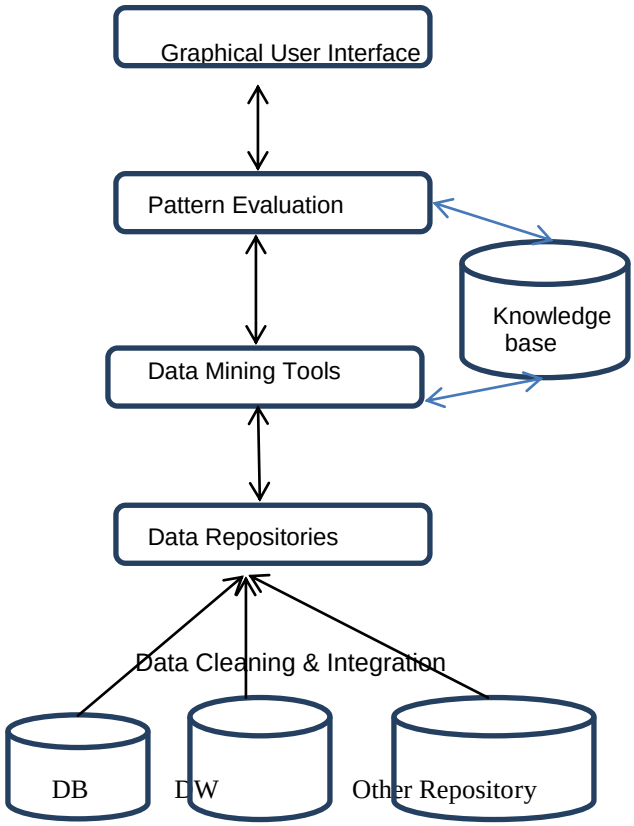

Fig 1.Knowledge Discovery in Database processes

II. ASSOCIATION RULE MINING

Association rule mining is to find out association rules thatsatisfy the predefined minimum support and confidence from agiven database. The problem is usually decomposed into two subproblems. One is to find those item sets whose occurrencesexceed a predefined threshold in the database; those item sets arecalled frequent or large item sets. The second problem is togenerate association rules from those large item sets with theconstraints of minimal confidence.

Association is the discovery of association relationships or correlations among a set of items. This problem was introduced in [5]. Let I = {i1, i2, …,im} be a set of binary attributes, called items. Let D a set of transactions and each transaction T is a set of items such that T⊆I. Let X be a set of items. A transaction T is said to contain X if and only if X⊆T. An association rule is an implication of the form X⇒Y, where X⊂ I, Y⊂ I, and X∩Y= ∅.

Furthermore, the rule X⇒Y is said to hold in the transaction set D with confidence c if there are c% of the transaction set D containing X also containing Y. The rule X⇒Y is said to have support s in the transaction set D if there are s% of transactions in D containing X∪Y. An example of an association rule is: "35% of transactions that contain bread also contain milk; 5% of all transactions contain both items". Here, 35% is called the confidence of the rule, and 5% the support of the rule [5, 16]. The selection of association rule is based on support and confidence. The confidence factor indicates the strength of the implication rules, i.e. the confidence for an association rule is the ratio of the number of transactions that contain X U Y to the number of transactions that contain X; whereas the support factor indicates the frequencies of the occurring patterns in the rule. i.e., the support for an association rule is the percentage of transactions in the database that contain X U Y. Given the database D, the problem of mining association rules involves the generation of all association rules among all items in the given database D that have support and confidence greater than or equal to the user specified minimum support and minimum confidence. Typically large confidence values and a smaller support are used. Rules that satisfy both minimum support and minimum confidence are called strong rules. Since the database is large and users concern about only those frequently purchased items, usually thresholds of support and confidence are predefined by users to drop those rules that are not so interesting or useful.

Generally, an association rules mining algorithm contains the following steps [7]:

- The set of candidate k itemsets is generated by 1-extensions of the large (k 1)itemsets generated in the previous iteration.
- Supports for the candidate k itemsts are generated by a pass over the database.
- Itemsets that do not have the minimum support are discarded and the remaining itemsets are called large k itemsets.

III. MAXIMAL FREQUENT ITEMSETS

Let I={i1, i2, . . . , in} be a set of items , its denoted by X ⊆I an item set, and it is call X a k-item set if the cardinality ofitem set X is k.

Let database T be a multi set of subsets of I, and letsupport(X) be the percentage of item set Y in T such that X ⊆Y .Informally, the support of an item set measures how often Xoccurs in the database. If support(X) <=minSup ,its called that X is a frequent item set , and its denote the set of all frequent item setsby FI.

A closed frequent item set is a frequent item set X such that thereexists no superset of X with the same support count as X. If X isfrequent and no superset of X is frequent, we say that X is amaximal frequent item set, and we denote the set of all maximalfrequent item sets by MFI.

IV. MINING ALGORITHM

Mining maximal frequent itemsets is of paramount relevance in many of data mining applications. The ''traditional'' algorithms address this problem through scanning databases many times. At present, most mining algorithms of maximal frequentitemsets are improved or derivative algorithms based on Apriorior FP-tree[6]. Thosealgorithms not only scan databases at least two times, which canincrease the usage of I/O resources, but also require storing largedata structures, which increase the usage of storage resources. Mining algorithms of maximalfrequent itemsets are improved or derivative algorithms based onthe Apriori or FP-tree. Recall that the underlying idea of the Apriori-based algorithm is to generate candidate itemsets by joiningfrequent itemsets that have been found and then check theirfrequencies.

Mining algorithms like Apriori or FP-tree algorithm are used general association rule and using this algorithm through they are created frequent item sets. So many algorithm are based on association rule but those algorithm general concept are follow Apriori algorithm and FP-tree algorithm.Max-Miner, Pincer-Search, Mafia, Depth-Project, SmartMiner, GenMax, are based onthe generic concept of the Apriorialgorithm and DMFIA,FPmax, SFP-Max, DFMFI, MFIM_P, FPmax, exploit the concept of the FP-tree.

A. Apriori Algorithm:[1,2]

Apriori involves a phase for finding patterns called frequentitemsets. A frequent itemset is a set of items appearing together in anumber of database records meeting a user-specified threshold.

Apriori employs a bottom-up search that enumerates every singlefrequent itemset. This implies in order to produce a frequent itemset of length, it must produce all of its subsets since they too must be frequent.This exponential complexity fundamentally restrictsApriori-like algorithms to discovering only short patterns.

B. Max-Miner[4,11]

The algorithm extracts only the maximal frequent itemsets.By extracting the maximal frequent itemsets and because any frequent itemset is a subset of a maximal frequent itemset, Max-Miner generates all the frequent itemsets. The algorithm combines a levelwise bottom-up traversal with a top-down traversal in order to quickly find the maximal frequent patterns.Then, all frequent patterns are derived from these ones and one last database scan is carried on to count their support.

Max-Miner is successful because it abandons a strict bottom-uptraversal of the search space, and instead always attempts to "lookahead" in order to quickly identify long frequent itemsets. Byidentifying a long frequent itemset early on, Max-Miner can pruneall its subsets from consideration. Max-Miner uses a heuristic totune its search in an effort to identify long frequent itemsets as earlyas possible. It also uses a technique that can often determine when anew candidate itemset is frequent before accessing the database.The idea is to use information gathered during previous databasepasses to compute a good lower-bound on the number oftransactions that contain the itemset.

C. Pincer-Search :[4]

Pincer algorithm which combines both bottom-up and top-down searches to identify frequent itemsets effectively.

All the itemsets are not explicitlyexamined. It classifies the data source into three classes as frequent, infrequent, and unclassified data.Bottom-up approach is the same as Apriori. Top-down search uses a new set called Maximum-Frequent-Candidate-Set (MFCS). It also uses another set called the Maximum Frequent Set (MFS) which contains all the maximal frequent itemsets identified during the process. Any itemset that is classified as infrequent in bottom-up approach is used to update MFCS. Any itemset that is classified as frequent in the top-down approach is usedto reduce the number of candidates in the bottom–up approach. When the process terminates, both MFCS and MFS are equal. This algorithm involves more data source scans in the case of sparse data sources.

D. Depth-Project :[5]

The algorithm finds frequent itemsets by using depth first search on a lexicographic tree of itemsets.DepthProject also mines only maximal frequent itemsets. It performs a mixed depth-first and breadth-first traversal of the itemset lattice. In the algorithm, both subset infrequency pruning and superset frequency pruning are used. The database is represented as a bitmap. Each row in the bitmap is a bitvector corresponding to atransaction and each column corresponds to an item. The number of rows is equal to thenumber of transactions, and the number of columns is equal to the number of items. By using the carefully designed counting methods, the algorithm significantly reduces the cost for finding the support counts.

E. AIS Algorithm :[1,5]

The AIS algorithm was the first algorithm proposed for mining association rules [5]. The algorithm consists of two phases. The first phase constitutes the generation of the frequent itemsets. The algorithm uses candidate generation to detect the frequent itemsets. This is followed by the generation of the confident and frequent association rules in the second phase. The main drawback of the AIS algorithm is that it makes multiple passes over the database. Furthermore, it generates and counts too many candidate itemsets that turn out to be small, which requires more space and wastes much effort that turned out to be useless.

F. SmartMiner :[6]

SmartMiner algorithm to find exactmaximalfrequent itemsets for large datasets. The SmartMiner algorithm first uses global and local tail information to augment dynamic reordering to reduce the search tree. Second, the passing of tail information eliminates the need of known MFI for superset checking. Smartminer does require superset checking that can bevery expensive.SmartMiner that at each step passestail information to guide the search for new MFI. SmartMiner using anaugmented heuristic and tailinformation has many benefits: it does not require supersetchecking, reduces the computation for counting support, andyields a small search tree. SmartMiner also reduces the number of support counting for determining the frequency of tail items andthus greatly saves counting time

G. GenMax :[7]

GenMax is a backtrack search based algorithm for mining maximal frequent itemsets. It uses progressive focusing to perform maximality checking, and diffset propagation to perform fast frequency computation.GenMax uses a number of optimizations to prune the searchspace. It uses a novel technique called progressive focusing to perform maximality checking, anddiffset propagation to perform fast frequency computation.

They have shown that GenMax is a highlyefficient method to mine the exact set of maximal patterns. GenMax representsthe database in a vertical TID set format like VIPER and uses diffset propagation to perform fast supportdegree counting. GenMax has the better performance in the largedata sets. GenMax is the method of choice for enumerating the exact set of maximal patterns.

H. MAFIA :[8]

MAFIA is an algorithm for mining maximal frequent itemsets from a transactional database (it has however the option to mine the closed sets as well). It is especially efficient when the itemsets in the database are very long. The search strategy integrates a depth-first traversal of the itemset lattice.

Mafia uses three pruning strategies to remove non-maximal sets. The first is the look-ahead pruningfirst used in MaxMiner. The second is to check if a new set is subsumed by an existing maximal set. The last technique checks if $t(X) \subseteq t(Y)$. If so X is considered together with Y for extension. Mafia uses vertical bit-vector data format, and compression and projection of bitmaps to improve performance. Mafia mines a superset of the **MFI**, and requires a post pruning step to eliminate non-maximal patterns.

I. FP-tree Algorithm :

FP-tree-based algorithm is to partition theoriginal database to smaller sub-databases by some partition cells,and then to mine itemsets in these sub-databases[6,16]. Unless no new itemsets can be found, the partition is recursively performed with the growth of partition cells. The FP-tree construction takes exactly two scans of the transaction database. The first scan collects the set of frequent items, and the second scan constructs the FP-tree. The cost of inserting a transaction *Trans* into the FP-tree is $O(|freq(Trans)|)$, where $freq(Trans)$ is the set of frequent items in *Trans*. We will show that the FP-tree contains the complete information for frequent-pattern mining.

J. DMFIA : [13]

The DMFIA is a breadth-first algorithm using a bidirectionalsearch strategy of top-down and bottom-up. The entire methodcompresses the database into a FP-tree, and counts support degreeby visiting the corresponding model of the tree.

K. FPmax :[14]

FPmax is an extension of the FP-Growth method for mining onlyMFI. During the mining process, FP-Tree (a tree structure) is used tostore the frequency information of the whole database. To test if afrequent itemset is maximal, another treestructure, called a MaximalFrequent Itemsets tree (MFI-tree), is utilized to keep track ofall maximal frequent itemsets. This structure makes FPmax aneffective vehicle in reducing the search time and the number ofsubset testing operations.

V. CONCLUSION

Association rule mining algorithms through improve the efficiency in large database when created frequent itemset. But in the large database there are a several problem like created more node using FP max Algorithm or it is take the more time to scanning the database. So using the Some Association rule mining algorithm through we can improved in time efficiency when scanning large database and this improvement is only show in large database.

REFERENCES

1. Hu, Y., & Han, R. X."An improved algorithm for mining maximal frequentpatterns," In Proceedings of international joint conference on artificial intelligence,2009, pp. 746–749.
2. Grahne, G., & Zhu, J. F,"Fast algorithms for frequent itemset mining using FPtrees", IEEE Transactions on Knowledge and Data Engineering, 17(10), 2005, pp. 1347–1362.
3. Chen, E. H., Cao, H. H., Li, Q., &Qian, T. Y,"Efficient strategies for toughaggregate constraint-based sequential pattern mining. Information Sciences" 2008,pp.178(6), 1498–1518.
4. Bayardo, R. J. "Efficiently mining long patterns from databases," In Proceeding of the ACM SIGMOD international conference on management of data, 1998, pp. 85–93.
5. Lin, D., &Kedem, Z. M. "Pincer-Search: an efficient algorithm for discovering the maximum frequent set." IEEE Transactions on Knowledge and DataEngineering, 2002, pp.14 (3), 553–566.
6. Agarwal, R. C., Aggarwal, C. C., & Prasad, V. V. V. " Depth first generation of long patterns," In Proceedings of the 6th ACM SIGKDD international conference on knowledge discovery and data mining , 2000,pp. 108–118.
7. Zhou, Q. H., Wesley, C., & Lu, B. J. "SmartMiner: A depth 1st algorithm guided by tail information for mining maximal frequent itemsets." In Proceedings of IEEE international conference on data mining ,2002,pp. 570–577.
8. Gouda, K., &Zaki, M. J,"Efficiently mining maximal frequent itemsets,"InProceedings of 1st IEEE international conference on data mining,2001, pp. 163–170.
9. Burdick, D., Calimlim, M., &Gehrke, J,"Mafia: A maximal frequent itemsetalgorithm for transactional databases" In Proceedings of 17th internationalconference on data engineering ,2001,pp. 443–452 .
10. Baralis, E., Cerquitelli, T., &Chiusano, S. "IMine: Index support for item set mining" IEEE Transactions on Knowledge and Data Engineering, 2009,pp.21(4), 493–506.
11. Chen, E. H., Cao, H. H., Li, Q., &Qian, T. Y." Efficient strategies for toughaggregate constraint-based sequential pattern mining. Information Sciences",2008,pp.178(6), 1498–1518.
12. RupaliHaldulakar and Prof. JitendraAgrawal, "Optimization of Association Rule Mining through Genetic Algorithm", International Journal on Computer Science and Engineering (IJCSE), Vol. 3 No. 3 , 2011, pp. 1252-1259.
13. Zaki, M. J.; Parthasarathy, S.; Ogihara, M.; and Li, W, "New Algorithms for Fast Discovery of association Rules,"InProc. of the Third Int'l Conf. on Knowledge Discovery inDatabases and Data Mining,1997,pp.283-286
14. Song, Y. Q., Zhu, Y. Q., & Sun, Z. H., "An algorithm and its updating algorithmbased on FP-tree for mining maximum frequent itemsets", Journal of Software, 14(9),2003 pp-1586–1592
15. Grahne, G., & Zhu, J. F."High performance mining of maximal frequentitemsets," In Proceedings of the 6th SIAM international workshop on highperformance data mining,, 2003,pp. 135–143.
16. Qin, L. X., & Shi, Z. Z,"SFP-Max: a sorted FP-tree based algorithm for maximalfrequent patterns mining", Journal of Computer Research and Development, 2005,pp.42(2),217–223.
17. Yan, Y. J., Li, Z. J., & Chen, H. W. " A depth-first search algorithm for miningmaximal frequent itemsets" Journal of Computer Research and Development,, 2005,pp. 462–467.