Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm

[1]Sanjeev Rao, [2]Priyanka Gupta
[1,2]Dept. of CSE, RIMT-MAEC, Mandi Gobindgarh, Punjab, india

Abstract
In this paper we present new scheme for extracting association rules that considers the time, number of database scans, memory consumption, and the interestingness of the rules. Discover a FIS data mining association algorithm that removes the disadvantages of APRIORI algorithm and is efficient in terms of number of database scan and time. The frequent patterns algorithm without candidate generation eliminates the costly candidate generation. It also avoids scanning the database again and again. So, we use Frequent Pattern (FP) Growth ARM algorithm that is more efficient structure to mine patterns when database grows.

Keywords
Data Mining, Association Rule Mining Algorithms, Apriori Algorithm, FP-Growth Algorithm, Unsupervised Learning, Early Pruning, etc.

I. Introduction

Data mining is the core process of "KNOWLEDGE DISCOVERY IN DATABASE". It is the process of extraction of useful patterns from the large database. To analyze the large amounts of collected information, the area of Knowledge Discovery in Databases (KDD) provides techniques which extract interesting patterns in a reasonable amount of time. Therefore, KDD employs methods at the cross point of machine learning, statistics and database systems. Data mining is the application of efficient algorithms to detect the desired patterns contained within the given data.

A. Association Rule Mining

Association rules mining are one of the major techniques of data mining and it is perhaps the most common form of local-pattern discovery in unsupervised learning systems. The technique is likely to be very practical in applications which use the similarity in customer buying behavior in order to make peer recommendations.

Association Rules will permit you to discover rules of the kind If X then (likely) Y where X and Y can be particular items ,values, words, etc. , or conjunctions of values, items, words, etc. (e. g.,if (Car=BMW and Gender=Male and Age<20) then (Risk=High and Insurance=High)).Data patterns and models can be mined from many different kinds of databases, such as Relational Databases, Data Warehouses, Transactional Databases, and Advanced Database Systems (Object-Oriented, Relational, Spatial and Temporal, Time-Series, Multimedia, Text, Heterogeneous, Legacy, Distributed, and WWW).

An association rule is composed of two item sets:

1. Antecedent or Left-Hand Side (LHS)
2. Consequent or Right-Hand Side (RHS)

Accompanied with frequency-based statistics, it describes the relationship between Support, Confidence and interestingness. The support and confidence are usually referred as interestingness measures of an association rule. Association rule mining is the process of finding all the association rules that pass the condition of min-support and min-confidence.

In order to mine these rules, first the support and confidence values have to be computed for all of the rules and then compare them with the threshold values to prune the rules with low values of either support or confidence.

II. Related Work

Frequent Itemset Mining (FIM) is an important data mining problem which detects frequent itemsets in a transaction database. It plays a fundamental role in many data mining tasks that attempt to find interesting patterns from databases, such as association rules, correlations, sequences, episodes, classifiers, clusters, etc. Many algorithms have been proposed to solve the problem. Most of them can be classified into two categories, candidate generation and pattern growth.

Apriori [Agrawal 1994], represents the candidate generation approach. Apriori is a Breadth First Search Algorithm (BFS) which generates candidate k+1-itemsets based on frequent k-itemsets. The frequency of an itemset is computed by counting its occurrence in each transaction.

FP-growth [Han 2000], is a representative pattern growth approach. It is a Depth First Approach (DFS) and uses a special data structure, FP-Tree, for compact representation of the original database. FP-growth detects the frequent itemsets by recursively finding all frequent 1-itemsets in the conditional pattern base that is efficiently constructed based on the node link structure associated with FP-Tree. FP-growth doesn't explicitly generates candidates; its detection of the item supports is equivalent to generating 1-itemset candidates implicitly.

III. Apriori Algorithm

Apriori algorithm (Agrawal et al. 1993), is the most classical and important algorithm for mining frequent itemsets. Apriori is used to find all frequent itemsets in a given database DB.

The key idea of Apriori algorithm is to make multiple passes over the database. It employs an iterative approach known as a breadth-first search (level-wise search) through the search space, where k-itemsets are used to explore (k+1)-itemsets.

In the beginning, the set of frequent 1-itemsets is found. The set of that contains one item, which satisfy the support threshold, is denoted by L1. In each subsequent pass, we begin with a seed set of itemsets found to be large in the previous pass. This seed set is used for generating new potentially large itemsets, called candidate itemsets, and count the actual support for these candidate itemsets during the pass over the data.

At the end of the pass, we determine which of the candidate itemsets are actually large (frequent), and they become the seed for the next pass. Therefore, L1 is used to find L2, the set of frequent 2-itemsets, which is used to find L3, and so on, until no more frequent k-itemsets can be found. Then, a very significant property called Apriori property is employed to reduce the search space, where the Apriori property is described as —"All nonempty subsets of a large itemset must also be large" or —"If a set is not large, then its superset can't be large either". This property belongs to a special category of properties called antimonotone in the sense that if a set cannot pass a test, all of its supersets will fail the same test as well.

A. APRIORI Algorithm

Apriori Algorithm can be used to generate all frequent itemset. A Frequent itemset is an itemset whose support is greater than some user-specified minimum support (denoted L_k, where k is the size of the itemset). A Candidate itemset is a potentially frequent itemset (denoted C_k, where k is the size of the itemset).

1. Pass 1

1. Generate the candidate itemsets in C_1
2. Save the frequent itemsets in L_1

2. Pass k

(i). Generate the candidate itemsets in C_k from the frequent itemsets in L_{k-1}

Join $L_{k-1}p$ with $L_{k-1}q$, as follows:

insert nto C_k

select $p.item_1$, $p.item_2$, . . . , $p.item_{k-1}$, $q.item_{k-1}$

from $L_{k-1}p$, $L_{k-1}q$

where, $p.item_1 = q.item_1$, . . . $p.item_{k-2} = q.item_{k-2}$, $p.item_{k-1} < q.item_{k-1}$

- Generate all (k-1)-subsets from the candidate itemsets in C_k
- Prune all candidate itemsets from C_k where, some (k-1)-subset of the candidate itemset is not in the frequent itemset L_{k-1}

(ii). Scan the transaction database to determine the support for each candidate itemset in C_k

(iii). Save the frequent itemsets in L_k.

B. Limitations of APRIORI Algorithm

Apriori algorithm, in spite of being simple and clear, has some limitation. It is costly to handle a huge number of candidate sets. For example, if there are10^4 frequent 1-item sets, the Apriori algorithm will need to generate more than10^7 length-2 candidates and accumulate and test their occurrence frequencies. Moreover, to discover a frequent pattern of size 100, such as {a1, a2. . . , a100}, it must generate 2^{100}-2 ˜ 10^{10} candidates in total. This is the inherent cost of candidate generation, no matter what implementation technique is applied. It is tedious to repeatedly scan the database and check a large set of candidates by pattern matching, which is especially true for mining long patterns. Apriori Algorithm Scans the database too many times, When the database storing a large number of data services, the limited memory capacity, the system I/O load, considerable time scanning the database will be a very long time, so efficiency is very low.

In order to overcome the drawback inherited in Apriori, an efficient FP-tree based mining method, FP-growth, which contains two phases, where the first phase constructs an FP tree, and the second phase recursively Researches the FP tree and outputs all frequent patterns.

IV. FP-Growth Algorithm

FP-growth algorithm is an efficient method of mining all frequent itemsets without candidate's generation. FP-growth utilizes a combination of the vertical and horizontal database layout to store the database in main memory. Instead of storing the cover for every item in the database, it stores the actual transactions from the database in a tree structure and every item has a linked list going through all transactions that contain that item. This new data structure is denoted by FP-tree (Frequent-Pattern tree) (Han et al 2000). Every node additionally stores a counter, which keeps track of the number of transactions that share the branch through that node. Also a link is stored, pointing to the next occurrence of the respective item in the FP-tree, such that all occurrences of an item in the FP-tree are linked together. Additionally, a header table is stored containing each separate item together with its support and a link to the first occurrence of the item in the FP-tree. In the FP-tree, all items are ordered in support descending order, because in this way, it is hoped that this representation of the database is kept as small as possible since all more frequently occurring items are arranged closer to the root of the FP-tree and thus are more likely to be shared.

The algorithm mine the frequent itemsets by using a divide-and-conquer strategy as follows: FP-growth first compresses the database representing frequent itemset into a frequent-pattern tree, or FP-tree, which retains the itemset association information as well. The next step is to divide a compressed database into set of conditional databases (a special kind of projected database), each associated with one frequent item. Finally, mine each such database separately. Particularly, the construction of FP-tree and the mining of FP-tree are the main steps in FP-growth algorithm.

A. FP–Growth Algorithm Steps

1. FP-Growth

Allows frequent itemset discovery without candidate itemset generation. Two step approach:

(i). Step 1

Build a compact data structure called the FP-tree built using 2 passes over the data-set.

(ii). Step 2

Extracts frequent itemsets directly from the FP-tree traversal through FP-Tree.

Table 1:

Algorithm: FP-growth
Input: DB: transaction database; Min_sup: the minimum support threshold
Output: frequent itemsets

(iii). Given

The transaction database with 10 transactions is shown in the following fig.

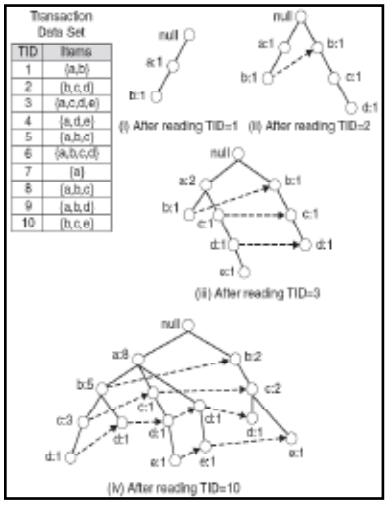

Fig. 1: Transaction Database

B. Advantages of FP-Growth Algorithm

The major advantages of FP-Growth algorithm is,

- Uses compact data structure
- Eliminates repeated database scan

FP-growth is faster than other association mining algorithms and is also faster than tree- Researching. The algorithm reduces the total number of candidate item sets by producing a compressed version of the database in terms of an FP-tree. The FP-tree stores relevant information and allows for the efficient discovery of frequent item sets.

The algorithm consists of two steps:

1. Compress a Large Database into a Compact, Frequent-Pattern tree (FP-tree) Structure

Highly condensed, but complete for frequent pattern mining and avoid costly database scans. Develop an efficient, FP-tree-based frequent pattern mining method (FP-growth)

2. Divide-and-Conquer Methodology

Decompose mining tasks into smaller ones and avoid candidate generation: sub-database test only.

FP-growth algorithm, its scalable frequent patterns mining method has been proposed as an alternative to the Apriori-based approach. This algorithm is faster than other algorithms. Several algorithms implicate the methodology of the FP-growth algorithm. Further improvements of FP-growth mining methods were introduced. (Grahne et al 2005, Gao 2007, Kumar et al. 2007) adapted the similar approach of (Han et al 2000) for mining the frequent itemsets from the transactional database.

V. Methodology & Results

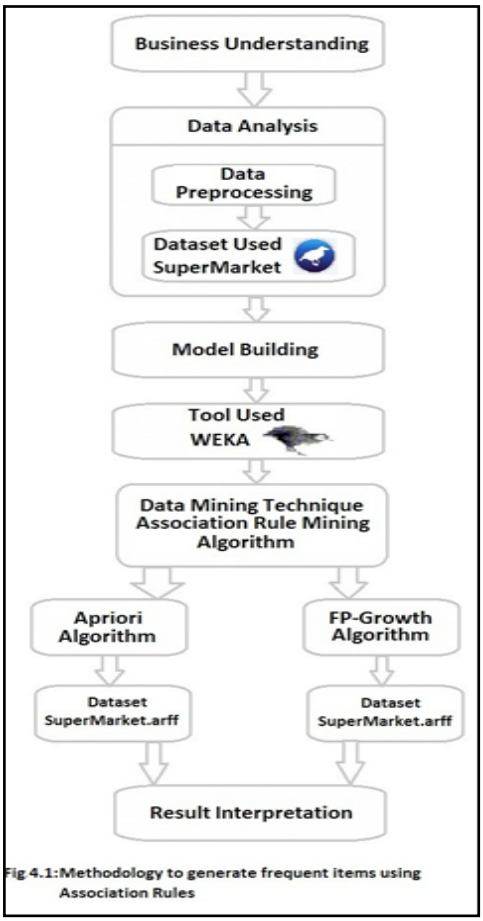

Fig. 2: Methodology to Generate Frequent Items using Associations Rules

Table 2:

S. No.	PROPERTIES		APRIORI ALGORITHM	FP –GROWTH ALGORITHM
1	Dataset Used		SuperMarket. arff	SuperMarket. arff
2	Size of Dataset		1.93 MB	1.93 MB
3	Number of transaction		4627	4627
4	Number of Columns / Items		217	217
5	Type of Dataset		Sparse	Sparse
6	Min Supp	Lower	0.1	0.1
		Upper	1.0	1.0
7	Min.Conf.		0.9	0.9
8	No. of Database Scans / Cycles performed		17	1
9	Memory Consumed (MB)		145 MB	157MB
10	Running Time (Secs)		128	3

A. Graphical Outputs

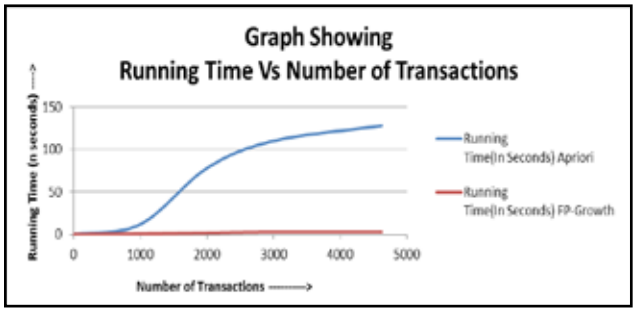

Fig. 3:

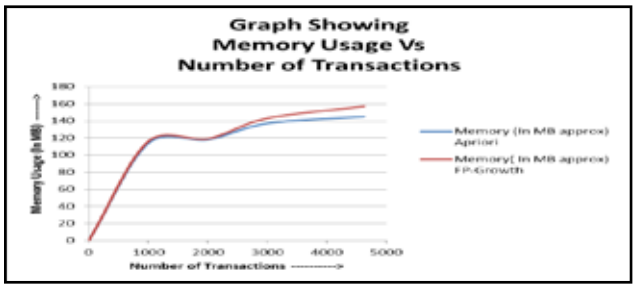

Fig. 4:

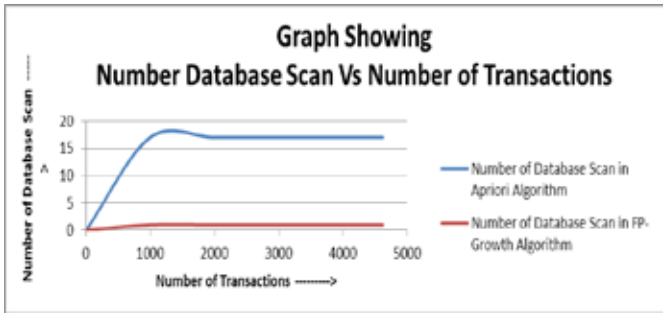

Fig. 5:

VI. Conclusion

In this paper, we presented the use of an ARM (Association rule mining) driven application is to manage retail businesses that provide retailers with reports regarding prediction of product sales trends and customer behavior. Our goal of research is to find a new scheme for finding the rules out of the transactional dataset which outperforms in terms of running time, number of database scan, memory consumption and the interestingness of the rules over the classical APRIORI Algorithm.

VII. Acknowledgement

Authors are greatly thankful to Mr.Philippe Fournier-Viger, Post-Doctoral Researcher, at Intelligent Database Laboratory, Dept. of Computer Science and Information Engg, National Cheng Kung University, Taiwan for giving his valuable time and resources that helped me to implement my research work. Also, we would like to thank our faculty members and Punjab Technical University, Jalandhar, India for providing valuable suggestions in this research work.

References

[1] Fayyad U. M., Piatetsky-Shapiro G., Piatetsky-Shapiro P. X.,"From data mining to knowledge discovery: an Overview" Advances in Knowledge Discovery and Data Mining, AAAI Press/MIT Press, pp. 1-36, 1996.

[2] Fayyad U. M., Piatetsky-Shapiro G., Smyth, P.,"From data mining to knowledge discovery in databases", AI Magazine Vol. 17, No. 3, pp. 37-54, 1996.

[3] Pujari A. K.,"Data mining techniques", Universities Press (India) Private Limited, 2001.

[4] Tan P.-N., Steinbach M., Kumar V.,"Introduction to data mining", Addison Wesley, 2006.

[5] Han, J., Kamber, M.,"Data mining concepts and techniques", Elsevier Inc., Second Edition, San Francisco, 2006.

[6] Han J., Pei J., Yin Y.,"Mining frequent patterns without candidate generations", In Proceeding of the ACM SIGMOD, pp. 1–12, 2000.

[7] Zaki, M. J., Parthasarathy, S., Ogihara, M., Li, W.," New algorithms for fast discovery of association rules", In 3rd Intl. Conf. on Knowledge Discovery and Data Mining, I997.

[8] Zaki, M.J.,"SPADE: An Efficient Algorithm for Mining Frequent Sequences", In Machine Learning, Kluwer Academic Publishers. Manufactured in The Netherlands, 42, pp. 31–60, 2001.

[9] Srikant R., Agrawal R.,"Mining quantitative association rules in large relational tables", In Proceeding of Association for Computing Machinery- Special Interest Group on Management of Data (ACM SIGMOD), pp. 1-12,1996.

[10] Agrawal R., Imielinski T., Swami A.N.,"Mining association rules between sets of items in large databases", In Proceedings ACM SIGMOD International Conference on Management of Data, Vol. 22, No. 2, of SIGMOD Record, Washington, pp. 207–216, 1993.

[11] Agrawal R., Srikant R.,"Fast algorithms for mining association rules", In Proceedings 20th International Conference on Very Large Data Bases (VLDB' 94), pp. 487-499, 1994.

[12] Agrawal R., Srikant R.,"Mining sequential patterns", In Proceeding of the 11th International Conference on Data Engineering, Taipei, Taiwan, pp. 3-14, 1995.

[13] Brin S. , Motwani R., Silverstein C.,"Beyond market baskets: Generalizing association rules to correlations", Data Mining and Knowledge Discovery Journal, Vol. 2, pp. 39-68, 1998.

[14] Kumar B.S., Rukmani K.V.,"Implementation of Web Usage Mining Using APRIORI and FP Growth Algorithms", in Int. J. of Advanced Networking and Applications Vol. 01, Issue, 06, pp. 400-404, 2010.

[15] Piatetsky-Shapiro, G.,"Discovery, analysis, and presentation of strong rules", in G. Piatetsky-Shapiro & W. J. Frawley, eds, "Knowledge Discovery in Databases", AAAI/MIT Press, Cambridge, MA, 1991.

[16] Cheng J., Ke Y., Ng W.,"Effective elimination of redundant association rules", Data Mining and Knowledge Discovery Journal, Vol. 16, pp. 221–249, 2008.

[17] J.Han, J.Pei, Y.Yin,"Mining Frequent Patterns without Candidate Generation" Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 16-18, 2000, Dallas, Texas, USA, 2000.

[18] Ramesh C. Agarwal, Charu C. Aggarwal, V.V.V. Prasad, "A tree projection algorithm for generation of frequent itemsets" Journal of Parallel and Distributed Computing, 2000.

[19] Jian Pei, Jiawei Han Lu, Shojiro Nishio, Shiwei Tang, Dongqing Yang,"H-Mine: Hyper-Structure Mining of Frequent Patterns in Large Databases", IEEE International Conference on Data Mining, 2001.

[20] C.Borgelt,"Paper: Keeping Things Simple: Finding Frequent Item Sets by Recursive Elimination", Workshop Open Source Data Mining Software (OSDM'05, Chicago, IL), 2005.

[21] Aiman Moyaid Said, Dr. P D D. Dominic, Dr. Azween B Abdullah,"A Comparative Study of FP-growth Variations". IJCSNS International Journal of Computer Science and Network Security, Vol. 9, No. 5, 2009.

[22] Balázes Rácz,"Nonordfp: An FP-Growth Variation without Rebuilding the FP-Tree", 2nd International Workshop on Frequent Itemset Mining Implementations, FIMI, 2004.

[23] Grahne O., Zhu J.,"Efficiently Using Prefix-trees in Mining Frequent Itemsets", In Proc. of the IEEE ICDM Workshop on Frequent Itemset Mining, 2004.

[24] Gao J.,"Realization of new association rule mining algorithm", In Proceeding of International Conference on Computational Intelligence and Security, IEEE CNF, pp. 201 – 204, 2007.

[25] Pramod Prasad et al,"Using Association Rule Mining for Extracting Product Sales Patterns in Retail Store Transactions", In International Journal on Computer Science and Engineering (IJCSE), Vol. 3, pp. 2177-2182, 2011.

[26] Frank M.H., Holmes G, Reutemann B. P. P, Witten I. H. "The WEKA Data Mining Software: An Update", in SIGKDD Explorations Vol. 11, Issue 1.

[27] Witten, Ian H., Eibe Frank,"Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations". Morgan Kaufmann, 1999.

[28] [Online] Available: http://www.wikipedia.com/datamining

[29] [Online] Available: http://www.en.wikipedia.org/wiki/Association_rule_learning

[30] [Online] Available: http://www. http://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Frequent_Pattern_Mining/The_FP-Growth_Algorithm

[31] [Online] Available: http://www. http://www.philippe-fournier-viger.com /spmf/index.php?link=documentation.php# allassociationrules

[32] [Online] Available: http://www. http://www.cs.rpi.edu/~zaki/software/
[33] [Online] Available: http://www. http://www.csc.liv.ac.uk/~frans/KDD/Software/FPgrowth/fpGrowth.html
[34] [Online] Available: http://www. http://www.csc.liv.ac.uk/~frans/KDD/Software/FPgrowth/FPtree.java
[35] [Online] Available: http://www. http://www.sigkdd.org/kddcup/index.php?section=1998&method=data
[36] [Online] Available: http://www. http://www.kdnuggets.com/software/associations.html
[37] [Online] Available: http://www. http://en.wikipedia.org/wiki/Weka_machine_learning
[38] [Online] Available: http://www. http://en.wikipedia.org/wiki/Weka_machine_learning#ARFF_file
[39] [Online] Available: http://www. http://www.cs.waikato.ac.nz/ml/weka/
[40] [Online] Available: http://www. http://sourceforge.net/projects/weka/files/weka-3-7-windows-jre/3.7.4/weka-3-7-4jre.exe/ download
[41] [Online] Available: http://www. www.sigkdd.org/kddcup/index.php?section=1998&method=data
[42] [Online] Available: http://www. http://kdd.ics.uci.edu/
[43] [Online] Available: http://www. http://www.kdnuggets.com/datasets/

Er. Sanjeev Rao presently working as Assistant Professor-CSE at RIMT-MAEC, Mandi Gobindgarh, Punjab, India. He has done B.Tech-IT degree from SUSCET, Mohali, India, M.Tech-CSE degree from SVIET, Banur, India. He is Oracle Certified Professional from Oracle University. His areas of interest are Databases (Oracle), Data warehousing & Data Mining, Cloud Computing and Software Engineering. He has published and presented many papers in International Conferences and Journals and attended various FDP/Workshop programs. He has both Industry and Teaching experience of about 5 years. Also he has delivered trainings in Oracle and expert lectures to Engineering students and to Corporates.

Er. Priyanka Gupta presently working as Assistant Professor-CSE at RIMT-MAEC, Mandi Gobindgarh, Punjab, India. She has done B.Tech-CSE degree from Punjab Technical University, Jalandhar, India, M.Tech-ICT (Internet & Communication Technology) from Punjabi University, Patiala, Punjab, India. Her areas of interest are Natural Language Processing, Data Mining and Cloud Computing. She has published and presented many papers in International Conferences and Journals and has attended various FDP/Workshop programs. She has more than 5 years of teaching experience.