International Journal of Technical Research and Applications e-ISSN: 2320-8163,
www.ijtra.com Volume 4, Issue 1 (January-February, 2016), PP. 17-21

AN EFFICIENT AND ENHANCE TOP K ASSOCIATION RULES MINING

Amardeep Kumar[1], Arvind Upadhyay[2]
[1] M.E Scholar Dept of Computer Science and Engineering
[2]Associate Professor
[1,2]IES IPS Academy Indore M.P
[1] akamardeep11@gmail.com
[2] upadhyayarvind10@gmail.com

Abstract: ***Association rule mining is used the most popular fiction in the field of research of data mining. This paper presents a survey of some most common techniques, which are frequently used for mining association rules from a data set.***
Association mining is a cardinal and advantageous researched data mining proficiency. However, depending on the alternative of the arguments (the minimum support and minimum confidence), current algorithms can become very slow and generate an exceeding huge amount of results or generate none or too few results, eliding useful information. This is a severe drawback because in practice users have circumscribed resources for taking apart the results and thus are often only interested in finding a sure amount of results, and excellence synchronization the arguments is time-consuming.

Keywords: ***support, confidence, Top K rules, Rule expansion.***

I. INTRODUCTION

Data mining, that is to boot cited as knowledge discovery in databases, has been recognized because the method of taking out non-trivial, implicit, antecedently unknown, and probably helpful data from knowledge in databases. The selective information employed in the mining method usually contains massive amounts of knowledge collected by computerized applications. Bar-code readers in retail stores, digital sensors in scientific experimentation, and substitute automation tools in engineering are best example typically generate enormous knowledge into databases in no time. Not to mention the aboriginal computing centric surrounding like internet access logs in net applications. These databases therefore work as ample and authentic sources for information generation and conformation. Meanwhile, the massive databases take exception for effective approaches for information discovery.
The ascertained information will be employed in many ways in corresponding applications. For exemplify, classify the oft times come out sets of things in a very retail info will be used to enhance the quality creating of merchandise placement or commercial. Discovering blueprint of client browsing and buying (from either client records or net traversals) could serve the modeling of user behaviors for client retention or customized services. The specified databases, like relational, transactional, spatial, temporal, or transmission database ones, we have a tendency to could get helpful info once the information discovery method if satisfactory mining techniques square criterion used.
Association innovation finds intimately correlate sets so the presence of some ingredient in an exceedingly frequent set can imply the presence of the remaining components (in identical set). Sequential pattern discovery finds temporal associations so not solely closely correlate sets however conjointly their relationships in time are uncovered.
Finding all the frequent patterns from the large databases sets may be a terribly long task. Though the frequency of a pattern may be determined by scanning the info once, the elements of the pattern can't be acknowledged ahead.

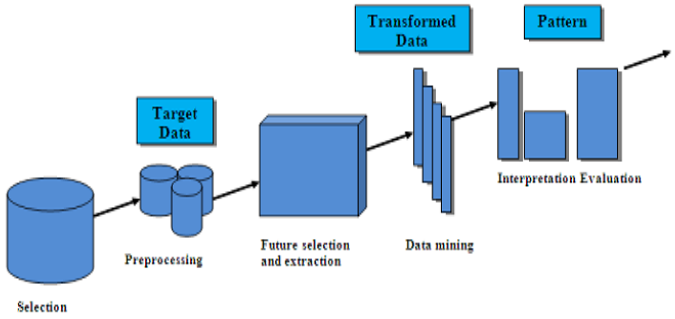

Figure1: Steps of Mining Association Rules

In addition, the mining algorithmic program should be climbable to handle databases of giant size. Whereas the reaction time is also tolerable for an algorithmic program to examine thousands of potential patterns against a little database having thousands of records, it can be intolerable against a info having immeasurable records. Though the frequency of a pattern is determined by scanning the info once, the elements of the pattern cannot be known before. Similarly, an algorithmic program that assumes the info has most a hundred elements may fail to mine any info having quite a hundred parts. Within the mining of frequent patterns in database context, the amount of elements and also the size of the info can be terribly massive. Any improper assumptions on info or main memory might presumably manufacture an

impractical algorithmic program that works well for tiny issues only.

II. BASIC THEORY

Association Rule Mining Association rules mining (Agrawal [1]) might be a widespread information discovery technique for locating associations between things from dealings data. Formally, a dealings data D is printed as a bunch of transactions T= and a bunch of things I=, where t1,t2,…,tn ⊆ I. The support of associate item set X ⊆ I for a data is denoted as sup(X) and is calculated because the type of transactions that contains X. the matter of mining association rules from a dealings data is to look out all association rules X→Y, such X, Y ⊆ I, X n Y=Ø, that the principles respect some stripped interest criteria. The two interest criteria at the beginning projected [1] are that deep-mined rules have a support larger or adequate a user-defined threshold minsup and a confidence larger or adequate a user-defined threshold minconf. The support of a rule X→Y is printed as sup(X ∪ Y) / |T|. The arrogance of a rule is printed as conf (X→Y) = sup(X ∪ Y) / sup(X). Since mode of T |T| = sup(X) for any set price of X ⊆ I, the relation conf(r) = sup(r) hold for any association rule r.

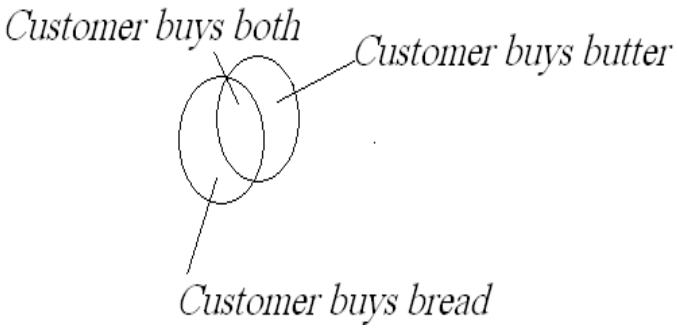

Fig.1. Probability of buying item X and Y

TABLE I. A transaction database

Transaction id	Item bought
10	A,B,C
20	A,C
30	A,D
40	B,E,F

From TABLE I and Fig.1.let itemset X={x1… xk} ,find all the rules X→Y with min confidence and support.
Support, s, probability that a transaction contains XUY
Confidence,c, conditional probability that a transaction having X also contains Y.
Let min_support = 50% and min_conf = 50%:

A → C (50%, 66.7%)

C →A (50%, 100%)

TABLE II. Example of mining association rules

Transaction id	Item bought
10	A,B,C,
20	A,C
30	A,D
40	B,E,F

Frequent Pattern	support
{A}	75%
{B}	50%
{C}	50%
{A,C}	50%

Min. support 50%,
Min. confidence 50%
For rule A ⇒ C,

Support = support ({A} U{C}) = 50%

Confidence = support ({A} U{C})/support ({A}) = 66.6%

The difficult job of mining association rules is to find all association rules in a database having a confidence not less than a user-outlined threshold minconf and a support no less than a user-defined threshold minsup. A major drawback that this method has not been addressed is how the user should choose the thresholds to generate a coveted amount of rules. To get the better of this problem we propose to mine the top k association rules, where k is the number of rules found in the association and it is set by the user.

III. LITERATURE SURVEY

First we will introduce some primitive and basic algorithms for association rule mining, Apriori serial approaches. Then another milepost, tree structured approaches will be explained. Finally this section will end with some peculiar issues of association rule mining, let in multiple level ARM, multiple dimension ARM, constraint based ARM and incremental ARM.

A. AIS algorithmic rule.

The AIS (Agrawal, Imielinski, Swami [1]) algorithmic rule was the primary algorithmic rule planned for mining association rule out [Agrawal et al. 1993[1]]. It specializes in up the standard of databases along with necessary practicality to method call support queries. during this algorithmic rule only 1 item sequent association rules ar generated, which implies that the ensuing of these rules solely contain one item, as an example we have a tendency to solely generate rules like X ∩ Y -> Z however not those rules as X -> Y ∩ Z. The databases were scanned over and over to induce the frequent itemsets in AIS.

The main disadvantage of the AIS algorithmic rule is just too several candidate itemsets that finally clad to be tiny ar generated, which needs extra space and wastes abundant effort that clad to be useless. At constant time this algorithmic rule needs too several passes over the full info.

B. Apriori Algorithm.

Apriori is an awesome upgrade in the historical backdrop of affiliation tenet mining, Apriori calculation was at first proposed by Agrawal in [Agrawal and Srikant 1994 [2]]. The AIS is only a straightforward methodology that requires numerous outputs over the database, creating numerous hopeful itemsets and putting away counters of every applicant while the vast majority of them prohibited to be not continuous. Apriori is more effective amid the hopeful era process for two reasons, Apriori participate in distinctive applicant's era approach and another pruning method. The Apriori calculation mitigates acquires the drawback of examining the whole databases again and again. it's upheld Apriori principle, a few novel calculations were appreciate with a few changes or change. as a rule there have been 2 approaches: one is to abbreviate the measure of disregards the whole data or trade the whole data with exclusively a piece of it upheld this successive itemsets, another methodology is to investigate totally distinctive types of pruning procedures to frame the measure of competitor itemsets a great deal of littler. Apriori-TID and Apriori-Hybrid [Agrawal and Srikant 1994[2]] , DHP [Park et al. 1995[15]], SON [Savesere et al. 1995[18]] or alterations of the Apriori principle.

A large portion of the calculations presented above are upheld the Apriori calculation and look at to improve the proficiency by making a few alterations, such as diminishing the amount of ignores the data ; decreasing the measurement of the database to be examined in every pass; pruning the hopefuls by very surprising strategies and misuse inspecting method. In any case, there are two bottlenecks of the Apriori algorithmic standard. One is that the intricate competitor era strategy that uses more often than not, region and memory. Another bottleneck is that the various sweep of the database data.

C. FP-Tree (Frequent Pattern Tree) algorithmic rule.

To interrupt the 2 bottlenecks of Apriori series algorithms, some works of association rule mining practice tree structure are designed. FP-Tree [Han et al. 2000 [11]], frequent pattern mining, is another milestone inside the event of association rule mining, that breaks the 2 bottlenecks of the Apriori. The frequent itemsets are generated with solely 2 passes over the information and with none candidate generation method. FP-Tree was introduced by Han et al in [Han et al. 2000[12]]. By avoiding the candidate generation method and fewer passes over the information, FP-Tree is AN order of magnitude quicker than the Apriori algorithmic rule. The frequent patterns generation method includes 2 sub processes: constructing the FT-Tree, and generating frequent patterns from the FP-Tree.

The potency of FP-Tree algorithmic rule account for three reasons, initial the FP-Tree may be a compressed illustration of the first information as a result of solely those frequent things are accustomed construct the tree, alternative unsuitable data are cropped. Conjointly by ordering the things consistent with their supports the overlapping components seem just once with totally different support count. Second this algorithmic rule solely scans the information double. The frequent patterns are generated by the FP-growth subroutine, constructing the conditional FP-Tree that contain patterns with specific suffix patterns, frequent patterns will be simply generated as shown in on top of the instance. Conjointly the computation value remittent dramatically. Thirdly, FP-Tree uses a divide and conquers methodology that significantly reduced the scale of the following conditional FP-Tree, longer frequent patterns are generated by adding a suffix to the shorter frequent patterns. In [Han et al. 2000[12]] [Han and I. M. Pei 2000[10]] there are examples maybe all the detail of this mining method.

Every algorithmic rule has his limitations, for FP-Tree it's troublesome to be utilized in AN interactive mining system. Throughout the interactive mining method, users could modification the edge of support consistent with the principles. but for FP-Tree the dynamic of support could result in repetition of the entire mining method. Another limitation is that FP-Tree is that it's not appropriate for progressive mining. Since as time goes on databases keep dynamic, new datasets could also be inserted into the information, those insertions may additionally result in a repetition of the entire method if we tend to use FP-Tree algorithmic rule.

D. Rapid Association Rule Mining (RARM).

RARM [Das et al. 2001[7]] is another association rule mining methodology that uses the tree structure to represent the initial information and avoids candidate generation method. RARM is claimed to be a lot of quicker than FP-Tree algorithmic program with the experiments result shown within the original paper. By exploitation the SOTrieIT structure RARM will generate giant 1- itemsets and 2-itemsets quickly while not scanning the information for the second time and candidate's generation. Just like the FP-Tree, each node of the SOTrieIT contains one item and also the corresponding support count. The big itemsets generation method is as follows. Preprocessing, the info is scanned to reconstruct the TrieIT, the process is analogous to the method of generation the FP-Tree. for every dealing all the attainable thing sets combos square measure extracted and for those items that square measure already within the TrieIT increase their support count by one, for people who still don't exist within the TrieIT the itemsets square measure inserted to the TrieIT with the

corresponding support count be one. The distinction between FPTree and TrieIT is that TrieIT solely will increase the support count of the leaf node things whereas FP-Tree will increase all the support counts on the trail of the itemsets. Since the TrieIT stores the support counts one by one, it needs larger memory area which cannot be glad, SOTrieIT (Support Ordered Trie Itemset) is introduced. To construct the SOTrieIT solely 1-itemsets and 2-itemsets square measure extracted from every dealing, the building method is that the same as within the construction of TrieIT, in the end the itemsets of a similar dealing were inserted the tree is ordered in a very drizzling order of support count, the SOTrieIT has solely 2 levels one is for 1-itemsets, another is for 2-itemsets.Since generating the big 2-itemsets is that the costliest method throughout the mining method, experiments in [Das et al. 2001[7]] showed that the potency of generating giant 1-itemsets and 2-itemsets within the SOTrieIT algorithmic program improves the performance dramatically, SOTrieIT is far quicker than FP-Tree, however SOTrieIT additionally faces a similar downside as FP-Tree.

E. Multiple thought Level ARM.

In reality, for several applications, it's tough to seek out robust association rules between knowledge things at low or primitive level of abstraction as a result of the meagerness of information in three-dimensional area [Han and Kamber 2000[9]]. Whereas robust association rules generated at the next thought level could also be wisdom to some users however it can also be novel to alternative users. Multiple level association rule mining is attempting to mine robust association rules among intra and inhume totally different levels of abstraction. for instance, besides the association rules between milk and ham, it will generalize those rules to relation between drink and meat, at constant time it may specify relation between sure complete of milk and ham. Researchers are wiped out mining association rule at multiple thought levels [Han 1995[9]], [Han and Fu 1995[10]], [Psaila and Lanzi 2000[17]].

F. Numerous Dimensional ARM.

Multiple dimensional affiliation principle mining is to revelation the connection between's very surprising predicts/properties. Every attribute/predict is named a dimension, such as: age, occupation and buys during this example. At a similar time multiple dimensional association rule mining issues all sorts knowledge of information} like Boolean data, categorical knowledge and numerical knowledge [Srikant and Agrawal 1996[20]]. The mining method is analogous to the method of multiple level association rule mining. first of all the frequent 1- dimensions square measure generated, then all frequent dimensions square measure generated supported the Apriori formula.

A handful research literature exists in the study of constraints based association rule mining [Ng et al. 1998[14]], [Pei and Han 2000[16]], [Bayardo et al. 1999[4]], [Srikant et al. 1997[20]], [Garofalakis et al. 1999[8]], [Klemettinen et al. 1994[13]], [Brin et al. 1997[5]], [Smythe and Goodman 1992[19]]. Constraints based association rule mining is to find all rules from a given data-set meeting all the user specified constraints. Apriori and its variants only employ two basic constraints: ***minimal support*** and ***minimal confidence***. However there are two points, one is some of the generated rules may be usefulness or not informative to individual users; another point is that with the constraints of minimal support and confidence those algorithms may miss some interesting information that may not satisfy them.

Some works have used the term "top-*k* association rules". But they are applied to mining streams or mining non-standard rules. [Webb05, You2010 [21]]

IV. CONCLUSION

Of all the mining functions within the information discovering method, frequent pattern mining is to search out the oftentimes occurred patterns. The live of frequent patterns may be a user-specified threshold that indicates the minimum occurring frequency of the pattern. We tend to could categorize recent studies in frequent pattern mining into the invention of association rules and therefore the discovery of consecutive patterns. This paper presented a review of the modern association rule mining techniques.

V. ACKNOWLEDGEMENTS

I am thankful to all faculties of my college and my friends who support me to accomplish this paper despite his or her busy schedule.

REFERENCES

[1] Agrawal, R., Imielinski, T., and Swami, A. N. 1993. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, P. Buneman and S. Jajodia, Eds. Washington, D.C., 207-216.

[2] Agrawal, R. and Srikant, R. 1994. Fast algorithms for mining association rules. In *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, J. B. Bocca, M. Jarke, and C. Zaniolo, Eds. Morgan Kaufmann, 487-499.

[3] Agrawal, R. and Srikant, R. 1995. Mining sequential patterns. In *Eleventh International Conference on Data Engineering*, P. S. Yu and A. S. P. Chen, Eds. IEEE Computer Society Press, Taipei, Taiwan, 3-14.

[4] Bayardo, R., Agrawal, R., and Gunopulos, D. 1999. Constraint-based rule mining in large, dense databases.

[5] Brin, S., Motwani, R., Ullman, J. D., and Tsur, S. 1997. Dynamic itemset counting and implication rules for market basket data. In *SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, Tucson, Arizona, USA*, J. Peckham, Ed. ACM Press, 255-264.

[6] Chen, M.-S., Han, J., and Yu, P. S. 1996. Data mining: an overview from a database perspective. *Ieee Trans. On Knowledge And Data Engineering 8*, 866-883.

[7] Das, A., Ng, W.-K., and Woon, Y.-K. 2001. Rapid association rule mining. In *Proceedings of the tenth international conference on Information and knowledge management*. ACM Press, 474-481.

[8] Garofalakis, M. N., Rastogi, R., and Shim, K. 1999. SPIRIT: Sequential pattern mining with regular expression constraints. In *The VLDB Journal*. 223-234.

[9] Han, J. 1995. Mining knowledge at multiple concept levels. In *CIKM*. 19-24.

[10]Han, J. and Fu, Y. 1995. Discovery of multiple-level association rules from large databases. In *Proc. of 1995 Int'l Conf. on Very Large Data Bases (VLDB'95), ZÄurich, Switzerland, Septem- ber 1995*. 420-431.

[11] Han, J. and Pei, J. 2000. Mining frequent patterns by pattern-growth: methodology and implications. *ACM SIGKDD Explorations Newsletter 2*, 2, 14-20.

[12] Han, J., Pei, J., and Yin, Y. 2000. Mining frequent patterns without candidate generation. In *2000 ACM SIGMOD Intl. Conference on Management of Data*, W. Chen, J. Naughton, and P. A. Bernstein, Eds. ACM Press, 1-12.

[13] Klemettinen, M., Mannila, H., Ronkainen, P., Toivonen, H., and Verkamo, A. I. 1994. Finding interesting rules from large sets of discovered association rules. In *Third International Conference on Information and Knowledge Management (CIKM'94)*, N. R. Adam, B. K. Bhargava, and Y. Yesha, Eds. ACM Press, 401-407.

[14] Ng, R. T., Lakshmanan, L. V. S., Han, J., and Pang, A. 1998. Exploratory mining and pruning optimizations of constrained associations rules. 13-24

[15] Park, J. S., Chen, M.-S., and Yu, P. S. 1995. An e®ective hash based algorithm for mining association rules. In *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, M. J. Carey and D. A. Schneider, Eds. San Jose, California, 175-186.

[16] Pei, J. and Han, J. 2000. Can we push more constraints into frequent pattern mining? In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM Press, 350-354.

[17] Psaila, G. and Lanzi, P. L. 2000. Hierarchy-based mining of association rules in data warehouses. In *Proceedings of the 2000 ACM symposium on Applied computing 2000*. ACM Press, 307-312.

[18] Savesere, A., Omiecinski, E., and Navathe, S. 1995. An efficient algorithm for mining association rules in large databases. In *Proceedings of 20th International Conference on VLDB.*

[19] Smythe and Goodman. 1992. An information theoretic approach to rule induction from databases. In *IEEE Transactions on Knowledge and Data Engineering*. IEEE Computer Society Press

[20] Srikant, R., Vu, Q., and Agrawal, R. 1997. Mining association rules with item constraints. In *Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining, KDD*, D. Heckerman, H. Mannila, D. Pregibon, and R. Uthurusamy, Eds. AAAI Press, 67-73.

[21] Webb and S. Zhang, "k-Optimal-Rule-Discovery," Data Mining and Knowledge Discovery, vol. 10, no. 1, 2005, pp. 39-79.

[22] Y. You, J. Zhang, Z. Yang and G. Liu, "Mining Top-k Fault Tolerant Association Rules by Redundant Pattern Disambiguation in Data Streams," Proc. 2010 Intern. Conf. Intelligent Computing and Cognitive Informatics, March 2010, IEEE Press, pp. 470-473