An Enhanced Apriori and Improved Algorithm for Association Rules

Mohit Ohri[1], **Komal Thakur**[2]

[1]*Department of Computer Applications, Innocent Heart Group of Institutes, Jalandhar, Punjab, India*
[2]*Department of Computer Engineering, DAV Institute of Engineering & Technology, Jalandhar, Punjab, India*

---***---

Abstract - *There are several mining algorithms of association rules. One of the most popular algorithms is Apriori that is used to extract frequent itemsets from large database and getting the association rule for discovering the knowledge. Based on this algorithm, this paper indicates the limitation of the original Apriori algorithm of wasting time for scanning the whole database searching on the frequent itemsets, and presents an improvement on Apriori by reducing that wasted time depending on scanning only some transactions. The paper shows by experimental results with several groups of transactions, and with several values of minimum support and minimum confidence that applied on the original Apriori and our implemented improved Apriori by means of the comparison with three parameters such as time Execution, Memory Consumed and Accuracy of Rules.*

Key Words: **(Apriori, Improved Apriori, Frequent itemset, Support, confidence,)….**

1. INTRODUCTION

With the progress of the technology of information and the need for extracting useful information of business people from dataset [7], data mining and its techniques is appeared to achieve the above goal. Data mining is the essential process of discovering hidden and interesting patterns from massive amount of data where data is stored in data warehouse, OLAP (on line analytical process), databases and other repositories of information [11]. This data may reach to more than terabytes. Data mining is also called (KDD) knowledge discovery in databases [3], and it includes an integration of techniques from many disciplines such as statistics, neural networks, database technology, machine learning and information retrieval, etc [6]. Interesting patterns are extracted at reasonable time by KDD's techniques [2]. KDD process has several steps, which are performed to extract patterns to user, such as data cleaning, data selection, data transformation, data preprocessing, data mining and pattern evaluation [4]. The architecture of data mining system has the following main components [6]: data warehouse, database or other repositories of information, a server that fetches the relevant data from repositories based on the user's request, knowledge base is used as guide of search according to defined constraint, data mining engine include set of essential modules, such as characterization, classification, clustering, association, regression and analysis of evolution. Pattern evaluation module that interacts with the modules of data mining to strive towards interested patterns. Finally, graphical user interfaces from through it the user can communicate with the data mining system and allow the user to interact.

2. ASSOCIATION RULE MINING

Association Mining is one of the most important data mining's functionalities and it is the most popular technique has been studied by researchers. Extracting association rules is the core of data mining [8]. It is mining for association rules in database of sales transactions between items which is important field of the research in dataset [6]. The benefits of these rules are detecting unknown relationships, producing results which can perform basis for decision making and prediction [8]. The discovery of association rules is divided into two phases [10, 5]: detection the frequent itemsets and generation of association rules. In the first phase, every set of items is called itemset, if they occurred together greater than the minimum support threshold [9], this itemset is called frequent itemset. Finding frequent itemsets is easy but costly so this phase is more important than second phase. In the second phase, it can generate many rules from one itemset as in form, if itemset {I1, I2, I3}, its rules are {I1–›I2, I3}, {I2–›I1, I3}, {I3–›I1, I2}, {I1, I2–›I3}, {I1, I3 –›I1}, {I2, I3–›I1}, number of those rules is n^2 -1 where n = number of items. To validate the rule (e.g. X–›Y), where X and Y are items, based on confidence threshold which determine the ratio of the transactions which contain X and Y to the transactions A% which contain X, this means that A% of the transactions which contain X also contain Y. minimum support and confidence is defined by the user which represents constraint of the rules. So the support and confidence thresholds should be applied for all the rules to prune the rules which it values less than thresholds values. The problem that is addressed into association mining is finding the correlation among different items from large set of transactions efficiency [8]. The research of association rules is motivated by more applications such as telecommunication, banking, health care and manufacturing, etc.

3. LIMITATIONS OF APRIORI ALGORITHM

Apriori algorithm suffers from some weakness in spite of being clear and simple. The main limitation is costly wasting of time to hold a vast number of candidate sets with much frequent itemsets, low minimum support or large itemsets. For example, if there are 10^4 from frequent 1- itemsets, it need to generate more than 10^7 candidates into 2-length which in turn they will be tested and accumulate [2]. Furthermore, to detect frequent pattern in size 100 (e.g.) v1, v2… v100, it have to generate 2^{100} candidate itemsets [1] that yield on costly and wasting of time of candidate generation. So, it will check for many sets from candidate itemsets, also it will scan database many times repeatedly for finding candidate itemsets. Apriori will be very low and inefficiency when memory capacity is limited with large number of transactions.

In this paper, we propose approach to reduce the time spent, accuracy and memory consumed for searching in database transactions for frequent itemsets.

4. THE IMPROVED ALGORITHM OF APRIORI

This section will address the improved Apriori, an example of the improved Apriori, the analysis and evaluation of the improved Apriori and the experiments.

4.1 The Improved Apriori

The improvement of algorithm can be described as follows:

1. Scanning the database and converting it into vertical data format.
2. Generating Trans_tokenSet from vertical data format and also maintaining a list for no. of iterations.
3. Finding the Frequent 1 itemset from the Trans_tokenSet i.e. the length of Trans_tokenSet of the item sets.
4. Sorting the itemset according to the ascending order of the Trans_tokenSet by its minimum support.
5. Gather these items as Keysets from the Trans_tokenSet.
6. Generate the Bit Table for each key that is available in Items keyset
7. Generate Subsume for each item in Items keyset
8. for each item in Items

```
If item. Subsume<> " "
If item. Support == min_sup then
FindItemsetsEqualsMinSup(item, item. Support)
 Else
FindItemsetsGreaterThanMinSup (item, item. Support)
End If
Else
If item. Support>min_sup
    AND Item_Sequence<Item.Length Then
        FindItemsetSubnumeNone(item)
Endif
Endif
```

4.2 An example of the improved Apriori

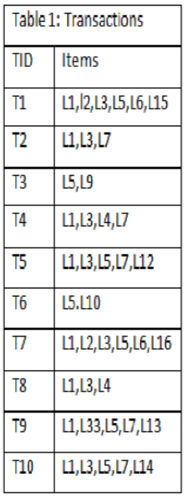

Table 1: Transactions	
TID	Items
T1	L1,l2,L3,L5,L6,L15
T2	L1,L3,L7
T3	L5,L9
T4	L1,L3,L4,L7
T5	L1,L3,L5,L7,L12
T6	L5.L10
T7	L1,L2,L3,L5,L6,L16
T8	L1,L3,L4
T9	L1,L33,L5,L7,L13
T10	L1,L3,L5,L7,L14

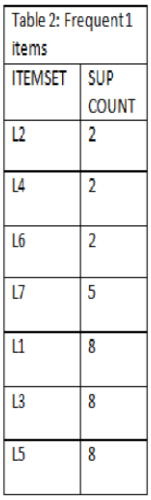

Table 2: Frequent1 items	
ITEMSET	SUP COUNT
L2	2
L4	2
L6	2
L7	5
L1	8
L3	8
L5	8

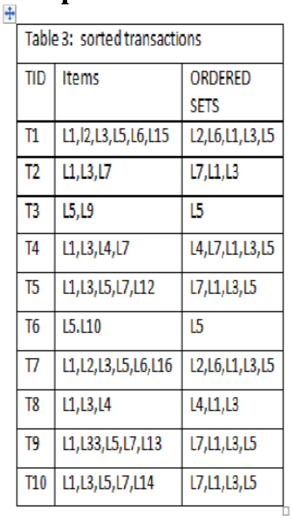

Table 3: sorted transactions		
TID	Items	ORDERED SETS
T1	L1,l2,L3,L5,L6,L15	L2,L6,L1,L3,L5
T2	L1,L3,L7	L7,L1,L3
T3	L5,L9	L5
T4	L1,L3,L4,L7	L4,L7,L1,L3,L5
T5	L1,L3,L5,L7,L12	L7,L1,L3,L5
T6	L5.L10	L5
T7	L1,L2,L3,L5,L6,L16	L2,L6,L1,L3,L5
T8	L1,L3,L4	L4,L1,L3
T9	L1,L33,L5,L7,L13	L7,L1,L3,L5
T10	L1,L3,L5,L7,L14	L7,L1,L3,L5

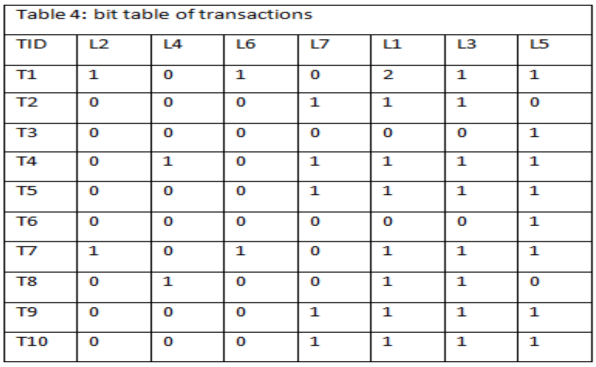

Table 4: bit table of transactions							
TID	L2	L4	L6	L7	L1	L3	L5
T1	1	0	1	0	2	1	1
T2	0	0	0	1	1	1	0
T3	0	0	0	0	0	0	1
T4	0	1	0	1	1	1	1
T5	0	0	0	1	1	1	1
T6	0	0	0	0	0	0	1
T7	1	0	1	0	1	1	1
T8	0	1	0	0	1	1	0
T9	0	0	0	1	1	1	1
T10	0	0	0	1	1	1	1

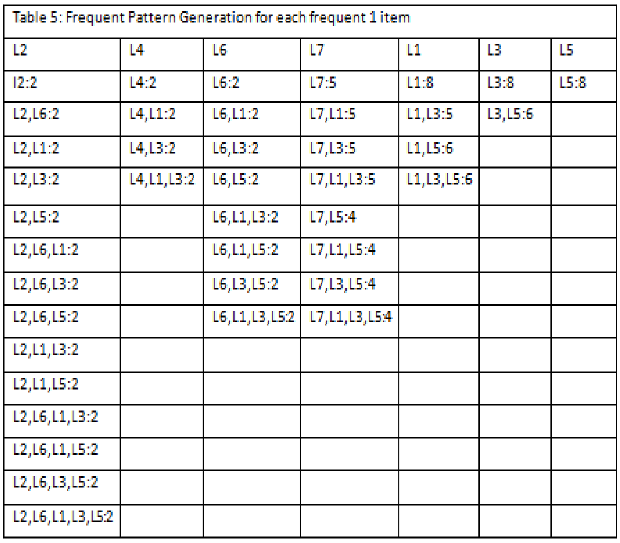

Table 5: Frequent Pattern Generation for each frequent 1 item

L2	L4	L6	L7	L1	L3	L5
I2:2	L4:2	L6:2	L7:5	L1:8	L3:8	L5:8
L2,L6:2	L4,L1:2	L6,L1:2	L7,L1:5	L1,L3:5	L3,L5:6	
L2,L1:2	L4,L3:2	L6,L3:2	L7,L3:5	L1,L5:6		
L2,L3:2	L4,L1,L3:2	L6,L5:2	L7,L1,L3:5	L1,L3,L5:6		
L2,L5:2		L6,L1,L3:2	L7,L5:4			
L2,L6,L1:2		L6,L1,L5:2	L7,L1,L5:4			
L2,L6,L3:2		L6,L3,L5:2	L7,L3,L5:4			
L2,L6,L5:2		L6,L1,L3,L5:2	L7,L1,L3,L5:4			
L2,L1,L3:2						
L2,L1,L5:2						
L2,L6,L1,L3:2						
L2,L6,L1,L5:2						
L2,L6,L3,L5:2						
L2,L6,L1,L3,L5:2						

The first experiment compares the time consumed of original Apriori, and our improved algorithm by applying the five groups of transactions in the implementation.

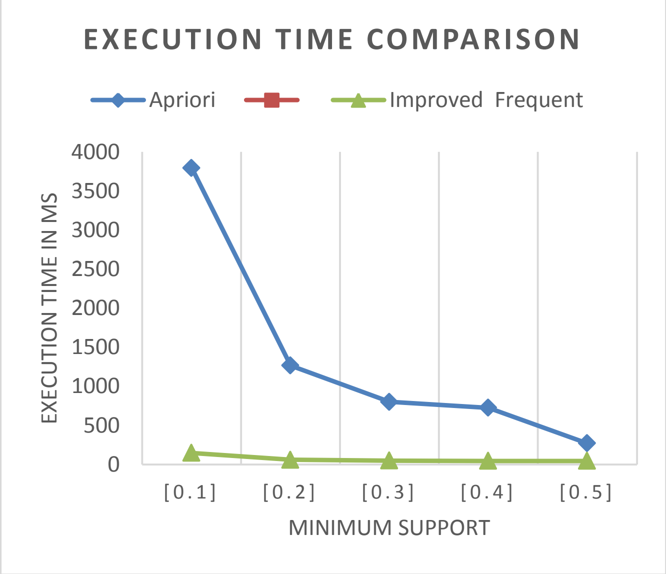

Chart 1: Time consuming comparison for different groups of transactions

The second experiment compares the accuracy of original Apriori, and our proposed algorithm by applying the one group of transactions through various values for minimum support in the implementation.

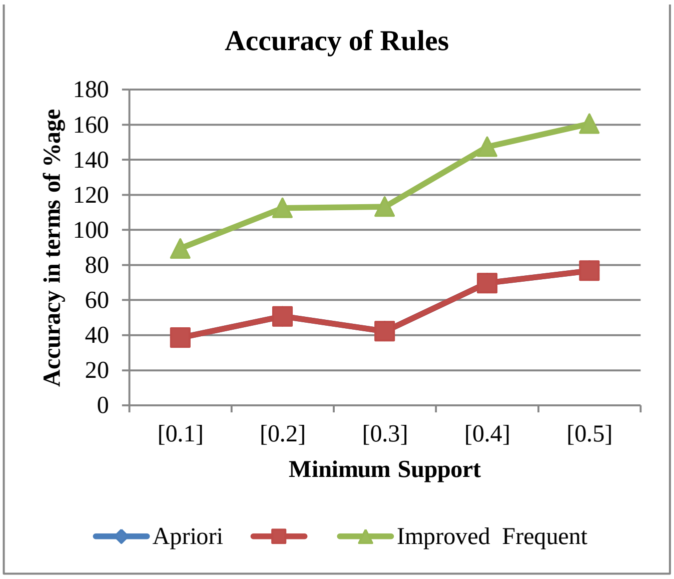

Chart 2: Accuracy comparison for different groups of transactions

5. CONCLUSION

In this paper, an improved Apriori is proposed through reducing the time consumed in transactions scanning for candidate itemsets by reducing the number of transactions to be scanned. Whenever the k of k-itemset increases, the gap between our improved Apriori and the original Apriori increases from view of time consumed, and whenever the value of minimum support increases, the gap between our improved Apriori and the original Apriori decreases from view of time consumed.

REFERENCES

[1] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, “Top 10 algorithms in data mining,” Knowledge and Information Systems, vol. 14, no. 1, pp. 1–37, Dec. 2007.

[2] S. Rao, R. Gupta, “Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm”, International Journal of Computer Science And Technology, pp. 489-493, Mar. 2012

[3] H. H. O. Nasereddin, “Stream data mining,” International Journal of Web Applications, vol. 1, no. 4, pp. 183–190, 2009.

[4] F. Crespo and R. Weber, “A methodology for dynamic data mining based on fuzzy clustering,”

Fuzzy Sets and Systems, vol. 150, no. 2, pp. 267–284, Mar. 2005.

[5] R. Srikant, “Fast algorithms for mining association rules and sequential patterns,” UNIVERSITY OF WISCONSIN, 1996.

[6] J. Han, M. Kamber,”Data Mining: Concepts and Techniques”, Morgan Kaufmann Publishers, Book, 2000.

[7] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, “From data mining to knowledge discovery in databases,” AI magazine, vol. 17, no. 3, p. 37, 1996.

[8] F. H. AL-Zawaidah, Y. H. Jbara, and A. L. Marwan, “An Improved Algorithm for Mining Association Rules in Large Databases,” Vol. 1, No. 7, 311-316, 2011

[9] T. C. Corporation, “Introduction to Data Miningand Knowledge Discovery”, Two Crows Corporation, Book, 1999.

[10] R. Agrawal, T. Imieliński, and A. Swami, “Mining association rules between sets of items in large databases,” in ACM SIGMOD Record, vol. 22, pp. 207–216, 1993

[11] M. Halkidi, “Quality assessment and uncertainty handling in data mining process,” in Proc, EDBT Conference, Konstanz, Germany, 2000.