DEVELOPING AN ENHANCED APRIORI ALGORITHM FOR FREQUENT PATTERN MINING

Komal Vihar Ramani[1], Dr. Paresh Tanna[2]

Article History: Received: 25-07-2023 Revised: 05-08-2023 Accepted: 25-08-2023

Abstract

It is now possible to store an enormous amount of information. This was not possible in the past. The technique of gleaning relevant information from large amounts of data. Such vast amounts of data have seen widespread to adoption of various data mining methodologies. It is useful in a variety of applications, including important fundamental leadership, financial speculation, medical conclusion, and so on. Data mining may function either as an illuminating or a predictive tool, depending on how it's used. One of the capabilities that data mining encompasses is known as affiliation manage mining. This postulation suggests a few of options for going ahead, including covering up affiliation administration mining, post mining, and affiliation administration mining. The task of locating the set that contains all of the subsequent item sets and developing standards that show promise is part of the process of developing affiliation rules. This suggestion suggests a method for calculating the continuous item sets in the circle are being generated by a single output taken from the exchanges database. The approach is outlined in the next paragraph. During this one database examination, the information about the item sets and the events that occurred is recorded in a table that is stored in the primary memory. Instead of looking at the plate in the process of figuring out the standard object sets, this table is looked at.

Keywords: *Enhanced apriorism algorithm, Frequent pattern mining*

[1]Research Scholar, RK University, Rajkot, Gujarat, India

[2]Professor, School of Engineering, RK University, Rajkot, Gujarat, India

1. Introduction

The technique of extracting useful information from massive volumes of data that have been stored in a data warehouse and the databases is a significant challenge. A significant amount of investigation is now being carried out in many different countries in order to get relevant information from the vast amounts of data stored in data warehouses. Throughout this process, a number of different algorithms have been presented in order to detect the relationships between the data that are stored in the database. This, in turn, leads to the mining of the pattern of relationship that may be found within the data. The process of knowledge discovery makes use of association rules. and for making effective management decisions inside an organization based on the findings of associations among data as a step towards making a smarter system (Divan Kirpan et al. 2021 & Cristobal 2010). These rules are used in the process of making a smarter system. In this context, Agarwal and Srikanth presented the first algorithm known as Apriori in the year 1994 with the intention of mining the frequent item set. Because of time constraints and the need for efficient algorithms, a significant amount of investigational work has been carried out in the domain of algorithms. The goal of this research is to develop efficient algorithms that require less time and fewer database scans with the purpose of mining frequently occurring item sets and association rules (J Han 2006). The principle of association is mostly focused on the identification of recurring item sets. Retail establishments typically make use of association rules as a tool to aid in marketing, advertising, inventory management, and the prediction of defects in communications networks. The following portions of this article are structured as follows: The Apriori Algorithm is described in Part 2, along with a functioning example. In the third part, provide more detail on the suggested method for the transaction reduction strategy, along with a working example. Section 4 includes conclusion (Agrawal et al. 1993 & U. Fayyad et al. 1996).

2. Background

The past few decades have seen the development of a large number of association rule algorithms, which can be broken down into two categories: (1) breadth-first search (BFS), which is also known as a candidate-generation-and-test approach such as Apriori, and (2) depth-first search (DFS), which is also known as a pattern-growth approach (Agrawal et al. 1994 & Hu Ji-Ming et al. 2006). Both of these categories can be broken down further into individual subcategories. Before it is possible to count the support values of the k-itemsets, it is necessary to first establish the support values of all of the itemsets that are a part of BFS (k minus 1). On the other hand, DFS will go downward in a recursive manner by adhering to the tree structure that was previously provided. As can be seen in figure 1, each of the algorithms may be separated from the others depending on the strategy that it employs to a) navigate the search space and b) determine the support values of the item sets (Deng et al. 2013). In addition to this, an algorithm could use certain optimizations in order to make the process even more expedient. The most well-known approach that falls into this category is known as the apriorism method. It was initially provided by this method, and it was the downward closure feature of itemset support that was being discussed. This quality is used in an additional way by Apriori, which is to exclude candidates. from consideration before calculating the number of people who support them if they have an uncommon subgroup (Yu Zhen Wang 2003). This optimization problem is made achievable as a result of BFS's guarantee that the support values of each and every subset of a candidate are determined in advance. A single pass over the database is all that is required for Apriori to do its counting of all candidates with cardinality k. The most important step is to search through each transaction in order to find potential prospects. In order to accomplish this goal, the research presented in presents a structure known as a hash tree. The hash-tree is traversed in a descending order using the entries in each transaction (T. C. Corporation 1999).

As they reach one of its leaves, they look for a group of potential candidates that all have the same prefix, which is an essential component of the transaction. They experience this pattern each time they reach a leaf. After this, a search is carried out using these candidates inside the transaction that has been encoded in the form of a bitmap (Chandrani Singh et al. 2011 & Hu Ji-Ming and Xian Xue-feng). The value of the counter that corresponds to the candidate in the tree is determined by whether or not the procedure is successful. increased.

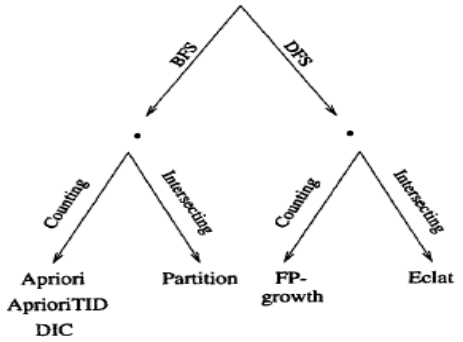

Figure 1: An Organizational Structure for the Different Algorithms

The Apriori Tidd methodology is an expansion of the original Apriori methodology.

- The need to use SQL to generate large itemsets served as the impetus for the development of the SETM approach; candidate itemsets are produced on-the-fly during the pass while data is being read. This goal was the driving force behind the development of the SETM algorithm.

- DIC is another iteration of the Apriori-Algorithm that has been developed. The clear divide that previously existed between counting and creating candidates is blurred by DIC. DIC will begin to generate more candidates based on a candidate once that candidate achieves its min-supp threshold, even if that candidate has not yet "seen" all of the transactions that have occurred.

- The Partition-Algorithm finds out what the support values are by looking at the intersections of sets.

For the Apriori Algorithm, Generating Frequent Sets of Items

The Apriori algorithm is the most well-known one for mining association rules. To determine the number of itemsets that have support, it employs a breadth-first search technique, and it makes use of a candidate generation function that takes use of the downward closure aspect of support (Hong Liu Yuanyuan Xia 2011 & Quang Yang 2011).

The Apriori technique makes use of a technique known as level-wise search, which is an iterative procedure. In this approach, (k+1)-item sets are investigated using k-item sets as the basis for the investigation. The first thing you need to do is find the group of one-item sets that are often referred to as L1. The next step is to make use of L1 in order to identify the two-item sets that occur more often in L2 (Jian Wang et al. 2012). Following that, the collection of standard 3-itemsets known as L3 is located with the assistance of L2, which is utilized. This process will continue to run in this manner until there are no further frequent k-item sets detected.

TABLE 1

TID	*ITEMS*
T1	I1,I2,I5
T2	I2,I4
T3	I2,I3
T4	I1,I2,I4
T5	I1,I3
T6	I2,I3
T7	I1,I3
T8	I1,I2,I3,I5
T9	I1,I2,I3

C_1		L_1	
Itemsets	*Support*	*Itemsets*	*Support*
I1	6	I1	6
I2	7	I2	7
I3	6	I3	6
I4	2	I4	2
I5	2	I5	2

C_2		L_2	
Itemsets	*Support*	*Itemsets*	*Support*
I1,I2	4	I1,I2	4
I1,I3	4	I1,I3	4
I1,I4	1	I1,I5	2
I1,I5	2	I2,I3	4
I2,I3	4	I2,I4	2
I2,I4	2	I2,I5	2
I2,I5	2		
I3,I4	0		
I3,I5	1		
I4,I5	0		

C_3		L_3	
Itemsets	*Support*	*Itemsets*	*Support*
I1,I2,I3	2	I1,I2,I3	2
I1,I2,I5	2	I1,I2,I5	2

Figure 2: Application of the Apriori Algorithm Allows for the Creation of Collections of Often Used Items

With the aid of the accompanying illustration, we will have no trouble understanding the core ideas behind the Apriori. A transactional database that has 9 transactions is shown in Table 1. A TID is a one-of-a-kind identifying number that is assigned to every transaction (Lan fang Lou et al. 2010 & Ekta Garg 2013). Let minimal support value be 2 (min sup=2)

Pseudo Code

```
Initialize: k := 1, C1 = all the 1- item sets;
read the database to count the support of C1 to dete
L1 := {frequent 1- item sets}; k:=2; //k represents t
while (Lk-1 ≠ ø) do
begin
Ck := gen_candidate_itemsets with
the given Lk-1
prune(Ck)
for all transactions t ϵ T do
increment the count of all candidates
in CK that are contained in t;
Lk := All candidates in Ck with
minimum support ;
k := k + 1;
end
Answer := Uk Lk
```

Currently Available Algorithms

The algorithm known as Generic Apriori

```
The general Apriori algorithm is:
T: Transactional data base
Ck: Candidate item set of size k
Lk: Frequent item set of size k
s: Support
Apriori(T, s)
L1 ← { large 1-item set that appear in more than or equal to s transactions }
k ← 2
While Lk-1 ≠ ϕ
Ck ← Join(Lk-1)
For each transaction t in T
For each candidate c in Ck
If(c c t) then
count[c]←count[c]+1
End If
End For
End For
Lk = ϕ
For each candidate c in Ck //Prune

If (count[c] >= s) then
Lk ← Lk U {c}
End If
End For
k ← k + 1
End While
Return Lk
End Apriori
```

While performing the frequency of each candidate in Ck is computed using the general Apriori technique by first searching through the transactional database. After calculating the frequencies for each candidate in a Ck, these frequencies are compared with support, and candidates whose frequencies are more supported are those whose frequencies are higher (Deng Jibing et al. 2010). Those who are lower than support is eliminated from consideration. This leads to the production of Lk as a consequence.

The universal Apriori algorithm suffers from the following deficiencies:

- A number of passes are made through the transactional database. This is due to the fact that each candidate from the candidate set (Ck) that is formed after the Join operation has to be verified for the existence of candidates in each and every transaction of the transactional database.

- The genera Apriori approach is not applicable if there are a significant number of transactions. the appropriate one to use.

Drawbacks of Apriori Algorithm

- The creation of a large number of seldom used item sets contributes to an increase in the space's complexity.

- Due to the enormous amount of itemsets that have been created, an excessive number of database scans are being performed. necessary.

- The amount of time and complexity required to complete a database scan grows proportionately with the size of the database.

3. Proposed Improved Version of the Apriori Algorithm (Enhanced Apriori Algorithm)

Each time a candidate item sets are formed using the traditional Apriori method, the algorithm is required to check the occurrence frequencies of the individual items (Xin-Hua Zhu 2010). Because of the high frequency of querying that will be caused by the manipulation including redundancy, an enormous quantity of resources will be wasted, either in terms of time or space (Zi Rong Yang 2008). As a result, the enhanced method It was hypothesized that mining the association rules might help in the process of building frequent k-item sets. Based on the conventional Apriori algorithm, this innovative method may rapidly identify groups of often occurring items and

remove the subset that does not fall into that category (M. Halkidi 2000). Before, after creating new candidates, the algorithm would make a determination as to whether or not these sets included frequently occurring items.

This enhancement will help reduce the number of times queries are run and the amount of space they take up (Ming Gao 2006). The enhanced The Apriori algorithm has the capability of mining common item sets. without the need for the production of new candidate sets.

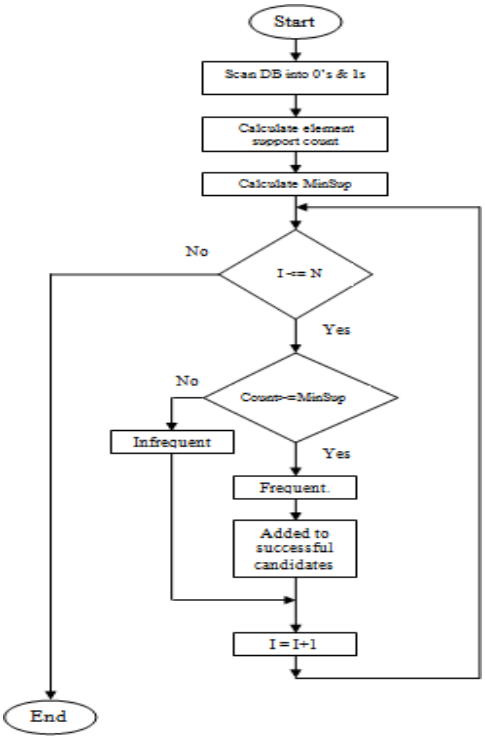

Figure 3: The Improved Apriori Algorithm That Is Being Proposed

Improved Apriori algorithm with deterministic-function pruning as its foundation

Enhanced Stages in the Algorithm

The new and better algorithm is presented in the stages that follow:

Input:

D is a database that contains transactions for the Min sup threshold, which is the minimum number of supporters needed.

Input: Database of transactions, type D

Output: Non-coincidental frequent itemsets

```
{
Ct=subset (C1, t);
For all candidates c ∈ Ct
c.count++
}
L1= Min_sup(C1);
For (k=2; Lk -1 ≠ φ;k + + )
{
Ck=Apriori (Lk-1);
For all transactions t ∈ D
{
Ct=subset (Ck,t);
For all candidates c ∈ Ct
c.count++
Lk= {MinSup(Ck)}
}
Return (∪k Lk )
}
```

1. During the first cycle of the algorithm, every item is considered a candidate for the one-item set C1, and as a result, each item is a member of that set. The computer simply looks at each and every transaction in order to get the total number of times each specific item has been purchased.

2. The set of frequent item sets, which is designated by the letter L1 and can be found by comparing the candidate count to the minimal support count, which is made up of candidate 1-item sets that are able to fulfil the conditions of minimum support, is created by doing so.

3. The procedure begins by generating a candidate set of 2-itemsetd in order to construct the set of frequent 2-itemsets, which is denoted by the letter L2. Next, the transactions in D are analyzed, and the support count of each candidate item set in C2 is obtained; this procedure is then followed by step 2, which is continued until all of the transactions in D have been processed.

4. Based on this, D2 may be derived from L2.

5. Using the data from step 2, generate candidates for category C3, then go to process 2 after scanning category D2 to

calculate the count of each C3 candidate.

6. After the first pass is complete, you should decide which of the candidate item sets are genuinely big. Such item sets should then serve as the germ of an idea for the subsequent round.

7. The germ of an idea for the subsequent round procedure is repeated until there are no bigger item sets to be discovered (Fig.1).

By creating subtests and checking to see whether they pass in the database, the updated Apriori algorithm cuts down on the quantity of database queries performed that are performed while the amount of duplication that occurs (Shao Chun Chang 2011 & A Fang Feng 2010). In order to generate a frequent item set, this technique requires in a shorter amount of time than the conventional Apriori approach does because of the reasons listed above.

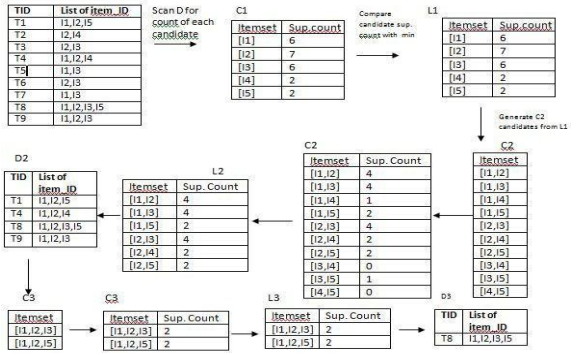

Figure 4: The Creation of a Candidate Item Set and a Frequent Item Set with an Example

Despite the fact that it is the first algorithm to be presented in the area of mining for frequent patterns, as time went on, a great deal of updated algorithms was devised to increase the effectiveness management of both one's time and one's memory as well as to eliminate the complexity of the process (Ji Ming Hu 2006). Counting the number of supporters for a candidate item set is one of the goals of the Apriori algorithm, and here we will propose an alternative method for doing so (Zhen Zhou 2009). Although Apriori primarily focuses on horizontal data layout, his technique is more suitable for vertical data layout, which is important given that Apriorism focus is about the horizontal arrangement of data. The concept of intersection, which originates in set theory, is put to use in this cutting-edge method that we have developed. In the Classical Apriori technique, in order to count the support of candidate set, each record is read one at a time, and the presence of each candidate is confirmed. This is done in order to determine how many records support the candidate set. In the event that a candidate does exist, the number of supporters will grow by one (Zheng et al. 2001). This method requires an iterative scan of the whole database for each candidate set. The length of each candidate set must be equal to or greater than the maximum length of the item set being considered for selection. It also requires a significant amount of time. The improved method, which makes use of the SQL intersect query, determines the total number of common transactions that are included in each member of the candidate set. This allows us to compute the support for the approach. As comparison to the traditional Apriori method, this one is much more time efficient (F. H. AL-Zawaidah 2011).

4. Results and Discussion

The tests were run on a system that has a 450 hard drive, a processor with an Intel Core i5 architecture, and 4 gigabytes of random-access memory (RAM). We put into practice the fundamental Apriori algorithm as well as the QR algorithm in order for the purpose of conducting the performance studies (Zhoushan 2010). Both The algorithm known as Apriori by itself and the Apriori method with substantial attribute selection were investigated, and the results were compared. An investigation has been carried out by altering a variety of characteristics, such as the amount of support and confidence, the number of records and the number of dimensions that are being used. The fundamental calculation behind Apriori:

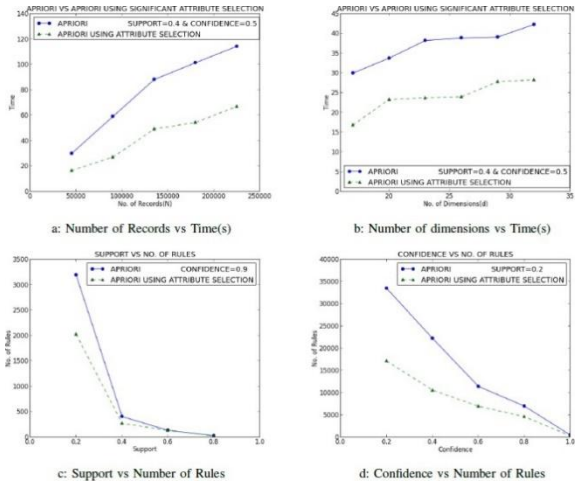

Figure 5: Results

One of the most effective association rule mining methods is the time-honored Apriori algorithm. The main drawback it has, however, is that it is not scalable, and as a result, it cannot handle large-scale datasets that have been collected of dimensions. In addition to this, the amount of time required grows exponentially with the size of the dataset. Experiments have shown that the standard Apriorism algorithm takes an infinitely long time compared to the Apriori algorithm that uses significant attribute selection when it comes to constructing association rules (R. Agrawal et al. 1993). There is (one) roughly a lessening of the around 40% time compared to the primary programming language when the records totaled in the count is modified, and there is about a 35% reduction when the number of dimensions is adjusted (Mao Sheng Dou 2009 & Rakesh Agrawal and Ramakrishnan Srikant 1994). This is as may be seen in table 2, as seen above.

Table 2: A Timeline (s) vs The quantity of Records

No of records	Apriori	Apriori with QR
45000	30	16.57
90000	59.11	27.17
135000	88.3	49.22
180000	101.48	54.46
225000	114.41	67.04

5. Conclusion

Throughout the course of the previous 10 years, a significant number of individuals have developed, implemented, and analyzed a variety of algorithms in an effort to tackle the issue of mining frequent item sets in the most effective manner possible. For instance, "An Improved Scaling Apriori for Association Rule Mining Efficiency" is a widely popular version of the Apriori algorithm. Its purpose is to maximize the effectiveness of mining association rules. But, when we examined the performance of both algorithms using the same implementation, we found that ours was far superior to his. We are able to get the conclusion that one of the primary goals of this fresh strategy is to shorten the amount of time it takes. As we shown above, the time required by the Enhance apriorism technique is much shorter than that required by traditional algorithms. In the event of a huge database, this is going to be really useful in terms of reducing the amount of time spent. This essential concept is without a doubt going to throw up a whole new door for forthcoming researchers who are interested in working within the scope of the industry of data mining.

References

Divan Kirpan, Slawomir Stankov, "Educational Data Mining for Grouping Students in E-learning System", In Proceeding of ITI 2012 34th Int. Conf. on Information Technology Interfaces, June 2012

Cristobal Romero," Educational Data Mining: A Review of the State of the Art", IEEE transaction on systems, MAN, and Cybernetics-part c: Application and Reviews, VOL. 40, NO. 6, November 2010

Agrawal, R., Imieliński, T. and Swami, A.N., Mining Association Rules between Sets of Items in Large Databases. In Proceedings of SIGMOD, 207- 16, 1993.

Zheng, Z., R. Kohavi and Mason, L., Real world performance of association rules. Sixth ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2(2),86-98, 2001.

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.

Deng Jibing, Hu JuanLi, Chi Hehua, Wu Jumbo, "An Apriori-based Approach for Teaching Evaluation", IEEE, 2010.

Xin-Hua Zhu, Ya-qigong Deng, Qing-ling Zeng," The Analysis on Course Grade of College-

wide Examination Based on Mixed Weighted Association Rules Mining Algorithm", ICCASM 2010.

Zhoushan, "Study and Analysis of Data Mining Technology in College Courses Students Failed", IEEE, 2010.

Chandrani Singh Dr. Arpita Gopal Santosh Mishra "Extraction and Analysis of Faculty Performance of Management Discipline from Student Feedback using Clustering and Association Rule Mining Techniques", IEEE 2011.

Hong Liu Yuanyuan Xia," Teaching Evaluation System Based on Association Rule Mining", IEEE 2011.

Quang Yang, Yanhong Hu, "Application of Improved Apriori Algorithm on Educational Information", IEEE 2011.

Jian Wang, Zhubin Lu, Wenhua Wu and Huzhou Li," The Application of Data Mining Technology based on Teaching Information", ICCSE 2012.

Lan fang Lou, Qingxian Pan, Xiuying Quiz," New Application of Association Rules in Teaching Evaluation System", IEEE 2010.

Deng Jibing, Hu JuanLi, Chi Hehua, Wu JueboAn, "Apriori-based Approach for Teaching Evolutionlike 2010.

Yu Zhen Wang. Analysis and Discussion of Web Data Mining. Development and application of computer. Vol 16:72-74(2003).

Zi Rong Yang. Study of Fields-oriented High Quality Information Retrieval Based on Web Data Mining. Guizhou University Master Thesis.2008.

Mao Sheng Dou. Research and application on association rules based data mining. Changchun University of Science and Technology Master Thesis.2009.

Ming Gao. The Research and Application on the Algorithms of Mining Association Rules. Shandong Normal University Master Thesis.2006.

Shao Chun Chang. Efficient frequent item set discovery methods and improved Apriori.Jiangsu University of Science and Technology Master Thesis.2011.

A Fang Feng. An optimization algorithm of association rules Apriori.Consumer guide. Vol 25:265-266(2010)

Ji Ming Hu. Research and Improvement on Apriorism' s Algorithm in Mining with Association Rules. Computer technology and development. Vol 16:99-102(2006)

Zhen Zhou. The research of Web data mining system based on E-commerce. Journal of Hunan Industry Polytechnic. Vol 9:58-59(2009)

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.

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

T. C. Corporation, "Introduction to Data Miningand Knowledge Discovery", Two Crows Corporation, Book, 1999.

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

M. Halkidi, "Quality assessment and uncertainty handling in data mining process," in Proc, EDBT Conference, Konstanz, Germany, 2000

Rakesh Agrawal and Ramakrishnan Srikant Fast algorithms for mining association rules in large databases. Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.

J Han, "Data Mining Concepts and Techniques "Second Edition. Morgan Kaufmann Publisher, 2006,pp.123-134

Hu Ji-Ming and Xian Xue-feng. The Research and Improvement of Apriorism for association rules mining [J]. Computer Technology and Development 2006 16(4) 99~104.

Zhuang Chen, Shebang Cai, Qulin Song and Chenglei Zhu, "An Improved Apriorism Algorithm based on Pruning Optimization and Transaction Reduction," AMISEC 2011, 2nd IEEE International Conference, pp1908-1911.

Ekta Garg(2013) A Survey On Improved Apriori Algorithm International Journal of Engineering Research & Technology (IJERT) Vol.