Int. J. Open Problems in Compt. Math, Vol. 2, No. 3, September 2009
ISSN 1998-6262; Copyright © ICSRS Publication, 2009
www.i-csrs.org

An Automated Association Rule Mining Technique With Cumulative Support Thresholds

C.S.Kanimozhi Selvi, and A.Tamilarasi,

Assistant Professor - CSE , Kongu Engineering College, Perundurai
kani_abi@yahoo.co.in
Professor - CSE , Kongu Engineering College, Perundurai

Abstract

A*ssociation rule mining is a task in data mining for discovering the hidden, interesting associations between items in the database. To find the relevant associations, the user has to specify support and confidence thresholds. These thresholds play an important role in deciding the number of appropriate rules found. User has many problems in specifying the appropriate thresholds, without the knowledge of itemsets and their frequency in the database. A high support threshold keeps away from generating more number of rules, but at the cost of losing interesting rules of low support. This paper proposes an approach to set suitable support thresholds for frequent itemset generation. Experimental results show that this approach produces the interesting rules without specifying the user specified support threshold.*

***Keywords**: Frequent itemset, Association rules, Confidence Lift Measure, Cumulative Support, Apriori*

1 Introduction

Mining association rules from a large database, has been an important task in the area of data mining to discover hidden, interesting associations that occur between various data items. Consider a transaction database, where each transaction is a set of items, and an association rule reveals the relationship between items. An association rule [1] is expressed in the form X → Y, where X, Y are itemsets. This rule exposes the relationship between the itemset X with the itemset Y. The interestingness of such a rule is measured by support and confidence. The support is measured as the percentage of transactions in the database that contain both X and Y itemsets. The confidence is the percentage of transactions in the database with itemset X also contains the itemset Y. To discover interesting association rule, the support and the confidence of the rule should satisfy a user-specified support threshold called *minsup* and a confidence threshold called *minconf*, respectively.

Association rule mining consists of two steps [1]:
1) Finding all the frequent itemsets that satisfies support thresholds.
2) Generating interesting association rules from these frequent itemsets.

The main difficulty in applying association rules mining is the setting of support threshold. Many approaches assume that all items in the database are of the same kind and have similar frequencies in the database but this assumption is not applicable in reality. In real life applications, some items appear very frequently in the database, while others hardly ever appear and the frequent itemsets are alone not interesting, but the rare items also. If a high support threshold is specified, it misses some interesting rules and if a low support threshold is used, it generates numerous unnecessary rules. Hence, the need for specifying the appropriate support threshold arises. A technique with automated support threshold which is appropriate at each level is one among the right choice to tackle this situation. To achieve this task, this paper proposes a technique to use automated supports for generating appropriate rules without missing interesting rules.

The proposed algorithm is based on the Confidence – Lift Measure specified in paper [16].This approach finds an initial minsup value by analyzing the itemsets and their frequency. It also proposes a cumulative support threshold on the subsequent levels based on the previous level support and the items considered in the current level.

The rest of the paper is organized as follows: Section 2 revisits the problem of association rule mining and explores the need for confidence-lift based automated support thresholds for the generation of association rules. Section 3 provides an insight into the modified association rule framework and explains the support specification and mining process for this model. Section 4 reports the experimental results on the IBM synthetic data upon rule set size. Finally, the conclusions are pointed out in Section 5.

2 An Overview of Related Work

Association rule mining is considered to be an interesting research area and studied widely [1-11] by many researchers. They mainly fall into two categories:

specification of user specified support and confidence thresholds or alternative measures to find the association rules. Most of the algorithms which follow level wise mining approach rely on support and confidence thresholds or the measures based on these thresholds. They are mainly varied in the usage of those thresholds. Algorithms like Apriori [1,12] and FP-Growth [13] employ the uniform support threshold at all levels. If a suitable support threshold is not specified the user would have to face two problems.

1. Generation of less number of rules upon specifying the high minimum support.
2. Generation of too many rules sometimes uninteresting upon specifying low minimum support. This requires stronger rule pruning techniques to be employed.

The paper [4] argues that the single support threshold for the whole database is inadequate, because it cannot capture the inherent nature and / or frequency differences of the items in the database. Therefore it suggests a model which uses reduced minimum support at each level. This approach also not solves the problem since the items exist at the same level may be of different kind and / or of different frequency.

Another solution lies in developing support constraints, which specify minimum support required for the group of similar itemsets. If more than one support constraint is satisfied by an itemset, then the one with minimum support value should be adopted. This has been studied in the paper [6]. This approach also has some problems as pointed out in [17]:

1. It assumes that the user has adequate domain knowledge.
2. Using the approach, one can analyze the nature of the items but can't able to know frequency of the items until all the items in the database are scanned and candidate items are generated.

Algorithms with alternative measures have also been proposed. Correlation based framework [10] is one such algorithm which uses contingency table to find positively correlated items. Although the correlation framework discovers strongly correlated items, the computation costs are too high and too many ineffectual itemsets would be generated [10]. In [15], mining high confidence associations without support constraints is proposed. The paper [11], the authors avoided the use of support measure and employed a combination of random sampling and hashing technique to find the interesting associations. The approaches without support thresholds are not suitable because is not suitable for capturing the real implications.

An effort is made by [14] for finding the N-most frequent itemsets. It is easier for the users to specify N to control the result, but leading to user intervention. Another paper [17], automatically calculates and uses the minimum support adaptively at each level. In [16] an automatic support model based on Confidence Lift measure is proposed for efficiently mining high-confidence and positive lift associations without consulting the users. Although the algorithm derives the minimum support dynamically from the item support, it also leaves out some interesting rules composed of more than two items because the specification is derived from frequent 2-itemsets as pointed out in [16].

This paper leaves the user free from specifying any constraints including support constraints. It proposes an approach to use an automatic support threshold called cumulative support threshold based on confidence-lift measure so that all frequent, interesting and meaningful rules would be generated. This avoids the user intervention and generates more appropriate and meaningful rules. Experiment results on synthetic data show that the proposed technique is effective.

3 Association Rule Mining Framework

As in [1], Consider a given transaction database T = $\{r_1, r_2, \ldots, r_n\}$, where each record $ri, 1 \leq i \leq n$ is a set of items from a set I of items, i.e., r$i \subset I$. The basic association rule model as follows:
Let I = $\{a_1, a_2 \ldots, a_m\}$ be a set of items. Let R be a set of records in the database, where each record r is a set of items such that $r \subseteq I$. An association rule is an expression of the form, A →B, where $A \subset I, B \subset I$, and $A \cap B = \phi$.

3.1 Proposed Model

This model adopts the apriori algorithm based on confidence-lift measure. Here, the minimum support threshold for an itemset is derived for every level of itemsets using cumulative cupport (cs) and minimum support (ms).The first level minimum support(ms) is identified using confidence lift support constraint specified in [16]. Items are sorted according to the increasing order of their support (sup).

3.1.1 Computation of Cumulative Support

An itemset $A = \{a_1, a_2, \ldots, a_k\}$, where $a_i \subset$ I is frequent if the support of A is greater than the lowest of all minimum support of items in A. Cumulative support is used to denote the collective supports of items in the previous level. It is derived from level 2 onwards.

Cumulative support(cs) for two itemsets is

$$cs(a_1,a_2)=\{ sup(a_1) * sup(a_2) \}$$

On subsequent levels the cumulative support of an itemset is calculated as the minimum of the cumulative support of its subsets. For an example consider the cumulative support of three itemset,

$$cs(a_1, a_2, a_3)=min \{ cs(a_1,a_2), cs(a_2, a_3), cs(a_1,a_3) \}$$

Cumulative Support (cs) for level two is identified using the support of each individual item. For third level cumulative support (cs) is identified using the second level cumulative support (cs) and so on.

3.1.2 Computation of Minimum Support

The databases T is scanned to obtain the support (sup) of each item and the minimum support (ms) is set according to the below equation**.** Let *ms*(*a*) denote the minimum support of an item *a, where* $a \subset I$ *,where* $a_1 \leq a_2 \leq a_3 \leq ... \leq a_k$

$$\begin{aligned} ms(a_i)= &sup(a_i)*max\{minconf,sup(a_{i+1})\}, \text{ if } 1 \le i \le n-1 \\ &sup(a_i), \text{ if } i=1 \end{aligned}$$

Let $ms(a_1,a_2)$ denote the minimum support of an transaction of itemset $\{a_1,a_2\}$.

$$ms(a_1, a_2) = min \{ sup(a_1, a_2), cs(a_1, a_2)\}$$

minimum support for three item sets is computed as follows:

$$ms(a_1, a_2, a_3)=min \{ sup(a_1, a_2, a_3), cs (a_1, a_2, a_3) \}$$

Minimum Support (ms) for level two is identified using the support (sup) and cumulative support (cs) of itemsets. For third level minimum support (ms) is identified using the support (sup) and cumulative support (cs) of itemsets. Similarly it is carried out for the remaining levels.

3.1.3 Selection of Frequent Itemsets

The frequent itemsets are generated based on the cumulative support value and the minimum support value at each level. The itemset is frequent if and only if its satisfies the below condition
The itemset (a, b) is frequent if and only if

$$cs(a, b) \geq \min\{ ms(a), ms(b)\}$$

The itemset (a, b, c) is frequent if and only if

$$cs(a, b, c) \geq \min\{ ms(a, b), ms(b,c)), ms(a,c)\}$$

and so on.

These conditions are applied on subsequent levels of candidate generation for each itemset.

4 Frequent Itemset Generation

1. Scan the database *D* to obtain the support of each item and set the minimum item supports. Set *L*1 as the sorted list of items in ascending order of their minimum item supports.
2. Apply apriori-gen in the subsequent candidate generation step to generate candidate 2-itemsets, while invoke *Ck*-gen for the other candidate *k*-itemsets, for *k* ≥3.

Algorithm Automated_Apriori

Input: Database T, and minimum confidence minconf
Output: L, frequent itemsets in T

```
Method:
L_1= find frequent 1 itemsets(R)
for (k = 2; L_{k-1} ≠Ø, k++) do
  C_k=candidate-gen (L_{k-1})
end
for each Record r in Є R do
  C_t = subset (C_k, r);
  for each candidate c Є Ct do c.count++;
  end
end
Compute ms (a) for all a ⊂ I; / compute the minimum support for all items*/
 L_1 =sort (I); /* ascending sort according to ms (a)*/
 for (k=2; L_{k-1} ≠ Φ; k++) do
     if k=2 then
       cs_2=cs_ gen (L_{k-1});
       ms_{2=} ms_ gen (cs_2);
```

```
        $L_2$=frequent($cs_2$, $ms_1$);
    else
        $cs_k$=cs_ gen ($L_{k-1}$);
        $ms_{k=}$ ms_ gen ($cs_k$);
        $L_k$=frequent ($cs_k$);
end for
        return L= $U_k$ $L_k$
```

```
Procedure Candidate-gen($L_{k-1)}$
for each itemset  $l_1$ Є $L_{k-1}$
  for each itemset $l_2$ Є $L_{k-1}$
      perform join operation l1    $l_2$
       if  has_infrequent_subset(c, $L_{k-1)}$
           prune c;
      else
          add c to $C_{k;}$
      end if
  end
 end
return $C_{k;}$
```

```
Procedure has_infrequent_subset(c, $L_{k-1)}$
for each (k-1) subset s of c
  if s is in $L_{k-1}$
   return false;
  else
  return true;
  end if
end
```

```
procedure cs_ gen ($L_{k-1}$)
         $C_{k=}$ apriori_ gen ($L_{k-1}$);/* joins $L_{k-1}$ with  $L_{k-1}$ */
          if k=2 then s=size of $L_{k-1}$
             for(i=0; I ≤ s; i++)
                  $cs_2$ ($a_{i,}$ $a_{i+1}$) ={ sup($a_i$)*sup($a_{i+1}$)}
          else
             for each itemset $A$= <.$a_1$, $a_2$, …, $a_k$> ∉ $C_k$ do
             for each ($k$-1)-subset $A'$=<$a_{i1}$, $a_{i2}$, …, $a_{ik-1}$.>of $A$ do
                   $cs_k$ (A) =min($cs_{k-1}$ ($a_{i1}$), $cs_{k-1}$ ($a_{i2}$), …, $cs_{k-1}$ ($a_{ik-1}$ ));
```

```
procedure ms_ gen ($cs_k$))
        for each itemset $A = <.a_1, a_2, \ldots, a_k> \in C_k$ do
            $ms_k = \min\{ sup(A), cs_k (A) \}$;
```

```
procedure frequent ($cs_k$);
        for each itemset $A = <.a_1, a_2, \ldots, a_k> \in C_k$ do
            for each ($k$-1)-subset $A' = <a_{i1}, a_{i2}, \ldots, a_{ik-1}.>$ of $A$ do
                if $cs_k (A) \geq \min\{ ms_{k-1} (a_{i1}), ms_{k-1} (a_{i2}), \ldots, ms_{k-1} (a_{ik-1}) \}$;
                  else delete $A$ from $C_k$;
```

Frequent item sets are generated based on the above algorithm. This algorithm generates most relevant, appropriate frequent itemsets.

5 Rule Generation

The proposed algorithm adopts the confidence based rule generation model of apriori [1] for rule generation. The confidence threshold can be used to find out the interesting rule set. The confidence of a rule is its support divided by the support of its antecedent. For example, the following rule $\{ a_1, a_2\} \rightarrow \{ a_3\}$ has confidence equivalent to support for $\{ a_1, a_2, a_3\}$ / support for $\{ a_1, a_2\}$. Association rules are generated as follows. For each frequent itemset (f), all non empty subsets of the frequent itemset(f) are generated. For every non empty subset s of f, the rule s →(f-s) is formed, if support-count (f) / support-count of (s)≥ min_conf.

After the generation of frequent items, the algorithm checks if it satisfies the min_conf threshold. If their confidence is larger than min_conf then they will be generated as interesting association rules. Otherwise, the rules will be discarded. The algorithm for rule generation is given below:

```
Algorithm: Rule_gen($L_k$, min_conf)
for each frequent itemset $f \in L_k$ do
for each nonempty subset $s$ of $f$ do
if $c.count(l)$/c.count($s$) $\geq$ min_conf then
output the rule $s \Rightarrow (f - s)$;
end
end
```

6 Experimental Results

In this section, the proposed Automated_Apriori algorithm obtains the frequent itemsets by finding cumulative Support and minimum support at each level. A minimum support of 50% is used at each level of candidate generation. To find the interesting association rules the user specifies the minimum confidence. Here the user can decide the minimum confidence level according to his need. We use different minimum confidence thresholds at each run.

We evaluate the proposed algorithm against CLS_Apriori[16]. The runs are examined based on the number of rules found. All experiments are performed on an Intel Pentium-IV with 256 MB RAM, running Windows 98.

We use two synthetic data sets generated from IBM synthetic data generator [1]: T10.I4.D100K and T40.I10.D100K. Characteristics of these two data sets are shown in Table 1.

Table 1 Dataset Characteristics

	T10I4D100K	T40I10D100K
Number of Transactions	1,00,000	1,00,000
Number of items	870	942
Minimum item frequency	0.001%	0.005%
Maximum item frequency	7.8%	28.738%
Average item frequency	0.11%	0.10%

Figure 6.1 Shows comparison of two algorithms. Automated_Apriori generates more number of rules than CLS_Apriori, while varying the minimum confidence levels. The whole process is automated and the user has to interrupt only for specifying the minimum confidence level.

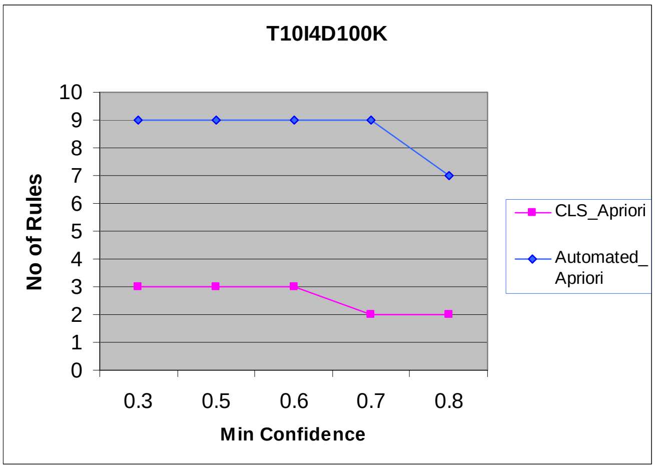

Figure 6.1 CLS_Apriori Vs Automated_Apriori

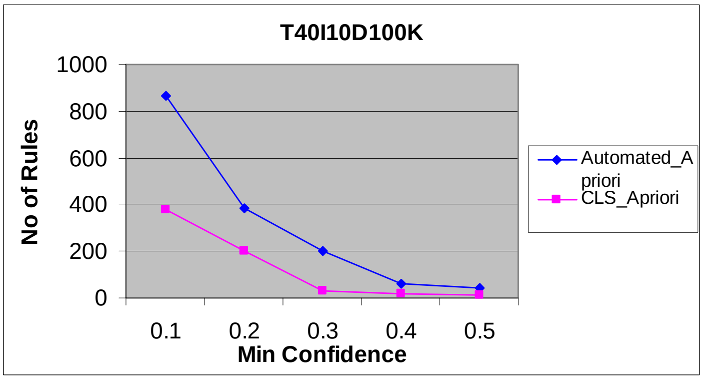

Figure 6.2 CLS_Apriori Vs Automated_Apriori

7 Conclusion

This paper investigates the problem of setting an appropriate support threshold for each itemset at each level. It proposes a confidence-lift-based support threshold which can be automatically set from the itemset support. Experimental results showed that the proposed model is performing well and generates relevant rules without missing interesting rules.

8 Open Problems

Most of the times, the end user will be in dilemma for selecting the appropriate interestingness measure. Hence, more automatic and intelligible association rule mining techniques need to be developed for the end users without the user specified support threshold so that the end users will not be in dilemma.

References

[1] R. Agrawal, T. Imielinski, and A. Swami, Mining association rules between sets of items in large databases.*Proceedings of 1993 ACM-SIGMOD International Conference on Management of Data* (1993) 207-216.

[2] S. Brin, R. Motwani, J.D. Ullman, and S. Tsur, Dynamic itemset counting and implication rules for marketbasket data. *Proceedings of 1997 ACM-SIGMOD International Conference on Management of Data* (1997) 207-216.

[3] J. Han and Y. Fu, Discovery of multiple-level association rules from large databases. *Proceedings of the21st International Conference on Very Large Data Bases* (1995) 420-431.

[4] B. Liu, W. Hsu, and Y. Ma, Mining association rules with multiple minimum supports. *Proceedings of 1999 ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining* (1999) 337-341.

[5] M.C. Tseng and W.Y. Lin, Mining generalized association rules with multiple minimum supports.*Proceedings of International Conference on Data Warehousing and Knowledge Discovery* (2001) 11-20.

[6] K. Wang, Y. He, and J. Han, Mining frequent itemsets using support constraints. *Proceedings of the 26th International Conference on Very Large Data Bases* (2000) 43-52.

[7] M. Seno and G. Karypis, LPMiner: An algorithm for finding frequent itemsets using length-decreasing support constraint. *Proceedings of the 1st IEEE International Conference on Data Mining* (2001).

[8] J. Li and X. Zhang, Efficient mining of high confidence association rules without support thresholds.*Proceedings of the 3rd European Conference on Principles and Practice of Knowledge Discovery in Databases* (1999).

[9] C.C. Aggarwal and P.S. Yu, A new framework for itemset generation. *Proceedings of the 17th ACM Symposium on Principles of Database Systems* (1998) 18-24.

[10] S. Brin, R. Motwani, and C. Silverstein, Beyond market baskets: generalizing association rules to correlations. *Proceedings of 1997 ACM-SIGMOD International Conference on Management of Data* (1997)265-276

[11] E. Cohen, M. Datar, S. Fujiwara, A. Gionis, P. Indyk, R. Motwani, J.D. Ullman, and C. Yang, Finding interesting associations without support pruning. *Proceedings of IEEE International Conference on Data Engineering* (2000) 489-499.

[12] R. Agrawal and R. Srikant, Fast algorithms for mining association rules. *Proceedings of the 20th International Conference on Very Large Data Bases* (1994) 487-499.

[13] J. Han, J. Pei, and Y. Yin. Mining Frequent Patterns without Candidate Generation, Proc. 2000 ACM-SIGMOD Int. Conf. on Management of Data (SIGMOD'00),Dallas, TX, May 2000.

[14] Ngan, S-C., Lam, T.,Wong, R.C-W. and Fu, A.W-C. (2005) ,Mining *N*-most interesting itemsets without support threshold by the COFI-tree, *Int. J. Business Intelligence and Data Mining,* Vol. 1, No. 1, pp.88–106.

[15] Y. Cheung and A. Fu. Mining frequent itemsets without support threshold: with and without item constraints. IEEE Trans. on Knowledge and Data Engineering, 16(9):1052–1069, 2004.

[16] Wen-Yang Lin and Ming-Cheng Tseng,*,* Automated support specification for efficient mining of interesting association rules , Journal of Information Science **,**Volume 32 , Issue 3 (June 2006) : 238 - 250 , 2006 ,ISSN:0165-5515

[17] C. S. Kanimozhi Selvi and A. Tamilarasi, Association Rule Mining with Dynamic Adaptive Support Thresholds for Associative Classification, Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007) Vol. 2, pp 76-80