2010 3rd International Conference on Computer and Electrical Engineering (ICCEE 2010)
IPCSIT vol. 53 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V53.No.2.92

Mining Condensed Non-Redundant Level-Crossing Approximate Association Rules

Zhao Yuhang[1+], Liu Jianbo[1] and Zhang Lei[2]

[1]School of Computer Science, Sichuan University

Chengdu, Sichuan Province, China

[2]School of Computer Science and Technology, China University of Mining and Technology

Xuzhou, Jiangsu Province, China

Abstract—In association rule mining one intractable problem is the huge number of the extracted rules, especially, in the case of level-crossing association rules. In this paper, aiming at the redundancy produced during level-crossing association rules mining, an approach for eliminating level-crossing approximate redundant rules is proposed. In the method, the redundancies are divided combination with the dataset's hierarchy or taxonomy into two categories: hierarchical Self-Redundancy and Inter-Redundancy, thus in the mining processing, deleting the Self-Redundant rules, removing the redundant rules from the Inter-Redundancy based on their definitions and characters in respective steps. The experiments show that the number of the extracted rules has been considerably reduced.

Keywords-component; level-crossing association rules; redundant rules; approximate basis

1. Introduction

An important issue in association rule mining is related to the huge number of the generated rules and the presence of many redundancies, since lots of the extracted rules can be replaced by other rules. Many efforts have been done in reducing the number of the extracted rules. One technique that uses the frequent closed itemsets instead of the traditional frequent itemsets to generate non-redundant association rules effectively reduces the number of rules [1,2,3]. The work in [4] provides a more concise and lossless representation of association rules, which is characterized by frequent closed itemsets and their generators. It proposes a min-max basis of association rules that contains the non-redundant association rules having minimal antecedent and maximal consequent. The work presented in [5] proves that the redundancy elimination based on the min-max basis does not reduce the belief of the extracted rules, and all rules can be deduced from the basis. Therefore, the basis is a lossless representation of association rules.

However, the previous work has only dealt with redundancy in single level datasets. For the redundancy of association rules, it is different between the multi-level datasets, namely hierarchical redundancy and the single level datasets. The original work in mining multi-level association rules presented in [6,7] primarily focused on finding frequent itemsets at each of level in the datasets and only brief work discussed about the removal of the hierarchical redundancy. Later work by Thakur, Jain & Paradasani in [8] proposed an approach for discovering level-crossing frequent itemsets which are composed of items from two or more different

[+] Corresponding author.
E-mail address: zhaoyuhang.scu@gmail.com

levels, but it did not eliminate the redundancies in the extracted rules. Recent work presented in [9,10] has extended the non-redundant approaches in single level to multi-level datasets to fill this grasp.

This paper proposes an extension of the previous work, which removes the redundancy of level-crossing approximate association rules that have a confidence less than 1. The redundancies can be divided into two categories: hierarchical Self-Redundancy and hierarchical Inter-Redundancy, through the use of the dataset's hierarchy or taxonomy. An approach is proposed to eliminate these redundancies in line with their definitions and characters in respective steps.

The paper is organized as follows. Section II discusses related work in multi-level datasets and the motivation of this paper. The categories of hierarchical redundancies and the approach for eliminating the redundancies are presented in Section III. Section IV shows the experiments and results. Lastly, the conclusions and the future work close the paper in Section V.

2. Related Work and Motivation

A multi-level dataset implies a concept tree through using of hierarchy or taxonomy (Figure 1). For mining multi-level association rules, each item of the multi-level dataset is encoded as a sequence of digits (Table I).

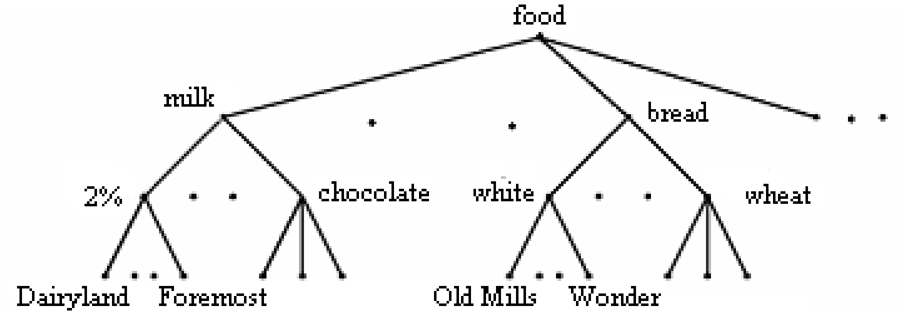

Fig.1. A taxonomy of multi-level dataset

TABLE I. AN ENCODED MULTI-LEVEL TRANSACTION TABLE

TID	Items
1	[1-1-1, 1-2-1, 2-1-1, 2-2-1]
2	[1-1-1, 2-1-1, 2-2-2, 3-2-3]
3	[1-1-2, 1-2-2, 2-2-1, 4-1-1]
4	[1-1-1, 1-2-1]
5	[1-1-1, 1-2-2, 2-1-1, 2-2-1, 4-1-3]
6	[1-1-3,3-2-3,5-2-4]
7	[1-3-1,2-3-1]
8	[3-2-3,4-1-1,5-2-4,7-1-3]

The earliest approaches for mining multi-level association rules are presented in [6,7], however, these have only extracted association rules which belong to the same hierarchical association rules at each of level in the datasets. The ancestor and the consequent in the rules belong to the same concept level, for instance, 'milk=>bread' and '2% milk=>wheat bread'.

The same hierarchical association rules are extracted at each concept level. For many applications, the rules do not necessarily need extracting or expending in a special level. For example, the itemset {2% Foremost milk, white bread} doesn't belong to the same hierarchical itemsets, because there is no such a concept level that can describe the brand and content of milk as well as the content of bread at the same time. Actually that is a level-crossing itemset as the first item belongs to the lowest level, while the second item is from a different concept level. The level-crossing concept is an affiliated concept to the same hierarchical concept in multi-level dataset, in the process of the level-crossing frequent itemsets, to find itemsets L_k (k≥2) in level l, a set of candidate k-itemsets is not only generated by joining L_{k-1} with itself, but also joining L_{k-1} with the 1-itemsets of each level i ($1 \leq i < l$). An approach is proposed to find level-crossing association rules by adapting existing multi-level association rules mining techniques [8].

The majority of the previous work focused on finding the frequent itemsets in multi-level datasets, and did not focused on removing the hierarchical redundancy of extracted association rules. Recent work in [9,10] proposes an approach for removing hierarchical redundancy in approximate basis association rules. This expands the techniques that reduce redundancy in single level datasets into multi-level datasets. However, it requires the identical consequents which confine the application of the method in two rules during eliminating redundant rules. To solve this problem, this paper makes a further work to remove the hierarchical redundancy in level-crossing approximate basis rule set.

3. Mining Condensed Non-Redundant Level-Crossing Approximate Association Rules

The minimum support and confidence thresholds are two measures of usefulness of rules. However, it doesn't guarantee that the rules with high support and confidence values actually convey new information. This section firstly presents the definitions and categories of hierarchical redundancies, and then an approach is proposed to eliminate the redundancies.

3.1. Hierarchical redundancy

To describe the hierarchical redundancy in approximate basis association rules, some definitions are proposed at first as follows.

Definition 1:(MinPartof relation) Let A and B be two itemsets of multi-level dataset, A and B have a relation of MinPartof if (1) the itemset A is made up of items where at least one item in A is descendant from the items in B and (2) the itemset B is made up of the items where at least one item in B is an ancestor of the item in A and (3) the other non-ancestor items in B are all present in itemset A, donated by *MinPartof*(A,B).

Definition 2: (MaxPartof relation) Let A and B be two itemsets of multi-level dataset, A and B have a relation of MaxPartof if (1) the itemset A is made up of items where at least one item in A is descendant form the items in B and (2) the itemset B is made up of the items where at least one item in B is an ancestor of the item in A and (3) there are not two items in A having the same ancestor item in B and (4) the other non-consequent items in A are all present in itemset B, donated by *MaxPartof*(A,B).

Based on Definition 1, for two itemsets A and B, if there exists the relation of *MinPartof*(A,B), some items in A can not be present in B. For example, in the itemsets {2% Foremost milk, white bread, wheat bread} and {2% milk, white bread}, the item '2% milk' belongs to more general/abstract concept level than '2% Foremost milk', they have an ancestor-descendant relationship, the two itemsets have a relation of *MinPartof*, though 'wheat bread' is not present in B. On the contrary, Definition 2 indicates that some items in B are not present in A if there exists *MaxPartof*(A,B) between A and B. Especially, the ancestor-descendant relationship is one kind of *Min/MaxPartof* relations.

Definition 3: (Hierarchical Self-Redundancy for Approximate Basis): Let R=X=>Y be a hierarchical approximate rule, R is considered as a hierarchically self-redundant rule if (1) two items in the ancestor itemsets X or (2) two items in consequent itemsets Y or (3) two items in X and Y have the relation of *MinPartof* /*MaxPartof*.

For example, '2% milk=>milk' is a self-redundant rule because the ancestor itemset and the consequent itemset have the relation of *MinPartof*, the rule does not convey any new information.

Definition 4: (Hierarchical Inter-Redundancy for Approximate Basis): Let R_1=X_1=>Y_1 with confidence C_1 and R_2=X_2=>Y_2 with the confidence C_2 be two approximate association rules. R_1 is redundant to R_2 if (1) the ancestor itemsets X_1 and X_2 satisfy *MinPartof*(X_1,X_2) or (2) the consequent itemsets Y_1 and Y_2 satisfy *MaxPartof*(Y_1,Y_2) and (3) the value of C_1 is less than or equal to the value of C_2.

Based on Definition 4, the rules R_1 is more restrictive than R_2, actually the information in R_1 is part of the information contained in R_2, having R_1 does not bring any new useful information to the users, thus R_1 is considered redundant to R_2. The exceptions to this would be if the confidence of R_1 is higher than R_2, the information in R_1 is more accurate than the information contained in R_2, though R_1 is more restrictive than R_2. We have more confidence in that R_1 is more correct than R_2, in this case, R_1 is not considered redundant.

Besides, if there are *MinPartof*(X_1,X_2) and *MaxPartof*(Y_2,Y_1) in the ancestors and consequents respectively at the same time, then in this case, both of the two rules are not redundant to each other.

Definition 4 is similar to Gavin Shaw's definition of Hierarchical Redundancy for Approximate Basis in [9,10]. However, Gavin Shaw's definition requires that a redundant rule and its corresponding non-redundant rule must have the same itemsets Y as the consequent, while Definition 4 only requires that the consequent itemsets of two approximate rules satisfy the relation of *MaxPartof*, this also ensures that the non-redundant rule has the maximal consequent. Therefore, more redundant association rules can be eliminated from the approximate association rules. In the Section 4 the experiments will make a further explanation.

3.2. Redundancy Elimination

1) Eliminating Hierarchical Self-Redundany

An adaptation of Apriori to generate level-crossing frequent itemsets is presented in [8]. To find the frequent k-itemsets in level l ($k \geq 2$, $l \geq 2$), a set of candidate k-itemsets should join (k-1)-itemset with itself ($L[l,k\text{-}1] \times L[l,k\text{-}1]$) and also join ($k$-1)-itemset with the 1-itemsets of each level i ($1 \leq i < l$), $C_k = L[l,k\text{-}1] \times (L[1,1] \cup L[2,1] \cup \ldots \cup L[l\text{-}1,1])$, then it indicates that a subset s and the rest itemset ($c \backslash s$) of the itemsets c have the relation of *MinPartof*/*MaxPartof*. Therefore, each generated k-itemsets c ($c \in C_k$) should be checked, if there is a relation of *MinPartof*/*MaxPartof* between s and ($c \backslash s$) then delete c from C_k.

2) Eliminating Hierarchical Inter-Redundancy

Based on the Definition 4, the approach for eliminating hierarchical Inter-Redundancy and extracting condensed Non-Redundant approximate basis rules is illustrated below.

Algorithm 1: **Condensed Reliable Approximate Basis with HRR**
Input: Set of frequent closed itemsets C and generators G
Output: Set of non-redundant approximate basis rules ApproxBasis

```
1.  ApproxiBasis ← $\Phi$
2.  for each $c \in C$
3.    for each $g \in G$ such that $\gamma(g) \subset c$
4.      nonRedundant = true
5.      if(conf(g=>(c\g)) ≥ minconf)
6.        if for all $c' \in C$ & for all $g' \in G$ such that $r(g') \subset c'$ &
           $g' \subseteq g$&(we have $\neg(g \supseteq ((c \backslash c') \cup g'))$ or
           $conf(g \Rightarrow (c \backslash g)) > conf(g' \Rightarrow (c' \backslash g'))$)
7.          for all $g_1 \in G, c_1 \in C$ where $g_1 \neq g$ do
8.            if (MinPartof($g$,$g_1$) or MaxPartof(($c \backslash g$),($c_1 \backslash g_1$)))
9.              if($conf(g \Rightarrow (c \backslash g)) \leq conf(g_1 \Rightarrow (c_1 \backslash g_1))$)
10.               nonRedundant = false
11.               break for loops at line 7
12.     else
13.       nonRedundant = false
14.     if(noRedundant)
15.       insert(r: g=>(c\g),conf(r)) into ApproxiBasis
16. return ApproxiBasis
```

Firstly, the algorithm generates an association rules by using the generators of the frequent closed itemsets, and then checks the rule with others, if the rule is not redundant to other rules, then it is added to the approximate basis set. Step 6 is a filter for each rule based on the definition of Reliable Approximate Basis proposed in [5], which ensures the non-redundant association rules having minimal antecedent and maximal consequent. To generate a rule, the algorithm needs to scan all generators to determine whether the rule is hierarchically redundant. Therefore, the complexity of the algorithm is $O(n^2)$, where n is the number of generators derived from the frequent itemsets.

4. Experiments and Result

The use of frequent closed itemsets has been successfully used in association rules mining [1,2,3]. This paper also makes use of this technique to extract level-crossing rules from multi-level datasets. Firstly, we implement the algorithm presented in [8] for finding the frequent level-crossing itemsets and the removal of the hierarchical Self-Redundancy, and then derive the frequent closed itemsets and generators from the frequent itemsets through using of CLOSE+ algorithm proposed in [4]. Finally, based on the frequent closed itemsets and generators, we extract level-crossing approximate association rules and eliminate the hierarchical Inter-Redundancy described in Section 3. The rules presented in Table II are the approximate association rules extracted from Table I when the minimum support is set to 4 for level 1 and 3 for levels 2 and 3 to discover the frequent itemsets, and the minimum confidence threshold is set to 0.5.

TABLE II. APPROXIMATE BASIS ASSOCIATION RULES EXTRACTED FROM TABLE I

No	Rules	Conf
1	[1-*-*]=>[2-*-*]	0.714
2	[1-*-*]=>[2-2-*]	0.571
3	[1-1-*,2-*-*]=>[1-2-*]	0.75
4	[1-1-*]=>[1-2-*,2-*-*]	0.5
5	[1-1-*]=>[1-2-*,2-2-*]	0.5
6	[1-1-*]=>[1-2-*,2-2-1]	0.5
7	[1-1-*]=>[1-2-*]	0.667
8	[1-1-*]=>[2-*-*]	0.667
9	[1-1-*]=>[2-1-*,2-2-*]	0.5
10	[1-1-*]=>[2-1-1,2-2-*]	0.5
11	[1-1-*]=>[2-2-*]	0.667
12	[1-1-1]=>[1-2-*]	0.75
13	[1-1-1]=>[2-*-*]	0.75
14	[1-1-1]=>[2-1-*,2-2-*]	0.75
15	[1-1-1]=>[2-1-1,2-2-*]	0.75
16	[1-2-*]=>[1-1-*,2-*-*]	0.75
17	[1-2-*]=>[1-1-*,2-2-*]	0.75
18	[1-2-*]=>[1-1-*,2-2-1]	0.75
19	[2-2-*]=>[1-*-*,2-1-*]	0.75
20	[2-2-*]=>[1-*-*,2-1-1]	0.75
21	[2-2-*]=>[1-1-*,1-2-*]	0.75
22	[2-2-*]=>[1-1-*,2-1-*]	0.75
23	[2-2-*]=>[1-1-*,2-1-1]	0.75

Table III shows the result of redundant rules and the corresponding non-redundant rules. The number of extracted rules is reduced by over 50 percent by using our approach. We also notice that , based on the Definition 4, rules 3 is not redundant to rules 7, as rules 3 is more correct and has more confidence than rules 7. Though the ancestors in rules 9 and rules 1 have the relation of *MinPartof*, rules 9 is not redundant to rules 1, because the consequent in rules 9 brings more information than rules 1. When 2-*-* is occurrence, it is difficult to say that 2-1-* and 2-2-* are co-occurrence, thus rules 9 is kept in the approximate basis rule set.

TABLE III. RESULT OF REDUNDANT RULES AND THE CORRESPONDING NON-REDUNDANT RULES

Non-Redundancy Rules	Redundancy Rules
1	2,8,11
4	5,6
9	10
14	15
16	17,18
19	20,22,23

We implement the algorithms Min-Max Approximate Basis with HRR (MMA with HRR) and Reliable Approximate Basis with HRR (RAB with HRR) presented in [10] to compare with our approach (CRAB with HRR) on the size of generated approximate basis rule set. We also use the same dataset used by Han & Fu [6,7] which has seven transactions and is named H to test these approaches. The results of experiments on H and Table I (T) are presented in Table IV.

TABLE IV. THE SIZE OF APPROXIMATE BASIS GENERATED BY EACH ALGORITHM

Data set	Approximate Basis				
	MMA	MMA with HRR	RAB	RAB with HRR	CRAB with HRR
T	23	23	23	22	11
H	36	27	35	26	11

5. Conclusion And Future Work

Lots of redundant rules are generated during association rule mining, which make too difficulty to use the result of data mining Aiming at the level-crossing redundant association rules, this paper proposes two kinds of redundancies: hierarchical Self-Redundancy and Inter-Redundancy through the use of the dataset's hierarchy or taxonomy, and gives an approach to eliminate these redundancies. The experiments show that our method effectively reduces the number of level-crossing approximate basis rule set.

This paper focuses on eliminating the hierarchical redundancy in multi-level datasets and improving the quality of level-crossing approximate association rules. The proposed algorithm is limited to small datasets due to efficiency problems of finding the frequent closed itemsets. Besides, for large datasets, with the $O(n^2)$ complexity, the method may have efficiency problems. These issues will be studied in our future work.

6. References

[1] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. "Efficient mining of association rultes using closed itemset lattices". Information Systems,24(1):25–46, 1999.

[2] M. J. Zaki. "Generating non-redundent association rules". In Proceedings of the KDD Conference, pages 34–43, 2000.

[3] M. J. Zaki. "Mining non-redundant association rules". Data Mining and Knowledge Discovery, 9:223–248, 2004.

[4] N. Pasquier, R. Taouil, Y. Bastide, G. Stumme, and L. Lakhal. "Generating a condensed representation for association rules". Journal of Intelligent Information Systems, 24(1):29–60, 2005.

[5] Yue Xu, Yuefeng Li, Gavin Shaw. "A Reliable Basis for Approximate Association Rules". IEEE Intelligent Informatics Bulletin. November 2008 Vol.9 No.1:25-31.

[6] J. Han & Y. Fu, "Discovery of Multiple-Level Association Rules from Large Databases", in 21st International Conference on Very Large Databases, Zurich, Switzerland, 1995, pp. 420-431.

[7] J. Han & Y. Fu, "Mining Multiple-Level Association Rules in Large Databases", IEEE Transactions on Knowledge and Data Engineering, Vol. 11, pp. 798-805, Sep/Oct., 1999.

[8] R. S. Thakur, R. C. Jain & K. P. Pardasani, "Mining Level-Crossing Association Rules from Large Databases", Journal of Computer Science, Vol. 12, pp. 76-81, 2006.

[9] Gavin Shaw, Yue Xu & Shlomo Geva. "Utilizing Non-Redundant Association Rules from Multi-Level Datasets". IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology.2008:681-684.

[10] Gavin Shaw, Yue Xu & Shlomo Geva. "Extracting Non-Redundant Approximate Rules from Multi-Level Datasets". Tools with Artificial Intelligence, 2008. ICTAI '08. 20th IEEE International Conference on:333 – 340.