

Generating Non-redundant Multilevel Association Rules Using Min-max Exact Rules

R. Vijaya Prakash¹, SSVN. Sarma², M. Sheshikala³

¹Department of Computer Science and Engineering, S R Engineering College, India

²Department of Computer Science and Engineering, Vaagdevi College of Engineering, India

³Department of Computer Science and Engineering, S R Engineering College, India

Article Info

Article history:

Received Sep 22, 2017

Revised Jul 13, 2018

Accepted Jul 29, 2018

Keyword:

Association Rule

Frequent Itemsets

Non-Redundant Rules

Reliable Rules

ABSTRACT

Association Rule mining plays an important role in the discovery of knowledge and information. Association Rule mining discovers huge number of rules for any dataset for different support and confidence values, among this many of them are redundant, especially in the case of multi-level datasets. Mining non-redundant Association Rules in multi-level dataset is a big concern in field of Data mining. In this paper, we present a definition for redundancy and a concise representation called Reliable Exact basis for representing non-redundant Association Rules from multi-level datasets. The given non-redundant Association Rules are loss less representation for any datasets.

*Copyright © 2018 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

R. Vijaya Prakash,

Department of Computer Science and Engineering,

S R Engineering College,

Anathasagar, Hasanparthy, Warangal, Telangana, India.

Email: vijprak@hotmail.com

1. INTRODUCTION

The huge amount of the extracted rules is a big problem for Association Rule mining [21]. Especially, many of the extracted rules are considered redundant since they produce the same meaning to the user or extracted rules can be replaced by other rules. Many efforts have been made on reducing the size of the extracted rule set. There are number of representations of frequent patterns have been proposed, one of them, is the closed itemsets, is of particular interest as they can be applied for generating non-redundant rules [10], [12], [18], [23]. The use of frequent closed itemsets presents a clear promise to reduce the number of extracted rules [13], [17], [19]. Multi-level datasets in which the items are not all at the same concept level contain information at different abstract levels.

The approaches used to find frequent itemsets in single level datasets miss information, as they only look at one level in the dataset. Thus techniques that consider all the levels are needed [6]-[9], [22]. However, rules derived from multi-level datasets can have the same issues with redundancy as those from a single level dataset. While approaches used to remove redundancy in single level datasets [13], [17], [19] can be adapted for use in one rule at a given level gives the same information as another rule at a different level. In this paper, we present a Reliable basis representation of non-redundant Association Rules. We then look into this hierarchical redundancy and propose an approach from which more non-redundant rules can be derived. We use the same definition of non-redundant rules in single level datasets, but to this definition we add a requirement that considers the different levels of the item(s) in determining the redundancy rule. By doing so, more redundant Association Rules can be eliminated. We also show that it is possible to derive all of the Association Rules, without lose of information.

The paper is organized as follows. Section 2 briefly discusses some related work, Section 3, we discuss the redundancy in Association Rules and present a definition to redundant rules. Experiments and results are presented in Section 4. Finally, Section 5 concludes the paper.

2. RELATED WORK

The approaches proposed in [13], [18] make use of the closure of the Galois connection [4] to extract non-redundant rules from frequent closed itemsets instead of from frequent itemsets. One difference between the two approaches is the definition of redundancy. The approach proposed in [18] extracts the rules with shorter antecedent and shorter consequent as well among rules which have the same confidence, while the method proposed in [13] defines that the non-redundant rules are those which have minimal antecedents and maximal consequents.

The definition proposed in [17] is like that of [13]. However, the requirement to redundancy is relaxed, and the lesser requirement makes more rules to be considered redundant and thus eliminated. Most importantly, [17] proved that the elimination of such redundant rules increases the belief to the extracted rules and the capacity of the extracted non-redundant rules for solving problems. However, the work mentioned above has only focused on datasets where all items are at the same concept level. Thus, they do not need to consider redundancy that can occur when there is a hierarchy among the items. A multi-level dataset is the one which has an implicit taxonomy or concept tree, like shown in Figure 1. The items in the dataset exist at the lowest concept level but are part of ahierarchical structure and organization.

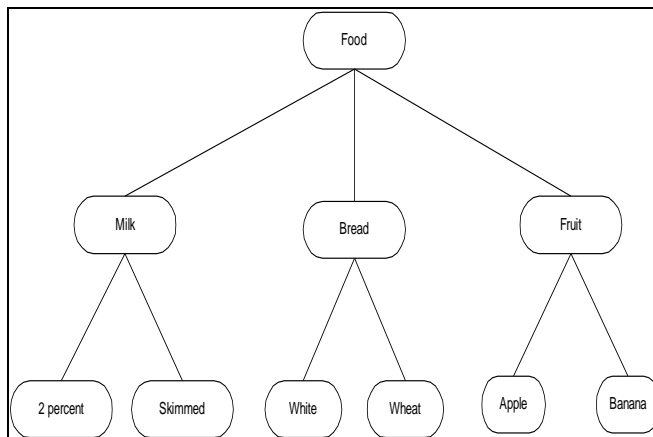


Figure 1. A Simple example of product taxonomy

In Figure 1 for an example, the frequent itemset {'Dairyland-2%-milk', 'white-bread'} is a cross-level itemset as the first item is from the lowest level, while the second item is from a different concept level. In fact the cross-level idea was an addition to the work being proposed. Further work proposed an approach which included finding cross-level frequent itemsets [15]. This later work also performs more pruning of the dataset to make finding the frequent itemsets more efficient. However, even with all this work the focus has been on finding the frequent itemsets as efficiently as possible and the issue of quality and/or redundancy in single level datasets. Some brief work presented by [5]-[6] discusses removing rules which are hierarchically redundant, but it relies on the user giving an expected confidence variation margin to determine redundancy. There appears to be a void in dealing with hierarchical redundancy in Association Rules derived from multi-level datasets. This work attempts to fill that void and show an approach to deal with hierarchical redundancy without losing any information.

From the beginning of Association Rule mining in [1], [3], [21], [23], [25], the first step has always been to find the frequent patterns or itemsets. The simplest way to do this is through the use of the Apriori algorithm [2]. However, Apriori is not designed to work on extracting frequent itemsets at multiple levels in a multi-level dataset. It is designed for use on single level datasets. But, it has been adapted for multi-level datasets.

One adaptation of Apriori to multi-level datasets is the ML_T2L1 algorithm [5]-[6]. The ML_T2L1 algorithm uses a transaction table that has the hierarchy information encoded into it. Each level in the dataset is processed individually. Firstly, level 1 analyzed for large 1-itemsets using Apriori. The list of level 1 large

1-itemsets is then used to filter and prune the transaction dataset of any item that does not have an ancestor in the level 1 large 1-itemset list and remove any transaction which has no frequent items. From the level 1 large 1-itemset list, level 1 large 2-itemsets are derived. Then level 1 large 3-itemsets are derived and so on, until there are no more frequent itemsets to discover at level 1. Since ML_T2L1 defines that only the items that are descendant from frequent items at level 1 can be frequent themselves, the level 2 itemsets are derived from the filtered transaction table. For level 2, the large 1-itemsets are discovered, from which the large 2-itemsets are derived and then large 3-itemsets etc. After all the frequent itemsets are discovered at level 2, the level 3 large 1-itemsets are discovered and so on. ML_T2L1 repeats until either all levels are searched using Apriori or no large 1-itemsets are found at a level.

As the original work shows [5]-[6], ML_T2L1 does not find cross-level frequent itemsets. We have added the ability for it to do this. At each level below 1, when large 2-itemsets or later are derived the Apriori algorithm is not restricted to just using the large n-1-itemsets at the current level, but can generate combinations using the large itemsets from higher levels. The only restrictions on this are that the derived frequent itemset(s) can not contain an item that has an ancestor-descendant relationship with another item within the same itemset and that the minimum support threshold used is that of the current level being processed.

3. GENERATION OF NON-REDUNDANT MULTI-LEVEL ASSOCIATION RULES

The use of frequent itemsets as the basis for Association Rule mining often results in the generation of many rules. More recent work has demonstrated that the use of closed itemsets and generators can reduce the number of rules generated [14],[16]-[18],[20]-[22]. Despite this, redundancy still exists in the rules generated from multi-level datasets even when using some of the methods designed to remove redundancy. This redundancy we call hierarchical redundancy. Here in this section we first introduce hierarchical redundancy in multi-level datasets and then we detail our work to remove this redundancy without losing information.

3.1. Hierarchical Redundancy

Whether a rule is interesting and/or useful is usually determined through the support and confidence values that it has. However, this does not guarantee that all of the rules that have a high enough support and confidence actually convey new information. Following is an example transaction table for a multi-level dataset Table 1.

Table 1. Simple Multi-level Transaction Dataset.

Transaction ID	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]

This simple multi-level transactional dataset has 3 levels with each item belonging to the lowest level. The item ID in the table store/holds the hierarchy information for each item. Thus, the item 1-2-1 belongs to the first category at level 1 and for level 2 it belongs to the second sub-category of the first level 1 category. Finally, at level 3 it belongs to the first subcategory of the parent category at level 2. From this transaction set we use the ML_T2L1 algorithm with the cross-level add-on and a minimum support value of 4 for level 1 and 3 for levels 2 and 3. From these frequent itemsets, the closed itemsets and generators are derived Table 2. The itemsets, closed itemsets and generators come from all three levels.

Finally, from the closed itemsets and generators the Association Rules can be generated. In this example, we use the ReliableExactRule approach presented in [17]-[18] to generate the exact basis rules. The discovered rules are from multiple levels and include cross-level rules. The ReliableExactRule approach can remove redundant rules, but as we will show, it does not remove hierarchy redundancy. The rules given in Table 3 are derived from the closed itemsets and generators in Table 3 when the minimum confidence threshold is set to 0.5 or 50% as shown in Table 3.

The ReliableExactRule algorithm lists all the rules in Table 3 as important and non-redundant. However, we argue that there are still redundant rules. This type of redundancy is beyond what the

ReliableExactRule algorithm was designed for. Looking at the rules in Table 3 we claim that rule 4 is redundant to rule 1, rule 7 is redundant to rule 5, rule 8 is redundant to rule 6 and rule 12 is redundant to rule 10. For example, the item 2-2-1 (from rule 4) is a child of the more general/abstract item 2-2-* (from rule 1). Thus rule 4 is in fact a more specific version of rule 1. Because we know that rule 1 says 2-2-* is enough to fire the rule with consequent C, whereas rule 4 requires 2-2-1 to fire with consequent C, any item that is a descendant of 2-2-* will cause a rule to fire with consequent C. It does not have to be 2-2-1. Thus rule 4 is more restrictive. Because 2-2-1 is part of 2-2-* having rule 4 does not actually bring any new information to the user, as the information contained in it is actually part of the information contained in rule 1. Thus rule 4 is redundant. We define hierarchical redundancy in exact Association Rules through the following definition

Table 2. Frequent Closed Itemsets and Generators Derived from the Frequent Itemsets in Table 1

Closed Itemsets	Generators
[1-*-*]	[1-*-*]
[1-1-*]	[1-1-*]
[1-1-1]	[1-1-1]
[1-*-, 2-2-*]	[2-2-*]
[2-*-, 1-1-*]	[2-*-, 1-1-*]
[1-1-*, 1-2-*]	[1-2-*]
[1-1-*, 2-2-*]	[2-2-*]
[1-*-, 2-2-1]	[2-2-1]
[2-*-, 1-1-1]	[2-*-, 1-1-1]
[1-2-*, 1-1-1]	[1-2-*, 1-1-1]
[1-*-, 2-1-*, 2-2-*]	[2-1-*]
[2-*, 1-1-*, 1-2-*]	[2-*-, 1-2-*]
[1-1-*, 1-2-*, 2-2-*]	[1-2-*, 2-2-*]
[1-1-*, 2-1-*, 2-2-*]	[2-1-*]
[1-*-, 2-1-1, 2-2-*]	[2-1-1]
[1-1-*, 2-1-1, 2-2-*]	[2-1-1]
[1-1-*, 2-2-1, 1-2-*]	[2-2-1]
[2-1-*, 1-1-1, 2-2-*]	[2-1-*, 1-1-1]
[2-2-*, 1-1-1, 2-1-1]	[2-1-1] [2-2-*, 1-1-1]

Table 3. Exact basis Association Rules Derived from Closed Itemsets and Generators in Table 2

No.	Rule	Supp
1	[2-2-*] ==> [1-*-*]	0.571
2	[1-2-*] ==> [1-1-*]	0.571
3	[2-2-*] ==> [1-1-*]	0.571
4	[2-2-1] ==> [1-*-*]	0.428
5	[2-1-*] ==> [1-*-, 2-2-*]	0.428
6	[2-1-*] ==> [1-1-*, 2-2-*]	0.428
7	[2-1-1] ==> [1-*-, 2-2-*]	0.428
8	[2-1-1] ==> [1-1-*, 2-2-*]	0.428
9	[2-2-1] ==> [1-1-*, 1-2-*]	0.428
10	[2-1-*] ==> [1-1-1, 2-2-*]	0.428
11	[2-2-*, 1-1-1] ==> [2-1-*]	0.428
12	[2-1-1] ==> [2-2-*, 1-1-1]	0.428
13	[2-2-*, 1-1-1] ==> [2-1-1]	0.428

Definition 1: Let $R1 = X1 \Rightarrow Y$ and $R2 = X2 \Rightarrow Y$ be two exact Association Rules, with exactly the same itemset Y as the consequent. Rule R1 is redundant to rule R2 if (1) the itemset X1 is made up of items where at least one item in X1 is descendant from the items in X2 and (2) the itemset X2 is entirely made up of items where at least one item in X2 is an ancestor of the items in X1 and (3) the other non-ancestor items in X2 are all present in itemset X1.

From this definition, if for an exact Association Rule $X1 \Rightarrow Y1$ there does not exist any other rule $X2 \Rightarrow Y2$ such that at least one item in X1 shares an ancestor-descendant relationship with X2 containing the ancestor(s) and all other items X2 are present in X1, then $X1 \Rightarrow Y1$ is a non-redundant rule. To test for redundancy, we take this definition and add another condition for a rule to be considered valid. A rule $X \Rightarrow Y$ is valid if it has no ancestor-descendant relationship between any items in itemsets X and Y. Thus, for example $1-2-1 \Rightarrow 1-2-*$ is not a valid rule, but $1-2-1 \Rightarrow 1-1-3$ is a valid rule. If this condition is not met by any rule $X2 \Rightarrow Y2$ when testing to see if $X1 \Rightarrow Y1$ is redundant to $X2 \Rightarrow Y2$, then $X1 \Rightarrow Y1$ is a non-redundant rule as $X2 \Rightarrow Y2$ is not a valid rule. Submit your manuscript electronically for review.

3.2. Generating Exact Basis Rules

As previous work has shown [14, 17-18] using frequent closed itemsets in the generation of Association Rules can reduce the quantity of discovered rules. Because we wish to remove redundancy on top of the redundancy already being removed, our approach uses the closed itemsets and generators to discover the non-redundant rules. [14, 17-18] have both proposed condensed/more concise bases to represent non-redundant exact rules. Exact rules refer to rules whose confidence is 1. The proposed approach will be extended to other rules (i.e., so called approximate rules). The following definitions outline these two bases:

Definition 2: For the Min-MaxExact (MME) basis, C is the set of the discovered frequent closed itemsets. For each closed itemset c in C , G_c is the set of generators for c . From this the exact basis for min-max is: $MMEHR = \{g \rightarrow c \mid c \in C, g \in G_c, g \neq c \text{ and there exists no rules } g' \rightarrow c' \text{ where } c' \in C, g' \in G_c, c \neq c', g' \neq c' \text{ and } g \text{ is descendant set of } g', g' \text{ has no ancestor or descendant of } c' \text{ or } g'\}$

Definition 8: (Reliable Exact Basis without Hierarchy Redundancy) Let C be the set of frequent closed itemsets. For each frequent closed itemset c , let G_c be the set of minimal generators of c . The Reliable exact basis is:

$REHR = \{g \rightarrow c \mid c \in C, g \in G_c, REHR = \{g \rightarrow c \mid c \in C, g \in G_c, \neg (c \text{ or } c' \cup g'), \text{ where } c' \in C, c' \subset c, g' \in G_c, \text{ and there exists no rules } g' \rightarrow c' \text{ where } c \neq c', g' \neq c' \text{ and } g \text{ is descendant set of } g', g' \text{ has no ancestor or descendant of } c' \text{ or } g'\}$. Thus the algorithms to extract non-redundant multi-level rules using either MinmaxExactHR or ReliableExactHR are given as follows:

Algorithm 1: MinmaxExactHR()

Input: C : a set of frequent closed itemsets G : a set of minimal generators. For $g \in G$, g .closure is the closed itemset of g .

Output: A set of non-redundant multilevel rules.

1. MinMaxExact: = ϕ
2. for each $k=1$ to v
3. for each k -generator $g \in G_k$
4. nonRedundant = true
5. if $g \neq g$ closure
6. for all $g' \in G$
7. if ($g' \neq g$)
8. if (g' is ancestor set of g) and ($(c' = c)$ or ($g = g'$)) and (g' is not ancestor set of c')
9. then nonRedundant = false
10. break
14. if nonRedundant = true
15. insert $\{(g \rightarrow c), g, \text{supp}\}$ in MinMaxExact
20. return MinMaxExact

Algorithm 2: ReliableExactHR ()

Input: C : a set of frequent closed itemsets G : a set of minimal generators. For $g \in G$, g .closure is the closed itemset of g .

Output: A set of non-redundant multilevel rules.

1. ReliableExact: = ϕ
2. for all $c \in C$
3. for all $g \in G_c$
4. nonRedundant = false
5. if $\forall c \in C$ such that $c' \subset c$ and $c, g' \in G_c$, we have $(c \text{ or } c') \cup g' \subseteq g$
6. then nonRedundant = true
7. else
8. nonRedundant = false
9. break
11. for all $g' \in G$
12. if $g' \neq g$
13. if (g' is ancestor set of g) and ($c' = c$ or $g' = g$) and (g' is not ancestor set of (c' or g') and (g' is not descendant set of (c' or g'))
14. then nonRedundant = true; break
19. if nonRedundant = true
20. insert $\{(g \rightarrow c \text{ or } g, g, \text{supp})\}$ in ReliableExact
24. return ReliableExact

3.3. Deriving Exact Rules from the Exact Basis Rules

The Min-MaxExact approach and ReliableExact approach have proven that they can deduce all of the exact rules from their basis set [17]. Comparing with the Min-MaxExact approach and ReliableExact approach, our work results in a smaller exact basis set by not only removing the redundant rules that are removed by the Min-MaxExact approach and ReliableExact approach, but also removing the hierarchically redundant rules. If we can recover all the hierarchically redundant rules, then we can derive all the exact rules by using the Min-MaxExact or ReliableExact recovery algorithm. This will ensure that all the exact rules can still be derived and by achieving this, our approach will be a lossless representation of the exact Association Rules.

The following algorithm is designed to recover the hierarchically redundant rules from the exact basis. By adding it to the algorithms used by Min-MaxExact and ReliableExact to derive the exact rules it is then able for the existing ReliableExact recovery algorithm to derive all the exact rules. This is because our algorithm will give them a basis set that includes the hierarchically redundant rules (which the ReliableExact approach would not have removed in the first place). The basic idea is that, for each exact basis rule, first from generators to construct all possible exact basis rules whose antecedent is a descendant of the exact basis rule (steps 4 to 7 in Algorithm 3). These rules are potential exact basis rules that might have been eliminated due to the ancestor-descendant relationship. Then check to make sure these potential rules are valid (steps 8 to 12), finally, from the potential exact rules to find exact basis rules. These exact basis rules have been eliminated due to the ancestor-descendant relationship (steps 13 to 18).

Algorithm 3: DeriveExactHR ()

Input: Set of exact basis rules denoted as *Exactbasis*, set of frequent closed itemsets *C* and generators *G*.

Output: Set of rules that covers the exact basis and the hierarchically redundant rules.

1. Recovered: = ϕ
2. $\forall r \in \text{Exact basis}$
3. CandidateBasis: = ϕ
4. for all generator *g* in *G*
5. if any of the item *x* in the antecedent *X* of rule *r*: $X \rightarrow Y$ is the ancestor of *g*.
6. then add all the possible subsets of *g* into *S*
8. for all *s* in *S*, check every, $x \in X$ if *x* doesn't have a descendant in *s*, add *x* to *s* to make *s* a descendant set of *X*
9. if *s* has no ancestors in *Y* and *s* has no descendants in *Y* and for all items $i \in s$ there are no ancestor-descendant relations with item $i' \in s$ and for all item $i \in Y$ there are no ancestor-descendant relation with item $i' \in Y$
10. then insert $s \rightarrow Y$ in CandidateBasis
13. for all $B \rightarrow D \in \text{CandidateBasis}$
14. if $B \cup D = \text{itemset } i \in C$ and $B = g \in G_i$
15. insert $\{B \rightarrow D, g, \text{supp}\}$ in Recovered
19. return Exactbasis \cup Recovered

4. EXPERIMENTS

Experiments were conducted to test and evaluate the effectiveness of the proposed hierarchically non-redundant exact basis and to confirm that it is also a lossless basis set. This section presents and details the experiments and their results.

4.1. Datasets

We used 6 datasets to test our approach to discover whether it reduced the size of the exact basis rule set and to test that the basis set was lossless, meaning all the rules could be recovered. These datasets were composed of 100, 200, 500, 2000 and 5000 transactions and are named A to F respectively. The key statistics for these built datasets are detailed in Table 4.

Table 4. Obtained Frequent itemsets using Exact Basis

Dataset	MME	MMEHR	RE	REHR
A	15	10	13	9
B	106	68	80	58
C	174	134	113	89
D	577	429	383	305
E	450	405	315	287
F	725	602	91	80

Where as the proposed algorithms MME reduced less number of rules compared to MMEHR. The MMEHR algorithms considers the cross level hierarchy and contains more data, at this level it reduced the more number of rules, it indicated that at hierarchy the data is duplicated and the proposed algorithms is reduced those replicated dat and produced more reliable and accurate rules.

Similarly, for RE and REHR algorithms has generated the different rules at with and without cross levels hierarchy. At cross level there will be more data so more rules are generated by REHR algorithm than the RE algorithm where as the ML_T2L1 algorithm has generated th same rules with and without cross level hierarchy. Thus the proposed work has generated more reliable and accurate algorithm than the ML_T2L1 algorithm.

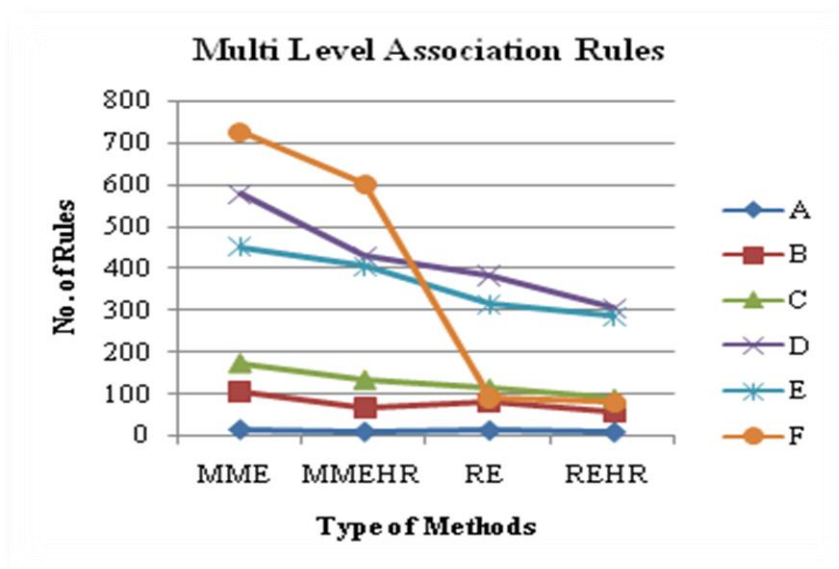


Figure 2. Multi Level Association Rules

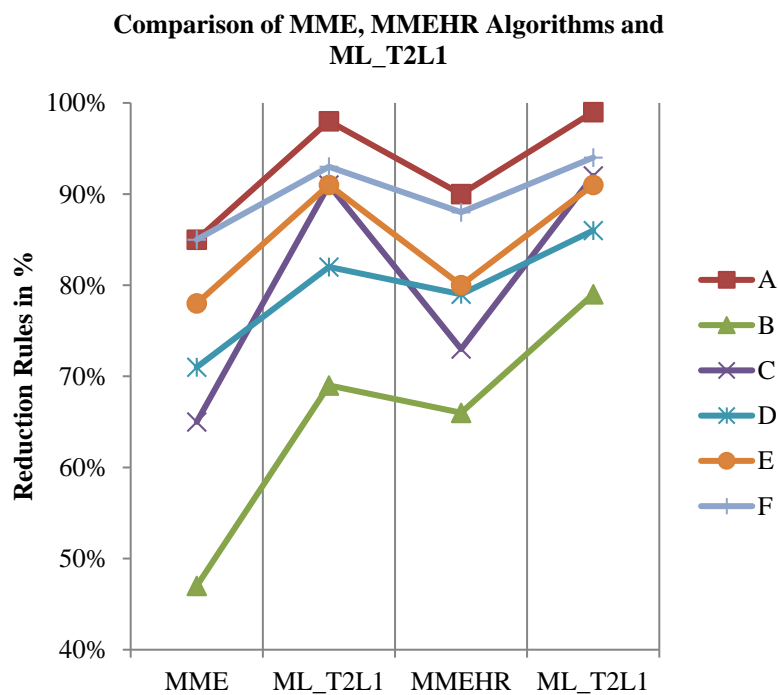


Figure 3. Comparison of MME, MMEHR Algorithms with ML_T2L1

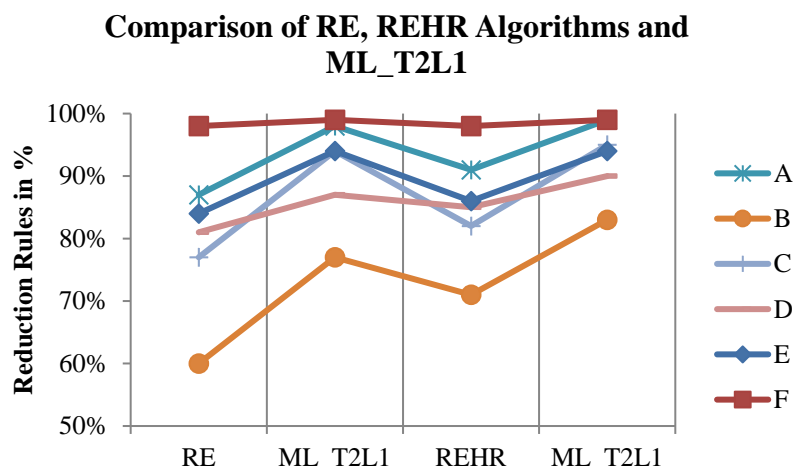


Figure 4. Comparison of RE, REHR Algorithms with ML_T2L1

5. CONCLUSION

Redundancy in Association Rules affects the quality of the information presented and this affects and reduces the use of the rule set. The goal of redundancy elimination is to improve the quality, thus allowing them to better solve problems being faced. Our work aims to remove hierarchical redundancy in multi-level datasets, thus reducing the size of the rule set to improve the quality and usefulness, without causing the loss of any information. We have proposed an approach which removes hierarchical redundancy through the use of frequent closed itemsets and generators. This allows it to be added to other approaches which also remove redundant rules, thereby allowing a user to remove as much redundancy as possible. The next step in our work is to apply this approach to the approximate basis rule set to remove redundancy there. We will also review our work to see if there are other hierarchical redundancies in the basis rule sets that should be removed and will investigate what should and can be done to further improve the quality of multi-level Association Rules.

REFERENCES

- [1] Bayardo, R.J., Agrawal, R. & Gunopulos, D. "Constraint-based rule mining in large, dense databases". *Data Mining and Knowledge Discovery*, Vol 4, pp. 217-240, 2000.
- [2] Berry, M.J.A. & Linoff, G.S. "Data Mining Techniques for Marketing Sales and Customer Support". *John Wiley and Sons*, 1997.
- [3] Brin, S., Motwani, R., Ullman, J.D. & Tsur, S. "Dynamic itemset counting and implication rules for market basket data" *Proceedings of the 1997 ACM SIGMOD Conference*, pp. 255-264, 1997.
- [4] Ganter, B. & Wille, R. "Formal Concept Analysis: Mathematical Foundations", *Springer-Verlag*, 1999.
- [5] Han, J. & Fu, Y. "Discovery of multiple-level Association Rules from large databases", *Proceedings of the 21st International Conference on Very Large Databases*, pp. 420-431, 1995.
- [6] Han, J. & Fu, Y. "Mining multiple-level Association Rules in large databases", *IEEE Transactions on Knowledge and Data Engineering*, Vol 11 pp. 798-805, 1999.
- [7] Han, J. & Fu, Y. "Mining multiple-level Association Rules in large databases", *IEEE Transactions on Knowledge and Data Engineering*, Vol 11, pp. 798-804, 2000.
- [8] Hong, T.P., Lin, K.Y. & Chien, B.C. "Mining fuzzy multiple-level Association Rules from quantitative data. *Applied Intelligence*", Vol. 18, pp. 79-90, 2003.
- [9] Kaya, M. & Alhajj, R. "Mining multi-cross-level fuzzy weighted Association Rules" *2nd International IEEE Conference on Intelligent Systems*, pp. 225-230, 2004.
- [10] Kryszkiewicz, M., Rybinski, H. & Gajek, M. "Dataless transitions between concise representations of frequent patterns", *Journal of Intelligent Information Systems*, Vol. 22, pp. 41-70, 2004.
- [11] Ng, R.T., Lakshmanan, V., Han, J. & Pang, A. "Exploratory mining and pruning optimizations of constrained Association Rules". *Proceedings of the SIGMOD conference*, pp. 13-24, 1998.
- [12] Pasquier, N., Bastide, Y., Taouil, R. & Lakhal, L. "Efficient mining of association rules using closed itemset lattices". *Journal of Intelligent Information Systems*, Vol. 24, pp. 25-46, 1999.
- [13] Pasquier, N., Taouil, R., Bastide, Y., Stumme, G. & Lakhal, L. "Generating a condensed representation for Association Rules", *Journal of Intelligent Information Systems*, Vol. 24, pp. 29-60, 2005.
- [14] Srikant, R., Vu, Q. & Agrawal, R. "Mining Association Rules with item constraints", *Proceedings of the KDD Conference*, pp. 67-73, 1997.

- [15] Thakur, R.S., Jain, R.C. & Pardasani, K.P. "Mining level-crossing Association Rules from large databases", *Journal of Computer Science*, pp. 76-81, 2006.
- [16] Wille, R. "Restructuring lattices theory: An approach based on hierarchies of concepts Ordered Sets", *Dordrecht-Boston*, 1982.
- [17] Xu, Y. & Li, Y. "Generating concise Association Rules", *Proceedings of the 16th ACM Conference on Conference on Information and Knowledge Management (CIKM07)*, pp. 781-790, 2007.
- [18] Zaki, M.J. "Generating nonredundent Association Rules", *Proceedings of the KDD Conference*, pp. 34-43, 2000.
- [19] Zaki, M.J. "Mining non-redundant Association Rules", *Data Mining and Knowledge Discovery*, Vol. 9, pp. 223-248, 2004.
- [20] Ziegler, C.N., McNee, S.M., Konstan, J.A & Lausen, G. "Improving recommendation lists through topic diversification", *Proceed. of the 14th Inter. World Wide Web Conference*, pp. 22-32, 2005.
- [21] K Rajendra Prasad, "Optimized High-Utility Itemsets Mining for Effective Association Mining Paper", *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 7, No. 5, pp 2911-2918.
- [22] Sushil Kumar Verma, R.S. Thakur, Shailesh Jaloree .Fuzzy Association Rule Mining based Model to Predict Students' Performance, *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 7, No. 4, pp. 2223-2231.
- [23] Harco Leslie Hendric Spits Warnars, "Using Attribute Oriented Induction High Level Emerging Pattern (AOI-HEP) to Mine Frequent Patterns", *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 6, No. 6, pp. 3037 – 3046.
- [24] Benaka Santhosha S, Chitra Kiran N, "A Systematic Review of Existing Data Mining Approaches Envisioned for Knowledge Discovery from Multimedia", *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 8, No. 2, pp. 908-916, 2018.
- [25] Leena Deshpande, M. Narsing Rao, "Concept Drift Identification using Classifier Ensemble Approach", *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 8, No. 1, pp. 19-25, 2018.