J Syst Sci Syst Eng
DOI: 10.1007/s11518-009-5098-x
ISSN: 1004-3756 (Paper) 1861-9576 (Online)
CN11-2983/N

CONCISE REPRESENTATIONS FOR ASSOCIATION RULES IN MULTI-LEVEL DATASETS*

Yue XU[1] Gavin SHAW[2] Yuefeng LI[3]

School of Information Technology, Queensland University of Technology, Brisbane, 4001, Australia

[1] *yue.xu@qut.edu.au* (✉) [2] *gavin.shaw@qut.edu.au* [3] *y2.li@qut.edu.au*

Abstract

Association rule mining plays an important role in knowledge and information discovery. Often for a dataset, a huge number of rules can be extracted, but many of them are redundant, especially in the case of multi-level datasets. Mining non-redundant rules is a promising approach to solve this problem. However, existing work (Pasquier et al. 2005, Xu & Li 2007) is only focused on single level datasets. In this paper, we firstly present a definition for redundancy and a concise representation called Reliable basis for representing non-redundant association rules, then we propose an extension to the previous work that can remove hierarchically redundant rules from multi-level datasets. We also show that the resulting concise representation of non-redundant association rules is lossless since all association rules can be derived from the representation. Experiments show that our extension can effectively generate multilevel non-redundant rules.

Keywords: Association rule mining, redundant association rules, closed itemsets, multi-level datasets

1. Introduction

The huge amount of the extracted rules is a big problem for association rule mining since it severely hinders the effective use of the discovered knowledge. Especially, many of the extracted rules are considered redundant since they produce no value to the user or can be replaced by other rules. Many efforts have been made on reducing the size of the extracted rule set. The approaches can be roughly divided to two categories, subjective approach and objective approach. In the subjective approach category, one technique is to define various interestingness measures and only the rules which are considered interesting based on the interesting measurements are generated (Berry & Linoff 1997, Brin et al. 1997). Another technique in this category is to apply constraints or templates to generate only those rules that satisfy the constraints or templates (Bayardo, Agrawal & Gunopulos 2000, Han & Fu 2000, Ng et al. 1998, Srikant, Vu & Agrawal 1997). In the objective approach category, the main technique is to construct concise representative bases of association rules without using user-dependent constraints. A concise

* Part of the paper was presented in the conference of IEEE SMC 2008.

representative basis contains much smaller number of rules and is considered lossless since all association rules can be derived from the basis. A number of concise representations of frequent patterns have been proposed, one of them, namely the closed itemsets, is of particular interest as they can be applied for generating non-redundant rules (Kryszkiewicz, Rybinski, & Gajek 2004, Pasquier et al. 1999, Zaki 2000). The notion of closed frequent itemset has its origins in the mathematical theory of Formal Concept Analysis (FCA) introduced in the early 80s' (Ganter & Wille 1999, Wille 1982). The use of frequent closed itemsets presents a clear promise to reduce the number of extracted rules and also provides a concise representation of association rules (Pasquier et al. 2005, Xu & Li 2007, Zaki 2004). However, these approaches have only dealt with redundancy in single level datasets. 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 (Han & Fu 1999, Hong, Lin & Chien 2003, Kaya & Alhajj 2004). 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 (Pasquier et al. 2005, Xu & Li 2007, Zaki 2004) can be adapted for use in multi-level datasets, they still fail to remove all of the redundancies, namely the redundancy of hierarchy, where one rule at a given level gives the same information as another rule at a different level.

In this paper, we present a concise representation basis of association rules, called Reliable basis. We then look into this hierarchical redundancy and propose an approach from which more concise non-redundant rules can be derived. We use the same definition for non-redundant rules, in which minimal antecedent and maximal consequents are desired. But to the definition we add a requirement that considers the level of the item(s) in the rule in determining redundancy. By doing so, more redundant association rules can be eliminated. We also show that it is possible to derive all of the association rules from this more concise set of basis rules and thus there is no loss of information in this basis set.

The paper is organized as follows. Section 2 briefly discusses some related work. In Section 3, we discuss the redundancy in association rules and present a definition to redundant rules. In Section 4, we first discuss the algorithms for generating frequent patterns in multilevel datasets and then we present the definition of hierarchical redundancy and introduce our approach for deriving multilevel non-redundant exact basis rules and recovering all the exact rules from the basis rules. Experiments and results are presented in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

The approaches proposed in Pasquier et al. (2005) and Zaki (2000) make use of the closure of the Galois connection (Ganter & Wille 1999) 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 Zaki (2000) extracts the rules with shorter antecedent and shorter consequent as well among rules which have the same confidence, while the method proposed in Pasquier et al. (2005) defines that the non-redundant rules are those which have minimal antecedents and maximal consequents. The definition proposed in Xu & Li (2007) is similar to that of Pasquier et al. (2005). However, the requirement to redundancy is relaxed, and the lesser requirement makes more rules to be considered redundant and thus eliminated. Most importantly, Xu & Li (2007) proved that the elimination of such redundant rules does not reduce the belief to the extracted rules and the capacity of the extracted non-redundant rules for solving problems will also not be reduced. 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 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 a hierarchical structure and organization. Thus for example, 'Old Mills' is an item at the lowest level of the taxonomy but it also belongs to the higher concept category of 'bread' and also the more refined category 'white bread'.

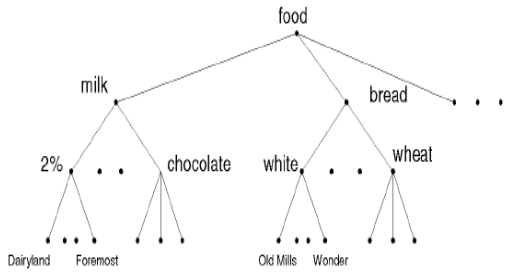

Figure 1 A Simple example of product taxonomy

Because of the hierarchical nature of a multi-level dataset, a new approach to finding frequent itemsets for multi-level datasets has to be considered. Work has been done in adapting approaches originally made for single level datasets into techniques usable on multi-level datasets. The paper in Han & Fu (1995) shows one of the earliest approaches proposed to find frequent itemsets in multi-level datasets and later was revisited in Han & Fu (1999). This work primarily focused on finding frequent itemsets at each of the levels in the dataset and did not focus heavily on cross-level itemsets (those itemsets that are composed of items from two or more different levels). Referring to 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 (Thakur, Jain & Pardasani 2006). 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 Han & Fu (1999) 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.

3. Redundancy in Association Rules

Let $I=\{I_1, I_2, \ldots, I_m\}$ be a set of m distinct items, t be a transaction that contains a set of items such that $t \subseteq I$, T be a database containing different identifiable transactions. An association rule is an implication in the form of $X \Rightarrow Y$, where $X, Y \subset I$ are sets of items called itemsets, and $X \bigcap Y = \varnothing$. Association rule mining is to find out association rules that satisfy the predefined minimum support (denoted as *minsupp*) and confidence (denoted as *mincof*) from a given database. The problem is usually decomposed into two subproblems: to find frequent itemsets and to generate association rules from those frequent itemsets. For the popular used Mushroom dataset (http://kdd.ics.uci.edu/), with minimal support 0.8 and minimal confidence 0.8, we can generate 88 association rules. Table 1 displays 20 of the 88 association rules.

Table 1 Non-redundant exact rules extracted from min-max exact basis
(Mushroom Dataset, minsupp=0.8, minconf=0.8)

Rule No.	Rules (supp, conf)
1	gill-attachment-f ⇒ veil-type-p (0.97415,1.0)
2	veil-color-w ⇒ veil-type-p (0.97538 ,1.0)
3	gill-attachment-f, veil-color-w ⇒ veil-type-p (0.97317,1.0)
4	gill-attachment-f, ring-number-o ⇒ veil-type-p (0.89808,1.0)
5	gill-spacing-c, veil-color-w ⇒ veil-type-p (0.81487,1.0)
6	gill-attachment-f, gill-spacing-c ⇒ veil-type-p, veil-color-w (0.81265,1.0)
7	gill-attachment-f, gill-spacing-c ⇒ veil-type-p (0.81265,1.0)
8	gill-attachment-f, gill-spacing-c, veil-type-p ⇒ veil-color-w (0.81265 ,1.0)
9	gill-attachment-f ⇒ veil-type-p, veil-color-w (0.97317,0.99899)
10	gill-attachment-f ⇒ veil-type-p, ring-number-o (0.89808,0.92191)
11	veil-color-w ⇒ gill-spacing-c, veil-type-p (0.81487,0.83544)
12	veil-color-w ⇒ gill-attachment-f, gill-spacing-c, veil-type-p (0.81265,0.83317)
13	gill-attachment-f, veil-color-w ⇒ gill-spacing-c, veil-type-p (0.81265,0.83506)
14	gill-attachment-f, veil-color-w ⇒ veil-type-p, ring-number-o (0.8971,0.92183)
15	gill-attachment-f, ring-number-o ⇒ veil-type-p, veil-color-w (0.8971,0.9989)
16	gill-spacing-c, veil-color-w ⇒ gill-attachment-f, veil-type-p (0.81265,0.99728)
17	gill-attachment-f ⇒ veil-color-w (0.97317,0.99899)
18	gill-attachment-f ⇒ ring-number-o (0.89808,0.92191)
19	gill-attachment-f, veil-color-w ⇒ gill-spacing-c (0.81265,0.83506)
20	gill-attachment-f, ring-number-o ⇒ veil-color-w (0.8971,0.9989)

The definition of closed itemsets comes from the closure operation of the Galois connection (Ganter& Wille 1999). $\forall\, i \in I$ and $\forall\, t \in T$, if item i appears in transaction t, then i and t has a binary relation δ denoted as $i\delta t$. The Galois connection of the binary relation is defined by the following mappings where $X \subseteq I$, $Y \subseteq T$, 2^I and 2^T are the power set of I and T, respectively:

$$\tau : 2^I \to 2^T, \tau(X) = \{t \in T \,|\, \forall i \in X, i\delta t\} \qquad (1)$$

$$\gamma : 2^T \to 2^I, \gamma(Y) = \{i \in I \,|\, \forall t \in Y, i\delta t\} \qquad (2)$$

$\tau(X)$ is called the transaction mapping of X. $\gamma(Y)$ is called the item mapping of Y. $\gamma \circ \tau(X)$, called the closure of X, gives the common items among the transactions each of which contains X.

Definition 1: (Closed Itemsets) Let X be a subset of I. X is a closed itemset iff $\gamma \circ \tau(X) = X$.

Definition 2: (Generators) An itemset $g \in 2^I$ is a generator of a closed itemset $c \in 2^I$ iff $c = \gamma \circ \tau(g)$ and $g \subset \gamma \circ \tau(g)$. g is said a minimal generator of the closed itemset set c if not $\exists\, g'$ such that $\gamma \circ \tau(g') = c$.

A challenge to association mining is the huge amount of the extracted rules. Recent studies have shown that using closed itemsets and generators to extract association rules can greatly reduce the number of extracted rules (Pasquier et al. 1999, Zaki 2000). However, considerable amount of redundancy still exists in the extracted association rules based on closed itemsets. Therefore, techniques are needed to remove the redundancy for generating high quality association rules. An important issue related to redundancy elimination is the definition of redundancy. The scope of the redundancy must be carefully and fairly defined so that the reduction won't cause information loss or reduce the belief to the resulting rules. Any information loss or belief degradation will cause the quality deterioration of the extracted rules, which makes the redundancy reduction not worthwhile. In this section, we start with some examples to show the existence of redundancy in extracted rules, following that we give a definition of redundant rules to be removed. We have proved that the elimination of the defined redundancy won't reduce the belief to the extracted non-redundant rules (Xu & Li 2007). In Section 4, we describe a concise representation of the defined non-redundant association rules in multi-level datasets and the algorithms to generate those non-redundant multi-level association rules.

3.1 Redundancy in Single Level Datasets

The rules in Table 1 are considered useful based on the predefined minimum support and confidence. However, some of the rules actually do not contribute new information. The consequent concluded by some rules can be obtained from some other rules without requiring more conditions but with higher or the same confidences. For example, in order to be fired the rules 5, 8, 13, and 20 in Table 1 require more conditions than that of rules 2, 6, 11, and 9, respectively, but conclude the same or less results which can be produced by rules 2, 6, 11, and 9. That means, without rules 5, 8, 13, and 20, we still can achieve the same result using other rules. Therefore, rules 5, 8, 13, and 20 are considered redundant to rules 2, 6, 11, and 9, respectively. Comparing to rules 2, 6, 11, and 9, the redundant rules 5, 8, 13, and 20 have a

longer or the same antecedent and a shorter or the same consequent, respectively, and the confidence of the redundant rules is not larger than that of their corresponding non-redundant rules. The following definition defines such kind of redundant rules.

Definition 3: (Redundant rules) Let $X \Rightarrow Y$ and $X' \Rightarrow Y'$ be two association rules with confidence cf and cf', respectively. $X \Rightarrow Y$ is said a redundant rule to $X'' \Rightarrow Y'$ if $X' \subseteq X$, $Y \subseteq Y'$, and $cf \leq cf'$.

Based on Definition 3, for an association rule $X \Rightarrow Y$, if there does not exist any other rule $X' \Rightarrow Y'$ such that the confidence of $X' \Rightarrow Y'$ is the same as or larger than the confidence of $X \Rightarrow Y$, $X' \subseteq X$ or $Y \subseteq Y'$, then $X \Rightarrow Y$ is non-redundant. In terms of the size of antecedent and consequent, Definition 3 is similar to Pasquier's definition of min-max association rules (Pasquier et al. 2005). However, Pasquier's definition requires that a redundant rule and its corresponding non-redundant rule must have identical confidence and identical support, while Definition 3 here only requires that the confidence of the redundant rule is not larger than that of its corresponding non-redundant rules.

3.2 Concise Representations

Developing concise and lossless representations is a promising way to improve the quality of the discovered associations. Pasquier et al. (2005) and Xu & Li (2007) proposed concise bases to represent non-redundant exact rules. Exact rules refer to rules whose confidence is 1, other rules are called Approximate rules. In this paper we focus on the redundancy elimination for multilevel exact rules.

Definition 4: (Min-max Exact Basis) 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 min-max exact basis is:

$$MinMaxExact = \{g \Rightarrow (c \setminus g) \mid c \in C, g \in G_c, g \neq c\}$$

Definition 5: (Reliable Exact Basis) 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:

$$ReliableExact = \{g \Rightarrow (c \setminus g) \mid c \in C, g \in G_c, \neg(g \supseteq ((c \setminus c') \cup g')), \text{where } c' \in C, c' \subset c, g' \in G_{c'}\}$$

Among the 88 rules extracted from the Mushroom dataset mentioned above, there are 17 exact rules. Based on the Minmax exact basis defined in Definition 4 (Pasquier et al. 2005), only 9 exact rules as displayed in Table 2 are extracted and considered non-redundant. However, according to the Reliable Exact basis defined in Definition 5, some of the rules extracted from the Min-max basis are redundant such as rules 5, 6 and 7 in Table 2 which are redundant to rules 1 and 2 in the same table. Using the Reliable Exact basis, only 6 non-redundant exact rules are extracted, rules 5, 6, and 7 in Table 2 are considered redundant and thus eliminated. We have proved the following theorem which can ensure the correctness of the Reliable Exact Basis (Xu & Li 2007).

Theorem 1 *Let $c \in C$ and C be the set of frequent closed itemsets, let $g \in G$ and G be the set of minimal generators of the closed itemsets*

in C, and $\gamma \circ \tau\ (g) \subset c$. $g \Rightarrow c \backslash g$ *is a non-redundant rule iff* $\forall\ c' \ni C$, $\forall\ g' \ni G$, $g' \subset g$, $\gamma \circ \tau\ (g') \subset c'$, *and* $(g \supseteq ((c \backslash c') \cup g'))$ *or conf*$(g \Rightarrow c \backslash g) >$ *conf*$(g' \Rightarrow c' \backslash g')$.

4. Generation of Non-Redundant Multi-level Association Rules

As mentioned above, recent work has demonstrated that the use of closed itemsets and generators can reduce the number of rules generated. This has helped to greatly reduce redundancy in the rules derived from single level datasets. 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. In this section, after a brief introduction to frequent pattern mining in multilevel datasets, we will discuss the hierarchical redundancy in multi-level datasets and then we detail our work to remove this redundancy without losing information.

4.1 Mining Frequent Patterns in Multilevel Datasets

The popular used pattern mining algorithm **Apriori** has been adapted for multi-level datasets. One adaptation of **Apriori** to multi-level datasets is the ML_T2L1 algorithm (Han & Fu 1995, 1999). 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 (the highest level in the hierarchy) is analysed 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 (thus contains only infrequent items when assessed using the level 1 large 1-itemset list). From the level 1 large 1-itemset list, level 1 large 2-itemsets are derived (using the filter dataset). Then level 1 large 3-itemsets are derived and so on, until there are no more frequent itemsets to discover

Table 2 Association rules (Mushroom Dataset, minsupp=0.8, minconf=0.8)

Rule No.	Rules (supp, conf)
1	gill-attachment-f $\Rightarrow$ veil-type-p (0.97415,1.0)
2	gill-spacing-c $\Rightarrow$ veil-type-p (0.8385,1.0)
3	veil-color-w $\Rightarrow$ veil-type-p (0.97538,1.0)
4	ring-number-o $\Rightarrow$ veil-type-p (0.92171,1.0)
5	gill-attachment-f, veil-color-w $\Rightarrow$ veil-type-p (0.97317,1.0)
6	gill-attachment-f, ring-number-o $\Rightarrow$ veil-type-p (0.89808, 1.0)
7	gill-spacing-c, veil-colo-w $\Rightarrow$ veil-type-p (0.81487 ,1.0)
8	gill-attachment-f, gill-spacing-c $\Rightarrow$ veil-type-p, veil-color-w (0.81265,1.0)
9	veil-color-w, ring-number-o $\Rightarrow$ gill-attachment-f, veil-type-p (0.8971, 1.0)

at level 1. Since ML_T2L1 defines that only the items that are descendant from frequent items at level 1 (essentially they must descend from level 1 large 1-itemsets) 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 (from the same filtered dataset) and so on. ML_T2L1 repeats until either all levels are searched using Apriori or no large 1-itemsets are a found at a level.

As the original work shows (Han & Fu 1995, 1999), ML_T2L1 does not find cross-level frequent itemsets. We have added the ability for it to do this. At each level below 1 (so starting at level 2) 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 (which is actually the lowest level in the itemset).

A second, more recent adaptation of Apriori for use in multi-level datasets is a top-down progressive deepening method by Thakur in (Thakur, Jain & Pardasani 2006). This approach was developed to find level-crossing association rules by extending existing multi-level mining techniques and uses reduced support and refinement of the transaction table at every hierarchy level. This algorithm works very similarly to ML_T2L1 presented previously in that it uses a transaction table which has the hierarchy encoded into it and each level is processed individually, one at a time. Initially, level 1 is processed, followed by level 2, 3 and so on until the lowest level is reached and processed, or a level generates no large 1-itemsets. At each level, the large 1-itemsets are first derived and are then used to filter / prune the transaction table (as described for ML_T2L1). This filtering happens at every level, not just level 1, like in ML_T2L1. Then large 2-itemsets, 3-itemsets etc are derived from the filtered table. When it comes to level 2 and lower, the itemsets are not restricted to just the current level, but can include itemsets from large itemset lists of higher levels. This is how level crossing association rules will be found. For the itemsets that span multiple levels, the minimum support threshold of the lowest level in the itemset is used as the threshold to determine whether the itemset is frequent / large. The two algorithms mentioned above have been used to generate frequent itemsets in our experiments.

The simple multi-level dataset as shown in Table 3 is used as an example to demonstrate the algorithms proposed in this paper. This simple multi-level 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 sub-category of the parent category at level 2. From this transaction set we use the ML T2L1 algorithm with the cross level add-on (as

described previously) and a minimum support value of 4 for level 1 and 3 for levels 2 and 3. Table 4 shows the discovered frequent itemsets. From these frequent itemsets the closed itemsets and generators are derived which are given in Table 5. The itemsets, closed itemsets and generators come from all three levels.

Table 3 A 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]
9	

Finally from the closed itemsets and generators the association rules can be generated either using the Min-max basis or Reliable exact basis. The rules given in Table 6 are derived from the closed itemsets and generators in Table 5 when the minimum confidence threshold is set to 0.5. Even though the Min-max Exact Basis and the Reliable Exact Basis approach can remove redundant rules, but as we will show, they do not remove hierarchy redundancy.

4.2 Hierarchical Redundancy

The Reliable Exact basis extracts the rules as important and non-redundant. However, we argue that there are still redundant rules. This type of redundancy is beyond what the Reliable Exact basis was designed for. Looking at the rules in Table 5 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 2-1-1 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

Table 4 Frequent itemsets

1-itemsets	2-items	3-itemsets
[1-*-*]	[1-*-*, 2-*-*]	[1-*-*, 2-1-*, 2-2-*]
[2-*-*]	[1-*-*, 2-1-*]	[2-*-*, 1-1-*, 1-2-*]
[1-1-*]	[1-*-*, 2-2-*]	[1-1-*, 1-2-*, 2-2-*]
[1-2-*]	[2-*-*, 1-1-*]	[1-1-*, 2-1-*, 2-2-*]
[2-1-*]	[2-*-*, 1-2-*]	[1-*-*, 2-1-*, 2-2-*]
[2-2-*]	[1-1-*, 1-2-*]	[1-1-*, 2-1-1, 2-2-*]
[1-1-1]	[1-1-*, 2-1-*]	[1-1-*, 2-2-1, 1-2-*]
[2-1-1]	[1-1-*, 2-2-*]	[2-1-*, 1-1-1, 2-2-*]
[2-2-1]	[1-2-*, 2-2-*]	[2-2-*, 1-1-1, 2-1-1]
	[2-1-*, 2-2-*]	
	[1-*-*, 2-1-1]	
	[1-*-*, 2-2-1]	
	[2-*-*, 1-1-1]	
	[1-1-*, 2-1-1]	
	[1-1-*, 2-2-1]	
	[1-2-*, 1-1-1]	
	[1-2-*, 2-2-1]	
	[2-1-*, 1-1-1]	
	[2-2-*, 1-1-1]	
	[2-2-*, 2-1-1]	
	[1-1-1, 2-1-1]	

Table 5 Frequent closed itemsets and generators derived from the frequent itemsets in table 4

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-*] [2-2-*, 1-1-1]
[2-2-*, 1-1-1, 2-1-1]	[2-1-1] [2-2-*, 1-1-1]

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 association rules through the following definition.

Definition 6: (Hierarchical Redundancy) Let $R_1 = X_1 \Rightarrow Y$ and $R_2 = X_2 \Rightarrow Y$ be two association rules, with exactly the same itemset Y as the consequent. Rule R_1 is redundant to rule R_2 if (1) the itemset X_1 is made up of items where at least one item in X_1 is descendant from the items in X_2 and (2) the itemset X_2 is entirely made up of items where at least one item in X_2 is an ancestor of the items in X_1 and (3) the other non-ancestor items in X_2 are all present in itemset X_1.

From this definition, if for an association rule $X_1 \Rightarrow Y_1$ there does not exist any other rule $X_2 \Rightarrow Y_1$ such that at least one item in X_1 shares an ancestor-descendant relationship with X_2 containing the ancestor(s) and all other items X_2 are present in X_1, then $X_1 \Rightarrow Y_1$ 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 $X_2 \Rightarrow Y_1$ when testing to see if $X_1 \Rightarrow Y_1$ is redundant to $X_2 \Rightarrow Y_1$, then $X_1 \Rightarrow Y_1$ is a non-redundant rule as $X_2 \Rightarrow Y_1$ is not a valid rule.

Table 6 Exact basis rules derived from closed itemsets and generators in table 5

Rule No	Rules	Support
1	[2-2-*] $\Rightarrow$ [1-*-*]	0.571
2	[1-2-*] $\Rightarrow$ [1-1-*]	0.571
3	[2-2-*] $\Rightarrow$ [1-1-*]	0.571
4	[2-2-1] $\Rightarrow$ [1-*-*]	0.428
5	[2-1-*] $\Rightarrow$ [1-*-*, 2-2-*]	0.428
6	[2-1-*] $\Rightarrow$ [1-1-*, 2-2-*]	0.428
7	[2-1-1] $\Rightarrow$ [1-*-*, 2-2-*]	0.428
8	[2-1-1] $\Rightarrow$ [1-1-*, 2-2-*]	0.428
9	[2-2-1] $\Rightarrow$ [1-1-*, 1-2-*]	0.428
10	[2-1-*] $\Rightarrow$ [1-1-1, 2-2-*]	0.428
11	[2-2-*, 1-1-1] $\Rightarrow$ [2-1-*]	0.428
12	[2-1-1] $\Rightarrow$ [2-2-*, 1-1-1]	0.428
13	[2-2-*, 1-1-1] $\Rightarrow$ [2-1-1]	0.428

4.3 Generating Exact Rules in Multilevel Datasets

We extend the Min-max basis and Reliable exact basis by including the condition specified in Definition 6 for generating the non-redundant multi-level association rules. Thus our modified approaches to deriving the exact basis rules are as follows:

Definition 7: (Min-max Basis without Hierarchy Redundancy) Let C be the set of frequent closed itemsets. For each frequent closed itemset c. The Min-max exact basis without hierarchy redundancy is:

$$MinMaxExactHR = \{g \Rightarrow (c \setminus g) \mid c \in C, g \in G_c, g \neq c, \text{ and there exists no rules } g' \Rightarrow (c' \setminus g') \text{ where } c' \in C, g' \in G_c, c \neq c', g' \neq c', \text{ and } (c \setminus g) = (c' \setminus g'), g \text{ is descendant set of } g', \text{ and } g' \text{ has no ancestors or descendants of } (c' \setminus g')\}$$

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:

$$ReliableExactHR = \{g \Rightarrow (c \setminus g) \mid c \in C, g \in G_c, \neg(g \supseteq ((c \setminus c') \cup g')), \text{ where } c' \in C, c' \subset c, g' \in G_c \text{ and there exists no rules } g' \Rightarrow (c' \setminus g') \text{ where } c' \in C, g' \in G_c, c \neq c', g' \neq c', \text{ and } (c \setminus g) = (c' \setminus g'), g \text{ is descendant set of } g', \text{ and } g' \text{ has no ancestors or descendants of } (c' \setminus g')\}$$

In the definitions above, "g is descendant set of g' " means that g contains at least one tem which is a descendant of the items in g' and the rest non-ancestor items in g' are present in g. Similarly, "g' is ancestor set of g " means that g' contains at least one item which is an ancestor of the items in g and the rest non-ancestor items in g' are present in 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 := ∅
2.  for each k=1 to v
3.    for each k-generator g ∈ G_k
4.      nonRedundant = true
5.      if g ≠ g.closure
6.        for all g' ∈ G
7.          if (g' ≠ g)
8.            if ( g' is ancestor set of g ) and
                (c' \ g') = (c \ g) and
                ( g' is not ancestor set of (c' \ g') )
                and
                ( g' is not descendant set of (c' \ g') )
9.              then nonRedundant = false
10.             break
11.           end if
12.         end if
13.       end for
14.       if nonRedundant = true
15.         insert { g ⇒ (c \ g) , g.supp} in
            MinMaxExact
16.       end if
```

17. end if
18. end for
19. end for
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. $ReliableExact := \varnothing$
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 $g' \in G'_c$, we have $\neg(g \supseteq ((c \setminus c') \bigcup g'))$
6. then *nonRedundant = true*
7. else
8. *nonRedundant = false*
9. break
10. end if
11. for all $g' \in G$
12. if $g' \neq g$
13. if (g' is ancestor set of g) and $(c' \setminus g') = (c \setminus g)$ and (g' is not ancestor set of $(c' \setminus g')$) and (g' is not descendant set of $(c' \setminus g')$)
14. then *nonRedundant = true*
15. break
16. end if
17. end if
18. end for
19. if *nonRedundant = true*
20. insert $\{ g \Rightarrow (c \setminus g)$,*g.supp*$\}$ in *ReliableExact*
21. end if
22. end for
23. end for
24. return *ReliableExact*

The complexity of the original Min-MaxExact is $O(n)$, where n is the number of generators derived from the frequent itemsets. For the algorithm Min-MaxExact without HR , before generating a rule, we need to scan all generators to determine whether it is hierarchically redundant. Therefore, the complexity of the algorithm Min-Max-Exact without HR is $O(n^2)$. For the original ReliableExact algorithm, the complexity is $O(n^2)$. Our modified algorithm ReliableExact without HR does not change its complexity, i.e., $O(n^2)$. For large datasets, with the $O(n^2)$ complexity, the two proposed methods may have efficiency problems. This issue will be addressed in our future work.

4.4 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 (Xu & Li 2007). 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 of 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 of 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. $Recovered := \varnothing$
2. $\forall r \in Exactbasis$
3. $CandidateBasis := \varnothing$
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 of the possible subsets of g into S
7. end for
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 decendants 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*
11. end if
12. end for
13. for all $B \Rightarrow D \in CandidateBasis$
14. if $B \bigcup D =$ itemset $i \in C$ and $B = g \in G_i$
15. insert $\{ B \Rightarrow D , g.supp\}$ in *Recovered*
16. end if
17. end for
18. end for
19. return $Exactbasis \bigcup Recovered$

5. Evaluation

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.

5.1 Datasets

We used 8 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. We used the same datasets used by Han & Fu (1999) and Thakur et al. (2006) which

had seven and eight transactions respectively and are named H1 and T1 respectively. We also used 5 randomly built datasets which were composed of 10, 20, 50, 200 and 500 transactions and are named T2 to T6 respectively. The key statistics for these built datasets are detailed in Table 7. The last dataset, BC, used in our experiments is based on the real world Book-Crossing dataset (http://www.informatik.unifreiburg.de/cziegler/ BX/) (Ziegler et al. 2005), from which we built a transactional dataset that contained 92,005 records and 270 inner and leaf categories with 2 concept levels.

Table 7 Dataset statistics

Dataset Parameters	T2	T3	T4	T5	T6
No. of transactions	10	20	50	200	500
Average no. of items per transaction	5	7	7	7	20
No. of items on the top concept level	5	10	10	10	10
No. of levels in the hierarchy	3	4	4	4	4
Average no. of child items a given item has	3	4	4	4	4

The experiments aim to find associations among the items in each of the datasets. The process to discover the association rules involves three steps. Firstly, the frequent itemsets are discovered through the use of minimal support values for each hierarchy level. We have implemented two approaches to find the frequent itemsets; Han & Fu's ML_T2L1 approach presented in Han & Fu (1999) with the addition to the base algorithm so as to find cross-level itemsets, and Thakur's algorithm (referred to as CLI) to find cross-level itemsets (along with normal itemsets) presented in Thakur, Jain and Pardasani (2006). Second, from the frequent itemsets, the frequent closed itemsets and generators are derived. We have implemented the CLOSE+ algorithm proposed by Pasquier et. al. (2005) to achieve this. Finally, the association rules are built. In these experiments we derive the rules using Pasquier's et. al. Min-MaxExact (referred to as MME) (Pasquier et al. 2005) and Xu & Li's ReliableExact approach (referred to as RE) (Xu & Li 2007), a modified version of Pasquier's et. al. work to include removing hierarchical redundancy (referred to as MMEHR) and a modified version of Xu & Li's work to include removing hierarchical redundancy (referred to as REHR).

5.2 Experiment Results

The primary objective of the experiments is to determine how well our proposed work performs at removing/reducing hierarchical redundancy in datasets even when other redundancy eliminating processes are included. The other objective is to ensure and demonstrate that this approach is lossless. We have defined our approach earlier in Section 4.3 to remove redundant rules in multi-level datasets and thus the exact basis should be smaller in size when it is utilized. We also confirm that our approach can recover all exact rules from multi-level datasets by comparing the modified versions of Min-MaxExact and ReliableExact (which include our work to remove hierarchically redundant rules) against unmodified versions for

each dataset to ensure that each recovers the same set of exact rules. Our approach to recover and derive all of the exact rules is detailed in Section 4.4. We also compare the size of the exact basis set generated by each of the four approaches to see what reduction in the basis set can be achieved. For all of the testing undertaken, the minimum confidence threshold for the association rules was set at 0.5. Tables 8 and 9 present the results obtained from each of the datasets showing the percentage reduction achieved.

As can be seen, the use of our approach reduces the exact basis rule set for all cases we tested. In some instances the basis set was only reduced by a few rules, but in other cases there was a more significant reduction in the size of the basis set. For example, in Table 8 for dataset T4 there was a reduction of 148 rules from 577 to 429, which is about 25.5%, and the reduction was around 46 to 47% for dataset H1 and nearly 36% for dataset T2. By using this approach we have successfully reduced the size of the exact basis and by doing so it may help to make it more possible to effectively use the extracted association rules without overwhelming a user. For the dataset BC derived from the Book-Crossing dataset, the number of rules generated is small. We believe that the small size of the discovered rule sets is due to the sparseness of the dataset, for instance, at the second level, out of all the 24,841,350 possible ratings (i.e., 92,005 * 270) there is only 427,422 actual ratings (where a rating indicates that a book in a category has been rated by the use). From the results, we can see that, despite the small rule sets, we still achieved 20% to 33% reduction.

For each test conducted we also checked the expanded exact association rules, i.e., to derive all exact rules from the exact basis. All four approaches were checked to ensure that they all derived the same number of expanded rules and that the sets were identical. For all of our tests this was the case. Thus, the results show that our approach, while reducing the size of the exact basis set does not lose any information and the expanded set of rules can be completely recovered.

Table 8 Results obtained using ML_T2LI with cross level add-on to extract frequent itemsets

Dataset	MME	MMEHR	Reduction (%)	RER	RERHR	Reduction (%)
H1	21	11	47	15	8	46
T1	15	10	33	13	9	31
T2	106	68	36	80	58	27
T3	174	134	23	113	89	21
T4	577	429	25	383	305	20
T5	450	405	10	315	287	9
T6	725	602	17	91	80	12
BC	56	45	20	18	12	33

Table 9 Results obtained using CLI with cross level add-on to extract frequent itemsets

Dataset	MME	MMEHR	Reduction (%)	RER	RERHR	Reduction (%)
H1	9	7	22	5	4	20
T1	2	1	50	2	1	50
T2	62	42	32	46	33	28
T3	44	39	11	29	26	10
T4	356	271	24	244	196	19
T5	180	174	3	121	116	4
T6	325	293	10	53	47	11
BC	54	43	20	18	12	33

6. 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.

Acknowledgment

The authors appreciate the referees for their valuable comments and suggestions.

References

[1] Bayardo, R.J., Agrawal, R. & Gunopulos, D. (2000). Constraint-based rule mining in large, dense databases. Data Mining and Knowledge Discovery, 4: 217-240

[2] Berry, M.J.A. & Linoff, G.S. (1997). Data Mining Techniques for Marketing Sales and Customer Support. John Wiley and Sons

[3] Brin, S., Motwani, R., Ullman, J.D. & Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. In: Proceedings of the 1997 ACM SIGMOD Conference, 255-264

[4] Ganter, B. & Wille, R. (1999). Formal Concept Analysis: Mathematical Foundations. Springer-Verlag

[5] Han, J. & Fu, Y. (1995). Discovery of multiple-level association rules from large databases. In: Proceedings of the 21st International Conference on Very Large Databases, 420-431

[6] Han, J. & Fu, Y. (1999). Mining multiple-level association rules in large databases. IEEE Transactions on Knowledge and Data Engineering, 11 (5): 798-805

[7] Han, J. & Fu, Y. (2000). Mining multiple-level association rules in large databases. IEEE Transactions on Knowledge and Data Engineering, 11: 798-804

[8] Hong, T.P., Lin, K.Y. & Chien, B.C. (2003). Mining fuzzy multiple-level association rules from quantitative data. Applied Intelligence, 18 (1): 79-90

[9] Kaya, M. & Alhajj, R. (2004). Mining multi-cross-level fuzzy weighted association rules. In: the 2nd International IEEE Conference on Intelligent Systems, 225-230

[10] Kryszkiewicz, M., Rybinski, H. & Gajek, M. (2004). Dataless transitions between concise representations of frequent patterns. Journal of Intelligent Information Systems, 22 (1): 41-70

[11] Ng, R.T., Lakshmanan, V., Han, J. & Pang, A. (1998). Exploratory mining and pruning otimizations of constrained association rules. In: Proceedings of the SIGMOD conference, 13-24

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

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

[14] Srikant, R., Vu, Q. & Agrawal, R. (1997). Mining association rules with item constraints. In: Proceedings of the KDD Conference, 67-73

[15] Thakur, R.S., Jain, R.C. & Pardasani, K.P. (2006). Mining level-crossing association rules from large databases. Journal of Computer Science, 76-81

[16] Wille, R. (1982). Restructuring lattices theory: An approach based on hierarchies of concepts. In: Rival, I. (ed.), Ordered Sets. Dordrecht-Boston

[17] Xu, Y. & Li, Y. (2007). Generating concise association rules. In: Proceedings of the 16th ACM Conference on Conference on Information and Knowledge Management (CIKM07), 781-790

[18] Zaki, M.J. (2000). Generating non-redundent association rules. In: Proceedings of the KDD Conference, 34-43

[19] Zaki, M.J. (2004). Mining non-redundant association rules. Data Mining and Knowledge Discovery, 9: 223-248

[20] Ziegler, C.N., McNee, S.M., Konstan, J.A. & Lausen, G. (2005). Improving recommendation lists through topic diversification. In: Proceedings of the 14th International World Wide Web Conference (WWW05), 22-32

Yue Xu received her Bachelor degree from the University of Science and Technology of China, Master degree from the Institute of Computing Technology, the Chinese Academy of Science, and a PhD from the University of New England, Australia. Currently she is a senior lecturer in the School of Information Technology, the Queensland University of Technology, Brisbane, Australia. Her current research interests include association rule mining, recommender systems, cross-language information retrieval, and Web

intelligence. She has published over 80 refereed papers, supervised two PhD students and one research master student to successful completion. Currently she is the principal supervisor of four PhD students and two research master students.

Gavin Shaw is a PhD candidate in the School of Information Technology, Queensland University of Technology, Brisbane, Australia. He obtained his bachelor (with first class honor) from Queensland University of Technology in 2006. His research interests include clustering, Web page feature extraction, and association rule mining. Currently he is focused on association rule mining in multi-level datasets for extracting non-redundant multi-level and cross-level association rules and application in recommender systems.

Yuefeng Li received his Bachelor degree in mathematics and a Master degree in computer science from Jilin University, China, and a PhD from Deakin University, Australia. Currently he is an Associate Professor in the School of Information Technology, Queensland University of Technology, Brisbane, Australia. His research interests include ontology learning, Web intelligence, information gathering, and data mining. He is an Associate Editor of the International Journal of Pattern Recognition and Artificial Intelligence and an Associate Editor of the IEEE Intelligent Informatics Bulletin. He has published over 100 refereed papers. He has supervised three PhD students and five research master students to successful completion. Currently he is the principal supervisor for five PhD students.