Multi-Tier Granule Mining for Representations of Multidimensional Association Rules

Yuefeng Li, Wanzhong Yang, Yue Xu
School of Software Engineering and Data Communications
Queensland University of Technology, Brisbane, QLD 4001, Australia
y2.li@qut.edu.au, w3.yang@student.qut.edu.au, yue.xu@qut.edu.au

Abstract

It is a big challenge to promise the quality of multidimensional association mining. The essential issue is how to represent meaningful multidimensional association rules efficiently. Currently we have not found satisfactory approaches for solving this challenge because of the complicated correlation between attributes. Multi-tier granule mining is an initiative for solving this challenging issue. It divides attributes into some tiers and then compresses the large multidimensional database into granules at each tier. It also builds association mappings to illustrate the correlation between tiers. In this way, the meaningful association rules can be justified according to these association mappings.

1. Introduction

Multidimensional association mining discusses two or more data dimensions or predicates [3]. Usually multidimensional association mining is designed for searching frequent predicate sets and that can be classified into inter-dimension and hybrid-dimension association rule mining.

We can obtain a huge amount of association rules using the existing data mining techniques. However, not all strong association rules are interesting to users [3]. Several approaches have been conducted in order to guarantee the quality of discovered knowledge: the concept of closed patterns [14] [15], non-redundant rules [17] [18], and constraint-based association rules [1] [4] [5] [10] [12] [16].

These approaches have significant performance for decreasing the number of association rules for transaction databases. However, they are not very efficient for representation of associations in very large multidimensional databases because we have to transfer multidimensional rule mining into single dimensional mining when we use these approaches.

Different to these approaches, in this paper we present the concept of granule mining (GM) in multidimensional databases for directly representations of associations between attributes, where a granule is a group of objects (transactions) that have the same attributes' values.

Basically attributes are divided by users into two groups: condition attributes and decision attributes, and decision tables can be used to represent the association between condition granules and decision granules [11] [8]. In cases of large number of attributes, however, decision tables become inefficient. Decision tables also cannot describe association rules with shorter premises (we call such rules general rules in this paper). To solve these drawbacks, in this paper we present multi-tier structures and association mappings to manage associations between attributes. It provides an alternative way to represent multidimensional association rules efficiently.

The remainder of the paper is structured as follows. We begin by introducing the concept of closed patterns and decision tables for granule mining in Section 2. In section 3, we discuss the relationship between granule mining and data mining. In Section 4, we introduce the multi-tier structure. In Section 5, we formalize association mappings for GM. In Section 6, we introduce the experiment results. The last section includes related work and conclusions.

2. Basic Definitions

Formally, a transaction database can be described as an information table ($\mathcal{T}$, $V^{\mathcal{T}}$), where $\mathcal{T}$ is a set of objects in which each record is a sequences of items, and $V^{\mathcal{T}} = \{a_1, a_2, \ldots, a_n\}$ is a set of selected items (or called attributes in decision tables) for all objects in $\mathcal{T}$. Each

Proceedings of the Sixth International Conference on Data Mining (ICDM'06)
0-7695-2701-9/06 $20.00 © 2006 IEEE

item can be a tuple, e.g., (*name*, *cost*, *price*) is a product item.

2.1 Closed Patterns

Definition 1. A set of items X is referred to as an *itemset* if $X \subseteq V^{\mathcal{T}}$. Let X be a itemset, we use $[X]$ to denote the *covering set* of X, which includes all objects t such that $X \subseteq t$, i.e., $[X] = \{t \mid t \in \mathcal{T}, X \subseteq t\}$.

Given an *itemset* X, its occurrence frequency is the number of objects that contain the *itemset*, that is $|[X]|$; and its support is $|[X]|/|\mathcal{T}|$. An itemset X is called *frequent pattern* if its support $\geq$ *min_sup*, a minimum support.

Definition 2. Given a set of objects Y, its *itemset* which satisfies

$$itemset(Y) = \{a \mid a \in V^{\mathcal{T}}, \forall t \in Y \Rightarrow a \in t\}.$$

Given a frequent pattern X, its *closure*

$$Closure(X) = itemset([X]).$$

From the above definitions, we have the following theorem (see [18]).

Theorem 1. Let X and Y be frequent patterns. We have

(1) $Closure(X) \supseteq X$ for all frequent patterns X;
(2) $X \subseteq Y \Rightarrow Closure(X) \subseteq Closure(Y)$.

Definition 3. An frequent pattern X is *closed* if and only if $X = Closure(X)$.

2.2. Decision Tables

Decision tables can be used for dealing with multiple dimensional databases in line with user constraints. Formally, users may use some attributes of a database; and they can divide these attributes into two groups: condition attributes and decision attributes, respectively. We call the tuple $(\mathcal{T}, V^{\mathcal{T}}, C, D)$ a *decision table* of $(\mathcal{T}, V^{\mathcal{T}})$ if $C \cap D = \emptyset$ and $C \cup D \subseteq V^{\mathcal{T}}$.

We usually assume that there is a function for every attribute $a \in V^{\mathcal{T}}$ such that a: $\mathcal{T} \rightarrow V_a$, where V_a is the set of all values of a. We call V_a the domain of a. C (or D) determines a binary relation $I(C)$ (or $I(D)$) on $\mathcal{T}$ such that $(t_1, t_2) \in I(C)$ if and only if $a(t_1) = a(t_2)$ for every $a \in C$, where $a(t)$ denotes the value of attribute a for object $t \in \mathcal{T}$. It is easy to prove that $I(C)$ is an equivalence relation, and the family of all equivalence classes of $I(C)$, that is a partition determined by C, is denoted by $\mathcal{T}/C$.

The classes in $\mathcal{T}/C$ (or $\mathcal{T}/D$) are referred to C-*granules* (or D-*granule*). The class which contains t is called C-*granule* induced by t, and is denoted by $C(t)$.

Object	*items*
t_1	a_1 a_2
t_2	a_3 a_4 a_6
t_3	a_3 a_4 a_5 a_6
t_4	a_3 a_4 a_5 a_6
t_5	a_1 a_2 a_6 a_7
t_6	a_1 a_2 a_6 a_7

Table 1 lists a part of an transition database, where $V^{\mathcal{T}} = \{a_1, a_2, \ldots, a_7\}$, $\mathcal{T} = \{t_1, t_2, \ldots, t_6\}$. We also can represent Table 1 as a decision table. Let a_1, a_2, a_3, a_4 and a_5 be the condition attributes and a_6 and a_7 be the decision attributes. Table 2 shows a decision table of Table 1, where $\mathcal{T}/C \cup D = \{g_1, g_2, g_3, g_4\}$ and $\boldsymbol{N_g}$ is the number of objects that are in the same granule.

Table 2. A decision table

Granule	a_1	a_2	a_3	a_4	a_5	a_6	a_7	N_g
g_1	1	1	0	0	0	0	0	1
g_2	0	0	1	1	0	1	0	1
g_3	0	0	1	1	1	1	0	2
g_4	1	1	0	0	0	1	1	2

Every granule in the decision table can be mapped into a decision rule [11], where we treat the presence and absence of items as the same position if we view the decision table as a multidimensional database. Therefore, we can obtain 4 decision rules in Table 2, and the first one can be read as the following decision rule:

$$a1 = 1 \wedge a2 = 1 \wedge a3 = 0 \wedge a4 = 0 \wedge a5 = 0$$
$$\rightarrow a6 = 0 \wedge a7 = 0$$

or in short $C(g_1) \rightarrow D(g_1)$ (or $C(t_1) \rightarrow D(t_1)$), where ^ means "*and*".

3. Data Mining and Granule Mining

Decision tables provide an efficient way to represent discovered knowledge. However, currently we can only obtain decision rules, a kind of very special association rules, in decision tables. To interpret what kinds of association rules in the decision tables, we present the concept of decision patterns.

Given a C-*granule* $cg = C(t)$, its *covering set* $[cg] = \{t' \mid t' \in \mathcal{T}, (t', t) \in I(C)\}$. Let cg be a C-*granule* and dg be a D-*granule*, we define $[cg \wedge dg] = [cg] \cap [dg]$.

For example, in Table 2 $g_1 = (a1 = 1 \wedge a2 = 1 \wedge a3 = 0 \wedge a4 = 0 \wedge a5 = 0 \wedge a6 = 0 \wedge a7 = 0) = C(g_1) \wedge D(g_1) = cg_1 \wedge dg_1$; therefore

$$[g_1] = [cg_1 \wedge dg_1] = [cg_1] \cap [dg_1]$$

$= \{d1, d5, d6\} \cap \{d1\} = \{d1\}$.

Table 3 illustrates the covering sets of granules, where (a) includes the covering sets of *C-granules*, (b) includes the covering sets of *D-granules*, and (c) includes the covering sets of *C∪D-granules*.

Table 3. Covering sets of granules

Granule	a_1	a_2	a_3	a_4	a_5	*covering set*
cg_1	1	1	0	0	0	$\{t1, t5, t6\}$
cg_2	0	0	1	1	0	$\{t2\}$
cg_3	0	0	1	1	1	$\{t3, t4\}$

(a) Covering sets of *C-granules*

Granule	a_6	a_7	*covering set*
dg_1	0	0	$\{t1\}$
dg_2	1	0	$\{t2, t3, t4\}$
dg_3	1	1	$\{t5, t6\}$

(b) Covering sets of *D-granules*

Granule	a_1	a_2	a_3	a_4	a_5	a_6	a_7	*covering set*
g_1	1	1	0	0	0	0	0	$\{t1\}$
g_2	0	0	1	1	0	1	0	$\{t2\}$
g_3	0	0	1	1	1	1	0	$\{t3, t4\}$
g_4	1	1	0	0	0	1	1	$\{t5, t6\}$

(c) Covering sets of *C∪D-granules*

Definition 4. Let X be a frequent pattern. We call it a decision *pattern* if $\exists g \in \mathcal{T}/C\cup D$ such that $X = \{a_i \in C\cup D \mid a_i(g) = 1\}$. We call X the derived decision pattern of g.

Theorem 2. Let $(\mathcal{T}, V^{\mathcal{T}}, C, D)$ be a *decision table*. We have

(1) $[C(t)] \supseteq [C\cup D(t)]$, for all $t \in \mathcal{T}$.

(2) The derived decision pattern of every granule $g \in \mathcal{T}/C\cup D$ is a closed pattern.

Proof: (1) is obvious in accordance with the definition of closure.

For (2), Let X be the derived pattern of g, that is, $X=\{a_i \in C\cup D \mid a_i(g) = 1\}$. From the definition of the granules, we know there is a object $t_0 \in [g]$ such that $X = \{a_i \in C\cup D \mid a_i(t_0) = 1\}$, that is $t_0 \in [X]$.

Given an item $a \in itemset([X])$, according to Definition 2 we have $a \in t$ for all $t \in [X]$, that is, $a \in t_0$ and also $a \in X$. Therefore, $Closure(X) = itemset([X]) \subseteq X$.

We also have $X \subseteq Closure(X)$ from Theorem 1, and hence we have $X = Closure(X)$. □

4. Multi-Tier Structures

In cases of large number of attributes, the decision tables become inefficient. Also, we cannot discover general rules in decision tables, for example, association rules with shorter premises. In addition, some decision rules may be meaningless. In this section, we present a multi-tier structure to manage the correlation between attributes in order to overcome these disadvantages of decision tables. We also clarify the meaning of meaningless in this section.

Let $\mathcal{T}/C$ be the set of condition granules and $\mathcal{T}/D$ be the set of decision granules. To describe the association between condition granules and decision granules, we can further divide the condition attributes into some groups in accordance with user constraints.

We assume that C_i and C_j are two subsets of condition attributes; and they satisfy:

$$C_i \cap C_j = \emptyset \text{ and } C_i \cup C_j = C.$$

Figure 1 illustrates the structure of the multi-tier granule mining, which describes the hierarchy of the possible associations between tiers, where $\mathcal{T}/C_i = \{cg_{i,1}, cg_{i,2}, \ldots, cg_{i,k}\}$, $\mathcal{T}/C_j = \{cg_{j,1}, cg_{j,2}, \ldots, cg_{j,m}\}$, and $\mathcal{T}/D = \{dg_1, dg_2, \ldots, dg_s\}$.

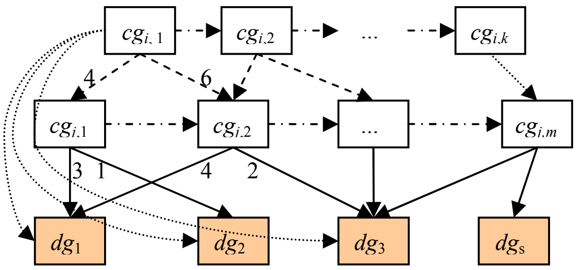

Figure 1. The hierarchy of the multi-tiers

The decision rules in the structure of multiple tiers can be illustrated as follows: $cg_{i,x} \wedge cg_{j,y} \rightarrow dg_z$

$(conf = \|[cg_{i,x} \wedge cg_{j,y} \wedge dg_z]\| / \|[cg_{i,x} \wedge cg_{j,y}]\|)$

Different to decision tables, we can obtain some general association rules with shorter premises as follows: $cg_{i,x} \rightarrow dg_z$, $(conf = \|[cg_{i,x} \wedge dg_z]\| / \|[cg_{i,x}]\|)$.

In Figure 1, we assume that

$\|[cg_{i,1} \wedge cg_{j,1} \wedge dg_1]\| = 3$, $\|[cg_{i,1} \wedge cg_{j,1} \wedge dg_2]\| = 1$

$\|[cg_{i,1} \wedge cg_{j,2} \wedge dg_1]\| = 4$, $\|[cg_{i,1} \wedge cg_{j,2} \wedge dg_3]\| = 2$

$\|[cg_{i,1} \wedge cg_{j,1}]\| = 4$, $\|[cg_{i,1} \wedge cg_{j,2}]\| = 6$, $\|[cg_{i,1}]\| = 10$.

According the above assumption and the structure in Figure 1, we have the following decision rules:

$$cg_{i,1} \wedge cg_{j,1} \rightarrow dg_1 \ (conf = 3/4 = 0.75)$$
$$cg_{i,1} \wedge cg_{j,1} \rightarrow dg_2 \ (conf = 1/4 = 0.25)$$
$$cg_{i,1} \wedge cg_{j,2} \rightarrow dg_1 \ (conf = 4/6 = 0.67)$$
$$cg_{i,1} \wedge cg_{j,2} \rightarrow dg_3 \ (conf = 2/6 = 0.33)$$

We also can obtain a general association rule with the shorter premise: $cg_{i,1} \rightarrow dg_1$ ($conf$ = 7/10 = 0.70).

Definition 5. A rule $cg_{i,x} \wedge cg_{j,y} \rightarrow dg_z$ is called meaningless if its confidence is less than or equals to the confidence of its general rule: $cg_{i,x} \rightarrow dg_z$.

Based on Definition 5, rule $cg_{i,1} \wedge cg_{j,2} \rightarrow dg_1$ ($conf$ = 4/6 = 0.67) is a meaningless rule since its confidence (0.67) is less than the confidence (0.70) of its general rule $cg_{i,1} \rightarrow dg_1$ ($conf$ = 7/10 = 0.70).

The rationale of the above definition is analogous to the definition of interesting association rules. If add extra evidence to a premise and obtain a weak conclusion, we can say the piece of evidence is meaningless.

5. Association Mappings

In this section, we firstly formalize the basic association in a decision table, and then we develop methods to derive other associations between granules in different tiers based on this basic association.

The basic association between condition granules and decision granules can be described as an association mapping $\Gamma_{i,j,d}$ such that $\Gamma_{i,j,d}(cg_{i,x} \wedge cg_{j,y})$ is a set of *D-granule* integer pairs. For example, using the granules in Figure 1, we have

$$\Gamma_{i,j,d}(cg_{i,1} \wedge cg_{j,1}) = \{(dg_1,3),(dg_2,1)\}.$$

Let N=|$\mathcal{T}$| and $dg_z \in \{dg | (dg, f) \in \Gamma_{i,j,d}(cg_{i,x} \wedge cg_{j,y})\}$, we can obtain a decision rule: $cg_{i,x} \wedge cg_{j,y} \rightarrow dg_z$ with the support and confidence:

$$sup(cg_{i,x} \wedge cg_{j,y} \rightarrow dg_z) = \frac{\sum_{(dg_z,f)\in\Gamma_{i,j,d}(cg_{i,x}\wedge cg_{j,y})} f}{N}$$

$$conf = |[cg_{i,x} \wedge cg_{j,y} \wedge dg_z]| \,/\, |[cg_{i,x} \wedge cg_{j,y}]|)$$
$$= \frac{\sum_{(dg_z,f)\in\Gamma_{i,j,d}(cg_{i,x}\wedge cg_{j,y})} f}{\sum_{(dg,f)\in\Gamma_{i,j,d}(cg_{i,x}\wedge cg_{j,y})} f}$$

The association mapping $\Gamma_{i,j}$ between $\mathcal{T}/C_i$ *and* $\mathcal{T}/C_j$ can be derived from association mapping $\Gamma_{i,j,d}$, where $\Gamma_{i,j}(cg_{i,x})$ is a set of C_j*-granule* integer pairs. The following is the equation that we can derive:

$$\Gamma_{i,j}(cg_{i,x}) = \{(cg_{j,y} \wedge f1) \mid \Gamma_{i,j,d}(cg_{i,x} \wedge cg_{j,y}) \neq \emptyset, f1 = \sum_{(dg,f)\in\Gamma_{i,j,d}(cg_{i,x}\wedge cg_{j,y})} f\}$$

For example, using the granules in Figure 1, we have:

$$\Gamma_{i,j}(cg_{i,1}) = \{(cg_{j,1},4),(cg_{j,2},6)\}.$$

It is more complicated to derive the association $\Gamma_{i,d}$ between $\mathcal{T}/C_i$ and $\mathcal{T}/D$ based on association $\Gamma_{i,j,d}$ and $\Gamma_{i,j}$. To simplify this process, we first review the composition operation that defined in [9].

Let P_1 and P_2 be sets of *D-granule* integer pairs. We call $P_1 \oplus P_2$ the *composition* of P_1 and P_2 which satisfies:

$$P_1 \oplus P_2 = \{(dg, f_1 + f_2) \mid (dg, f_1) \in P_1, (dg, f_2) \in P_2\} \cup \{(dg, f) \mid dg \in (gname(P_1) \cup gname(P_2)) - (gname(P_1) \cap gname(P_2)), (dg, f) \in P_1 \cup P_2\},$$

where $gname(P_i) = \{dg \mid (dg, f) \in P_i\}$.

The operands of $\oplus$ are interchangeable, therefore we can use $\oplus\{P_1, P_2, P_3\}$ to be the short form of $(P_1 \oplus P_2) \oplus P_3$. The result of the composition is still a set of *D-granule* integer pairs.

Let $\Gamma_{i,d}$ be the association mapping between $\mathcal{T}/C_i$ *and* $\mathcal{T}/D$, we have the following equation for it:

$$\Gamma_{i,d}(cg_{i,x}) = \oplus\{\Gamma_{i,j,d}(cg_{i,x} \wedge cg_{j,y}) \mid (cg_{j,y}, f1) \in \Gamma_{i,j}(cg_{i,x})\}$$

for all $cg_{i,x} \in \mathcal{T}/C_i$.

Algorithm 5.1 (Construction of Multi-Tiers)
Input parameters: ($\mathcal{T}$, $V^{\mathcal{T}}$, C, D, C_i, C_j).
Output: Association mappings.
***Method* 1:** (Evaluate $\Gamma_{i,j,d}$).
  $\mathcal{T}/C = \emptyset$;
  for (each $g \in \mathcal{T}/C \cup D$) *//start from a decision table*
    if ($C(g) \in \mathcal{T}/C$) *//notice g=$C(g) \wedge D(g)$*
      $\Gamma_{i,j,d}(C(g)) = \Gamma_{i,j,d}(C(g)) \cup \{(D(g), N_g)\}$;
    else {
      $\mathcal{T}/C = \mathcal{T}/C \cup \{C(g)\}$;
      $\Gamma_{i,j,d}(C(g)) = \{(D(g), N_g)\}$; }

***Method* 2:** (Evaluate $\Gamma_{i,j}$)
  $\mathcal{T}/C_i = \emptyset$;
  for (each $cg \in \mathcal{T}/C$) {
    $f1$= 0; *//notice cg=$C_i(cg) \wedge C_j(cg)$*
    for ((dg, f)$\in \Gamma_{i,j,d}(cg)$) $f1 = f1 + f$;
    if ($C_i(cg) \in \mathcal{T}/C_i$)
      $\Gamma_{i,j}(C_i(cg)) = \Gamma_{i,j}(C_i(cg)) \cup \{(C_j(cg), f1)\}$;
    else {
      $\mathcal{T}/C_i = \mathcal{T}/C_i \cup \{C_i(cg)\}$;
      $\Gamma_{i,j}(C_i(cg)) = \{(C_j(cg), f1)\}$ } };

***Method* 3:** (Evaluate $\Gamma_{i,d}$)
  for (each $cg_{i,x} \in \mathcal{T}/C_i$) {
    $\Gamma_{i,d}(cg_{i,x}) = \emptyset$;
    for (($cg_{j,y}, f1$)$\in \Gamma_{i,j}(cg_{i,x})$)
      $\Gamma_{i,d}(cg_{i,x}) = \Gamma_{i,d}(cg_{i,x}) \oplus \Gamma_{i,j,d}(cg_{i,x} \wedge cg_{j,y})$ };

For example, using the information in Figure 1 we have:

$\Gamma_{i,d}(cg_{i,1}) = \{(dg_1, 3), (dg_2, 1)\} \oplus \{(dg_1, 4), (dg_3, 2)\}$

= {(dg_1, 7), (dg_2, 1), (dg_3, 2)}.

Algorithm 5.1 describes the main procedure of the construction of the multi-tier structure. It includes three methods for calculating the three kinds of association mappings.

The time complexity of Algorithm 5.1 is determined by Method 1 because $|\mathcal{T}/C_i| \leq |\mathcal{T}/C|$. In Method 1, checking $C(g) \in \mathcal{T}/C$ takes $O(|\mathcal{T}/C|)$, so the time complexity of the algorithm is $O(n \times |\mathcal{T}/C|)$, where n is the number of granules in the decision table, and the basic operation is the comparison between granules. Since $|\mathcal{T}/C| \leq n$, the time complexity of Algorithm 5.1 $\leq O(n^2)$.

After constructed the multi-tier structure, we can easily obtain decision rules and general rules by traversal of the multi-tier structure. Pruning meaningless decision rules can also be simply implemented by removing pairs from the corresponding association mapping. For example, given a condition granule *cg*, based on its general rule, we might remove pairs in $\Gamma_{i,j,d}$ (*cg*) if they are the conclusions of meaningless rules.

6. Experiments

We simulate the data in a real multiple store environment where a fact table of sales can be described using one star schema including multiple dimensions: customer dimension, time dimension, data dimension, store dimension and production dimension. In our current experiment, we use time and production dimensions only. A product includes *name*, *cost* and *price* attributes.

The fact table of sales in a financial year includes 26,590 transaction records and 5000 different products. We view each product as an item. We set one to an item for a transaction if it appears in the transaction; otherwise we set up zero to it.

We first select 300 most frequent products as items (attributes) {*a*1, *a*2, … , *a*300}. We also choice 162 products C from the 300 products, which profits are more than 50% (*price* > *1.5* × *cost*) as condition attributes; and select 35 products D from the 300 products, which profits are less than or equal to 20% (*1.2* × *cost* > *price*) as decision attributes.

After compressed the transaction records, we obtain a decision table which includes 2486 granules, and hence we can generate 2486 decision rules.

The condition attributes are further divided into two tiers: C_i-tier the products that profits are more than 90% (*price* > *1.9* × *cost*); and C_j-tier the products that the profits are in [90%, 50%), (*1.9* × *cost* ≥ *price* > *1.5* × *cost*). The products are now classified into three tiers: *high* profit C_i*-tier*, *medium* profit C_j*-tier*, and *low* profit *D-tier*. The association between *high* profit products and *low* profit *produces* can also be described as general rules.

Figure 2 illustrates the numbers of granules and attributes in the three tiers. To compare to decision table, the multi-tier structure is extremely impressive since only very small amounts of granules are useful for generating rules.

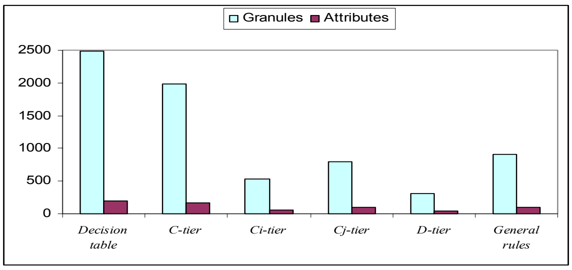

Figure 2. Granules and attributes in multi-tiers.

Figure 3 depicts the percentages (57.9%) of meaningless decision rules that can be pruned. It is also shows the percentages (68.9%) of transactions that are covered by these meaningless decision rules.

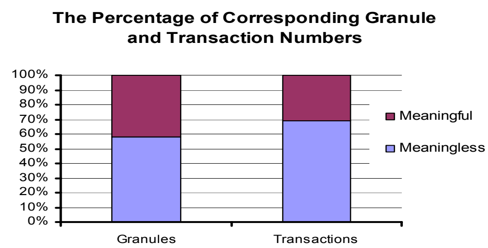

Figure 3. The percentage of meaningless decision rules.

In summary, the results demonstrate that the multi-tier structure uses only a very small space to store meaningful multi-dimensional association rules. It is a very efficient and promising alternative of decision tables for representations of multidimensional association rules.

7. Related Work

As mentioned in the introduction, several approaches have been conducted for the quality of discovered knowledge. We also noticed another interesting research, which discussed the similarity between patterns to discover the real useful patterns [13].

For multidimensional association mining, Han et al. in [3] [4] summarized the possible techniques in accordance with the corresponding treatments of quantitative attributes. Habitually, most current researchers on multidimensional association mining endeavor to use the existing efficient algorithms for single dimensional mining. Lee et al. presented an approach for multidimensional constraints [6]. It checked constraint during FP-tree constructions. The approach firstly grouped products at the same cost and price into an item, and view the product table as a set of transactions.

In this paper, we concentrate on inter-dimension association mining. Different to other ideas, we want to describe the associations in multidimensional databases based on some abstractions (granules).

In the beginning, rough set theory looked like an adequate tool for this question, and can be used to describes the knowledge in information tables [2] [7]. Further, rough set based decision tables [11] presented by Pawlak can be used to represented some sorts of association rules. Li and Zhong [8] also presented a structure to disconnect the condition granules and decision granules in order to improve the efficiency of generating associate rules from decision tables.

Rough set theory is elegant, and has a clear logical semantics. However, it lacks the accurateness when we use it to deal with the associations between granules in multiple tiers.

8. Conclusions

In this research, we present a multi-tier granule mining approach in order to provide a foundational framework for efficiently representations of multidimensional association rules. We demonstrate that it is a significant replacement of decision tables. We also prove that granules in decision tables are kinds of closed patterns. In addition, we present the definition of meaningless decision rules. The definition can also be justified in the multi-tier structure.

Acknowledgments

This paper was partially supported by Grant DP0556455 from the Australian Research Council. The authors also wish to thank Professor Ning Zhong at Maebashi Institute of Technology, Japan for his valuable comments.

References

[1] C. Bucila, J. Gehrke, D. Kifer, W. White, "DualMiner: a Dual-Pruning algorithm for Itemsets with constraints," *ACM-SIGKDD*, Alberta, Canada, 2002, 42-51.

[2] J. W. Guan, D. A. Bell, D. Y. Liu, "The rough set approach to association rules," *ICDM*, USA, 2003, 529-532.

[3] J. Han and M. Kamber, "*Data Mining: Concepts and Techniques"*, Morgan Kaufmann Publishers, 2000.

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

[5] C. K.-S. Leung, L.V.S. Lakshmanan, R. T. Ng, "Exploiting succinct constraints using FP-trees," *ACM-SIGKDD Explorations*, 2002, 40-49.

[6] A. J. T. Lee, W. Lin, C. Wang, "Mining association rules with multi-dimensional constraints," *The Journal of Systems and Software*, 2006, 79-92.

[7] Y. Li, "Extended random sets for knowledge discovery in information systems", *9th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, China, 2003, 524-532.

[8] Y. Li and N. Zhong, "Interpretations of association rules by granular computing," *ICDM*, USA, 2003, 593-596.

[9] Y. Li and N. Zhong, "Mining ontology for automatically acquiring Web user information needs," *IEEE Transactions on Knowledge and Data Engineering*, 2006, **18(4):** 554-568.

[10] R.T.Ng, L.V.S. Lakshmanan, J. Han, A. Pang, "Exploratory mining and pruning optimizations of constrained associations rules," *ACM-SIGMOD*, Seattle, Washington, USA, 1998, 13-24.

[11] Z. Pawlak, "In pursuit of patterns in data reasoning from data, the rough set way," *3rd International Conference on Rough Sets and Current Trends in Computing*, USA, 2002, 1-9.

[12] J. Pei, J. Han, L.V.S. Lakshmanan, "Mining frequent itemsets with convertible constraints," *17th International Conference on Data Engineering*, Heidelberg, Germany, 2001, 433-442.

[13] S. Tsumoto and S. Hirano, "Visualization of rule's similarity using multidimensional scaling", *ICDM*, Melbourne, Florida, USA, 2003, 339-346.

[14] P. Tzvetkov, X. Yan and J. Han, "TSP: Mining top-K closed sequential patterns," *ICDM*, Melbourne, Florida, USA, 2003, 347-354.

[15] S.-T. Wu, Y. Li, Y. Xu, B. Pham and P. Chen, "Automatic pattern taxonomy exatraction for Web mining," *IEEE/WIC/ACM International Conference on Web Intelligence*, Beijing, China, 2004, 242-248.

[16] G. I. Webb and S. Zhang, "K-optimal rule discovery," *Data Mining and Knowledge Discovery*, 2004, **10:** 39-79.

[17] Y. Xu, and Y. Li, "Mining for useful association rules using the ATMS," *CIMCA2005, Vienna, Austria*, Nov. 2005, 271-276.

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