Summarization of Association Rules in Multi-tier Granule Mining

Yuefeng Li, *Member, IEEE,* Jingtong Wu

***Abstract*—It is a big challenge to find useful associations in databases for user specific needs. The essential issue is how to provide efficient methods for describing meaningful associations and pruning false discoveries or meaningless ones. One major obstacle is the overwhelmingly large volume of discovered patterns. This paper discusses an alternative approach called multi-tier granule mining to improve frequent association mining. Rather than using patterns, it uses granules to represent knowledge implicitly contained in databases. It also uses multi-tier structures and association mappings to represent association rules in terms of granules. Consequently, association rules can be quickly accessed and meaningless association rules can be justified according to the association mappings. Moreover, the proposed structure is also an precise compression of patterns which can restore the original supports. The experimental results shows that the proposed approach is promising.**

***Index Terms*—knowledge discovery in databases, association rule mining, granule mining, pattern mining, decision rules, support restoration.**

I. Introduction

THE association mining consists of two phases: pattern mining and rule generation. Many efficient algorithms have been developed for pattern mining; However, the challenging issue for pattern mining is not efficiency but interpretability, due to the huge number of patterns generated by the mining process [33], [18]. Frequent closed patterns partially alleviate the redundancy problem. Recently, many experiments [29], [36], [13], [16] have proved that frequent closed patterns are good alternative of terms for representing text features. Several approaches for pattern post-processing have also been proposed recently. Pattern compression [30], pattern deploying [29] and pattern summarization [33], [24] were proposed to summarize patterns.

The phase of rule generation is to find interesting rules based on discovered patterns and a minimum confidence, which is also a time consuming activity that can generate many redundant rules. The approaches for pruning redundant rules can be roughly divided into two categories, the subjective based approach and objective approach. The former is to find rules that satisfy some constraints or templates [7], [2]. The later is to construct concise representations of rules without applying user-dependent constraints [35], [31].

There are several obstacles when we consider using association mining in applications: the overwhelmingly large volume of discovered patterns and rules, false discoveries, the lack of semantic information along with the mining process, and the incompleteness of knowledge coverage. Frequent association mining has been extended to multilevel association mining, which uses concept hierarchies or taxonomy trees to find rules [8]. The leaves of a taxonomy tree represent items at the lowest level of abstraction. Using a top-down strategy, at each level, frequent patterns are calculated based on accumulated counts. Recently, mining flipping correlations [1] has been proposed to find positive and negative correlations in taxonomy trees. Another paradigm is the filtered-top-k association discovery [28] which used three parameters: a user specified measure of how potential interesting an association is, filters for discarding inappropriate associations, and k the number of associations to be discovered.

One important finding is that the use of closed patterns can greatly reduce the number of extracted rules; however, a considerable amount of redundancy still remains [32]. Therefore, the size of the set of closed patterns need to be further reduced. The summarization approaches can achieve this purpose. But the summarization approaches are loss methods that they carry errors when restoring the support of original patterns from the compressed patterns. Moreover, both the closed patterns and summarization approaches do not annotate the patterns with semantic information.

Based on our knowledge, currently there are three different approaches for the interpretation of discovered knowledge based on some sorts of semantic annotations: an OLAP based visualization method [17], a generating semantic annotation method [18] and multi-tier structures [15], [14], which used "granules" instead of "patterns" and "rules", and defined meaningless rules based on the relationship between long rules and their general ones (short rules).

In previous research we have found that granules were also a compressed representation. Thus, in this paper, we explore the capability of multi-tier structures for estimating supports for patterns without information loss. This paper proposes the concepts and definitions to illustrate the relationship between patterns and granules. We also presents a method to estimate patterns' support based on granules. A set of experiments has been conducted and the experimental results show that the proposed approach is promising.

The remainder of the paper is structured as follows. Section II discusses related work. Section III and IV introduces basic concepts of granules and the multi-tier structures and describes the basic and derived association mappings. Section V presents the definition of association mappings and discusses their properties. Section VI then presents the support estimation

Y. Li and J. Wu are with the School of Electrical Engineering and Computer Science, Queensland University of Technology, Australia, Brisbane, QLD 4001.
E-mail: y2.li@qut.edu.au, j3.wu@student.qut.edu.au

discussion for pattern and granule based methods. Section VII evaluates the proposed approach and the last section is the conclusion.

II. Related Work

Pattern mining played an important role for the development of association mining. Many efficient algorithms have been developed for pattern mining [6], [9] in transaction databases. Pattern mining has also been developed for mining frequent itemsets in multiple levels [5], [6], and constraint-based techniques [19], [3], [12], [11], [23].

Since these approaches produce a huge volume of patterns, a new major challenging issue for pattern mining is how to present and interpret discovered patterns. Several approaches have been developed for this issue. A concise representation of patterns is a lossless representation, for example, non-derivable patterns [4], condensed patterns [22], maximal patterns, closed patterns, and regular patterns [25]. Pattern post-processing was also presented recently, for example, pattern compression [30], pattern deploying [29], [13] and pattern summarization [33], [27], [10], [24].

A transaction database can be formally described as an information table (T, V^T), where T is the set of transactions, and $V^T = \{a_1, a_2, \ldots, a_n\}$ is the set of items (or called attributes) for all transactions in T.

Let α be an *itemset*, a subset of V^T. Its *coverset* is the set of all transactions (or objects) $t \in T$ such that $\alpha \subseteq t$, and its support is $\frac{|coverset(\alpha)|}{|T|}$. An itemset α is called *frequent pattern* if its support $\geq min\_sup$, a minimum support. Given a set of transactions (objects) Y, its *itemset* denotes the set of items (attributes) that appear in all the objects of Y. For a pattern α, its closure $closure(\alpha) = itemset(coverset(\alpha))$.

A pattern α is *closed* if and only if $\alpha = closure(\alpha)$. Closed patterns can be summarized into pattern profiles [33] by clustering the patterns with respect to KL-divergence , and a pattern's support can be estimated by using pattern profiles.

Let $T' = \bigcup_{1 \leq i \leq m} T_{\alpha_i}$, where T_{α_i} is the coverset of pattern α_i. A profile M is a triple $\langle pr, \phi, \rho \rangle$, where pr is a probability distribution vector of the items in this profile; ϕ is called master pattern which is the union of a set of patterns (α_1, α_2, ... , α_m); and ρ is the support of the profile which equals to $\frac{|T'|}{|T|}$.

The profile based summarization can largely reduce the pattern number, however, it has following limitations. Firstly, a pattern is possibly covered by multiple profiles. Secondly, it is lack of error guarantee in the support estimation. To achieve a result with less error, a greater number of profiles is required that can reduce the performance of pattern summarization. Finally, the estimation sometimes falsely mark some infrequent patterns as frequent ones, or vise versa.

The concepts of decision rules and granules are well acceptable in the rough set community [20]. Rough set theory has been developed to deal with vagueness for reasoning precisely about approximations of vague concepts. Decision rules have been used for rule-based classification [26], and the construction of decision trees and flow graphs [21].

The advantage of using decision rules is to reduce the two-phases of association mining (pattern mining and rule generation) into one process. In this research, we develop granule mining into multi-tier granule mining in order to identify meaningless rules and efficiently access association rules for user specific needs.

TABLE I
An information table

Object(*Transaction*)	*Items* (*Attributes*)
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 II
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

III. Decision Table and Two-Tier Structure

In the multi-tier granule mining, the information table is firstly compressed into a decision table for a selected set of attributes by using the Group By operation. The decision table is then represented into a two-tier structure based on a partition of attributes, which classifies the set of attributes into condition attributes and decision attributes, and describes the associations between condition granules and decision granules. The two-tier structure can be further derived into different multi-tier structures to summarize all possible associations between granules based user selected attributes and tiers.

Formally, the decision table of a information table (T, V^T) is denoted as a tuple of (T, V^T, C, D) if $C \cap D = \emptyset$ and $C \cup D \subseteq V^T$. C and D are two groups of attributes which are conditions and decision attributes respectively.

Usually, it is assumed (see [21]) that there is a function for every attribute $a \in V^T$ such that $a : T \rightarrow V_a$, where V_a is the set of all values of a. We call V_a the domain of a. Let B be a subset of V^T. B determines a binary relation $I(B)$ on T such that $(t_1, t_2) \in I(B)$ if and only if $a(t_1) = a(t_2)$ for all $a \in B$, where $a(t)$ denotes the value of attribute a for object $t \in T$. It is easy to prove that $I(B)$ is an equivalence relation, and the family of all equivalence classes of $I(B)$ is denoted by $U = T/B$. We call each equivalence class in U a *granule*. The granule in U that contains transaction t is denoted by $B(t)$. Let $U_C = T/C$ and $U_D = T/D$, granules in U_C or U_D are also referred to *C-granules* or *D-granules*, respectively.

Table I list out a sample transaction table, where $V^T = \{a_1, a_2, ..., a_7\}$ and $T = \{t_1, t_2, ..., t_6\}$. Let a_1 to a_5 be the

TABLE III
C-Granules

Condition Granule	a_1	a_2	a_3	a_4	a_5	*coverset*
cg_1	1	1	0	0	0	$\{t_1, t_5, t_6\}$
cg_2	0	0	1	1	0	$\{t_2\}$
cg_3	0	0	1	1	1	$\{t_3, t_4\}$

TABLE IV
D-Granules

Decision Granule	a_6	a_7	*coverset*
dg_1	0	0	$\{t_1\}$
dg_2	1	0	$\{t_2, t_3, t_4\}$
dg_3	1	1	$\{t_5, t_6\}$

condition attributes and a_6, a_7 be the decision attributes, then table I can be grouped by V^T into a decision table as shown in table II, where $T/C \cup D$ = $\{g_1, g_2, g_3, g_4\}$. Based on this definition, we also have the condition and decision granules as listed out in table III and IV.

In this paper, a relation R_B between U and T is used to describe the relationships between granules and transactions in formal concept analysis [34]. That is, given a transaction $t \in T$ and a granule $g \in U$, we say g is induced by t or t has the property g if $g = B(t)$ (also written as tR_Bg).

Let B be a subset of V^T and $U = T/B$, and granule $g \in U$ be induced by transaction t. Its covering set $coverset(g) = \{t' | t' \in T, t'R_Bg\}$. Let granule $g = cg \wedge dg$, where cg is a *C-granule* and dg is a *D-granule*. We can easily prove that $coverset(g) = coverset(cg) \cap coverset(dg)$. Table III and IV also list out the coversets for the sample *C-granule* and *D-granule*.

The smallest granules only contain one single attribute, we also call them primary granules. A large granule can be generated from some smaller granules by using logic operation "and", $\wedge$. Every granule in the decision table can be mapped into an *association rule* (or called *decision rule*), where the antecedent is a *C-granule* which consists of attributes in C, and the consequent is a *D-granule* which consists of attributes in D. The decision rules can also be regarded as larger granules generated by the condition and decision granules. For instance, the granules g_1, g_2, g_3 and g_4 shown in table II can be generated by the *C-granules* and *D-granules* as follows:

$g_1 = cg_1 \wedge dg_1$;
$g_2 = cg_2 \wedge dg_2$;
$g_3 = cg_3 \wedge dg_2$;
$g_4 = cg_1 \wedge dg_3$.

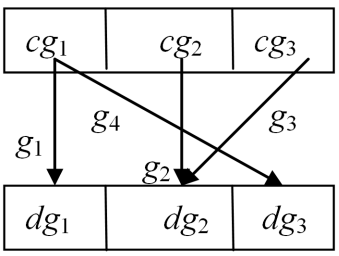

Fig. 1. A 2-tier structure

With these definitions of condition and decision granules as well as the relations between them, a 2-tier structure can be built. Fig. 1 illustrates a 2-tier structure to describe the relationship between these granules in Table II, III and IV. The links (arrows) also represent the associations (decision rules) between condition granules and decision granules. Based on the 2-tier structure, varieties of multi-tier structures and mappings can be derived. The details will be discussed in the following two sections.

IV. MULTI-TIERS STRUCTURE

In this section, we first discuss the concept of multi-tier structures. We also define the concept of general rules (i.e., rules with shorter antecedents) of decision rules in order to clarify the meaning of meaningless in granule mining. At last, we present the method to estimate patterns' support based on granules.

To describe more associations between granules, we can further divide the condition attributes into some categories in accordance with what users want. For example, let C_i and C_j be two subsets of C, which satisfy $C_i \cap C_j = \emptyset$ and $C_i \cup C_j = C$, hence a *C-granule* cg can be divided into a C_i granule cg_i and C_j granule cg_j and have $cg = cg_i \wedge cg_j$.

A multi-tier structure can be describes as a pair (H, A), where *H* is a set of granule tiers and *A* is a set of association mappings that illustrate the associations between granules in different tiers.

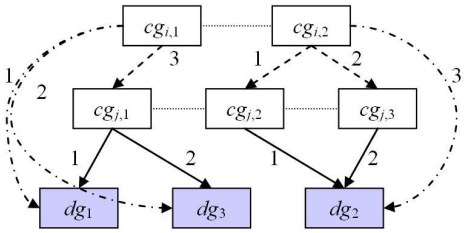

Fig. 2. An example of a multi-tier structure

Fig. 2 illustrates a 3-tier structure, where *C-granules* are divided into C_i*-granules* and C_j*-granules* (i.e., the first two levels in the figure), and we have $H = \{C_i, C_j, D\}$. The C_i tier includes C_i*-granules* = $\{cg_{i,1}, cg_{i,2}, \ldots, cg_{i,k}\}$, the C_j tier includes C_j*-granules* = $\{cg_{j,1}, cg_{j,2}, \ldots, cg_{j,r}\}$, and the D tier includes *D-granules* = $\{dg_1, dg_2, \ldots, dg_v\}$, where $k = 2$, $r = 3$ and $v = 3$.

The 3-tier structure in Fig. 2 includes three association mappings (arrows), Γ_{cd}, Γ_{ij}, and Γ_{id} (i.e., $A = \{\Gamma_{cd}, \Gamma_{ij}, \Gamma_{id}\}$), which show the linkages between *C-granules* and *D-granules* (e.g., the solid arrows), C_i*-granules* and C_j*-granules*, and C_i*-granules* and *D-granules*, respectively. These association mappings can be used to generate association rules.

Given a *C-granule* cg_k and a C_i*-granule* $cg_{i,x}$, $\Gamma_{cd}(cg_k)$ includes all possible associations (links and their strengths) between cg_k and *D-granules*; $\Gamma_{ij}(cg_{i,x})$ includes all possible associations between $cg_{i,x}$ and C_j*-granules*; and $\Gamma_{id}(cg_{i,x})$ includes all possible associations between $cg_{i,x}$ and *D-granules*.

The *link strength* between granule cg_k and granule dg_z is defined as

$$lstrength(cg_k, dg_z) = |coverset(cg_k \wedge dg_z)|$$

which is the number of transactions that have the property "$cg_k \wedge dg_z$".

As defined above, the rule "$cg_k \rightarrow dg_z$" is a decision rule (or association rule), where cg_k is its antecedent and dg_z is its consequent (note the following concepts are also applicable for "$cg_{i,x} \rightarrow dg_z$"). Its *support* is

$$\frac{|coverset(cg_k \wedge dg_z)|}{|T|} = \frac{1}{N} lstrength(cg_k, dg_z)$$

and its *confidence* is

$$\frac{|coverset(cg_k \wedge dg_z)|}{|coverset(cg_k)|} = \frac{lstrength(cg_k, dg_z)}{|coverset(cg_k)|} \quad (1)$$

where $N = |T|$, the total number of transactions.

Different to decision tables, we can discuss general association rules (rules with shorter premises) of decision rules in a multi-tier structure.

Let cg_k be a *C-granule*, dg_z be a *D-granule* and $cg_k = cg_{i,x} \wedge cg_{j,y}$. We call "$cg_{i,x} \rightarrow dg_z$" (or "$cg_{i,y} \rightarrow dg_z$") a *general rule* of rule "$cg_k \rightarrow dg_z$".

Especially in the multi-tier structure, we can define the term "meaningless" for a decision rule based on selected tiers. We call "$cg_k \rightarrow dg_z$" *meaningless* if its confidence is less than or equal to the confidence of its a general rule.

The rationale of this definition is analogous to the definition of interesting association rules, where $\alpha \rightarrow \beta$ is an interesting rule if $P(\beta|\alpha)$ (conditional probability) is greater than $P(\beta)$. If we add a piece of extra evidence to a premise and obtain a weak conclusion, we can say the piece of evidence is meaningless.

V. ASSOCIATION MAPPINGS

In the last section, we discussed a three tiers structure (H, A), where $H = \{C_i, C_j, D\}$, $C_i \cup C_j = C$ and $C_i \cap C_j = \emptyset$, and $A = \{\Gamma_{cd}, \Gamma_{ij}, \Gamma_{id}\}$. Association mappings are used to describe the association relationships between granules in different tiers. They can be used to enumerate all association rules between the associated granules. Usually, there are many possible pairs (C_i, C_j) such that $C_i \cup C_j = C$ and $C_i \cap C_j = \emptyset$, and C_i and C_j can be further divided into smaller sets. Therefore, it is necessary using derived association mappings (e.g., Γ_{id}) for efficient rule generations in multi-tier structures.

A. Basic Association Mapping

The basic association mapping is the mapping between granules from two tiers. For example, the mappings between the condition and decision granules are basic mappings. As the previous definitions, let $U = T/V_T$, $U_C = T/C$ and $U_D = T/D$ to be the set of granules, condition and decision granules. Also let $g_1 \in U_C$ and let $g_2 \in U_D$. Then based on Eq.(1) and Section IV, we have

$$\begin{aligned} lstrength(g_1, g_2) &= |coverset(g_1 \wedge g_2)| \\ &= |coverset(g_1) \cap coverset(g_2)| \\ &= |\{t \in T | tR_C g_1 \ and \ tR_D g_2\}|. \end{aligned}$$

The basic associations between *C-granules* and *D-granules* can be described as a basic association mapping Γ_{cd} such that $\Gamma_{cd}(g)$ is a set of *D-granule* link-strength pairs for all $g \in U_C$. Formally, Γ_{cd} is defined as $\Gamma_{cd} :: U_C \rightarrow 2^{U_D \times I}$, which satisfies

$$\begin{aligned} \Gamma_{cd}(g) = \ & \{(dg, lstrength(g, dg)) | dg \in U_D, \\ & \{t \in T | tR_C g \ and \ tR_D dg\} \neq \emptyset\} \end{aligned}$$

for all granules $g \in U_C$, where I is the set of all integers.

Obviously, supports and confidences of association rules can be easily calculated based on the basic association mapping. Let $g_1 \in U_C$, $g_2 \in U_D$, and "$g_1 \rightarrow g_2$" be a decision rule, its support and confidence can be derived as follows:

$$\begin{aligned} sup(g_1 \rightarrow g_2) &= \frac{1}{N} lstrength(g_1, g_2) \\ &= \frac{1}{N} \sum_{(g_2, ls) \in \Gamma_{cd}(g_1)} ls \end{aligned}$$

$$\begin{aligned} conf(g_1 \rightarrow g_2) &= \frac{lstrength(g_1, g_2)}{|coverset(g_1)|} \\ &= \frac{\sum_{(g_2, ls) \in \Gamma_{cd}(g_1)} ls}{\sum_{(g, ls) \in \Gamma_{cd}(g_1)} ls} \end{aligned}$$

B. Derived Association Mappings

The very interesting property of the multi-tier structures is that we can derive many association mappings based on the basic association mapping rather than using the original set of transactions. This property is significant on time complexities for rule generations.

To simplify the process of deriving, we first consider the method for deriving association mapping Γ_{ij} between C_i-*granules* and C_j-*granules* based on the basic association Γ_{cd}, where $\Gamma_{ij}(g)$ is a set of C_j-*granule* integer pairs, which satisfies

$$\Gamma_{ij} :: U_i \rightarrow 2^{U_j \times I}$$

and

$$\begin{aligned} \Gamma_{ij}(g_i) = \ & \{(g_j, lstrength(g_i, g_j)) | g_j \in U_j, \\ & \{t \in T | tR_i g_i \ and \ tR_j g_j\} \neq \emptyset\} \end{aligned}$$

for all granules $g_i \in U_i$, where $C_i \cup C_j = C$, $C_i \cap C_j = \emptyset$, $U_i = T/C_i$ (the set of C_i-*granules*), $U_j = T/C_j$ (the set of C_j-*granules*), and R_i and R_j are relations between U_i and T, and U_j and T, respectively.

We can also derive the association mapping Γ_{id} between C_i-*granules* and *D-granules* based on the association mappings Γ_{ij} and Γ_{cd}, which satisfies

$$\Gamma_{id} :: U_i \rightarrow 2^{U_D \times I}$$

and

$$\begin{aligned} \Gamma_{id}(g_i) = \ & \{(dg, lstrength(g_i, dg)) | dg \in U_D, \\ & \{t \in T | tR_i g_i \ and \ tR_D dg\} \neq \emptyset\} \end{aligned}$$

for all granules $g_i \in U_i$.

Fig. 3 illustrates the relations between these association mappings. In this figure, the set of condition attributes are split into two sets: C_i and C_j, and the *C-granules* (U_C) are also correspondingly compressed into C_i-*granules* (U_i) and C_j-*granules* (U_j). As defined before, Γ_{cd} is used to describe the association relationship between U_C and U_D. Association mapping Γ_{ij} is used to describe the association relationship between U_i and U_j, and association mapping Γ_{id} is used to describe the association relationship between U_i and U_D.

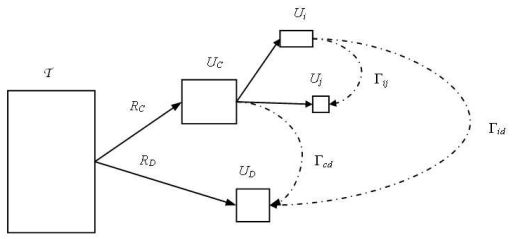

Fig. 3. Relations for derived association mappings

The relationship between the basic mapping and derived mappings can be defined by the following definitions:

Let $C \subseteq B \subseteq V_T$, then the relationship between the granules in U_C = T/C and granules in U_B = T/B can also be defined. A granule $g \in U_C$ is called a $generalized\ granule$ of granule g' $\in U_B$ if $\forall t \in coverset(g') \Rightarrow tR_C g$ (i.e., $coverset(g') \subseteq coverset(g)$). This is denoted as $g' \succcurlyeq g$ for the generalized relationship between g' and g.

Then for all $g \in U_C$, the relation between the coverset of g and its generalized granule g' is formally denoted as the following equation:

$$coverset(g) = \bigcup_{\{g' \in U_B | g' \succcurlyeq g\}} coverset(g') \tag{2}$$

VI. Support Estimation

The support estimation is originally proposed to provide a method to restore the support for the patterns that summarized into limited number of profiles or compressed representation. Given a pattern, usually its support can not be obtained directly from the profiles or compression. Thus, the support of the given pattern only can be estimated through the corresponding restore calculation using the information stored in the profiles or representatives. Moreover, because the profiles are loss summarization, a measure called restoration error is used to examine the precision of the estimated support. This measure is also applied to the estimated support calculated through granules.

A. Support estimation for summarization

After the closed frequent patterns are summarized into the profiles, the support for a pattern needs to be retrieved through the calculation from the profile information. Because one pattern can be covered by multiple profiles, then the maximum result is selected as estimated support. Formally, for a given pattern α_k, its estimated support can be calculated as follows:

$$\hat{s}(\alpha_k) = max_M(\rho_M \times \prod_{a_i \in \alpha_k} pr_M(a_i = 1)) \tag{3}$$

which selects the maximum one from all profiles M that include α_k.

One method to measure the accuracy of the estimated support is to measure the average relative error between the estimated support and the original support. Formally, given a summarization or compression of the original patterns and a set of testing pattern set T = $\{\alpha_1, \alpha_2, ..., \alpha_l\}$, the quality of this summarization or compression can be evaluated through the average relative error, called restoration error denoted as J, defined as follow:

$$J = \frac{1}{|T|} \sum_{\alpha_k \in T} \frac{|s(\alpha_k) - \hat{s}(\alpha_k)|}{s(\alpha_k)} \tag{4}$$

where T = $\{\alpha_1, \alpha_2, ... , \alpha_l\}$ is a given test set of patterns. $s(\alpha_k)$ is the real support of the pattern α_k, while the $\hat{s}(\alpha_k)$ is the estimated support calculated by the pattern profiles or granules.

In the actual calculation, the original pattern sets can be used as the testing set so that the restoration error measures difference between the real support and the estimated support. The smaller the error rate is, the closer is the estimated support to the actual support. It is obvious that if the restoration error is zero, the estimated support equals to the actual support.

B. Support estimation for granules

In terms of multi-tier structure of granules, the estimated support is calculated through the granules and association mappings. The estimated support can be calculated by the granule support if the given pattern is derived by the granule or the support can be calculated through the link strength of the association mappings between granules that containing the pattern. In some circumstance, the estimated support calculated through the multi-tier structure can achieve a zero restoration error rate.

There are several different calculation to obtained the estimated support from the multi-tier structure of granules according to what is the definition of the current multi-tier structure and which tiers of granule are containing the given pattern.

The first case is to estimate the support for a pattern with a decision table. Let G be the decision table of information table(T, V^T), then the estimated support is calculated solely through sum of support of the granules containing the pattern. The equation to calculate the estimated support for a given pattern α is as follow:

$$\hat{s}_1(\alpha, G) = \frac{\sum_{g \in G, \alpha \subseteq g} \sup(g)}{\sum_{g_i \in G} \sup(g_i)} = \frac{\sum_{g \in G, \alpha \subseteq g} \sup(g)}{|T|}$$

The second case is calculating the estimated support with a two tier structure. For a 2-tier structure, let CG be the set of C-granules and DG be the set of D-granules. Given a pattern α, it can be divided into two patterns α_1 and α_2 such that α_1 = $\alpha \cap C$ and α_2 = $\alpha \cap D$, respectively. Then the estimation support is calculated as the summary of granules support if pattern α only contained by the granules in one tier. Or it can be calculated through the link strength of the association mappings between the granules that containing α_1 and α_2. The

equations for the estimated support calculation are as follow:

$$\hat{s}_2(\alpha, CG, DG) =$$

$$\begin{cases} \frac{1}{|T|} \sum\limits_{g \in CG, \alpha_1 \subseteq g} \sup(g) = \hat{s}_1(\alpha_1, CG) & \text{if } \alpha_2 = \emptyset \\ \frac{1}{|T|} \sum\limits_{g \in DG, \alpha_2 \subseteq g} \sup(g) = \hat{s}_1(\alpha_2, DG) & \text{if } \alpha_1 = \emptyset \\ \frac{1}{|T|} \sum\limits_{\alpha_1 \subseteq g_1 \in CG, \alpha_2 \subseteq g_2 \in DG} \textit{lstrengh}(g_1 \rightarrow g_2) & \text{otherwise} \end{cases} \quad (5)$$

For other cases with the n-tier structures, the estimated support can be calculated by different methods depending on how many tiers of granules that the given pattern is derived from. There are three categories of the calculation method for the estimated support in the multi-tier structure. The first case is that the given pattern is only contained by granules in only one tier, then the support can be calculated by using the supports of the granules in the corresponding tier. The second case is for the patterns which are contained by the granules of two tiers in the multi-tier structure. The calculation for the support in such cases can use the link strength of the mappings between the two granules to obtain the support. This calculation is done directly in the current multi-tier structure. The third method is for the patterns that are contained by the granules from three or more tiers. To get the support for such patterns, the calculation needs to use the mapping informations from the 2-tier structure to compute the support through Eq.(5).

To demonstrate the estimated support calculation from the multi-tier structure, here uses a 3-tier structure as an example to illustrate the calculation details. Let a 3-tier structure be $H = \{C_i, C_j, D\}$ containing three sets of granule that are C_iG, C_jG and DG respectively. Then a pattern α can be divided into three patterns, namely $\alpha_1 = \alpha \cap C_iG$, $\alpha_2 = \alpha \cap C_jG$ and $\alpha_3 = \alpha \cap DG$. If only one of α_1, α_2 or α_3 is non-empty, then the support is calculated through the sum of support of only one set of granules as follow:

$$\hat{s}_3(\alpha, C_iG, C_jG, DG) =$$

$$\begin{cases} \frac{1}{|T|} \sum\limits_{g \in C_iG, \alpha_1 \subseteq g} \sup(g) = \hat{s}_1(\alpha_1, C_iG) \text{ if } \alpha_2, \alpha_3 = \emptyset \\ \frac{1}{|T|} \sum\limits_{g \in C_jG, \alpha_2 \subseteq g} \sup(g) = \hat{s}_1(\alpha_2, C_jG) \text{if } \alpha_1, \alpha_3 = \emptyset \\ \frac{1}{|T|} \sum\limits_{g \in DG, \alpha_3 \subseteq g} \sup(g) = \hat{s}_1(\alpha_3, DG) \text{ if } \alpha_1, \alpha_2 = \emptyset \end{cases} \quad (6)$$

For the second case, that is one of α_1, α_2 and α_3 is empty, then the support is calculated using the link strength of the mappings of $\Gamma_{i,j}$, $\Gamma_{i,d}$ or $\Gamma_{j,d}$ as follow:

$$\hat{s}_3(\alpha, C_iG, C_jG, DG) =$$

$$\begin{cases} \frac{1}{|T|} \sum\limits_{\alpha_1 \subseteq g_1 \in C_iG, \alpha_2 \subseteq g_2 \in C_jG} \textit{lstrengh}(g_1 \rightarrow g_2) \text{if } \alpha_3 = \emptyset \\ \frac{1}{|T|} \sum\limits_{\alpha_1 \subseteq g_1 \in C_iG, \alpha_3 \subseteq g_3 \in DG} \textit{lstrengh}(g_1 \rightarrow g_3) \text{if } \alpha_2 = \emptyset \\ \frac{1}{|T|} \sum\limits_{\alpha_2 \subseteq g_2 \in C_jG, \alpha_3 \subseteq g_3 \in DG} \textit{lstrengh}(g_2 \rightarrow g_3) \text{if } \alpha_1 = \emptyset \end{cases} \quad (7)$$

Finally, if none of α_1, α_2 or α_3 is empty, then it is a case of the third category. Then the support is calculated through the mappings of $\Gamma_{c,d}$. In order to use these mappings, the division of α need to be modified. That is, let $\alpha_1 \cup \alpha_2 = \alpha \cap CG$ where $CG = C_i \cup C_j$ such that the support can be obtained by using a modified equation of Eq.(5). The equation used for this calculation is as follow:

$$\begin{aligned} &\hat{s}_3(\alpha, C_iG, C_jG, DG) = \\ &\frac{1}{|T|} \sum_{Cond} \textit{lstrengh}((g_1 \wedge g_2) \rightarrow g_3) \\ &Cond : \alpha_1 \subset g_1 \in C_iG, \alpha_2 \subset g_2 \in C_jG, \\ &\qquad C_iG \cup C_jG = CG, \alpha_3 \subseteq g_3 \in DG \end{aligned} \quad (8)$$

Regarding the quality of the estimation, when using the two tier structure to calculate the estimated support, it can achieve the zero restoration error rate because the two tier structure is a lossless compression. Further, for the multi-tier structure has more than two tiers, it also can achieve the zero error rate when using only the mappings of granules from two tiers or the calculation is performed via the basic 2-tier structure.

Theorem 1: For a given pattern α and a multi-tier structure $H = \{C, D\}$, the estimated support calculated through H equals to the original support of α. That is, $\hat{s}(\alpha, H) = Sup_\alpha$.

Proof: For a pattern α, let $\alpha = \alpha_1 \cup \alpha_2$ such that $\alpha_1 \cap \alpha_2 = \emptyset$. Then for the support of α, we have

$$Sup_\alpha = |coverset(\alpha_1) \cap coverset(\alpha_2)|.$$

Assume $\alpha_1 \subset g_1$ and $\alpha_2 \subset g_2$, and $g_1 \in CG$ and $g_2 \in DG$. According to Eq.(2), we have:

$$coverset(\alpha_1) = \bigcup_{g_{1,i} \in CG} coverset(g_{1,i})$$

and

$$coverset(\alpha_2) = \bigcup_{g_{2,i} \in CG} coverset(g_{2,i}).$$

Meanwhile, the link strength of the mapping from g_1 to g_2 is:

$$\begin{aligned} lstrength(g_1 \rightarrow g_2) &= |coverset(g_1 \wedge g_2)| \\ &= |coverset(g_1) \cap coverset(g_2)|. \end{aligned}$$

Moreover, we have

$$\begin{aligned} &\sum_{g_1 \in CG, g_2 \in DG} lstrength(g_1 \rightarrow g_2) \\ &= |\bigcup_{g_1 \in CG, g_2 DG} (coverset(g_1) \cap (coverset(g_2))| \\ &= |\bigcup_{g_{1,i} \in CG} coverset(g_{1,i}) \cap \bigcup_{g_{2,i} \in CG} coverset(g_{2,i})| \\ &= |coverset(\alpha_1) \cap coverset(\alpha_2)| \\ &= Sup_\alpha. \end{aligned}$$

Therefore, we have $Sup_\alpha = \hat{s}(\alpha, H)$. ■

VII. Experiments and Discussion

Foodmart 2005 data collection contains two databases: SQL Database and OLAP Database. The data used in this experiment is the customer sales data from the OLAP database (see *http://www.e-tservice.com/*), which includes four data cubes. The Warehouse and Sales cube is used in our experiments, which contains four measures and we used the unit-sales measure. The Product dimension used in the Warehouse and Sales cube, consists of eight levels which are All, Product family,

TABLE V
THE TIERS AND THEIR ATTRIBUTES

Levels	Attributes			
2-tiers	C			D
3-tiers	C_i		C_j	D
	Drink	Food1	non-consumable	Food2
4-tiers	$C_{i,1}$	$C_{i,2}$	C_j	D
	$A_1..A_4$	$A_5..A_{11}$	$A_{12}..A_{16}$	$A_{17}..A_{23}$

Product department, Product category, Product subcategory, Brand and Product. In the experiments, we only use the top three levels: All, Product family, and Product department.

There are total 23 attributes in the *Product Department* level. These attributes are categorized into 4 product families: *Drink* (Alcoholic Beverages, Baking Goods, Beverages, Dairy), *Non-Consumable* (Carousel, Checkout, Health and Hygiene, Household, Periodicals), *Food* 1 (Baked Goods, Breakfast Foods, Canned Foods, Canned Products, Deli, Eggs, Frozen Foods) and *Food* 2 (Meat, Packaged Foods, Produce, Seafood, Snack Foods, Snacks, Starchy Foods).

The transactions used in the experiments are the customers' purchase records stored in the fact table of unit sales. Every transaction is the record of one day purchase of one customer for all products which is sum up to product categories. To build up the decision table and multi-tier structures of granules, the transactions of the Unit sales are transformed into an information table using the following procedure. If the customer purchases one or more products from that product department, the value of the attribute in the product department level is set to 1; otherwise, the value is set to 0. The total number of transactions in the information table is 53,700.

The experiments test the proposed solution from several aspects, including space and time complexities, and the restoration error rate of estimated support. We use two baseline models to compare with the proposed theory. The first baseline model is the decision table. The attributes are viewed as two groups: condition and decision attributes. The second baseline model is a pattern summarization model [33], which used pattern profiles to estimate the support of any pattern (see Eq.(3) and (4)).

1) Space and time complexity: In the experiments, the information table is transformed into a decision table first. Multi-tier structures are then constructed based on this decision table and the semantic information of attributes. Table V shows a special definition of the multi-tier structures, where the semantic relation between attributes are considered. As in Table V, there are three multi-tier structures: a two-tier structure (C and D), a three-tier structure (C_i, C_j and D), and a four-tier structure ($C_{i,1}$, $C_{i,2}$, C_j and D).

We also made other 16 definitions of multi-tier structures by grouping the 23 attitudes in different combinations. For each definition, a 2-tier structure (C and D) is built firstly. Then from it a 3-tier structure is built by dividing the C tier into two smaller C_i and C_j. Then the C_i tier is further divided into tier $C_{i,1}$ and $C_{i,2}$ to generate a 4-tier structure (C_1, C_2, C_j and D).

Fig 4 depicts the trends of total granule numbers in the multi-tier structures when the number of tiers increases. It is obvious that in most of the test, the number of granules drops largely with the tier increases. Table VI shows the number of patterns in the information table based on different minimum support values.

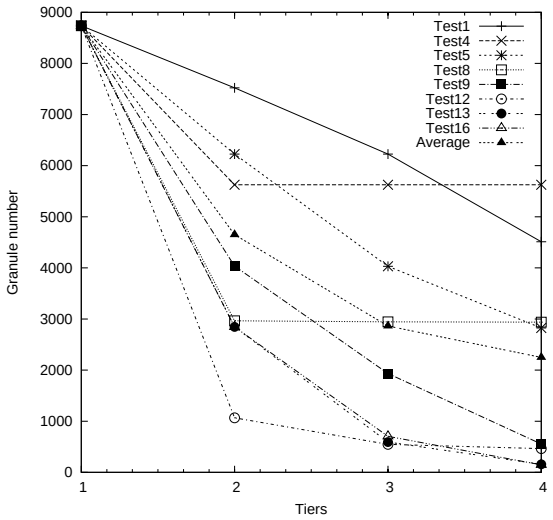

Fig. 4. Granule numbers under different tier settings

TABLE VI
FREQUENT PATTERN NUMBER

Pattern type	$min\_support$	Number of patterns
Frequent pattern	1	56707
Closed pattern	1	15859
Frequent pattern	5	12217
Closed pattern	5	10963
Frequent pattern	50	1486
Closed pattern	50	1486

Comparing with the multi-tier structures, pattern mining gets a large amount patterns if the $min\_support$ is not big enough. However, when the $min\_support$ is big enough (e.g., 50 in this example), pattern mining will lose many large patterns. Different from the pattern mining, multi-tier structures can use a very small space to contain all the possible associations for the chosen data attributes.

Table VII shows the results of the runtime tests. It is obviously that the time used by multi-tier structures is much less than that of pattern mining. Only when the minimum support is set to a very large number of occurrence, the time to obtain the frequent patterns looks acceptable. Fig 5 also obviously shows these differences between the two approaches.

The results also reflect that the time used to create new tiers from smaller granules is less than that from larger granules.

TABLE VII
RUNTIME

Granule		Pattern	
Multi-tier structure	Time(ms)	min_sup	Time(ms)
Decision table	19140	1	1.078e+007
2-tier	6765	5	1.131e+006
3 tier from 2 tier	2593	50	122672
4 tier from 3 tier	171		

For example, the time used to generate a four-tier structure from a three-tier structure is 171 ms, while it takes 2593 ms for constructing the three-tier structure from a two-tier structure. These results show that the proposed theory has achieved the remarkable performance.

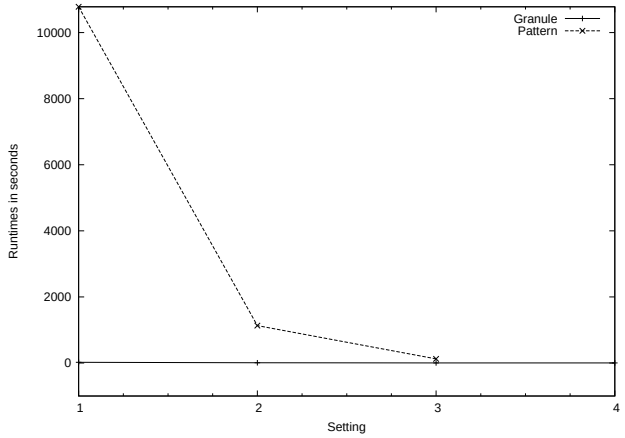

Fig. 5. Runtime of granule and pattern approach

2) Restoration error rate and meaningless rules: The pattern summarization model uses all closed patterns, which are generated from the whole information table with a minimum support of 5, as the input patterns. There are 10963 closed patterns in total. The restoration error rate J is calculated by using Eq.(4). Several tests are carried out with the different number of profiles. The number of profiles is set to 200, 500, 750 and 1000 respectively. Fig 6 shows the results for the error rate of the pattern summarization model vs. granule mining. The results reflect that when using small number of profiles such as 200, 500 and 750, the restoration error is much higher than using granules. To be noticed, using a two tier structure to calculate the support for all patterns (see Eq.(5), (6), (7) and (8)), the J values can remain as zero. This result proves the discussion in section VI-B that the support estimated by the granules in a two tier structure equals to the pattern's original support.

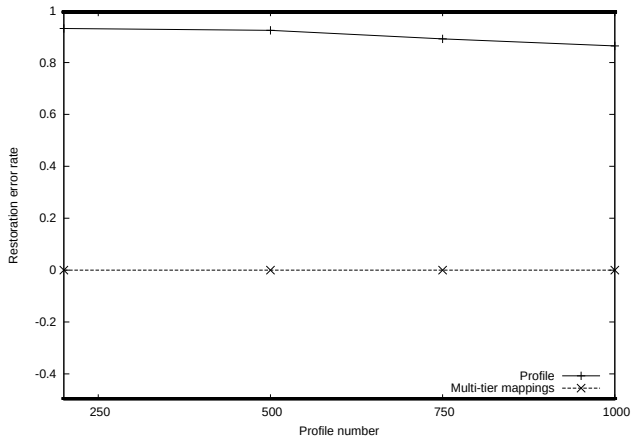

Fig. 6. Estimated support error rate

The multi-tier structures also provide a special feature for pruning some meaningless rules. Based on the definitions, in these experiments, we generate general rules first for the 16 definitions of multi-tier structures. We then filter out the meaningless rules based on their general rules. We found that the rules contain about 30% meaningless rules in average.

VIII. Conclusion

Multi-tier granule mining provide an efficient way to represent and summarize association rules between granules based user selected attributes and tiers. This paper continues the development of multi-tier structures. It presents formalizes concepts of association mappings and a method to 'estimate patterns' support based on related granules and the multi-tier structures. Moreover, it conducts a set of experiments on Foodmart 2005 data collection to test the proposed method. Compared with pattern summarization, the proposed multi-tier granule mining achieves the best performance with zero restoration error rate. The experimental results also show that the multi-tier structures can use a very small space to store the possible associations, and the multi-tier structures can be created efficiently. This research provides a promising alternative approach to find useful associations in databases for user specific needs.

References

[1] M. Barsky, S. Kim, T. Weninger, and J. Han. Mining flipping correlations from large datasets with taxonomies. In *Proceedings of the VLDB endowment, Vol. 5, No. 4*, pages 370–381, 2011.

[2] R. Bayardo, R. Agrawal, and D. Gunopulos. Constraint-based rule mining in large, dense databases. *Data Mining and Knowledge Discovery*, 4:217–240, 2000.

[3] C. Bucila, J. Gehrke, D. Kifer, and W. White. Dualminer: a dual-pruning algorithm for itemsets with constraints. In *Proc. of KDD'02*, 42–51, 2002.

[4] T. Calders and B. Goethals. Mining all non-derivable frequent itemsets. In *Proc. of 2002 PKDD*, 74–85, 2002.

[5] J. Han and Y. Fu. Discovery of multiple-level association rules from large databases. In *Proc. of VLDB'95*, 420–431, 1995.

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

[7] J. Han, L. V. Lakshmanan, and R. T. Ng. Constraint-based multidimensional data mining. *IEEE Computer*, 32(8):46–50, 1999.

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

[9] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proc. of SIGMOD'00*, 1–12, 2000.

[10] R. Jin, M. Abu-Ata, Y. Xiang, and N. Ruan. Effective and efficient itemset pattern summarization: Regression-based approaches. In *Proc. of KDD-08*, 399–407, 2008.

[11] A. J. T. Lee, W. Lin, and C. Wang. Mining association rules with multi-dimensional constraints. *Journal of Systems and Software*, 79(1):79–92, 2006.

[12] C. K.-S. Leung, L. V. S. Lakshmanan, and R. T. Ng. Exploiting succinct constraints using fp-trees. *SIGKDD Explorations*, 4(1):40–49, 2002.

[13] Y. Li, A. Algarni, and N. Zhong. Mining positive and negative patterns for relevance feature discovery. In *Proceeding of KDD'10*, pp. 753–762, 2010.

[14] Y. Li, W. Yang, and Y. Xu. Multi-tier granule mining for representations of multidimensional association rules. In *Proceedings of ICDM'06*, pp. 953–958, 2006.

[15] Y. Li and N. Zhong. Interpretations of association rules by granular computing. *ICDM-03*, pages 593–596, 2003.

[16] Y. Li, X. Zhou, P. Bruza, Y. Xu, and R. Y. Lau. A Two-Stage Decision Model for Information Filtering. *Decision Support Systems*, 52(3):706–716, 2012.

[17] B. Liu, K. Zhao, J. Benkler, and W. Xiao. Rule interestingness analysis using olap operations. In *Proceedings of KDD'06*, pp. 297–306, 2006.

[18] Q. Mei, D. Xin, H. Cheng, J. Han, and C. Zhai. Generating semantic annotations for frequent patterns with context analysis. In *Proc. of KDD'06*, 337–346, 2006.
[19] R. T. Ng, L. V. S. Lakshmanan, J. Han, and A. Pang. Discovery of multiple-level association rules from large databases. In *Proc. of SIGMOD'98*, 13–24, 1998.
[20] Z. Pawlak. In pursuit of patterns in data reasoning from data - the rough set way. *RSCTC-02*, 1–9, 2002.
[21] Z. Pawlak. Decision trees and flow graphs. In *Proceedings of RSCTC'06*, pp. 1–11, 2006.
[22] J. Pei, G. Dong, W. Zou, and J. Han. On computing condensed frequent pattern bases. In *Proceedings of ICDM'02*, pp. 378–385, 2002.
[23] J. Pei and J. Han. Constrained frequent pattern mining: a pattern-growth view. *SIGKDD Explorations*, 4(1):31–39, 2002.
[24] A. K. Poernomo and V. Gopalkrishnan. Cp-summary: a concise representation for browsing frequent itemsets. In *Proceedings of KDD'09*, pp. 687–696, 2009.
[25] S. Ruggieri. Frequent regular itemset mining. In *Proceedings of KDD'10*, pp. 263–272, 2010.
[26] A. Skowron, H. Wang, A. Wojna, and J. G. Bazan. Multimodal classification: case studies. *Transactions on Rough Sets V*, LNCS 4100:224–239, 2006.
[27] C. Wang and S. Parthasarathy. Summarizing itemset patterns using probabilistic models. In *Proceedings of KDD-06*, pp. 730–735, 2006.
[28] G. Webb. Filtered-top-k association discovery. *WIREs Data Mining and Knowledge Discovery*, 1(3):183–192, 2011.
[29] S.-T. Wu, Y. Li, and Y. Xu. Deploying approaches for pattern refinement in text mining. In *Proceedings of ICDM'06*, pp. 1157–1161, 2006.
[30] D. Xin, J. Han, X. Yan, and H. Cheng. Mining comressed frequent-pattern sets. In *Proceedings of VLDB'05*, pp. 709 – 720, 2005.
[31] Y. Xu and Y. Li. Generating concise association rules. In *Proceedings of CIKM'07*, pp. 781–790, 2007.
[32] Y. Xu, Y. Li, and G. Shaw. Reliable representations for association rules. *Data and Knowledge Engineering*, 70(6):555–575, 2011.
[33] X. Yan, H. Cheng, J. Han, and D. Xin. Summarizing itemset patterns: A profilebased approach. In *Proceedings of KDD-05*, pp. 314–323, 2005.
[34] Y. Y. Yao. A comparative study of formal concept analysis and rough set theory in data analysis. In *Proceedings of RSCTC'04*, pages 59–68, 2004.
[35] M. J. Zaki. Mining non-redundant association rules. *Data Mining and Knowledge Discovery*, 9:223–248, 2004.
[36] N. Zhong, Y. Li, and S.-T. Wu. Effective pattern discovery for text mining. *IEEE Transactions on Knowledge and Data Engineering*, 24(1):30 –44, 2012.