Two FCA-Based Methods for Extracting Concepts and Corresponding Concept Lattices from Hybrid Relations

Yuxia LEI*[a], Jingying TIAN[b], Feng JIANG[c]

[a] *School of Information Science and Engineering,* Qufu Normal University, Rizhao 276826, China
[b]*School of Architectural Engineering,* Rizhao Politechnic, Rizhao, China
[c]College of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China

***Abstract* — This paper addresses the important problems of efficiently processing those hybrid databases which allow complex values such as relations in columns for relations. Two methods based on Formal Concept Analysis (FCA) are considered for extracting concepts and corresponding concept lattices from hybrid relations. The first one is based on an insertion operation, which transforms a hybrid relation to a new complex relation called a target relation. From the target relation, concepts with richer intents and a denser concept lattice can be extracted. The second one is based on a partition-insertion operator and needs interconnections among sub-relations in a hybrid relation to mining concepts. The insertion operator is semantics-preserving and extent-preserving. The second method is shown to be more computationally efficient. In addition, this paper analyzed structural connections among concept lattices in detail, which can be used to construct a complex concept lattice from other simple ones. The distributed method can reduce time complexity of constructing concept lattices.**

Keywords - hybrid relations; data mining; formal concepts; concept lattices; structural connections

I. INTRODUCTION

In many real-world knowledge discovery problems, researchers have to deal with complex descriptions such as set, strings, numerical intervals, which are different from binary data-tables. In order to process these complex data, some hybrid databases are proposed, e.g., HadoopDBs [1], F1-DBs[13], and hybrid relational databases[2;16; 21]. On the other hand, the semantics of data can be advantageously exploited to discover relevant patterns which are a concise and semantically rich representation of data[22]. Patterns can be concepts, association rules, and so on. In this paper, we address the specific problems of processing those hybrid databases which preserve the relational database model and allow complex values such as relations in columns for relations, and present alternate ways to extract formal concepts from hybrid relations.

In standard relations, domains of attributes consist of atomic values. This means that the elements of such domains must be simple values such as integers, dates, or strings of characters, which are not further decomposed in working with them. However, in many practical database systems, domains may contain such values as relations, sets, or any other complex objects, and data often have been stored by hybrid databases[2;16; 21]. We call these hybrid databases hybrid relational databases. For example, in many master student management systems, each student can be characterized by personal information, such as gender and student identification number, as well as by some relational information such as academic transcript and paper appraisal results, and for each master student the value on the attribute academic transcript is also a relation. We call academic transcript a relational attribute. Formally, a hybrid relation R is a triple (U,A$\cup$M,I) such that for any $(r,a)\in U\times A$, $I(r,a)=r(a)\in D_a$ is an atomic value, and for any $(r,m)\in U\times M, I(r,m)=r(m)\in D_m$ is a relation $R_{rm}=(V_{rm},E_{rm},I_{rm})$ .That is, domains of attributes in A consist of atomic values, and ones of attributes in M consist of relations. Obviously, a hybrid relation R is connected to many relations R_{rm} through relational attributes, and we respectively call R and R_{rm} a main relation and sub-relations, where $m\in M$. In a sub-relation, tuples and attributes are called sub-tuples and sub-attributes, respectively. How to speedy find potential concepts in hybrid relations is interesting and important.

Formal concept analysis (FCA) focuses on the concept lattice induced by a binary relation between a pair of sets (called objects and attributes, respectively). A node of concept lattices is an objects/attributes pair, called a (formal) concept, consisting of two parts: the extent (objects the concept covers) and intent (attributes describing the concept). The line diagram corresponding to a concept lattice vividly unfolds generalization/specialization relationship among concepts [9]. Recently, concept lattices have already been successfully applied to a wide range of scientific disciplines including knowledge discovery[4-6;11-19;21-26;33;34], information retrieval[8;27;31], rough set theory[14;28;32], and multi-adjoint relation analysis[17; 23].

While the classical FCA problem only considers binary relations, i.e., objects being described by Boolean attributes [9], many practical relations include some complex attributes values such as relations. Thus, a main axis of research on FCA has aimed at integrating further attribute types such as numerical, categorical and taxonomic into the initial framework, either by conceptual scaling or logic scaling back to binary attributes[3;12;29;30] or by extending the definition of the Galois connection[10;18]. Within this axis, a particular trend has investigated the processing of relations whose descriptions go beyond the limits of propositional logics. In order to process such complex relations, several attempts have been made to introduce relations into the FCA field. However, some proposed approaches invariably look at relations as an intra-concept construct, typically relating two parts of the concept description, and therefore can only lead to the discovery of coarse-grained patterns. Other approaches are to enhance the (object × attribute) data representation with a new dimension, while these approaches invariably apply scaling methods. One of the disadvantages of scaling methods is that it is very sensitive to user's selection of scale attributes [3]. A very small difference in the definition of scale attributes may lead to a large difference in the resulted concept lattices. Instead of using scaling methods, we extend the definition of the Galois connection that underlies the lattice structure. Hence, the method can obtain a unique concept lattice for a relation. In section 3, we will find that a hybrid relation can also be treated as a many-valued context of FCA.

Following the second type relation analysis methods, we design a new approach towards extracting concepts from hybrid relations, which does not apply to any scaling method. Actually, each hybrid relation involves one-to-many relations. Undoubtedly, taking into account sub-relations for concept construction will widen the range of potentially interesting generalizations from the raw data. Therefore, it is natural to transform these relations into a single relation. As an approach towards the discovery of concepts from hybrid relational databases, we regard the elements of sub-tuples × sub-attributes as new attributes and insert these pairs into a corresponding main relation, and obtain a new relation called a target relation from which we can extract concepts and concept lattice. On the one hand, the insertion operation only increases the dimension of attributes in a main relation and does not change the values on each original attributes for every tuple. Hence, the insertion operation is semantics- preserving. On the other hand, as the new attributes may yield new concept extents, and hence enhancing the dimension of attributes means refining the closure system of concept extents of a relation. Therefore, the insertion operation is also extents-preserving. However, this method results in the following disadvantage: with the increase of dimensions of attributes, the node number of the constructed concept lattice usually increases enormously, which means that more computational time is consumed. In order to reduce time complexity of generating concept lattices, we replace straightly constructing the concept lattice of a target relation by constructing it from the concept lattice of a main relation and concept lattices of sub-relations. Hence, it is necessary for us to analyze structural connections among these concept lattices.

Our main contributions in this paper are to propose an approach towards constructing concepts and concept lattices from hybrid relational databases, and to illustrate the approach proposed with two kinds significant hybrid relations. The key idea is to use the concept lattice of a main relation as a basic conceptual hierarchy and to insert concepts of sub-relations into it, and the insertion operation gives rise to a richer concept lattice.

The paper is organized as follows. Section 2 briefly describes some basic notions such as concepts, relations, hybrid relations, and concept lattices. Section 3 describes concepts and concept lattices in hybrid relations. Section 4 and section 5 present an approach towards the discovery of concepts in two kinds of hybrid relations, which is based on structural connections among concept lattices. Finally, section 6 concludes the paper.

II. PRELIMINARIES

A. Some Basic Notations in Formal Concept Analysis

In FCA[9], a context is defined as a triple $K=(G,M,I)$ where G and M are sets and $I \subseteq G \times M$. The elements of G and M are called objects and attributes, respectively. For any $g \in G$ and $m \in M$, $(g,m) \in I$ implies that the object g possesses the attribute m. The relation I induces two maps α and β between the power set $\mathrm{P}(G)$ of G and the power set $\mathrm{P}(M)$ of M. For a set $X \in \mathrm{P}(G)$ of objects, $\alpha(X)$ is defined as:

$$\alpha(X)=\{m \in M : \forall g \in X(g\,\mathrm{Im})\}, \tag{1}$$

which is the set of attributes common to the objects in X. Dually, for a set $Y \in \mathrm{P}(M)$ of attributes, $\beta(Y)$ is defined as:

$$\beta(Y)=\{g \in G : \forall \mathrm{m} \in Y(g\,\mathrm{Im})\}, \tag{2}$$

which is the set of objects which have all attributes in Y.

For any $X \in \mathrm{P}(G)$ and $Y \in \mathrm{P}(M)$, the pair (X,Y) is called a (formal) concept if $\alpha(X)=Y$ and $\alpha(Y)=X$, where X and Y are called the extent and the intent of the concept, respectively. The set of all concepts, the set of extents and the set of intents in K are denoted $\mathrm{L}(K)$, $\mathrm{L}_E(K)$ and $\mathrm{L}_I(K)$, respectively. If C is a concept in K, we respectively use Ext(C) and Int(C) representing the extent and the intent of C. If $(X_1,Y_1),(X_2,Y_2) \in \mathrm{L}(K)$ are concepts, (X_1,Y_1) is called a sub-concept of (X_2,Y_2), provided that $X_1 \subseteq X_2$ (which is equivalent to $Y_2 \subseteq Y_1$), denoted by $(X_1,Y_1)\preceq(X_2,Y_2)$. In this case, (X_2,Y_2) is a super-concept of (X_1, Y_1). The relation $\preceq$ is a hierarchical order of the concepts, which produces a lattice structure in $\mathrm{L}(K)$, called the concept lattice of K, also denoted by $\mathrm{L}(K)$.

The above context K is binary. As many practical applications involve non-binary data, many-valued contexts have been introduced in FCA. A many-valued context $\overline{K} = (G, M, W, I)$ consists of sets G, M, W and a ternary relation I between G, M and W for which it holds that $(g, m, w) \in I$ and $(g, m, v) \in I$ always imply $w = v$. The elements of G, M and W are called objects, attributes and attribute values, respectively. A tuple (g, m, w) is interpreted as the object g has a value w for the attribute m.

B. Relations in Relational Databases

Actually, a many-valued context can be regarded as a relation with the column containing the objects being a primary key[29]. In the RDM, a relation is described by a relation schema $S = R(a_1, a_2, ..., a_n)$, where $a_i (1 \le i \le n)$ represent attributes. Each attribute a is associate with a domain D_a, which is the set of possible values for the attribute a. A relation (R, A) is denoted by a set of tuples $U = \{r : r = (v_1, v_2, ..., v_n) \in R\}$, where $A = (a_1, a_2, ..., a_n)$, and r is a tuple such that for every $1 \le i \le n, v_i \in D_{a_i}$. An equivalent way to view such a tuple $r \in U$ is as a map from $A = (a_1, a_2, ..., a_n)$ to $\bigcup_{a \in A} D_a$ such that $r(a) \in D_a$ [7;14;20]. Thus we can further represent a relation R by a triple (U, A, I), where U is a non-empty set of tuples, and I is a map from $U \times A$ to $\bigcup_{a \in A} D_a$ such that for any $((r, a) \in U \times A$, $I(r, a) = r(a) \in D_a$. A relation (U, A, I) can be thought of as representing a table with rows corresponding to U, columns corresponding to A, and table entries at the intersection of rows and columns containing values in domains.

C. Hybrid Relations

Hybrid relations preserve the relational model and allow complex values such as relations in their columns [21]. Formally, a hybrid relation R is a triple $(U, A \bigcup M, I)$ such that for any $(r, a) \in U \times A$, $I(r, a) = r(a) \in D_{\mathrm{a}}$ is an atomic value, and for any $(r, \mathrm{m}) \in U \times A$, $I(r, \mathrm{m}) = r(\mathrm{m}) \in D_{\mathrm{m}}$ is a relation R_{rm}. R and R_{rm} are called the main relation and sub-relations, respectively. Tuples and attributes in a main relation are called main tuples and main-attributes, respectively. Correspondingly, tuples and attributes in sub-relations are called sub-tuples and sub-attributes, respectively.

This paper focuses on those simple hybrid relations in which only one column consists of relations, for many complex hybrid relation can be processed by similar methods. Formally, we mainly discuss those hybrid relations that they are represented a triple R=$(U, A \bigcup \{b\}, I)$ such that for any $(r, a) \in U \times A$, $I(r, a) \in D_{\mathrm{a}}$ is atomic value, and for any $r \in U$, $I(r, \mathrm{b}) \in D_{\mathrm{b}}$ is a relation R_{r}=(V, E, I_r). We discuss structural connections among concept lattices, in order to speedy extract concepts from multiple relations involved in hybrid relations. For this purpose, it is helpful to categorize these simple hybrid relations into two classes: first-kind hybrid relations and second-kind ones. The main difference of them is as follows: each main tuple r in a second-kind hybrid relation is related to all columns of the sub-relation R_{r}, while each main tuple r in a first-kind hybrid relation is only related to one column of the sub-relation R_{r}. A first-kind hybrid relation R is a triple $(U, A \bigcup \{b\}, I)$, where there is a universe V and a set $B \bigcup \{c\}$ of sub-attributes such that for each $r \in U$, $I(r, b)$ is a relation $R_r = (V, B \bigcup \{c\}, I_r)$ satisfying the following conditions:

- for any $v \in V$ and $d \in B$,
  $I_r(v, d) \in D_d$,and $I_r(v, c) \in D_c$;
- for any $t \in U$ and $d \in B$, $I_r(s, d) \in I_t(s, d)$;
- for any $r, s \in U$,
  $$I_r(r, b) = I(s, b) \Leftrightarrow R_r = R_s \Leftrightarrow I_r = I_s .$$

The second condition means that all tuples in U share a common value on each attribute in B. The third condition means that R and $R_r (r \in U)$ must satisfy two functional dependencies $\{b\} \to V \times \{c\}$ and $V \times \{c\} \to \{b\}$.

Example 1. Let $A = \{a\}$, $B = \{b_1, b_2\}$, $V = \{v_1, v_2\}$.

$$R = \begin{array}{c|cc} & a & b \\ \hline r & v_{ra} & v_{rb} \\ s & v_{sa} & v_{sb} \\ \vdots & \vdots & \vdots \end{array} \qquad where$$

$$v_{rb} = \begin{array}{c|ccc} & b_1 & b_2 & c \\ \hline v_1 & w_{11} & w_{12} & w_{1r} \\ v_2 & w_{21} & w_{22} & w_{2r} \end{array} \qquad v_{sb} = \begin{array}{c|ccc} & b_1 & b_2 & c \\ \hline v_1 & w_{11} & w_{12} & w_{1s} \\ v_2 & w_{21} & w_{22} & w_{2s} \end{array}$$

A second-kind hybrid relation R is a triple $(U, A \bigcup \{b\}, I)$, where there is a universe V and a set B of sub-attributes such that for each $r \in U$, $I(r, b) = v_{rb} \in D_b$ is a relation R_{r}=(V, B, I_r) satisfying the following conditions:

- for any $v \in V$ and $b' \in B$, $I_r(v,b') \in D_{b'}$;
- for any $r,s \in U$,

$$I(r,b) = I(s,b) \Leftrightarrow R_r = R_s \Leftrightarrow I_r = I_s .$$

The second condition means that R and $R_r (r \in U)$ must satisfy two functional dependencies $\{b\} \to V \times B$ and $V \times B \to \{b\}$.

Example 2. Let $A = \{a\}$, $B = \{b_1, b_2\}$, $V = \{v_1, v_2\}$.

$$R = \begin{array}{c|cc} & a & b \\ \hline r & v_{ra} & v_{rb} \\ s & v_{sa} & v_{sb} \\ \vdots & \vdots & \vdots \end{array} \quad where$$

$$v_{rb} = \begin{array}{c|cc} & b_1 & b_2 \\ \hline v_1 & w_{r11} & w_{r12} \\ v_2 & w_{r21} & w_{r22} \end{array} \qquad v_{sb} = \begin{array}{c|cc} & b_1 & b_2 \\ \hline v_1 & w_{s11} & w_{s12} \\ v_2 & w_{s21} & w_{s22} \end{array}$$

III. CONCEPTS AND CORRESPONDING CONCEPT LATTICES IN HYBRID RELATIONS

Given a relation $R = (U, A, I)$, for any tuple $r \in U$ and any attribute $a \in A$, if the tuple r has a value $v \in D_a$ for the attribute a, then we say r having a property (a,v). On the other hand, in FCA, for any many-valued context $\overline{K} = (G, M, W, I)$, we can omit the third component W from the quadruple $\overline{K}$ by assuming that for every $m \in M$, m has a domain D_m and $W = \bigcup_{m \in M} D_m$. Thus we can represent $\overline{K}$ by a triple (G, M, J), where J is a map from $G \times M$ to $\bigcup_{m \in M} D_m$ such that for any $g \in G$ and $m \in M$, $J(g,m) \in D_m$, and $J(g,m) = w \Leftrightarrow (g,m,w) \in I$ Hence, we can treat $R = (U, A, I)$ as a many-valued context of FCA, and can apply the theory of FCA to relations.

In classical FCA, the definition of concepts completely depends on a single binary relation. Thus, the definition of concepts in hybrid relations can share the same form. Next we define concepts of (hybrid) relations in a straightforward way. Given a (hybrid) relation $R = (U, A, I)$, for a set $X \subseteq U$ of tuples, we define $\alpha(X)$ as follows:

$$\alpha(X) = \{(a,v) \in A \times \bigcup_{a \in A} D_a : \forall r \in X (I(r,a) = v)\}, \quad (3)$$

which represents the set of properties common to the tuples in X. Correspondingly, for a set $Y \subseteq A \times \bigcup_{a \in A} D_a$ of properties, we define

$$\beta(Y) = \{r \in U : \forall (a,v) \in Y (I(r,a) = v)\}, \quad (4)$$

which represents the set of tuples having all properties in the set Y.

A concept in R is defined as a pair $[X,Y]$ with $X \subseteq U$ and $Y \subseteq A \times \bigcup_{a \in A} D_a$ such that $[X,Y]$ is maximal with the characteristic $X \times Y \subseteq I$, that is, $\alpha(X) = Y$ and $\beta(Y) = X$. The sets X and Y are called the extent and the intent of the concept, respectively. The hierarchical order of concepts is formalized by

$$[X_1, Y_1] \leq [X_2, Y_2] \Leftrightarrow X_1 \subseteq X_2 \Leftrightarrow Y_2 \subseteq Y_1 . \quad (5)$$

If $[X_1, Y_1] \leq [X_2, Y_2]$, then $[X_1, Y_1]$ is a sub-concept of $[X_2, Y_2]$. Dually, $[X_2, Y_2]$ is a super-concept of $[X_1, Y_1]$. The set of all concepts in R together with the order $\leq$ is always a lattice, called the concept lattice of R, denoted by $\mathrm{L}(R)$. Similar to the concept lattice of a binary relation, the concept lattice $\mathrm{L}(R)$ is also a complete lattice. Specially, for any tuple r, the pair $[\beta\alpha(\{r\}), \alpha(\{r\})]$ is a concept, called the tuple concept of r. Correspondingly, for any property (a,v), the pair $[\beta(\{(a,v)\}), \alpha\beta(\{(a,v)\})]$ is also a concept, called the property concept of (a,v).

Generally, there are two alternative methods for extracting concepts from multiple relations involved in a hybrid relation. One method is to transform multiple relations into a single relation and to extract concepts from the single relation. The other is to construct new concepts from some concepts of original relations. We will only discuss these two methods for analyzing hybrid relations, in order to construct richer concepts from concepts of a main relation and concepts of its sub-relations.

IV. FIRST-KIND HYBRID RELATIONS ANALYSIS METHODS

Because each hybrid relation is linked to many sub-relations, a natural way of analyzing hybrid relations would be to construct concepts that reflect commonalities both in individual attributes and in sub-relations. Alternative approach is to transform these relations into a single relation called a target one. This section describes two kinds of methods: insertion and partition-insertion, and analyzes the structural connections among concept lattices, which are the effects of these methods in the concept lattice. Based on these connections, one can construct the concept lattice of the target relation.

DOI 10.5013/ IJSSST.a.17.27.35 ISSN: 1473-804x online, 1473-8031 print

A. *Transformation of First-Kind Hybrid Relations*

Given a first-kind hybrid relation $R=(U, A\bigcup\{b\}, I)$, every tuple $r\in U$ has a sub-relation $R_r=(V, B\bigcup\{c\}, I_r)$ as the value v_{rb} for the attribute b, and these sub-relations share a same set V of sub-tuples and a same set $B\bigcup\{c\}$ of sub-attributes. Because FCA only functions a single relation, we first need to find a method for constructing a single relation which summaries all relations involved in a hybrid relation. In order to reach this goal, we insert all sub-relations into a corresponding main relation. Formally, we regard the elements of $V\times(B\bigcup\{c\})$ as new attributes and insert these pairs into R, and then obtain a target relation $R^{'}=(U^{'}, A^{'}, I^{'})$, where $U^{'}=U$; $A^{'}=A\bigcup V\times B\bigcup V\times\{c\}$; $I^{'}$ is defined as follows: for any $r\in U$ and $a\in A^{'}$,

$$I^{'}(r,a)=\begin{cases} I(r,a) & \text{if } a\in A \\ I_r(v,b) & \text{if } a=(v,b)\in V\times B \\ I_r(v,c) & \text{if } a=(v,c)\in V\times\{c\} \end{cases}$$

By the insertion method, we can transform Example 1 into the following table:

$$R' = \begin{array}{c|ccccccc} & a & (v_1,b_1) & (v_1,b_2) & (v_2,b_1) & (v_2,b_2) & (v_1,c) & (v_2,c) \\ \hline r & v_{ra} & w_{11} & w_{12} & w_{21} & w_{22} & w_{1r} & w_{2r} \\ s & v_{sa} & w_{11} & w_{12} & w_{21} & w_{22} & w_{1s} & w_{2s} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{array}$$

B. *Connections between* $\mathrm{L}(R)$, $\mathrm{L}(R^{'})$ *and* $\mathrm{L}(R_r)$

Next we mainly discuss the effects of the insertion operation in concept lattices, that is, the structural connections among $\mathrm{L}(R)$, $\mathrm{L}(R_r)$ and $\mathrm{L}(R^{'})$.

The upper insertion operation actually is to replace the attribute b by the elements of $V\times(B\bigcup\{c\})$. On the one hand, the operation does not change the tuples in U and the values for all attributes in A. On the other hand, if some tuples share a same value for the attribute b, then they share a same value for every element of $V\times(B\bigcup\{c\})$. Hence, the operation is semantic-preserving. Because all tuples in U also keep constant, every property extent of R is also a property extent of $R^{'}$ and, since every concept extent is the intersection of property extents, we obtain that every extent of R is an extent of $R^{'}$. Hence, the operation is also extent-preserving, that is, the operation refines the closure system of concept extents of the relation R. Formally, if $[X,Y]$ is a concept in R, then $[X,\alpha^{(R^{'})}(X)]$ must be a concept in $R^{'}$. Therefore, there exists a natural order-embedding map ρ between the two concept lattices $\mathrm{L}(R)$ and $\mathrm{L}(R^{'})$: for any concept $[X,Y]\in\mathrm{L}(R)$, $\rho([X,Y])=[X,\alpha^{(R^{'})}(X)]$. The map is also supermum-preserving.

Given a concept $[X,Y]$ in a relation R, we want to make $\alpha^{(R^{'})}(X)$ more clear. For the purpose, it is helpful to partition $A^{'}$ into three subset A, $V\times B$ and $V\times\{c\}$, and $\alpha^{(R^{'})}(X)$ accordingly consists of three components. Firstly, for all tuples, because the values on every attribute in A keep constant, the properties in Y with the form $(a,v)\in A\times\bigcup_{a\in A}D_a$ are inherited into $\alpha^{(R^{'})}(X)$. Secondly, all tuples share the same value on every attribute in $V\times B$, so all such properties are in $\alpha^{(R^{'})}(X)$. Thirdly, $\alpha^{(R^{'})}(X)$ also contains such properties with the form $((v_i,c),w_i)$ common to X, namely, all tuples in X share a same value $w_i\in D_c$ for every attribute $(v_i,c)\in V\times\{c\}$.

As every property extent of R is also a property extent of $R^{'}$ and, since every concept extent is the intersection of property extents, we have the following proposition:

Proposition 4.1 Every extent of R is also an extent of $R^{'}$.

This proposition means that the closure system of extents of a main relation is a subset of the one of the corresponding target relation. Thus, we can obtain a natural order-embedding map between $\mathrm{L}(R)$ and $\mathrm{L}(R^{'})$. Because Y is an intent of R, Y either contains a property with the form (b,w_b) or not, where $w_b\in D_b$. Hence, we distinguish the following two cases to prove proposition4.2: one is that Y contains a property with the form (b,w_b); the other is that Y does not contain any property with the form (b,w_b).

Proposition 4.2 There is a map ρ from $\mathrm{L}(R)$ to $\mathrm{L}(R^{'})$, which is supermum-preserving and order-embedding. For any concept $[X,Y]\in\mathrm{L}(R)$, ρ is defined as follows:

$$\rho([X,Y])=\begin{cases} [\phi,\alpha^{(R^{'})}(\phi)] & \text{if } X=\phi \\ [X, Y\bigcup W\bigcup P] & \text{if } X\neq\phi \text{ and } Y\bigcap\{b\}\times D_b=\phi \\ [X,(Y-W_b)\bigcup W\bigcup P] & \text{if } X\neq\phi \text{ and } Y\bigcap\{b\}\times D_b\neq\phi \end{cases}$$

where,

$$W=\{((v_i,b_j),w_{ij})\in(V\times B)\times(\textstyle\bigcup_{b^{'}\in B}D_{b^{'}}):\exists r\in U(I_r(v_i,b_j)=w_{ij})\}$$

, $W_b=\{(b,w_b)\in\{b\}\times D_b:\forall r\in X(I(r,b)=w_b)\}$,

$P=\{((v_i,c),w_i)\in(V\times\{c\})\times D_c:\forall r\in X(I_r(v_i,c)=w_i)\}$.

We mainly divide two steps to prove the proposition: we first show that there exists such a map, and then concretely describe the $\alpha^{(R^{'})}(X)$, where X is an extent of a main relation R.

(1) For any concept $[X,Y]\in L(R)$, by proposition 4.1 we realize easily that the pair $[X,\alpha^{(R')}(X)]$ is a concept in the target relation R'. Hence, we can define a map

$$L(R)\to L(R'),[X,Y]\mapsto [X,\alpha^{(R')}(X)] \quad (6)$$

The map is supermum-preserving and order-embedding. On the one hand, let T be a limit index set, given some concepts $[X_{t\in T},Y_{t\in T}]\in L(R)$, then there are the following inference links:

$$\begin{aligned}&\rho(\wedge_{t\in T}[X_t,Y_t])\\ &=\rho[\bigcap_{t\in T}X_t,\alpha^{(R)}(\bigcap_{t\in T}X_t)]\\ &=[\bigcap_{t\in T}X_t,\alpha^{(R')}(\bigcap_{t\in T}X_t)]\\ &=\wedge_{t\in T}\rho([X_t,Y_t])\end{aligned} \quad (7)$$

Thus, the map ρ is supermum-preserving. On the other hand, for any two concepts $[X_1,Y_1]$, $[X_2,Y_2]\in L(R)$, we can obtain the following inference links:

$$\begin{aligned}&[X_1,Y_1]\preceq[X_2,Y_2]\\ &\Leftrightarrow X_1\subseteq X_2\\ &\Leftrightarrow \rho([X_1,Y_1])\preceq\rho([X_2,Y_2]).\end{aligned}$$

Thus, ρ is an order-embedding.

(2) If Y does not contain any property with the form (b,w_b), then $\alpha^{(R')}(X)=Y\bigcup W\bigcup Q$. Otherwise,

$$\alpha^{(R')}(X)==(Y-W_b)\bigcup W\bigcup P.$$

Case 1: $Y\bigcap\{b\}\times D_b=\phi$. In this case, we can realize easily that $X\neq\phi$. Hence, it is feasible to define the set Q. If $Y=\phi$, then we can infer easily that $X=U$ holds, and thus $[U,W\bigcup Q]$ is a concept in R'.Assume $Y\neq\phi$, we must show that $Y\bigcup W\bigcup Q=\alpha^{(R')}(X)$. On the one hand, $Y\bigcup W\bigcup Q\subseteq\alpha^{(R')}(X)$ can be easily proved. On the other hand, if for any an attribute $a\in A$, there is a $v\in D_a$ such that $(a,v)\in Y$, then $\alpha^{(R')}(X)=Y\bigcup W\bigcup Q$ holds. Or else, we assume that there exists an tuple $a_0'\in\alpha^{(R')}(X)$ such that at $a_0'\notin Y\bigcup W\bigcup Q$. From $a_0'\notin Y\bigcup W\bigcup Q$,we can infer that there exists an attribute $a_0'\in A\times(\bigcup_{a\in A}D_a)-Y$, and further there exist $a_1\in A$ and $w_{a1}\in D_{a1}$ satisfying $a_0'=(a_1,w_{a1})$. However, from $a_0'=(a_1,w_{a1})\in\alpha^{(R')}(X)$, we also can infer that $I'(r,a_1)=I(r,a_1)=w_{a1}$ holds for all $r\in X$, and then obtain $a_0'=(a_1,w_{a1})\in\alpha^{(R)}(X)=Y$. This leads to a contradiction. Consequently, $\alpha^{(R')}(X)=Y\bigcup W\bigcup Q$.

Case 2: Y contains a property with the form (b,w_b). If $X=\phi$, then Y is a set of all properties in R. Because of $W_b\subseteq Y$, we know that $[\phi,(Y-W_b)\}\bigcup W\bigcup P]$ is a concept in the relation R'. If $X\neq\phi$, then $|W_b|=1$. Let $W_b=\{(b,w_b):\forall r\in X(I(r,b)=w_b)\}$. Obviously, all tuples in X share a same value w_i on each attribute $(v_i,c)\in V\times\{c\}$. Hence, it is feasible to define the set P. Assume $X\neq\phi$ .A similar argument to Case 1 can be used to show that $(Y-\{(b,w_b)\})\bigcup W\bigcup P=\alpha^{(R')}(X)$ holds.

For two binary relations, we provide a proposition for deciding whether there exists an order-embedding map between two concept lattices. Similarly, we have the following proposition, which describes a necessary and sufficient condition to decide whether there exists an order-embedding map between two concept lattices $L(R)$ and $L(R')$. The proposition means that $L(R)$ is isomorphic to a suborder of $L(R')$.

Proposition 4.3 Given a first-kind hybrid relation $R=(U,A_0,I)$, where $A_0=A\bigcup\{b\}$. Let a corresponding target relation $R'=(U,A',I')$. The map

$$\varphi:\varphi([X,Y])=\vee_{r\in X}[\beta^{(R')}\alpha\Big|^{(R')}(r),\alpha^{(R')}(r)]$$

between $L(R)$ and $L(R')$ is order-embedding if and only if

(4.1) There are maps

$\lambda:U\to L(R')$ and

$\chi:A_0\Big|\times(\bigcup_{a\in A_0}D_a)\to L(R')$; and

(4.2) The maps λ and χ satisfy the following condition:

$$I(r,a)=w\in D_a\Leftrightarrow\lambda(r)\preceq\chi(a,w).$$

If $\varphi:\varphi([X,Y])=\vee_{r\in X}[\beta^{(R')}\alpha^{(R')}(r),\alpha^{(R')}(r)]$ is an order-embedding, then we construct the following maps:

$\gamma:U\to L(R)$,

$\gamma\mapsto[\beta\alpha(r),\alpha(r)]$ and

$\mu:A_0\times\bigcup_{a\in A_0}D_a\to L(R)$,

$(a,w)\mapsto[\beta((a,w)),\alpha\beta((a,w))]$,

and further construct the maps $\lambda=\varphi o\gamma$ and $\chi=\varphi o\mu$, which precisely have the properties specified in (4.1) and (4.2).

If, conversely, (λ,χ) is a pair of maps with $I(r,a)=w \Leftrightarrow \lambda(r)\underline{\text{p}}(a,w)$, then the map between L (R) and L $(R^{'})$:

$$\varphi([X,Y]) = \vee_{r\in X}[\beta^{(R^{'})}\alpha^{(R^{'})}(r),\alpha^{(R^{'})}(r)]$$

is order-preserving. We show that φ is, moreover, an order-embedding. If $[X_1,Y_1],[X_2,Y_2]\in$ L (R) are concepts and if $[X_1,Y_1]$ ¾ $[X_2,Y_2]$, then $X_1\neq\phi$ and $Y_1\neq\phi$. On the one hand,

$[X_1,Y_1]$ ¾ $[X_2,Y_2]$.

$\Rightarrow \exists s\in X_1 \exists (a,w)\in Y_2 I(s,a)\neq w$

$\Rightarrow \lambda(s)$ ¾ $\chi(a,w)$

On the other hand, $I(r,a)=w \Leftrightarrow \lambda(r)$ ° $\chi(a,w)$ holds for all $r\in X_2$

$\Rightarrow \vee\{\lambda r : r\in X_2\}$ ° $\chi(a,w)$

$\Rightarrow \lambda(s)$ ¾ $\vee\{\lambda r : r\in X_2\}$

$\Rightarrow \vee\{\lambda p : p\in X_1\}$ ¾ $\vee\{\lambda g : g\in X_2\}$

$\Rightarrow \varphi[X_1,Y_1]$ ¾ $\varphi[X_2,Y_2]$.

Proposition 4.2 means that there exists an order-embedding map between L (R) and L $(R^{'})$. Generally, the map is not an isomorphism. The reason is as follows: when inserting these sub-relations into R , we do not change the tuples in U but add the dimension of attributes in the relation R , and thereby refine the closure system of concept extents. Consequently, the insertion operation generally results in some new concepts. Obviously, every concept in R can be expanded to a concept in $R^{'}$. Conversely, there are some interesting connections between concepts in $R^{'}$ and ones in R .

Proposition 4.4 For any $C'\in$ L $(R^{'})$, there exist two concepts C_1 and $C_2\in$ L (R) satisfying $\beta(C_1)\preceq C^{'}\preceq\beta(C_2)$.

(1) If $C'=$• $^{(R^{'})}$, then $\rho($• $^{(R)})=$• $^{(R^{'})}$ holds. Assume C p • $^{(R)}$, we can obtain $\rho(C)$ ° C' ° $\rho($• $^{(R)})$. Similarly, $C'=\perp^{(R^{'})}$ holds.

(2) Assume $C'\notin\{\perp^{(R^{'})},$• $^{(R^{'})}\}$. If there exists a concept $C\in$ L (R) , then there exist the supermum C_1 of C and the infimum C_2 of C satisfying $\rho(C_1)$ ° C' ° $\rho(C_2)$. Assume that $\rho(C^{''})\neq\rho(C)$ hold for any concept $C\in$ L (R) . If, $C^{'}\in$ L (R) , there exist inverse images C_1 and C_2 of an infimum and an supermum of C' , respectively, then $\rho(C_1)$ ° C' ° $\rho(C_2)$ holds. Otherwise, we continually observe whether there exists an inverse images of an infimum of infimums of $C^{''}$. If there is an inverse image, then the proposition holds. If not, we can continue the process. Because the concept lattice of $R^{'}$ is finite, the process will stop. The similar method can be used to the supermum.

By proposition 4.2, we know that the insertion operation refines the closure system of concept extents of a main relation. Because all tuples in U share a same value for every attribute $(v_i,b_j)\in V\times B$, each new extent must be the property extent of a property in $(V\times\{c\}\times D_c)$ or the intersection of some property extents in $(V\times\{c\}\times D_c)$. These properties are exactly the common information of some concepts in sub-relations, namely, the intersection of intents (extents) of these concepts is not empty. Thus, we call the Cartesian product of the intersection of extents and one of intents the common information of these concepts. Formally, given some concepts $[A_t,B_t](t\in T)$ of sub-relations, the common information of these concepts is the set $(\bigcap_{t\in T}A_t)\times(\bigcap_{t\in T}B_t)$. Consequently, some relevant concepts of sub-relations can uniquely determine a concept extent Ext(C), and the common information of these concepts are components of the intent Int(C), where C is a concept in the target relation.

Next we present some propositions describing connections between L (R_r) and L $(R^{'})$, which aim to illustrate how to lead to new extents and to make the intents richer, where $R_r(r\in U)$ and $R^{'}$ are sub-relations and a target relation, respectively. Our key idea is as follows: observing all sub-relations R_r , we can define an equivalence relation on U in the light of different values on the attribute c for all $v_i\in V$. Formally, for any $v_i\in V$, and r , $s\in U$, we define a relation θ_{v_i} on U such that $r\theta_{v_i}s\Leftrightarrow I_r(v_i,c)=I_s(v_i,c)$. Obviously, θ_{v_i} is an equivalence relation on U . Thus, we can obtain a corresponding equivalence relation θ_{v_i} on U for any pair (v_i,c) , and further obtain a corresponding set U/θ_{v_i} of equivalence classes. Obviously, each class in U/θ_{v_i} is a property extent of $R^{'}$. Hence, the intersection of equivalence classes of any finite different equivalence relations is a concept extent of $R^{'}$. Proposition 4.5 describes what a property extent is, and corollary 4.1-4.3 present methods for determining all (new) extents of the relation $R^{'}$, and proposition 4.6 means that concepts common to all sub-relations can be mapped onto the maximal concept in $R^{'}$.

Proposition 4.5 Let $[r]\theta_{v_i}=\{s\in U : r\theta_{v_i}s\}$. $[r]\theta_{v_i}$ is the equivalence class of θ_{v_i} containing r , is the extent of a property $((v_i,c),I_r(v_i,c))$.

There are the following connections between equivalence classes of every equivalence relation θ_{v_i} on U and concepts of $L(R_r)$, $r \in U$.

Corollary 4.1 For any $r \in U$ and $v_i \in V$. Let $S_{v_i r}=\{s \in U : \exists C \in L(R_s)(v_i \in Ext(C))\} \wedge (c, I_r(v_i, c)) \in Int(C))\}$ then $S_{v_i r}=[r]\theta_{v_i}$ and $((v_i, c), I_r(v_i, c)) \in \alpha^{R'}(S_{v_i r}))$ hold, where $Ext(C)$ and $Int(C)$ represent the extent and the intent of a concept C, respectively.

Corollary 4.2 For any $r \in U$, $\bigcap_{v_i \in V} S_{v_i r}$ is an extent of a target relation R'.

Corollary4.3 Let $V=\{v_1, v_2, \cdots, v_n\}$ and the corresponding equivalence relations on U be $\theta_{v1}, \theta_{v2}, \cdots, \theta_{vn}$. For every n-tuple element $N \in U/\theta_{v1} \times U/\theta_{v2} \times \cdots \times U/\theta_{vn}$, the intersection of any finite component*(s)* of N is a concept extent of R'.

Proposition 4.6 Let $S=\bigcap_{r \in U} L(R_r)$. A map :

$$S \to L(R'),\ [X,Y] \mapsto [U, \alpha^{(R')}(U)]$$

is infimum *(supermum)-preserving* and *order-preserving.*

Based on above propositions, we can construct the concept lattice $L(R')$ from the concept lattices $L(R)$ and $L(R_r)(r \in U)$. The key idea is that $L(R)$ is used as a basic conceptual hierarchy to support construction of $L(R')$, that is, inserting $L(R_r)(r \in U)$ into $L(R)$ can lead to a richer concept lattice. The specific descriptions are as follows: assume the concepts $[A_t, B_t](t \in T)$ determine an extent of the relation R', then the extent is the set $s=\{r \in U : \exists t \in T(A_t, B_t) \in R_r\}$. If S is a tuple extent, then $\bigcup_{t \in T} A_t \times B_t$ are components of the intent $\alpha^{(R')}(S)$. If $S \in L_E(R)$ and $|S| \geq 2$, then $(\bigcap_{t \in T} A_t) \times (\bigcap_{t \in T} B_t)$ are components of the intent $\alpha^{R'}(S)$. If $S \notin L_E(R)$, then the pair $[S, (\bigcap_{t \in T} A_t)] \times (\bigcap_{t \in T} B_t)]$ is a new concept, that is, $[S, (\bigcap_{t \in T} A_t)] \times (\bigcap_{t \in T} B_t)] \in L(R')$. The process will give rise to richer concept lattice.

Now, we briefly discuss connections between $L(R_r)$ and $L(R)$.

Proposition 4.7 A map:

$$L(R_r) \to L(R),\ [X,Y] \mapsto [\beta(b, v_b), \alpha\beta(b, v_b)]$$

is infimum*(supermum) -preserving* and *order-preserving*.

Algorithm 1: Extracting Concepts from A Hybrid Relation

Algorithm for Extracting Concepts from A Hybrid Relation
Input: a hybrid relation $R=(U, A \cup \{b\}, I)$
Output: all concepts of R'
Process
Step 1: Generating concept lattices of the main relation/sub-relations
Step 2: For each sub-relation R_r, $r \in U$
For all tuple concepts $[A_t, B_t] \in L(R_r), (t \in T)$
$[\{r\}, \alpha^R(r) \cup (\bigcup_{t \in T}(A_t B_t))]$ is a concept in R'
Step 3: For any tuple concepts $[A_t, B_t]$ of sub-relations
let $S=\{r \in U : \exists t \in T(A_t, B_t) \in R_r\}$, and assume that $|S| \geq 2$
If $S \notin L_E(R)$
Then $[S, (\bigcap_{t \in T} A_t)] \times (\bigcap_{t \in T} B_t)]$ is a concept in R'
Else $[(\bigcap_{t \in T} A_t)] \times (\bigcap_{t \in T} B_t)]$ are components of $\alpha^{(R')}(S)$

C. *Experimental results*

In order to make the above propositions clearer, we use the information table of master students of some university. Each student can be characterized by personal information, such as gender and student identification number, as well as by some relational information such as academic transcript. Each value on the attribute academic transcript is a sub-relation, which consists of three courses.

TABLE 1: COMPARISON OF ORIGINAL RELATION, TARGET RELATION BASED ON INSERTION METHOD

	Concepts			Edges			Height of the lattice		
Datasets									
|U|	4	10	25	4	10	25	4	10	25
|A|	3	5	5	3	5	5	3	5	5
The original relation	7	28	82	10	59	189	4	4	6
The target relation	9	33	118	12	71	236	4	5	7

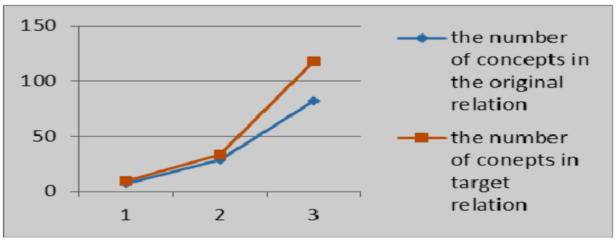

Figure 1: A comparison of the numbers of concepts in 3 databsets

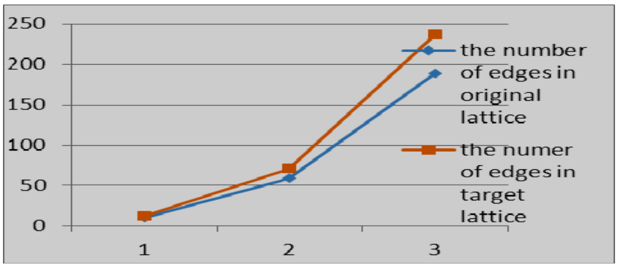

Figure 2: A comparison of the numbers of edges in 3 databsets.

D. *First-kind Hybrid Relation Analysis Based on Partition-Insertion Method*

In section 4.1, we introduce the insertion method, which regards the elements of $V \times (B \bigcup \{c\})$ as new attributes of tuples in U . However, in each sub-relation R_r , the values on all sub-attributes in B for every $v_i \in V$ are not relevant to r . What are relevant to r are the values of sub-tuples in V on the sub-attribute c . Thus, we first partition the set $(B \bigcup \{c\})$ into two sets B and $\{c\}$, and then only insert all elements of $V \times \{c\}$ into R . We call this process partition-insertion method.

Formally, R can be transformed into two relations $R^{''} = (U^{''}, A^{''}, I^{''})$,and $R^{'''} = (U^{'''}, A^{'''}, I^{'''})$ where

- $U^{''} = U$; $U^{'''} = V$; $A^{''} = A \bigcup (V \times c)$; $A^{'''} = B$;
- $I^{''}$ is defined as follows: for any $r \in U$, $a \in A \bigcup \{V \times c\}$,

$$I^{''}(r,a) = \begin{cases} I(r,a) & \text{if } a \in A \\ I_r(v,c) & \text{if } a = (v,c) \in V \times c \end{cases}$$

$I^{'''} = I_r(V \times B)$. For some $r \in U$. Notice that this definition is meaningful, because for any r , $r^{'} \in U$, $I_r(V \times B) = I_{r^{'}}(V \times B)$.

By the partition-insertion method, we can transform Example 1 into the following tables:

$$R'' = \begin{array}{c|ccc} & a & (v_1,c) & (v_2,c) \\ \hline r & v_{ra} & w_{1r} & w_{2r} \\ s & v_{sa} & w_{1s} & w_{2s} \\ \vdots & \vdots & \vdots & \vdots \end{array} \qquad R''' = \begin{array}{c|cc} & b_1 & b_2 \\ \hline v_1 & w_{11} & w_{12} \\ v_2 & w_{21} & w_{22} \end{array}$$

Proposition 4.8 Every extent of R is an extent of $R^{''}$.

Proposition 4.9 A map ρ from $L(R)$ to $L(R'')$ is supermum-preserving and order-embedding. The map ρ is defined as follows:

$$\rho([X,Y]) = \begin{cases} [\phi, \alpha^{(R^{'})}(\phi)] & \text{if } X = \phi \\ [X, Y \bigcup Q] & \text{if } X = \phi and Y \bigcap \{b\} \times D_b = \phi \\ [\phi, (Y - W_b) \cup P] & \text{if } X = \phi and Y \bigcap \{b\} \times D_b \neq \phi \end{cases}$$

Where, P and W_b are same to ones in proposition 4.2.

Proposition 4.10 A map:

$$\mathrm{L}(R') \rightarrow \mathrm{L}(R''),\ [X,Y] \mapsto [X, Y - W_0]$$

is an isomorphism.

Now, we present connections between $\mathrm{L}(R''')$ and $\mathrm{L}(R')$. In the relation $R^{'''} = (V, B, I^{'''})$, the values of every sub-tuple $v_i \in V$ on all sub-attributes in B are not relevant to tuples in U . Hence, we can map the concept lattice $\mathrm{L}(R''')$ onto the maximal concept $[U, \alpha^{(R^{'})}(U)]$ in $R^{'}$, and the map is infimum(supermum)-preserving and order-preserving, as described in proposition 4.11.

Proposition 4.11 The map :

$$\mathrm{L}(R''') \rightarrow \mathrm{L}(R'),\ [X,Y] \mapsto [[U, \alpha^{(R^{'})}(U)]]$$

is infimum (supermum)-preserving and order-preserving.

So far, we discussed two kinds of methods for analyzing first-kind hybrid relations. The main contents are structural connections among $\mathrm{L}(R)$, $\mathrm{L}(R')$ and $\mathrm{L}(R_r)$, and ones among $\mathrm{L}(R)$, $\mathrm{L}(R'')$ and $\mathrm{L}(R_r)$.

V. SECOND-KIND HYBRID RELATIONS ANALYSIS METHODS

A similar method can be used to analyze second-kind hybrid relations.

A. *Transformation of Second-kind Hbrid Relations*

Given a second-kind hybrid relation $R = (U, A \bigcup \{b\}, I)$, for each sub-relation $R_r = (V, B, I_r)$, we regard the elements of $V \times B$ as new attributes and insert them into R , and then obtain a target relation $R^{'} = (U, A \bigcup (V \times B), I^{'})$, where $I^{'}$ is defined as follows: for any $r \in U$ and $a \in A^{'}$,

$$I'(r,a) = \begin{cases} I(r,a) & \text{if } a \in A \\ I_r(v,b) & \text{if } a = (v,b) \in V \times B \end{cases}$$

For instance, Example 2 can be transformed into R':

$$R' = \begin{array}{c|ccccc} & a & (v_1,b_1) & (v_1,b_2) & (v_2,b_1) & (v_2,b_2) \\ \hline r & v_{ra} & w_{r11} & w_{r12} & w_{r21} & w_{r22} \\ s & v_{sa} & w_{s11} & w_{s12} & w_{s21} & w_{s22} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{array}$$

B. Connections between $\mathrm{L}(R)$, $\mathrm{L}(R')$ *and* $\mathrm{L}(R_r)$

Similar to the insertion method described in section 4.1, we can obtain some similar propositions. Proposition 5.1 means that the insertion method is extent- preserving. Hence, there is a natural order-embedding map between the two concept lattices $\mathrm{L}(R)$ and $\mathrm{L}(R')$ .Proposition 5.2 concretely present the map, that is, it provides a method for determining a concept $[X,\alpha^{(R')}(X)]$ in a relation R' from every concept $[X,Y]$ in a relation R . The method can be described as follows:

(1) If all tuples in X share a same value on the attribute b, then they share a same value on every attribute $(v_i,b_j)\in V\times B$, and we denote the set of all these properties by S. Thus, we obtain an intent $\alpha^{(R')}(X)$ by replacing the property (b,w_b) by S in Y ;

(2) If there exist tuples in X having different values on the attribute b, then we obtain an intent $\alpha^{(R')}(X)$ by merging new properties common to the tuples in X to Y .

Contrasted to first-kind hybrid relations, second-kind hybrid relations have more semantics information among tuples, and hence there are denser concept lattices.

Proposition 5.1 Any concept extent of R is a concept extent of R'

Proposition 5.2 A map $\rho:\mathrm{L}(R)\rightarrow\mathrm{L}(R')$ between two concept lattices $\mathrm{L}(R)$ and $\mathrm{L}(R')$ is supermum-preserving and order-embedding. The map ρ is defined as follows:

$$\rho([X,Y])=\begin{cases}[\phi,\alpha^{(R')}(\phi)] & \text{if } X=\phi\\ [X,Y\bigcup P] & \text{if } X=\phi and Y\bigcap\{b\}\times D_b=\phi\\ [\phi,(Y-W_b)\cup P] & \text{if } X=\phi and Y\bigcap\{b\}\times D_b\neq\phi\end{cases}$$

where, P and W_b are same to ones in proposition 4.2.

When the elements of $V\times B$ are inserted into the relation R , the values on every attribute $a\in A$ for tuples in U remain constant. Hence, every new extent must be an extent of a property or the intersection of some extents of some properties in $(V\times B)\times D$, where $D=\{0,1,2,3,4,5\}$. Now, we analyze connections among concepts in sub-relations and ones in R', in order to explain how to rich concept intents and to establish new concept extents.

Our basic idea is as follows: by observing all sub-relations R_r , $r\in U$, we can define a relation $(v_i,b_j)\in V\times B$ on U for every pair $(v_i,b_j)\in V\times B$. Formally, for any $r,s\in U, r\theta(v_i,b_j)s$ if and only if $I_r(v_i,b_j)=I_s(v_i,b_j)$. Obviously, $\theta(v_i,b_j)$ is an equivalence relation on U . Thus, we obtain an equivalence relation $\theta(v_i,b_j)$, and further obtain a set of equivalence classes $U/\theta(v_i,b_j)$. Every equivalence class is exactly a property extent of the relation R' . Consequently, the intersection of equivalence classes of any finite different equivalence relations is also a concept extent of the relation Rt. The above descriptions can be formalized by the following propositions. Proposition 5.3 describes what every property extent is. Corollary 5.4-5.6 present methods for finding (new) extents of the relation R', and proposition 4.15 means that common concepts of all sub-relations can be mapped onto the maximal concept of the relation R'.

Proposition5.3 For any $(v_i,b_j)\in V\times B$, let

$$[r]\theta(v_i,b_j)=\{s\in U: r\theta(v_i,b_j)s\},$$

then $[r]\theta(v_i,b_j)$, which is an equivalence class of $\theta(v_i,b_j)$ containing r is the extent of a property $((v_i,b_j),I_r(v_i,b_j))$.

There are the following connections between equivalence classes of $\theta(v_i,b_j)$ and concepts in $R_r(r\in U)$:

Corollary5.1 For any $r\in U$ and $(v_i,b_j)\in V\times B$. Let the set

$$S_{(v_i,b_j)r}=\{s\in U:\exists C\in\mathrm{L}(R_S)(v_i\in Ext(C)\wedge(b_j,I_r(v_i,b_j))\in Int(C))\}$$

then $S_{(v_i,b_j)r}=[r]\theta_{(v_i,b_j)}$ and $((v_i,b_j),I_r(v_i,b_j))\in\alpha^{(R')}(S_{(v_i,b_j)r})$ hold in R'.

Corollary 5.2 For any $r\in U$, $\bigcap_{(v_i,b_j)\in V\times B} S_{(v_i,b_j)r}$ is an extent of R'.

By corollary 5.1 and proposition 5.3, for any $r\in U$, $(v_i,b_j)\in V\times B$, we know that $S_{(v_i,b_j)r}$ is an extent of R', and hence the intersection of all set is also an extent of R'.

Corollary5.3 Let $V=\{v_1,v_2,\ldots,v_n\}$, $B=\{b_1,b_2,\ldots,b_m\}$. Assume the corresponding equivalence relations on U are $\theta_{(v_1,b_1)}$, $\theta_{(v_2,b_1)}$, . . . , and $\theta_{(v_n,b_m)}$,then for any n-tuple $N\in U/\theta_{(v_1,b_1)}\times U/\theta_{(v_2,b_1)}\times\ldots\times U/\theta_{(v_n,b_m)}$, the intersection of any finite components in N is also a concept extent of the relation R'.

Proposition 5.4 Let $S=\bigcap_{r\in U}\mathrm{L}(R_r)$, then a map

$$\delta: S\rightarrow\mathrm{L}(R'),\ [X,Y]\mapsto[U,\alpha^{(R')}(U)]$$

is infimum(supermum)-preserving and order-preserving.

Based on above propositions, we can construct the concept lattice $\mathrm{L}(R')$ from the concept lattices $\mathrm{L}(R)$ and $\mathrm{L}(R_r)(r\in U)$. The key idea is that $\mathrm{L}(R)$ is used as a basic conceptual hierarchy to support construction of $\mathrm{L}\left(R^{'}\right)$, that is, inserting $\mathrm{L}\left(R_r\right)(r \in U)$ into $\mathrm{L}\left(R\right)$ can lead to a richer concept lattice. The process will give rise to richer concept lattice, which exactly is the concept lattice of the relation $R^{'}$.

So far, we have analyzed two kind of simple hybrid relations: first-kind hybrid relations and second-kind ones. Our method can be used to process those complex hybrid relations that they are represented a triple $(U, A \cup M, I)$.

VI. CONCLUSIONS

The paper proposed an approach towards analyzing some complex relations in hybrid relational databases. Using the approach, we can extract formal concepts from multiple relations. The approach has the following characteristics. Firstly, with the suitable redefined Galois Connection, we not only avoids the problems such as having to use many attribute scales and the arbitrariness of user's definition of a scale in scaling methods, but can speedy process multiple relations. Thus, compared to scaling methods, our method is more straight-forward. Secondly, by inserting the elements of sub-tuples × sub- attributes into a main relation R, we add the dimension of attributes in R, and hence not only expand concept intents of the main relation R, but refine the closure systems of concept extents of the main relation R. Consequently, there exists a natural order-embedding map from $\mathrm{L}(R)$ into $\mathrm{L}(R')$, which is also supermum-preserving.

By the insertion method, we analyzed two kinds of hybrid relations, which include relational information. We first transform all sub-relations into a main relation without losing any information, and then obtain a target relation. In the paper, we mainly analyzed structural connections among concept lattices. As an important connection, some relevant concepts in sub-relations can determine some concept extent of the corresponding target relation, and the common information of these concepts is transformed into the intent. By using the connection, we can speedy determine all new concept extents resulted in by the insertion operation.

Introducing FCA into hybrid relations analysis has some important advantages. Firstly, FCA offers a formalization method by mathematizing concepts that are understood as units of thought constituted by their extent and intent. Each concept in a relation is defined as maximal sets of tuples that share the same properties. This representation method perfectly reflects the human conceptual thinking. Secondly, a line diagram not only vividly exhibits the generalization/specialization relationship among concepts in a relation, but compactly visualizes multiple relations (consisting of a main relation and its sub-relations) in a hybrid relational database. Thirdly, based on the extracted concept lattices, one can not only extract some interesting rules or function dependence rules which have quite good or more better classified effect than other sorters(14), but more effectively process information retrieval problems than using other conventional information retrieval methods[11].

Several problems remain unsolved. One of the interesting problems is how to analyze those hybrid relations with columns having null values or whose sub-relations having different sub-tuples.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous referees for their careful reading and helpful comments, which led to an improved presentation of the manuscript. This work is supported by the Natural Science Foundation of China under Grant No. 61572284 and 61502272, the Promotive Research Fund for Excellent Young and Middle-aged Scientists of Shandong Province under Grant No. BS2014DX004 and BS2014DX005.

REFERENCES

[1] A AZZA, B P KAMIL, A DANIEL, et al., "HadoopDB: An Architectural Hybrid of Map Reduce and DBMS Technologies for Analytical Workloads", VIDB09(2009), pp, 922-933.2009.

[2] S BASHIR, A RAUF BAIG, "Performance Analysis of Frequent Item set Mining Using Hybrid Database Representation Approach", Multi-topic Conference, INMIC'06, IEEE 2006, pp 237-243, 2006.

[3] R BELOHAYEK, J KONECNY, Scaling, "Granulation, and Fuzzy Attributes in Formal Concept Analysis", FUZZ-IEEE 2007, IEEE International Conference on Fuzzy Systems, pp 1-6, 2007.

[4] R BELOHLAYEK, BERNARD DE BAETS, J KONECNY, "Granularity of Attributes in Formal Concept Analysis", Information Sciences, Vol. 260, pp.149-170, 2014.

[5] R BERGHAMMER, M WINTER, "Decomposition of Relations and Concept Lattices", Fundamenta Informaticae, Vol.126, pp 37-82, 2013.

[6] P CIMIANO, A HOTHO, S STAAB, "Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis", Journal of Artificial Intelligence Research, Vol. 24, pp 305-339, 2005.

[7] E F CODD, "A Relational Model of Data for Large Shared Databases", Communications of the ACM, Vol. 13, pp 377-387,1970.

[8] P EKLUND, J DUCROU, F DAU, "Concept Similarity and Related Categories in Information Retrieval using Formal Concept Analysis", International Journal of General Systems, Vol.41, pp 826-846,2012.

[9] B GANTER, R WILLE, "Formal Concept Analysis: Mathematical Foundations", Springer-Verlag, 1999.

[10] R GUGISCH, "Many-Valued Context Analysis Using Descriptions", H.Delugach and G.Stumme(Eds.): ICCS'01, LNAI 2120, pp 157-168,2001.

[11] K Y HU, Y C LU, C Y SHI, "An Integrated Mining Approach for Classification and Association Rule Based on Concept Lattice", Journal of Software, Vol. 11, pp 1478-1484,2000. (in Chinese)

[12] M HUCHARD, M ROUANE HACENE, C ROUME, P VALTCHEY, "Relational Concept Discovery in Structured Datasets", Ann Math ArtifIntell, Vol. 49, pp 39-76,2007.

[13] S JEFF, V RADEK, S BART, et al., "F1: a distributed SQL database that scales", Proceedings of the VLDB Endowment, Vol.6, pp 1068-1079,2013.

[14] F JIANG, Y F SUI, C G CAO, "Formal Concept Analysis in Relational Database and Rough Relational Database", Fundamenta Informaticae, Vol. 80, pp 435-451,2007.

[15] P JONAS, O K SERGEI, I I DMITRY, D GUIDO, "Formal Concept Analysis in knowledge processing: A survey on models and techniques", Expert Systems with Applications, Vol.40, pp 6601-6623,2013.

[16] R K JONATHAN, C P NOLAH, W C MARKUS, "WholeCellSimDB: A hybrid Relational/HDF Database for Whole-cell Model Predictions", Database, 2014, pp 1-8, 2014.

[17] C D JUAN, M JESS, "Multi-adjoint relation equations: Definition, properties and solutions using concept lattices", Information Sciences, Vol. 253, pp 100-109, 2013.

[18] M KAYTOUE, S O KUZNETSOY, A NAPAPOLI, S DUPLESSIS, "Mining Gene Expression Data with Pattern Structures in Formal Concept Analysis", Information Sciences, Vol.181, pp 1989-2001,2011.

[19] J KO¨TTERS, "Concept Lattices of a Relational Structure", Lecture Notes in Computer Science, Vol. 7735, pp 301-310,2013.

[20] Y X LEI, Y F SUI, C G CAO, "Normalized-scale Relations and Their Concept Lattices in Relational Databases", Fundamenta Informaticae, Vol. 93, pp 393-409, 2009.

[21] Y X LEI, Y F SUI, B X CAO, "Formal Concept Analysis in Hybrid Relational Databases", International Review on Computers and Software, Vol. 7, pp 2904-2910, 2012.

[22] K LE´ONARD, M ROKIA, B ABDE´LILAH, V JEAN, "Generalized Pattern Extraction from Concept Lattices", Ann Math Artif Intell, Vol. 72, pp 151-168,2014.

[23] C M EUGENIA, M JESS MEDINA, R P ELOISA, "Attribute reduction in multi-adjoint concept lattices", Information Sciences, Vol. 294, pp 41-56, 2015.

[24] R MISSAOUI, L NOURINE, Y RENAUD, "Computing Implications with Negation from a Formal Context", Fundamenta Informaticae, Vol. 115, pp 357-375,2012.

[25] R H MOHAMED, H MARIANNE, N AMEDEO, V PETKO, "Relational Concept Analysis: Mining Concept Lattices from Multi-relational Data", Annals of Mathematics and Artificial Intelligence, Vol. 67, pp 81-108, 2013.

[26] A MUTLU, P SENKUL, Y KAYURUCU, "Improving the scalability of ILP- based multi-relational concept discovery system through parallelization", Knowledge-Based Systems, Vol.27, pp 352-368, 2012.

[27] A NEZNANOY, O SERGEI, KUZNETSOY, "Information Retrieval and Knowledge Discovery with FCART", In: Claudio Carpineto, Sergei O. Kuznetsov, Amedeo Napoli, Eds., Proceedings of the Workshop Formal Concept Analysis Meets Information Retrieval (FCAIR 2013), pp 74-82, 2013.

[28] J POELMANS, D I IGNATOV, S O KUZNETSOY, G.DEDENE, "Fuzzy and Rough Formal Concept Analysis: a survey", International Journal of General Systems, Vol.43, pp 105-134, 2014.

[29] S PREDIGER, G. STUMME, "Theory-driven Logical Scaling: Conceptual Information Systems meet Description Logics", http://citeseer.ist.psu.edu/246950.html.

[30] S PREDIGER, "Logical Scaling in Formal Concept Analysis", LNCS 1257, pp 332-341, 1997.

[31] A E QADI, D. ABOUTAJDINE, Y ENNOUARY, "Formal Concept Analysis for Information Retrieval", International Journal of Computer Science and In- formation Security, Vol.7, pp 119-125, 2010.

[32] K S QU, Y H ZHAI, J Y LIANG, D Y LI. "Representation and Extension of Rough Set Theory Based on Formal Concept Analysis", Journal of Software, Vol. 18, pp 2174-2182, 2007.(in Chinses)

[33] O SERGEI, KUZNETSOY, "Fitting Pattern Structures to Knowledge Discovery in Big Data", P. Cellier, F. Distel, and B. Ganter (Eds.): ICFCA 2013, LNAI 7880, pp 254-266, 2013.

[34] D X WANG , Q XIE, et al., "Analysis of Association Rule Mining on Quantitative Concept Lattice", Lecture Notes in Computer Science, Vol.7530, pp 142-149, 2012..