Research Article

A Novel Concept Acquisition Approach Based on Formal Contexts

Ting Qian and Ling Wei

School of Mathematics, Northwest University, Xi'an, Shaanxi 710069, China

Correspondence should be addressed to Ling Wei; wl@nwu.edu.cn

Received 28 June 2014; Accepted 12 July 2014; Published 23 July 2014

Academic Editor: Yunqiang Yin

Copyright © 2014 T. Qian and L. Wei. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As an important tool for data analysis and knowledge processing, formal concept analysis (FCA) has been applied to many fields. In this paper, we introduce a new method to find all formal concepts based on formal contexts. The amount of intents calculation is reduced by the method. And the corresponding algorithm of our approach is proposed. The main theorems and the corresponding algorithm are examined by examples, respectively. At last, several real-life databases are analyzed to demonstrate the application of the proposed approach. Experimental results show that the proposed approach is simple and effective.

1. Introduction

Formal concept analysis (FCA), proposed by Wille in 1982 [1], is a field of applied mathematics based on the mathematization of concept and conceptual hierarchy. It thereby activates mathematical thinking for conceptual data analysis and knowledge processing. FCA starts with a formal context defined as a triple containing an object set, an attribute (property) set, and a binary relation between the object set and the attribute set. A formal concept is a pair (object subset, attribute subset) induced by the binary relation, and a concept lattice is an ordered hierarchical structure of formal concepts. A formal context in FCA corresponds to a special information system with input data being two-valued in rough set theory [2].

Most of the researches on FCA concentrate on the following topics: construction and pruning algorithm of concept lattices [3, 4]; relationship between FCA and rough sets [5–10]; acquisition of rules [11, 12]; reduction of concept lattices [6, 10, 13]. FCA is also proved to be useful in many fields, such as the organization of web search results on a hierarchical structure of concepts based on common topics [14], information retrieval [15, 16], hierarchical analysis of software code [17–19], visualization in software engineering [19, 20], detecting suspects in human trafficking [21], analysis of questionnaire data [22], and mining gene expression data

[23]. Further references to applications of FCA can be found in [14, 24].

Formal concepts are very important notions of FCA. And intents and extents are also very important elements of formal concepts. The set of intents (extents) is isomorphic to the corresponding concept lattice under the order relationship “ $\supseteq$ ” (“ $\subseteq$ ”). So, if the set of intents is determined, the corresponding concept lattice is identified. Thus, obtaining all intents or extents is very important. Generally, the basic way to obtain all intents or extents is via their definitions. If there are n objects, then we should calculate 2^n times to obtain all intents. Obviously, the computational costing is very huge. To solve this problem, we give a new method to obtain all intents. And correspondingly, the formal concepts are determined.

This paper is organized as follows. In Section 2, we briefly review some basic notions related to FCA. In Section 3, a novel concept acquisition approach is introduced and some related conclusions are given. In Section 4, the corresponding algorithm is proposed and experimental results are shown to illustrate the validity of our method. Finally, conclusions are drawn in Section 5.

2. Preliminaries

In this section, we recall some basic notions and properties in FCA.

Definition 1 (see [24]). A formal context (G, M, I) consists of two sets G and M and a relation I between G and M . The elements of G are called the objects and the elements of M are called the attributes of the context. In order to express that an object g is in a relation I with an attribute m , we write gIm or $(g, m) \in I$ and read it as “the object g has the attribute m .”

With respect to a formal context (G, M, I) , Ganter and Wille [24] defined a pair of dual operators for any $A \subseteq G$ and $B \subseteq M$ by

$$\begin{aligned} A^* &= \{m \in M \mid gIm \ \forall g \in A\}, \\ B' &= \{g \in G \mid gIm \ \forall m \in B\}. \end{aligned} \quad (1)$$

A formal context is called canonical if $\forall g \in G, g^* \neq \emptyset, g^* \neq M$, and $\forall m \in M, m' \neq \emptyset, m' \neq G$. We assume that all the formal contexts we study in the sequel are finite and canonical.

Let (G, M, I) be a formal context. $\forall A_1, A_2, A \subseteq G, \forall B_1, B_2, B \subseteq M$; the following properties hold.

- (1) $A_1 \subseteq A_2 \Rightarrow A_2^* \subseteq A_1^*, B_1 \subseteq B_2 \Rightarrow B_2' \subseteq B_1'$.
- (2) $A \subseteq A^{*'} , B \subseteq B'^*$.
- (3) $A^* = A^{*'} , B' = B'^*$.
- (4) $A \subseteq B' \Leftrightarrow B \subseteq A^*$.
- (5) $(A_1 \cup A_2)^* = A_1^* \cap A_2^*, (B_1 \cup B_2)' = B_1' \cap B_2'$.
- (6) $(A_1 \cap A_2)^* \supseteq A_1^* \cup A_2^*, (B_1 \cap B_2)' \supseteq B_1' \cup B_2'$.

If $A^* = B$ and $B' = A$, then (A, B) is called a formal concept, where A is called the extent of the formal concept and B is called the intent of the formal concept. For any $g \in G$, a pair $(g^{*'}, g^*)$ is a formal concept and is called an object concept. Similarly, for any $m \in M$, a pair (m', m'^*) is a formal concept and is called an attribute concept. The family of all formal concepts of (G, M, I) forms a complete lattice that is called the concept lattice and is denoted by $L(G, M, I)$. For any $(A_1, B_1), (A_2, B_2) \in L(G, M, I)$, the partial order is defined by

$$(A_1, B_1) \leq (A_2, B_2) \iff A_1 \subseteq A_2 (\iff B_1 \supseteq B_2). \quad (2)$$

And the infimum $\wedge$ and supremum $\vee$ of (A_1, B_1) and (A_2, B_2) are defined by

$$\begin{aligned} (A_1, B_1) \wedge (A_2, B_2) &= (A_1 \cap A_2, (B_1 \cup B_2)^{*'}), \\ (A_1, B_1) \vee (A_2, B_2) &= ((A_1 \cup A_2)^{*'}, B_1 \cap B_2), \end{aligned} \quad (3)$$

respectively.

Definition 2 (see [10]). Let (G, M, I) be a formal context. $|G| < \infty$ and $|M| < \infty$. Denote $L_M(G, M, I) = \{Y \mid Y \subseteq M \text{ and } Y \text{ is an intent of } (G, M, I)\}$.

Example 3 (see [10]). Table 1 is a formal context (G, M, I) . $G = \{1, 2, 3, 4, 5, 6\}$ is an object set and $M = \{a, b, c, d, e\}$ is an attribute set. The corresponding concept lattice $L(G, M, I)$ is shown in Figure 1, in which every set is denoted directly by listing its elements except G, M , and $\emptyset$.

FIGURE 1: $L(G, M, I)$ of Example 3.

TABLE 1: A formal context (G, M, I) .

G	a	b	c	d	e
1	$\times$		$\times$	$\times$	$\times$
2	$\times$		$\times$		
3		$\times$			$\times$
4		$\times$			$\times$
5	$\times$				
6	$\times$	$\times$			$\times$

3. A Novel Concept Acquisition Approach

The basic way to obtain all intents or extents is via their definitions. If there are n objects, then we should calculate 2^n times to get all intents. Obviously, the amount of computation is very large. So our paper presents a new approach to solve the problem. In this section, we give this new method and some theorems to explain its rationality and validity.

Before giving the method, we firstly propose a related definition.

Definition 4. Let (G, M, I) be a formal context. $|G| < \infty, |M| < \infty$. Denote $\alpha_n = \{X \mid X \subseteq G \text{ and } |X| = n\}$, $\beta_n = \{Y \mid Y \subseteq M, Y = X^* \text{ and } X \in \alpha_n\}$, $n = 1, 2, \dots, |G|$, where $|\cdot|$ presents the cardinal of a set.

Since the method in this paper is aimed at obtaining all intents, we use subsets α_n of G to determine subsets β_n of M . On the contrary, if we want to obtain all extents, the subsets of M can be used to determine subsets of G . This point has been illustrated in the sequel.

Theorem 5. β_1 is an intent of an object concept.

Proof. The proof is immediately obtained from Definitions 1 and 4. $\square$

Theorem 6. *If there exists n ($n = 1, 2, \dots, |G|$) such that $\beta_{n+1} \subseteq \beta_n$, then $\beta_{n+2} \subseteq \beta_{n+1}$.*

Proof. Suppose $Y \in \beta_{n+2}$. By Definition 4, there exists $X_{n+2} \in \alpha_{n+2}$ such that $Y = X_{n+2}^*$.

Since $X_{n+2} \neq \emptyset$, there exists $a \in X_{n+2}$ such that $Y = X_{n+2}^* = (X_{n+2} \setminus \{a\} \cup \{a\})^* = (X_{n+2} \setminus \{a\})^* \cap \{a\}^*$. Noting that $|X_{n+2} \setminus \{a\}| = n+1$, we have $X_{n+2} \setminus \{a\} \in \alpha_{n+1}$. Moreover, from $\beta_{n+1} \subseteq \beta_n$, we know that there exists $X_n \in \alpha_n$ satisfying $(X_{n+2} \setminus \{a\})^* = X_n^*$; that is, $Y = X_n^* \cap \{a\}^* = (X_n \cup \{a\})^*$.

Now, we discuss two cases to prove $Y \in \beta_{n+1}$.

The one case is that $a \notin X_n$. In this case, $|X_n \cup \{a\}| = n+1$. Thus, $Y = X_n^* \cap \{a\}^* = (X_n \cup \{a\})^* \in \beta_{n+1}$.

The other one is that $a \in X_n$. In this case, $Y = X_n^* \cap \{a\}^* = X_n^*$. Because $X_{n+2} \setminus X_n \neq \emptyset$, there exists $b \in X_{n+2} \setminus X_n$ such that $X_n \subseteq X_n \cup \{b\} \subseteq X_n \cup X_{n+2}$. Therefore, we have $X_n^* \supseteq (X_n \cup \{b\})^* \supseteq (X_n \cup X_{n+2})^*$; that is, $X_n^* \supseteq (X_n \cup \{b\})^* \supseteq X_n^* \cap X_{n+2}^*$. That means, $Y \supseteq (X_n \cup \{b\})^* \supseteq Y$. Therefore, we have $Y = (X_n \cup \{b\})^*$. Thereby, we can obtain $Y \in \beta_{n+1}$.

To sum up the above two cases, $\beta_{n+2} \subseteq \beta_{n+1}$ holds. $\square$

Theorem 6 guarantees the convergence of Algorithm 2 involved in the sequel.

Corollary 7. *If there exists n ($n = 1, 2, \dots, |G|$) such that $\beta_{n+1} \subseteq \beta_n$, then for any $m, m \geq n$, we have $\beta_{m+1} \subseteq \beta_m \subseteq \beta_n$.*

Proof. According to the condition $\beta_{n+1} \subseteq \beta_n$, we have $\beta_{n+2} \subseteq \beta_{n+1}$ by Theorem 6. Using Theorem 6 repeatedly, we can easily obtain the following results: $\beta_{m+1} \subseteq \beta_m \subseteq \beta_{m-1} \subseteq \beta_{m-2} \subseteq \dots \subseteq \beta_{n+1} \subseteq \beta_n$. $\square$

Theorem 8. *Suppose $Y \in \beta_k$, $1 \leq k \leq |G|$, $1 < m < k$; if $Y \notin \beta_m$, then we have $Y \notin \beta_{m-1}$.*

Proof. We will adopt the proof by contradiction.

Suppose $Y \in \beta_{m-1}$; there is $X_{m-1} \in \alpha_{m-1}$ satisfying $Y = X_{m-1}^*$; according to the condition and Definition 4, there is $X_k \in \alpha_k$ satisfying $Y = X_k^*$.

Because $X_k \setminus X_{m-1} \neq \emptyset$, there exists $a \in X_k \setminus X_{m-1}$ such that $|X_{m-1} \cup \{a\}| = m$; that is, $X_{m-1} \cup \{a\} \in \alpha_m$. Obviously, $X_{m-1} \subseteq X_{m-1} \cup \{a\} \subseteq X_{m-1} \cup X_k$, and thus, $X_{m-1}^* \supseteq (X_{m-1} \cup \{a\})^* \supseteq (X_{m-1} \cup X_k)^*$. That is, $X_{m-1}^* \supseteq (X_{m-1} \cup \{a\})^* \supseteq X_{m-1}^* \cap X_k^*$. That means $Y \supseteq (X_{m-1} \cup \{a\})^* \supseteq Y$. Therefore, we have $Y = (X_{m-1} \cup \{a\})^*$. From Definition 4, we know that $Y \in \beta_m$. It is a contradiction with $Y \notin \beta_m$.

Therefore, $Y \notin \beta_{m-1}$ holds. $\square$

Corollary 9. *Suppose $Y \in \beta_k$, $1 \leq k \leq |G|$, and $Y \notin \beta_{k-1}, \beta_k \subseteq \beta_{k-1}$; then for any $m, 1 \leq m < k$, $Y \notin \beta_m$.*

Proof. Because $Y \in \beta_k$, and $Y \notin \beta_{k-1}$, we have $Y \notin \beta_{k-2}$ by Theorem 8. Using Theorem 8 repeatedly, we have $Y \notin \beta_m$. $\square$

Theorem 10. *Suppose (G, M, I) is canonical; then $\beta_{n+1} \subseteq \beta_n$ if and only if $\cup_{i=1}^n \beta_i \cup \{M\} = L_M(G, M, I)$.*

Proof.

Necessity. Suppose $\beta_{n+1} \subseteq \beta_n$.

TABLE 2: Living beings and water (G, M, I) .

G	b	c	d	e	f	g	h	i
1	$\times$					$\times$		
2	$\times$					$\times$	$\times$	
3	$\times$	$\times$				$\times$	$\times$	
4		$\times$				$\times$	$\times$	$\times$
5	$\times$		$\times$		$\times$			
6	$\times$	$\times$	$\times$		$\times$			
7		$\times$	$\times$	$\times$				
8		$\times$	$\times$		$\times$			

For any $Y \in \cup_{i=1}^n \beta_i \cup \{M\}$, if $Y = \{M\}$, then it is evident that $Y \in L_M(G, M, I)$. If $Y \neq \{M\}$, then there exists n_0 such that $Y \in \beta_{n_0}$. By Definition 4, there exists $X_{n_0} \in \alpha_{n_0}$ such that $Y = X_{n_0}^*$. Obviously, $Y \in L_M(G, M, I)$. Since Y is arbitrary and $M \in L_M(G, M, I)$, we have $\cup_{i=1}^n \beta_i \cup \{M\} \subseteq L_M(G, M, I)$.

For any $Y \in L_M(G, M, I)$, by Definition 2 and properties of the operator $*$, we have $Y = (Y^*)^*, Y^* \subseteq G$. Without loss of the generality, we can suppose $|Y^*| = m$. If $m \leq n$, then $Y \in \beta_m \subseteq \cup_{i=1}^n \beta_i$ by Definition 4. If $m > n$, from above Corollary 7, we have $Y \in \beta_m \subseteq \beta_n \subseteq \cup_{i=1}^n \beta_i$. Since $M \in \cup_{i=1}^n \beta_i \cup \{M\}$ and Y is arbitrary, we obtain $\cup_{i=1}^n \beta_i \cup \{M\} \supseteq L_M(G, M, I)$.

Therefore, $\cup_{i=1}^n \beta_i \cup \{M\} = L_M(G, M, I)$.

Sufficiency. We assume $\beta_{n+1} \not\subseteq \beta_n$ and prove $\cup_{i=1}^n \beta_i \cup \{M\} \neq L_M(G, M, I)$.

If $\beta_{n+1} \not\subseteq \beta_n$, then there is $Y \in \beta_{n+1}$, but $Y \notin \beta_n$. From Corollary 9, for any $m, 1 \leq m < n$, we have $Y \notin \beta_m$. So, $Y \notin \cup_{i=1}^n \beta_i$. (G, M, I) is canonical and $Y \in \beta_{n+1}$, so $Y \neq M$. Thus, $Y \notin \cup_{i=1}^n \beta_i \cup \{M\}$.

On the other hand, since $Y \in \beta_{n+1}$, by Definition 4, there exists $X_{n+1} \in \alpha_{n+1}$, such that $Y = X_{n+1}^*$. By the definition of $L_M(G, M, I)$, $Y \in L_M(G, M, I)$.

That means there exists one set Y such that $Y \notin \cup_{i=1}^n \beta_i \cup \{M\}$, but $Y \in L_M(G, M, I)$. Therefore, $\cup_{i=1}^n \beta_i \cup \{M\} \neq L_M(G, M, I)$. $\square$

Theorem 10 gives a sufficient and necessary condition and computation method to find $L_M(G, M, I)$. Now, the process to calculate all intents is summarized as follows. *Step 1.* Calculate α_1 and β_1 by Definition 4. *Step 2.* Calculate α_2 and β_2 by Definition 4. If $\beta_2 \subseteq \beta_1$, then the set of intents is $\beta_1 \cup \{M\}$. Otherwise, we proceed Step 3. *Step 3.* Calculate α_3 and β_3 by Definition 4. If $\beta_3 \subseteq \beta_2$, then the set of intents is $\beta_1 \cup \beta_2 \cup \{M\}$. Otherwise, calculate β_i ($1 \leq i \leq |G|$) continuously. The computation needs to stop at β_{n+1} which exactly meets $\beta_{n+1} \subseteq \beta_n$. Meanwhile, the set of intents is $\cup_{i=1}^n \beta_i \cup \{M\}$.

The merit of our method is that we do not need to calculate all β_i , $1 \leq i \leq |G|$ and the computation needs only to stop at β_{n+1} which exactly meets $\beta_{n+1} \subseteq \beta_n$. Now all the intents have been found and there is no extra computing.

In the following, we use an example in the literature [24] to examine the main results about the new method to find all intents of formal concepts.

The formal context in Table 2 is a minor revision of the famous example, a film “Living Beings and Water” [24]. Since we require all the formal contexts in this paper are canonical, we delete the attribute a (water) from the original formal context. The objects are living beings mentioned in the film and are denoted by $G = \{1, 2, 3, 4, 5, 6, 7, 8\}$, where 1 is leech, 2 is bream, 3 is frog, 4 is dog, 5 is spike-weed, 6 is reed, 7 is bean, and 8 is maize. And the attributes in $M = \{b, c, d, e, f, g, h, i\}$ are the properties which the film emphasizes: b : lives in water, c : lives on land, d : needs chlorophyll to produce food, e : two seed leaves, f : one seed leaf, g : can move around, h : has limbs, and i : suckles its offspring.

The corresponding concept lattice $L(G, M, I)$ of this formal context is shown in Figure 2.

We calculate α_i and β_i ($i = 1, 2, 3, \dots, 8$) firstly:

$$\alpha_1 = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}\},$$

$$\beta_1 = \{\{b, g\}, \{b, g, h\}, \{b, g, c, h\}, \{c, g, h, i\}, \{b, d, f\},$$

$$\{b, c, d, f\}, \{c, d, e\}, \{c, d, f\}\},$$

$$\alpha_2 = \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{1, 5\}, \{1, 6\}, \{1, 7\}, \{1, 8\}, \{2, 3\}, \\ \{2, 4\}, \{2, 5\}, \{2, 6\}, \{2, 7\}, \{2, 8\}, \{3, 4\}, \{3, 5\}, \{3, 6\}, \\ \{3, 7\}, \{3, 8\}, \{4, 5\}, \{4, 6\}, \{4, 7\}, \{4, 8\}, \{5, 6\}, \{5, 7\}, \\ \{5, 8\}, \{6, 7\}, \{6, 8\}, \{7, 8\}\},$$

$$\beta_2 = \{\{b, g\}, \{g\}, \{b\}, \emptyset, \{b, g, h\}, \{g, h\}, \{c, g, h\}, \{b, c\}, \{c\}, \\ \{b, d, f\}, \{d\}, \{d, f\}, \{c, d\}, \{c, d, f\}\},$$

$$\alpha_3 = \{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 5\}, \{1, 2, 6\}, \{1, 2, 7\}, \{1, 2, 8\}, \\ \{2, 3, 4\}, \{2, 3, 5\}, \{2, 3, 6\}, \{2, 3, 7\}, \{2, 3, 8\}, \{3, 4, 5\}, \\ \{3, 4, 6\}, \{3, 4, 7\}, \{3, 4, 8\}, \{4, 5, 6\}, \{4, 5, 7\}, \{4, 5, 8\}, \\ \{5, 6, 7\}, \{5, 6, 8\}, \{6, 7, 8\}, \{1, 3, 4\}, \{1, 3, 5\}, \{1, 3, 6\}, \\ \{1, 3, 7\}, \{1, 3, 8\}, \{2, 4, 5\}, \{2, 4, 6\}, \{2, 4, 7\}, \{2, 4, 8\}, \\ \{3, 5, 6\}, \{3, 5, 7\}, \{3, 5, 8\}, \{4, 6, 7\}, \{4, 6, 8\}, \{5, 7, 8\}, \\ \{1, 4, 5\}, \{1, 4, 6\}, \{1, 4, 7\}, \{1, 4, 8\}, \{2, 5, 6\}, \{2, 5, 7\}, \\ \{2, 5, 8\}, \{3, 6, 7\}, \{3, 6, 8\}, \{4, 7, 8\}, \{1, 5, 6\}, \{1, 5, 7\}, \\ \{1, 5, 8\}, \{2, 6, 7\}, \{2, 6, 8\}, \{3, 7, 8\}, \{1, 6, 7\}, \\ \{1, 6, 8\}, \{2, 7, 8\}, \{1, 7, 8\}\},$$

$$\beta_3 = \{\{b, g\}, \{g\}, \{b\}, \emptyset, \{b, g, h\}, \{g, h\}, \{c, g, h\}, \\ \{b, c\}, \{c\}, \{b, d, f\}, \{d\}, \{d, f\}, \{c, d\}, \{c, d, f\}\}. \quad (4)$$

Similarly, we can calculate $\beta_4, \beta_5, \dots, \beta_8$ and find $\beta_2 \supseteq \beta_3 \supseteq \beta_4 \supseteq \beta_5 \supseteq \dots \supseteq \beta_8$. And we can also know $\beta_2 \not\supseteq \beta_1$. In fact, we only need to calculate $\beta_1, \beta_2, \beta_3$. Once we have $\beta_3 \subseteq \beta_2$, but $\beta_2 \not\subseteq \beta_1$, the computation can be stopped.

FIGURE 2: $L(G, M, I)$.

According to Theorem 10, the set of all the intents is $\beta_1 \cup \beta_2 \cup \{M\}$; that is, $L_M(G, M, I) = \{\{b, g\}, \{g\}, \{b\}, \emptyset, \{b, g, h\}, \{g, h\}, \{c, g, h\}, \{b, c\}, \{c\}, \{b, d, f\}, \{d\}, \{d, f\}, \{c, d\}, \{c, d, f\}, \{b, g, c, h\}, \{c, g, h, i\}, \{b, c, d, f\}, \{c, d, e\}, M\}$. These results are easily examined from Figure 2.

4. Algorithms and Experiments

4.1. Algorithms. Algorithm 1 is given based on Definition 1 completely.

Algorithm 2 is based on our approach presented by Theorem 10. Comparing with Algorithm 1, we add a condition to terminate the program.

The time complexity of Algorithm 2 is analyzed as follows.

Denote $N = \min\{|G|, |M|\}$; by Definition 4, we know the time complexity of Step I in Algorithms 1 or 2 is $O(C_N^i)$. So we can get two matters as follows.

- (1) The time complexity of algorithm is $O(2^N)$.
- (2) Suppose that Algorithm 2 will be terminated in the k th step; then the time complexity of Algorithm 2 is $O(\sum_{i=1}^k (C_N^i))$ by Theorem 10. We can easily get $O(\sum_{i=1}^k (C_N^i)) \leq O(2^N)$.

We present an example demonstrating performance of Algorithm 2. The database “patient and Ill symptoms” showed in Table 3 comes from UCI Machine Learning Repository [25]. Suppose there are 12 patients which are denoted by $1, \dots, 12$ and 8 symptoms of patients which are denoted by $a, \dots, h$, where a is headache, b is fever, c stands for painful limbs, d represents swollen glands in neck, e is cold, f is stiff neck, g is rash, and h is vomiting. Input the formal context and run the program; we obtain the set of all intents when $n = 2$:

$$\{\emptyset, \{b\}, \{a, b\}, \{b, e\}, \{g\}, \{b, e, g\}, \{a, b, f, h\}, \{a, b, c, e\}, \\ \{a, b, c, d, e, f, g, h\}\}.$$

```

(1) input context,
(2) step = 1,
(3) get first step set (),
(4) intent set = former step set,
(5) step++,
(6) later step set =  $\{\beta : \beta = \text{make arbitrary two ranks meet in the context}\}$ ,
(7) while (step  $\leq$  max rank of context) {former step set = later step set;
    intent set = union of later step set and intent set; step++;},
(8) output the set of intents.

```

ALGORITHM 1

```

(1) input context,
(2) step = 1,
(3) get first step set (),
(4) intent set = former step set,
(5) step++,
(6) later step set =  $\{\beta : \beta = \text{make arbitrary two ranks meet in the context}\}$ ,
(7) while (step  $\leq$  max rank of context) {if (later step set is not the subset of
    former step set) {former step set = later step set; intent set = union of
    later step set and intent set; step++;} else {return}},
(8) output the set of intents.

```

ALGORITHM 2

TABLE 3: Patients and ill symptoms (G, M, I).

G	a	b	c	d	e	f	g	h
1	X	X	X		X			
2	X	X				X		X
3		X			X		X	
4	X	X				X		X
5	X	X	X		X			
6		X			X		X	
7		X			X		X	
8							X	
9	X	X	X		X			
10							X	
11	X	X	X		X			
12	X	X				X		X

4.2. Experimental Results. In this section, we conduct some experiments to compare Algorithm 2 with Algorithm 1. In the experiments, four real life databases we selected are as follows:

- (1) Living beings and water [24] introduced in Section 4.1.
- (2) Patients and ill symptoms [25] introduced in Section 4.1.
- (3) *Bacterial Taxonomy* [26]. Data are presented for 6 species most of whom having data for more than one strain and 16 phenotypic characters (0 and 1). The species are *Escherichia coli* (*ecoli*), *Salmonella typhi* (*styphi*), *Klebsiella pneumoniae* (*kpneu*), *Proteus*

TABLE 4: A contrast between algorithms.

Data	$ G $	$ M $	$ I $	Time 1	Time 2	Efficiency
1	8	8	19	733	686	6.4%
2	12	8	9	733	718	2.1%
3	17	16	53	921	733	20.4%
4	130	6	23	1504	1217	19.1%

vulgaris (*pvul*), *Proteus morganii* (*pmor*), and *Serratia marcescens* (*smar*). The phenotypic characters are H2S, MAN, LYS, IND, ORN, CIT, URE, ONP, VPT, INO, LIP, PHE, MAL, ADO, ARA, and RHA.

- (4) *Membership of Developing Countries in Supranational Group* [24]. In this data, 130 developing countries are objects. Six properties (group of 77, nonaligned, least developed countries, most seriously affected countries, Organization of Petrol Exporting Countries, and African Caribbean and Pacific Countries) are attributes.

The results are shown in Table 4 and Figure 3, where Time 1 and Time 2 are the running time of Algorithms 1 and 2, respectively. $|I|$ presents the number of intents and the efficiency is equivalent to $(\text{Time 1} - \text{Time 2})/\text{Time 1}$. It can be seen that Algorithm 2 is much more efficient than Algorithm 1 along with the increase of $|I|$.

5. Conclusion

To find new methods to solve the difficult problems of the concept lattice construction is a hot problem. Constructing

FIGURE 3: Database analysis.

concept lattices is a novel research branch for data processing and data analysis. Different methods play essential roles in different problems. This paper first defines some basic notions. Based on the basic notion of intents, we obtain a new judgment method of finding all intents of formal concepts. Moreover, an example is given to explain the feasibility of this method. At last, we give the corresponding algorithm of this method and do the experiments to illustrate the effectiveness of this method.

For Algorithm 2, we have the following discussion which can be applied to real application. We can compare $|G|$ with $|M|$ of a formal context. If $|G| \leq |M|$, then we use subsets of G to determine subsets of M and output the set of intents. Otherwise, according to the duality principle, the subsets of M can be used to determine subsets of G and output the set of extents. We will improve the corresponding algorithm of this method in the future.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work is supported by Grants from the National Natural Science Foundation of China (no. 11371014, no. 11071281, and no. 61202206).

References

[1] R. Wille, "Restructuring lattice theory: an approach based on hierarchies of concepts," in *Ordered Sets*, I. Rival, Ed., pp. 445–470, Reidel, Dordrecht, The Netherlands, 1982.

[2] Z. Pawlak, "Rough sets," *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[3] L. K. Guo, F. P. Huang, Q. G. Li, and G. Q. Zhang, "Power contexts and their concept lattices," *Discrete Mathematics*, vol. 311, no. 18–19, pp. 2049–2063, 2011.

[4] S. O. Kuzetsov and S. A. Obiekov, "Comparing performance of algorithms for generation concept lattices," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 14, pp. 189–216, 2002.

[5] R. Bělohlávek, "Similarity relations in concept lattices," *Journal of Logic and Computation*, vol. 10, no. 6, pp. 823–845, 2000.

[6] W. X. Zhang, Y. Y. Yao, and Y. Leung, *Rough Set and Concept Lattice*, Xi'an Jiantong University Press, Xi'an, China, 2006.

[7] R. E. Kent, "Rough concept analysis: a synthesis of rough sets and formal concept analysis," *Fundamenta Informaticae*, vol. 27, no. 2–3, pp. 169–181, 1996.

[8] J. Saquer and J. S. Deogun, "Concept approximations based on rough sets and similarity measures," *International Journal of Applied Mathematics and Computer Science*, vol. 11, no. 3, pp. 655–674, 2001.

[9] J. J. Qi, L. Wei, and Z. Z. Li, "A partitional view of concept lattice," in *Lecture Notes in Computer Science*, vol. 10, pp. 74–83, 2005.

[10] L. Wei, *Reduction Theory and Approach to Rough Set and Concept Lattice*, Xi'an Jiaotong University, Xi'an, China, 2005.

[11] R. Godin, R. Missaoui, and H. Alaoui, "Incremental concept formation algorithms based on Galois (concept) lattices," *Computational Intelligence*, vol. 11, no. 2, pp. 246–267, 1995.

[12] B. Ganter, G. Stumme, and R. Wille, *Formal Concept Analysis: Foundations and Applications*, Springer, Berlin, Germany, 2005.

[13] W. X. Zhang, L. Wei, and J. J. Qi, "Attribute reduction in concept lattice based on discernibility matrix," in *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, vol. 3642 of *Lecture Notes in Computer Science*, pp. 157–165, Springer, Berlin, Germany, 2005.

[14] C. Carpineto and G. Romano, *Concept Data Analysis: Theory and Applications*, John Wiley and Sons, New York, NY, USA, 2004.

[15] W. C. Cho and D. Richards, "Improvement of precision and recall for information retrieval in a narrow domain: reuse of concepts by formal concept analysis," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI '04)*, pp. 370–376, September 2004.

[16] Y. Okubo and M. Haraguchi, "Finding conceptual document clusters with improved top-N formal concept search," in *Proceeding of the IEEE/WIC/ACM International Conference on Web Intelligence (WI '06)*, pp. 347–351, Hong Kong, China, December 2006.

[17] U. Dekel and Y. Gil, "Visualizing class interfaces with formal concept analysis," in *Proceedings of the 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA '03)*, pp. 288–289, Anaheim, Calif, USA, 2003.

[18] G. Snelting and F. Tip, "Understanding class hierarchies using concept analysis," *ACM Transactions on Programming Languages and Systems*, vol. 22, no. 3, pp. 540–582, 2000.

[19] P. Tonella, "Using a concept lattice of decomposition slices for program understanding and impact analysis," *IEEE Transactions on Software Engineering*, vol. 29, no. 6, pp. 495–509, 2003.

[20] V. Ganapathy, D. King, T. Jaeger, and S. Jha, "Mining security-sensitive operations in legacy code using concept analysis," in *Proceedings of the 29th International Conference on Software Engineering (ICSE '07)*, pp. 458–467, May 2007.

- [21] J. Poelmans, P. Elzinga, D. I. Ignatov, and S. O. Kuznetsov, "Semi-automated knowledge discovery: identifying and profiling human trafficking," *International Journal of General Systems*, vol. 41, no. 8, pp. 774–804, 2012.
- [22] R. Belohlavek, E. Sigmund, and J. Zaczal, "Evaluation of IPAQ questionnaires supported by formal concept analysis," *Information Sciences*, vol. 181, no. 10, pp. 1774–1786, 2011.
- [23] M. Kaytoue, S. O. Kuznetsov, A. Napoli, and S. Duplessis, "Mining gene expression data with pattern structures in formal concept analysis," *Information Sciences*, vol. 181, no. 10, pp. 1989–2001, 2011.
- [24] B. Ganter and R. Wille, *Formal Concept Analysis: Mathematical Foundations*, Springer, Berlin, Germany, 1999.
- [25] A. Asuncion and D. J. Newman, *UCI Machine Learning Repository*, University of California, School of Information and Computer Sciences, Irvine, Calif, USA, 2007.
- [26] A. H. Fielding, *Clustering and Classification Techniques for the Biosciences*, Cambridge University Press, London, UK, 2002.