Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 917639, 6 pages
http://dx.doi.org/10.1155/2014/917639

Research Article

On the Calculation of Formal Concept Stability

Hui-lai Zhi

School of Computer Science and Technology, Henan Polytechnic University, No. 2001, ShiJi Avenue, Jiaozuo 454000, China

Correspondence should be addressed to Hui-lai Zhi; zhihuilai@126.com

Received 27 January 2014; Revised 15 June 2014; Accepted 29 June 2014; Published 10 July 2014

Academic Editor: Heping Zhang

The idea of stability has been used in many applications. However, computing stability is still a challenge and the best algorithms known so far have algorithmic complexity quadratic to the size of the lattice. To improve the effectiveness, a critical term is introduced in this paper, that is, minimal generator, which serves as the minimal set that makes a concept stable when deleting some objects from the extent. Moreover, by irreducible elements, minimal generator is derived. Finally, based on inclusion-exclusion principle and minimal generator, formulas for the calculation of concept stability are proposed.

1. Introduction

When making scientific hypotheses about the cause of a natural phenomenon, some data should be gathered randomly to certain extent. A best hypothesis should be independent of this randomness, which means the hypothesis is not determined by any particular piece of data. In Kuznetsov's research, this sort of independence was called stability [1, 2], and it was used in many occasions such as succinct representation of lattice based taxonomies [3–5], jackknife estimation towards sample functions [6], and the work of Carnap on inductive logic [7].

However, computing stability of a concept was proved to be a #P-complete problem and the best algorithms known so far have algorithmic complexity quadratic to the size of the lattice [8]. In this paper we will reconsider this problem and give a new method to improve the computational efficiency.

2. Basic Definitions in FCA

Formal concept analysis (FCA) is a generally appropriate framework for building categories defined as object sets sharing some attributes, irrespectively of a particular domain of application [9–13]. This presents a convincing formal model of the philosophical notion of a concept characterized extensionally by the set of entities and intensionally by the set of attributes they have in common.

Before proceeding, we briefly recall the FCA terminology and properties [12, 13]. Given a formal context $K = (G, M, I)$, where G is called a set of objects, M is called a set of attributes, and the binary relation $I \subseteq G \times M$ specifies which objects have which attributes, the derivation functions $f(\cdot)$ and $g(\cdot)$ are defined for $A \subseteq G$ and $B \subseteq G$ as follows:

$$f(A) = \{m \in M \mid \forall g \in A : gIm\};$$
$$g(B) = \{g \in G \mid \forall m \in B : gIm\}. \quad (1)$$

$f(A)$ is the set of attributes common to all objects of A and $g(B)$ is the set of objects sharing all attributes of B, respectively.

A formal concept of the extent K is a pair (A, B), where $A \subseteq G$, $B \subseteq M$, $f(A) = B$, and $g(B) = A$. The set A is called the extent and B is called the intent of the concept (A, B).

A concept (A, B) is subconcept of (C, D) if $A \subseteq C$ (equivalently, $D \subseteq B$). In this case, (C, D) is called a superconcept of (A, B). We write $(A, B) \leq (C, D)$ and define the relations $\geq$, $<$, and $>$ as usual. If $(A, B) \leq (C, D)$ and there is no (E, F) such that $(A, B) < (E, F) < (C, D)$, then (A, B) is a lower neighbor of (C, D) and (C, D) is an upper neighbor of (A, B); notation: $(A, B) \prec (C, D)$ and $(C, D) \succ (A, B)$.

The set of all concepts ordered by relation $\leq$ forms a lattice, which is denoted by $B(K)$ and called the concept lattice of the context K. The relation defines the covering graph of $B(K)$.

Let $K = (G, M, I)$ be a formal context and $A, A_1, A_2 \subseteq G$, $B, B_1, B_2 \subseteq M$. Then the following propositions hold:

(1) $A_1 \subseteq A_2 \Rightarrow f(A_2) \subseteq f(A_1)$, $B_1 \subseteq B_2 \Rightarrow g(B_2) \subseteq g(B_1)$,

(2) $A \subseteq g(f(A))$, $B \subseteq f(g(B))$,

(3) $f(A) = f(g(f(A)))$, $g(B) = g(f(g(B)))$,

(4) $A \subseteq g(B) \Leftrightarrow B \subseteq f(A) \Leftrightarrow A \times B \subseteq I$.

3. Stability Calculation Based on Minimal Generator

Before proceeding, let us give two symbols which will be used frequently in the following discussion. For a given set A,

(1) $|A|$ indicates the cardinality of A;

(2) 2^A indicates the power set of A.

Definition 1 (see [1, 2, 8]). Let $K = (G, M, I)$ be a formal context and (A, B) a formal concept of K. The stability index σ of (A, B) is defined as follows:

$$\sigma(A, B) = \frac{|\{C \subseteq A \mid f(C) = B\}|}{2^{|A|}}. \quad (2)$$

Proposition 2. *Let $K = (G, M, I)$ be a formal context and (A, B) be a formal concept of K. If there is a set $A_1 \subseteq A$ with $f(A_1) = B$, then $f(A_2) = B$ where $A_1 \subseteq A_2 \subseteq A$.*

Proof. Given $A_1 \subseteq A_2 \subseteq A$, then $f(A) \subseteq f(A_2) \subseteq f(A_1)$. Since (A, B) is a concept and $f(A_1) = B$, then $f(A_2) = B$ is valid. □

Proposition 3. *Let A_1, A be two sets and $A_1 \subseteq A$. Then $|\{A_2 \mid A_1 \subseteq A_2 \subseteq A\}| = 2^{|A-A_1|}$.*

Proof. Noting A_1 is the subset of A_2 with $A_2 = A_1 \cup s$ where $s \in 2^{A-A_1}$, it follows that $|\{A_2 \mid A_1 \subseteq A_2 \subseteq A\}| = 2^{|A-A_1|}$. □

Definition 4 (see [14]). Let $K = (G, M, I)$ be a formal context and (A, B) a formal concept of K. If there is a subset R of A which makes $f(R) = B$, and for any $R_1 \subset R$, $f(R_1) \neq B$, then we call R the minimal generator of concept (A, B).

Remark 5. It is worthwhile to note that for a given concept, it may have more than one minimal generator. A minimal generator of a formal concept is a minimal set that makes the intent of a concept stable.

Theorem 6. *Let $K = (G, M, I)$ be a formal context and (A, B) a formal concept of K. If (A, B) has only one minimal generator R, then*

$$\sigma(A, B) = 2^{-|R|}. \quad (3)$$

Proof. Since R is the minimal generator of (A, B), by Propositions 2 and 3, we obtain $|\{C \mid R \subseteq C \subseteq A\}| = 2^{|A-R|}$. Then by Definition 1, we have $\sigma(A, B) = |\{C \mid R \subseteq C \subseteq A\}|/2^{|A|}$, and a simple manipulation leads to the conclusion $\sigma(A, B) = 2^{-|R|}$. □

Theorem 7. *Let $K = (G, M, I)$ be a formal context and (A, B) a formal concept of K. If (A, B) has a family of minimal generators $\{R_i\}_{i\in 1,\ldots,n}$, then*

$$\sigma(A, B) = \left(\sum_{i=1}^{n} 2^{|A-R_i|} - \sum_{1\le i<j\le n} 2^{|A|-|R_i \cup R_j|} + \cdots + (-1)^n 2^{|A-R_1\cup R_2\cup\cdots\cup R_n|}\right)\left(2^{|A|}\right)^{-1}. \quad (4)$$

Proof. Since $\{R_i\}_{i\in 1,\ldots,n}$ is the family of the minimal generators of (A, B), by Propositions 2 and 3, we have $|\{C_i \mid R_i \subseteq C_i \subseteq A\}| = 2^{|A-R_i|}$.

According to inclusion-exclusion principle, we can show that $|\bigcup_{i=1}^{n}\{C_i\}| = \sum_{i=1}^{n} 2^{|A-R_i|} - \sum_{1\le i<j\le n} 2^{|A|-|R_i\cup R_j|} + \cdots + (-1)^n 2^{|A-R_1\cup R_2\cup\cdots\cup R_n|}$. By Definition 1, it follows that $(A, B) = |\bigcup_{i=1}^{n}\{C_i\}|/2^{|A|} = (\sum_{i=1}^{n} 2^{|A-R_i|} - \sum_{1\le i<j\le n} 2^{|A|-|R_i\cup R_j|} + \cdots + (-1)^n 2^{|A-R_1\cup R_2\cup\cdots\cup R_n|})/2^{|A|}$. □

Corollary 8. *Let $K = (G, M, I)$ be a formal context and (A, B) a formal concept of K. If (A, B) has only two minimal generators R_1 and R_2, then*

$$\sigma(A, B) = \frac{2^{|A-R_1|} + 2^{|A-R_2|} - 2^{|A-R_1\cup R_2|}}{2^{|A|}}. \quad (5)$$

Followed by the above discussion, the only thing left to get the stability index of a formal concept is to find its minimal generator, which will be discussed in Section 4.

4. Minimal Generator Computation

Definition 9 (see [15]). Let $K = (G, M, I)$ be a formal context. Object g is called a full attributes object if $f(g) = M$. Dually, attribute m is called the largest common attribute if $g(m) = G$.

Definition 10 (see [16]). Let $K = (G, M, I)$ be a formal context. Object g is called a reducible object if there exists a series of objects $\{g_i\}_{i\in T}$ that makes $\bigcap_{i\in T} f(g_i) = f(g)$ where T is the index set. Dually, attribute m is called a reducible attribute if there exists a series of attributes $\{m_i\}_{i\in T}$ that makes $\bigcap_{i\in T} g(m_i) = g(m)$ where T is the index set.

Definition 11. Let $K = (G, M, I)$ be a formal context. K is called a purified formal context, provided that there exists no full attributes object, no largest common attribute, no reducible object, and no reducible attribute.

Definition 12. Let $K = (G, M, I)$ be a formal context. A concept is called an object concept if it has the form $(g(f(g)), f(g))$, $g \in G$, and g is called its object label. Dually, a concept is called an attribute concept if it has the form $(g(m), f(g(m)))$, $m \in M$, and m is called its attribute label.

Theorem 13. *Let $K = (G, M, I)$ be a purified formal context. Then any object concept of K must be an upper bound irreducible element and vice versa.*

Proof. Assume that an object concept (A, B) is not an upper bound irreducible element; then (A, B) has at least two lower neighbors, and denoting them by $(A_t, B_t)_{t\in T}$, T is the index set. Since (A, B) is an object concept, there exists an object g such that $f(g) = B$. By basic theorem on concept lattice, we have $B = \bigcap_{t\in T} B_t = \bigcap_{t\in T} f(A_t)$. By the fact that $f(A) = B$, it follows that $f(A) = \bigcap_{t\in T} f(A_t)$, which means (A, B) is a irreducible object, and this is contradict to the definition of purified formal context. The reverse implication is proved in much the same way. Hence, this theorem holds. □

Theorem 14. *Let $K = (G, M, I)$ be a purified formal context. If a concept is an object concept, then its minimal generator consists of its object label.*

Proof. The theorem is immediate from Definitions 4 and 12. □

Definition 15. Let $A = \{g_t\}_{t\in T}$ (T is the index set) be a set of objects. Function simplification simp( ) on set A is defined as

$$\operatorname{simp}(A) = A - \left\{g_i \mid \exists g_j, f\left(g_j\right) \subseteq f\left(g_i\right)\right\}. \quad (6)$$

Lemma 16. *Let $A = \{g_t\}_{t\in T}$ (T is the index set) be a set of objects. Then $f(simp(A)) = f(A)$.*

Proof. Without loss of generality, we assume that there exist two objects g_i and g_j with $f(g_i) \subseteq f(g_j)$. Then $f(g_i) \cap f(g_j) = f(g_i)$, so a simple manipulation leads to the equation $f(A) = f(\bigcup_{t\in T} g_t) = \bigcap_{t\in T} f(g_t) = \bigcap_{t\in T-i} f(g_t) = f(\bigcup_{t\in T-i} g_t) = f(\operatorname{simp}(A))$. □

Lemma 17. *Let $K = (G, M, I)$ be a purified formal context and suppose (A, B) is not an object concept. Then minimal generator of (A, B) is contained in the union of the minimal generators of its any two lower neighbors.*

Proof. As (A, B) is not an object concept, there exist at least two lower neighbors, denoting them by $(A_t, B_t)_{t\in 1,\ldots,n}$ with their minimal generators C_i, respectively. By the fact that $A = \bigcup_{t=1}^{n} A_t$, it follows that $f(A) = \bigcap_{t=1}^{n} f(C_t)$. Moreover, for any $i \neq j$, we can see

$$f\left(C_i\right) \cap f\left(C_j\right) \supseteq \bigcap_{t=1}^{n} f\left(A_t\right) = f\left(A\right). \quad (7)$$

Suppose ⊃ of the above expression holds. Then there exists a concept (A', B') with $B' = f(C_i) \cap f(C_j)$, such that $(A, B) > (A', B') > (A_t, B_t)$, for any $t \in 1, \ldots, n$, and this is contradiction to the assumption that $(A_t, B_t)_{t\in 1,\ldots,n}$ are the lower neighbors of (A, B). Hence, the theorem is proved. □

Theorem 18. *Let $K = (G, M, I)$ be a purified formal context. If (A, B) is not an object concept, then its minimal generator is the simplification of the union of its any two lower neighbors' minimal generators.*

Proof. By Lemmas 16 and 17 and in light of Definitions 4 and 15, the proof is trivial. □

Table 1: Formal context "biology and water."

	a	b	c	d	e	f	g	h	i
1	*	*					*		
2	*	*					*	*	
3	*	*	*				*	*	
4	*		*				*	*	*
5	*	*		*		*			
6	*	*	*	*		*			
7	*		*	*	*				
8	*		*	*		*			

Based on Theorems 14 and 18, we get the following recursive formula on the calculation of the minimal generator R of a given concept (A, B).

If (A, B) is an object concept, then R consists of its object label.

Otherwise, R is the simplification of union of its any two lower neighbors' minimal generators.

5. Example and Analysis

Example 1. Consider descriptions of several objects in Table 1, which is the well known formal context "biology and water" in [16].

The corresponding concept lattice of the above formal context "biology and water" is sketched by Figure 1.

For convenience, in the representation of a concept we omit the curly braces and commas. For example, we use (1234, ag) instead of ({1, 2, 3, 4}, {a, g}).

Considering the stability index of concept (1234, ag), there are two steps.

Step 1 (find minimal generator). As (1234, ag) is not an object concept, its minimal generator is determined by its lower neighbors (234, agh) and (123, abg). So we must investigate the minimal generators of (234, agh) and (123, abg), respectively. This downward recursive call continues until object concept is encountered. The integral computation process is illustrated by Figure 2, in which seven concepts are visited with concept (1234, ag) involved.

Step 2 (compute stability index). According to Theorem 6, it follows that

$$\sigma\left(\left(1234, \mathrm{ag}\right)\right) = 2^{-|\{1,4\}|} = 0.25. \quad (8)$$

Given a formal context (G, M, I), time complexity of concept lattice construction is $O(|G|^2|M|L)$ (let L be the number of concepts contained in concept lattice) [17]. When calculating the minimal generator of a given concept, we need to recursively traverse its subconcepts. The number of the visited concepts is apparently less than L, so the time complexity of getting minimal generator of a given concept is $O(|G|^2|M|L^2)$.

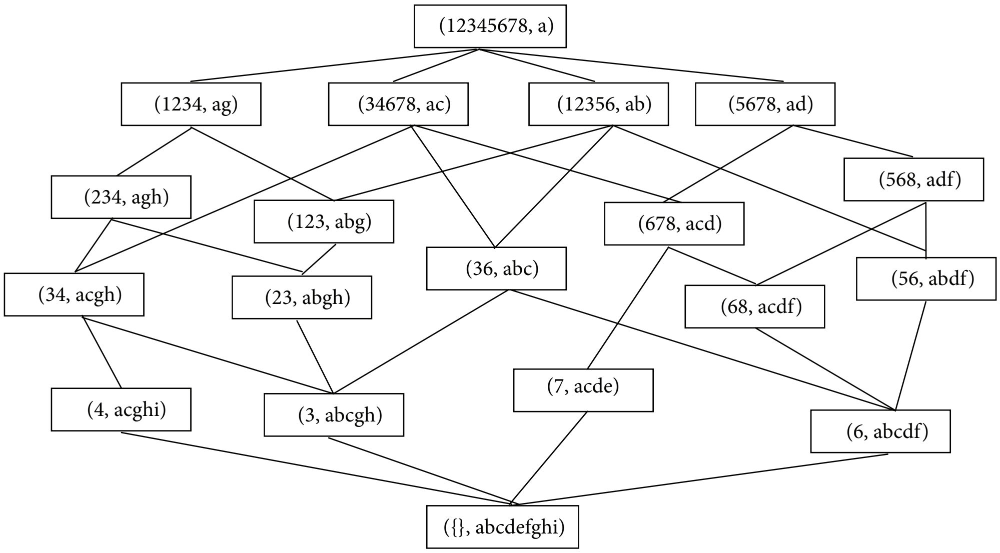

Figure 1: Concept lattice of the context "biology and water."

Table 2: A comparison between our method and [2] with respect to the number of concepts visited.

	(1234, ag)	(34678, ac)	(12356, ab)	(5678, ad)	(234, agh)	(123, abg)
Our method	5	7	6	6	5	1
Method in [2]	7	8	7	7	5	3
	(36, abc)	(678, acd)	(568, adf)	(34, acgh)	(23, abgh)	(68, acdf)
Our method	3	3	3	3	2	1
Method in [2]	3	4	4	3	2	2
	(56, abdf)	(4, acghi)	(3, abcgh)	(7, acde)	(6, abcdf)	
Our Method	1	1	1	1	1	
method in [2]	2	1	1	1	1	

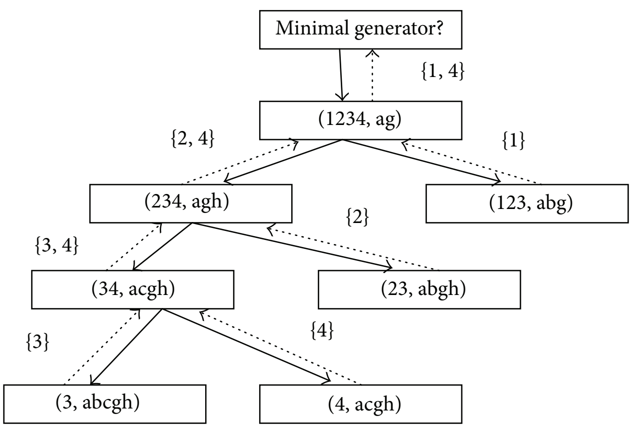

Figure 2: Minimal generator calculation process. (The solid arc represents downward recursive call, and the set besides the dotted upward arc represents the return value.)

Since stability index of a given concept can be derived directly by Theorems 6 and 7, the time complexity of calculating stability of a given concept is still $O(|G|^2|M|L^2)$.

Although upper estimate is the same as the already known algorithm [2], our method is more effective and there is a major reason to account for this.

Let C be a concept; its integral stability index is derived by summing over stability index $J_j(C)$ with level j ranging from 2 to $|A|-1$. Apparently each level must be taken into consideration which means all the subconcepts of C must be visited [2]. But in our method, integral stability index is determined only by its minimal generator. When calculating the minimal generator of C, instead of visiting all the subconcepts of it, the downward recursive traverse will terminate if object concept is encountered, and all the subconcepts of object concept will not be visited any more.

By using Example 1, a comparison between our method and [2] is conducted with respect to the number of concepts visited, and the result is shown in Table 2.

If we randomly select a concept from Table 2, we can see that the average number of concepts visited by our method is 50/17, whereas the number is 61/17 while using the method in [2].

Finally, our method shows its advantage when computing stability indexes of all concepts. The procedure is depicted by Algorithm 1.

Apparently, in Algorithm 1 every concept is visited only once, so time complexity of this algorithm is still $O(|G|^2|M|L^2)$. But if using the method in [2], we have to compute stability index of each concept one by one, and

Compute stability indexes of all concepts of a given formal context
Input: formal context $K = (G, M, I)$
Output: stability indexes of all concepts of $L(K)$
Begin
(1) construct concept lattice $L(K)$;
(2) deem the concept lattice $L(K)$ as an undirected graph and traverse the concept lattice upwards from the minimal concept of $L(K)$ by using breadth first search;
(3) if the current visited concept lattice is an object concept, then the minimal generator is its object label; else, its minimal generator is determined by its any two lower neighbors' minimal generator;
(4) compute stability indexes based on minimal generator by using the formulas in Theorems 6 and 7.
End.

Algorithm 1

thus the overall time complexity is $L * O(|G|^2|M|L^2)$, that is, $O(|G|^2|M|L^3)$, which is greater than that of Algorithm 1.

6. Conclusions

The purpose of this paper is to find a more effective way to calculate stability of a given concept. The critical step of our method is to find minimal generator, and a recursive method is given with the irreducible elements as the ending condition. In light of minimal generator, stability can be computed by using inclusion-exclusion principle.

A valuable further work may be how to extend the results to the setting of heterogeneous information system or to the setting of fuzzy formal context.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The work presented in this paper is supported by Doctorial Foundation of Henan Polytechnic University (B2011-102). The author also gratefully acknowledges the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] S. O. Kuznetsov, "Stability as an estimate of the degree of substantiation of hypotheses derived on the basis of operational similarity," *Nauchno-Tekhnicheskaya Informatsiya Ser.2 (Automat. Document. Math. Linguist)*, no. 12, pp. 21–29, 1990.

[2] S. O. Kuznetsov, "On stability of a formal concept," *Annals of Mathematics and Artificial Intelligence*, vol. 49, no. 1–4, pp. 101–115, 2007.

[3] S. O. Kuznetsov, S. Obiedkov, and C. Roth, "Reducing the representation complexity of lattice-based taxonomies," in *Proceedings of the 15th International Conference on Conceptual Structures (ICCS '07)*, U. Priss, S. Polovina, and R. Hill, Eds., vol. 4604 of *Lecture Notes in Artificial Intelligence*, pp. 241–254, Springer, 2007.

[4] C. Roth, S. Obiedkov, and D. G. Kourie, "On succinct representation of knowledge community taxonomies with formal concept analysis," *International Journal of Foundations of Computer Science*, vol. 19, no. 2, pp. 383–404, 2008.

[5] C. Roth, S. Obiedkov, and D. Kourie, "Towards concise representation for taxonomies of epistemic communities," in *Proceedings of the 4th International Conference on Concept Lattices and Their Applications*, pp. 240–255, Tunis, Tunisia, 2006.

[6] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*, CBMS-NSF Regional Conference Series in Applied Mathematics, Philadelphia, Pa, USA, 1982.

[7] R. Carnap, *The Logical Foundations of Probability*, University of Chicago Press, Chicago, Ill, USA, 1962.

[8] M. A. Babin and S. O. Kuznetsov, "Approximating concept stability," in *Formal Concept Analysis: Proceedings of the 10th International Conference, ICFCA 2012, Leuven, Belgium, May 7–10, 2012*, F. Domenach, D. Ignatov, and J. Poelmans, Eds., vol. 7278 of *Lecture Notes in Artificial Intelligence*, pp. 7–15, 2012.

[9] V. K. Finn, "On machine-oriented formalization of plausible reasoning in the style of F. Bacon-J. S. Mill," *Semiotika i Informatika*, no. 20, pp. 35–101, 1983.

[10] S. O. Kuznetsov, "Mathematical aspects of concept analysis," *Journal of Mathematical Sciences*, vol. 80, no. 2, pp. 1654–1698, 1996.

[11] B. Ganter and S. O. Kuznetsov, "Formalizing hypotheses with concepts," in *Proceedings of the 8th International Conference on Conceptual Structures (ICCS '00)*, vol. 1867 of *Lecture Notes in Computer Science*, pp. 342–356, 2000.

[12] R. Wille, "Concept lattices and conceptual knowledge systems," *Computers and Mathematics with Applications*, vol. 23, no. 6–9, pp. 493–515, 1992.

[13] R. Wille, "Restructuring lattice theory: an approach based on hierarchies of concepts," in *Ordered Sets*, vol. 83 of *NATO Advanced Study Institutes Series*, pp. 445–470, Springer, Berlin, Germany, 1982.

[14] B. Ganter and S. O. Kuznetsov, "Hypotheses and version Spaces," in *Proceedings of the 10th International Conference on Conceptual Structures (ICCS '03)*, A. de Moor, W. Lex, and B. Ganter, Eds., vol. 2746 of *Lecture Notes in Artificial Intelligence*, pp. 83–95, Dresden, Germany, 2003.

[15] W. Hong, J. Mao, J. Yu, and J. Song, "The complete definitions of attributes and abstract description of attribute features of the

formal context," *ICIC Express Letters*, vol. 7, no. 3, pp. 997–1003, 2013.

[16] B. Ganter and R. Wille, *Formal Concept Analysis: Mathematical Foundation*, Springer, Berlin, Germany, 1999.

[17] S. O. Kuznetsov and S. A. Obiedkov, "Comparing performance of algorithms for generating concept lattices," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 14, no. 2-3, pp. 189–216, 2002.