Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 1789121, 11 pages
https://doi.org/10.1155/2018/1789121

Research Article

The Hierarchies of Multivalued Attribute Domains and Corresponding Applications in Data Mining

Yuxia Lei ,[1] Yushu Yan,[1] Yonghua Han,[1] and Feng Jiang[2]

[1] *School of Information Science and Engineering, Qufu Normal University, Rizhao 276826, China*
[2] *College of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China*

Correspondence should be addressed to Yuxia Lei; yx_lei@126.com

Received 30 March 2018; Accepted 4 July 2018; Published 28 August 2018

Academic Editor: Tianyong Hao

In mobile computing, machine learning models for natural language processing (NLP) have become one of the most attractive focus areas in research. Association rules among attributes are common knowledge patterns, which can often provide potential and useful information such as mobile users' interests. Actually, almost each attribute is associated with a hierarchy of the domain. Given an relation R = (U, A) and any cut α_a on the hierarchy for every attribute a, there is another rough relation R_Φ, where $\Phi = (\alpha_a : a \in A)$. This paper will establish the connection between the functional dependencies in R and R_Φ, propose the method for extracting reducts in R_Φ, and demonstrate the implementation of proposed method on an application in data mining of association rules. The method for acquiring association rules consists of the following three steps: (1) translating natural texts into relations, by NLP; (2) translating relations into rough ones, by attributes analysis or fuzzy k-means (FKM) clustering; and (3) extracting association rules from concept lattices, by formal concept analysis (FCA). Our experimental results show that the proposed methods, which can be applied directly to regular mobile data such as healthcare data, improved quality, and relevance of rules.

1. Introduction

With the rapid growth in use of mobile devices, more and more mobile generated data is in great need of processing. A large amount of valuable content exists in natural text such as web pages, news feeds, and Twitter/WeChat messages. Natural language processing (NLP) techniques have proven to be useful in dealing with the information overload problem in the mobile environment, for example, news summarization, question answering, and information extraction and retrieval. In these areas, machine learning models for NLP are one of the important research contents [1], in which association rules are common knowledge patterns. Association rules can often provide potential and useful information for mobile clients, contributing to automatically providing personalized recommended services.

In the process of knowledge acquisition from natural texts, we frequently encounter multivalued attributes such as spatial locations and security policies in mobile environments. Actually, for each multivalued attribute, there are different levels of partitions or fuzzy partitions in its domain, which are forbidden in the domain. If we take the different partitions as new values and then obtain attribute values of different granularity. It will be meaningful to discover the attribute dependence of different granularity.

Formal concept analysis (FCA) is an effective tool for knowledge representation, acquisition, and knowledge discovery. 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 [2]. Recently, concept lattices have already been successfully applied to a wide range of scientific disciplines including knowledge representation [3–5], knowledge discovery [6–8], knowledge reduction [9–11], hybrid relation analysis [12], wireless sensor network [13], and information retrieval [14].

In order to process natural texts, we firstly extract some formal objects and attributes using NLP, secondly translate the texts into relations, and thirdly process the relations using FCA. However, with the growth of the size of the relation, the number of concepts and association rules grows in an exponential manner. Hence, it is necessary to reduce contexts before applying FCA. Several researchers have used matrix approximation techniques such as singular value decomposition (SVD) [15] and nonnegative matrix factorization (NMF) [11] for reducing the size of the context. However, matrix DR methods are based on the expensive eigenvalue computations and hence are known for their high computational complexity. Cluster analysis can well be used as method for data reduction under the notion of concept decomposition (CD). With its lesser computational complexity, FKM [15] is proved to be an alternative method for reducing the dimensionality of context, thereby controlling the size of concept lattices. Actually, there are different levels of partitions or fuzzy partitions [16, 17] in each multivalued attribute domain. Therefore, each attribute is associated with a hierarchy of the domain of the attribute. Based on attribute analysis and FKM, this paper proposes a method of acquiring association rules, which can validly reduce the number of association rules. The method firstly translates attribute values into different abstract hierarchies, which are new attribute values, and then generates association rules with FCA. Experimental results on heart disease data show that the proposed method can improve quality and relevance of association rules, which can be applied directly to regular mobile data such as healthcare data and spatial locations, contributing to providing precise personalized recommended services.

The paper is organized as follows: the next section gives the basic notations of FCA and FKM; the third section gives the hierarchies of domains, discusses the possible connection between the functional dependencies of R and R_Φ, and gives a sufficient and necessary condition for a functional dependence $B \longrightarrow C$ holding in both R_Φ and R_Ψ if Ψ is below Φ; the fourth section provides some experiments, in order to verify the feasibility of the method. Finally, Section 5 concludes the paper.

2. Some Basic Notations

2.1. Translation from Natural Texts into Relations. Before using FCA to extract association rules, we need to translate natural texts into a knowledge frame and then merge the related frames into a relation. Our method can be described as follows: firstly, we create some attribute thesaurus. For example, we create an attribute thesaurus that describes people such as age, sex, height, weight, date of birth, hobbies, occupation, etc. Secondly, using a semiautomatic approach, we translate text knowledge into a frame and then merge some related frames into a relation.

2.2. Formal Concept Analysis. This section will introduce some basic notations in FCA [2]. A (formal) context is defined as a triple $K = (G, M, I)$, where G and M are sets and $I \subseteq G \times M$ is a binary relation. For any $X \subseteq G$ and $Y \subseteq M$, the pair (X, Y) is called a (formal) concept if (1) Y is the set of attributes common to the objects in X and (2) X is the set of objects having all attributes in Y. X and Y are called the (concept) extent and the (concept) intent of the concept, respectively. There are two kinds of special concepts: object concepts and property concepts. Given an object $g \in G$, the object concept of g is the smallest concept having g in its extent. Correspondingly, given an attribute $m \in M$, the attribute concept of m is the greatest concept having m in its intent.

The line diagram corresponding to a concept lattice can vividly unfold generalization-specialization relationship among concepts. The labeling can be simplified considerably by putting down each object and each attribute only once. Thus, the concept lattices can be described by the line diagrams with reduced labeling. In a line diagram, the name of an object g is always attached to the circle representing the smallest concept with g in its extent; dually the name of an attribute m is always attached to the circle representing the largest concept with m in its intent. This allows us to read the map I from the diagram: an object g has an attribute m if and only if there is an ascending path from the circle labeled by g to the circle labeled by m. The extent of a concept consists of all tuples whose labels are below in the diagram and the intent consists of all properties attached to concepts above in the hierarchy. Thus, we can easily extract association rules with 100% confidence from the line diagrams, and the *stem base* of the attribute implications is nonredundant and complete [2].

As many practical applications involve nonbinary data, multivalued contexts have been introduced in FCA. A multivalued context $K = (G, M, W, I)$ 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 object g that has value w for attribute m. Actually, a multivalued context can be regarded as a relation with the column containing the objects being a primary key. In the RDM, a relation is described by a relation schema $S = R(m_1, m_2, \ldots, m_n)$, where m_i $(1 \le i \le n)$ represent attributes. Each attribute m is associated with a domain D_m, which is the set of possible values for the attribute m. A relation (R, A) is denoted by a set of tuples $U = \{r\colon r = (v_1, v_2, \ldots, v_n) \in R\}$, where $A = (m_1, m_2, \ldots, m_n)$ and g is a tuple such that for every $1 \le i \le n$, $v_i \in D_{mi}$. An equivalent way to view such a tuple $g \in U$ is as a map from A to $\prod_{a\in A} D_m$ such that $g(m) \in D_m$ [11]. Thus we can further represent a relation R by a triple (U, A, I), where I is a map from $U \times A$ to $\bigcup_{a\in A} D_m$ such that, for any $(g, m) \in U \times A$, $I(g, m) = g(m) \in D_m$. 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.

2.3. Fuzzy K-Means Clustering. Fuzzy k-means (FKM) [18] partitions a set of t-dimensional vectors $X = \{X_1, X_2, \ldots, X_n)$ into k-clusters, where $X_j = \{X_{1j}, X_{2j}, \ldots, X_{tj})$ represents the jth sample. For X_j and the ith cluster center v_i, there is a membership degree u_{ji} indicating to what degree sample X_j belongs to v_i, $i = 1, 2, \ldots, k$. Thus, there is a fuzzy

Table 1

Category	Systolic(mmHg)	Diastolic(mmHg)
Normal value	120	80
Normal high value	120 ~139	80 ~89
High blood pressure	140	90
(Hypertension) stage 1	140 ~159	90 ~99
(Hypertension) stage 2	160 ~179	100 ~109
(Hypertension) stage 3	180	110
Simple systolic hypertension	140	89

partition matrix $U = (u_{ij})_{d\times k}$. The FKM algorithm is based on minimizing the objective function J_{fuzz} defined as

$$J_{fuzz} = \sum_{j=1}^{d} \sum_{i=1}^{k} {u_j}^m {d_i}^2 \qquad (1)$$

where d_j is the Euclidean distance between X_j to the cluster center v_i. The exponent m in (1) is called fuzzifier parameter and it defines the fuzziness of the clustering. The formulae of u_{ji} and v_i are

$$u_{ji} = \frac{1}{\sum_{p=1}^{k} \left(d_j/d_p\right)^{1/(m-1)}}, \qquad v_i = \frac{\sum_{j=1}^{d} {u_{ji}}^m X_j}{\sum_{j=1}^{d} {u_{ji}}^m} \qquad (2)$$

where $m \neq 1$ and $i = 1, \ldots, k$.

Based on the above discussion, the FKM algorithm can be summarized as follows.

Step 1. Choose the number of clusters k, degree of fuzziness m, and a threshold value e. Initialize the fuzzy partition matrix U.

Step 2. Compute the cluster centers v_i $(i = 1, 2, \ldots, k)$, according to (2).

Step 3. Compute the Euclidean distance d_{ji} from the sample X_j to the cluster center v_i according to the Euclidean distance. Then calculate all uji using (2) and update fuzzy partition matrix U.

Step 4. Compute the objective function J_{fuzz} using (1). Verify whether the function converges or the difference between the two adjacent values of objective function is less than the given threshold value e, then stop. Otherwise repeat from the Step 2.

3. Attribute Analysis and the Reducts in Rough Relations

3.1. Attribute Analysis. For attributes, there are often different criteria for division, for example, Chinese blood pressure categories and Chinese age categories (≥18 years old), which can be found in “Guidelines for the Prevention and Control of Hypertension in China” (2005 Revision).

Table 2

Category	Sub-category	Age range
Youth	Puberty	18-28
	Mature period	29-40
Middle aged	Strong period	41-48
	Robust period	49-55
	Adjustment period	56-65
Old age	Initial old period	66-72
	Middle old period	73-84
	Old period	85

Chinese blood pressure categories are described as shown in Table 1.

Chinese age categories are described as shown in Table 2.

Generally, attributes can be divided into the three types. Type 1: there is a category; type 2: there is no category, but there are reference criteria for the classification of attributes; type 3: attribute values are never category or classification criteria for reference. In Algorithm 1, the subscript of R in step 3 is not obvious. Therefore, we followed the methods of Lei et al. 2016 [19] and propose a method for extracting association rules, which has the following steps: (1) analyzing attribute types and the structures of domains, (2) generating different hierarchies of attribute values with FKM clustering, (3) translating original relations into rough ones, (4) generating the concept lattice, and (5) extracting the association rules from concept lattices. The method can be described as in Algorithm 1.

3.2. The Reducts in Rough Relations. In a rough relation in the rough relation databases, each attribute $a \in A$ is associated with a equivalence relation θ_a on domain D_a. We denote the corresponding partition of θ_a on D_a by P_a: ${X^a}_1, {X^a}_2, \ldots, {X^a}_k$ for some natural number k.

Definition 1 (a rough relation). A rough relation R is a subset of $\prod_{a\in A} P(D_a)$ such that, for every $x \in R$, every $a \in A$, and every $1 \le i \le k$, $|x(a) \cap {X^a}_i| \neq 1$, where $P(D_a)$ is the power set of D_a.

Hence, in a rough relation R, a tuple x takes multivalues at attributes, satisfying certain conditions, where the conditions are given in terms of equivalence relations on domains of

Algorithm for extracting associate rules from a relation
Input: a relation R = (U, A)
Output: association rules satisfying given the minimum support and the minimum confidence
Process
Step 1: For any attribute a ∈ A, generating a hierarchy H_a of its domain by traditional standards or FKM clustering method.
Step 2: Given a cut $\alpha_a = V_{1a} \cup V_{2a} \cup \ldots \cup V_{ka}$ in H_a, then there is a set $\Phi = (\alpha_a\colon a \in A)$ of cuts,
translating the relation R into a rough relation R., where $r(a) = V_{ia}$ in R. if $r(a) \in V_{ia}$ in R.
Step 3: Translating R. into a binary relation R′,
Step 4: Generating the concept lattice from R′,
Step 5: Extracting association rules satisfying given the minimum support and the minimum confidence.

Algorithm 1: Extracting association rules from a relation.

attributes. A rough relation R is reduced to be a normal relation if every attribute has $P(D_a)$ as the domain, instead of D_a.

Given an attribute a, there is a hierarchy $T_a = (S_a, \subseteq)$, where S_a is a set of subsets of D_a and ⊆ is a binary relation on S_a, such that (1) $D_a \in S_a$; (2) for any $v \in D_a$, $\{v\} \in S_a$; (3) $(S_a, \subseteq)$ is a tree.

Definition 2. A cut α is a subset of S_a such that for any path σ from the root to a leaf, $|\alpha \cap \sigma| = 1$; given two cuts α and β, we say that α is above β, denoted by $\alpha \gg \beta$, for any path σ from the root to a leaf, $S_{\beta,\sigma} \subseteq S_{\alpha,\sigma}$, where $S_{\alpha,\sigma}$ and $S_{\beta,\sigma}$ are the unique ones in $\alpha \cap \sigma$ and $\beta \cap \sigma$, respectively.

Given a cut α on T_a, there is an equivalence relation $\theta_{\alpha,a}$ on D_a such that, for any $u, v \in D_a$, $u\theta_{\alpha,a}v$ iff there is a unique $s \in \alpha$ such that $u, v \in s$. We use $[u]_{a,\alpha}$ to denote the equivalence class of $\theta_{\alpha,a}$ containing u.

Let Φ be a cut vector $(\alpha_a : a \in A)$, where α_a is a cut on T_a. Given two cut vectors $\Phi = (\alpha_a : a \in A)$ and $\Psi = (\beta_a : a \in A)$, we say that Φ is above Ψ, denoted by $\Phi \gg \Psi$, if for every $a \in A$, $\alpha_a \gg \beta_a$. Given a relation R and a cut vector $\Phi = (\alpha_a : a \in A)$, there is a relation R_Φ such that for any tuple $x \in R$ and attribute $a \in A$, if $x(a) = u$ in R then $x(a) = [u]_{a,\alpha}$ in R_Φ. Given a relation R and a cut vector Φ, define a relation $\theta_{R,\Phi}$ such that, for any $x, y \in R$, $x\theta_{R,\Phi}y$ iff $x(a) = [u]_{a,\alpha}$ and $y(a) = [v]_{a,\alpha}$ in R_Φ and $[u]_{a,\alpha} = [v]_{a,\alpha}$, where $x(a) = u$ and $y(a) = v$ in R.

Proposition 3. *Given a relation R and a cut vector Φ, $\theta_{R,\Phi}$ is an equivalence relation on U.*

Proposition 4. *Given a relation R and two cut vectors $\Phi = (\alpha_a : a \in A)$ and $\Psi = (\beta_a : a \in A)$, if $\Phi \gg \Psi$ then $\theta_{R,\Psi}$ is a refinement of $\theta_{R,\Phi}$.*

Proof. For any $x, y \in R$, assume that $x\theta_{R,\Psi}y$. Then, for any $a \in A$, $[x(a)]_{a,\beta_a} = [y(x)]_{a,\beta_a}$. Because β_a is a refinement of α_a, $[x(a)]_{a,\alpha_a} = [y(a)]_{a,\alpha_a}$. That is, for any $a \in A$, $[x(a)]_{a,\alpha_a} = [y(a)]_{a,\alpha_a}$, i.e., $x\theta_{R,\Phi}y$. □

Definition 5. Given a relation R and subsets $B, C \subseteq A$, if, for any $x, y \in R$, if $x(a) = y(a)$ for every $a \in B$ then $x(c) = y(c)$ for every $c \in C$, we say that C depends on B in R, denoted by $R| = B \longrightarrow C$.

Assume that $R| = B \longrightarrow C$. By the sense of functional dependencies, we define a function $f : \prod_{b\in B} D_b \longrightarrow \prod_{c\in C} D_c$ such that, for any $v \in \prod_{b\in B} D_b$, if there is a $x \in R$ such that $x(b) = v(b)$ for every $b \in B$ then $f(v) = u \in \prod_{c\in C} D_c$, where u is defined as follows: for any $c \in C$, $u(c) = x(c)$.

By the assumption that $R| = B \longrightarrow C$, f is a function. We say that f witnesses that $R| = B \longrightarrow C$. There are two special cut vectors $\Phi_\perp$ and Φ_T defined as follows: for any $a \in A$, $\Phi_{\perp,a} = \{u\colon u \in D_a\}$, $\Phi_{T,a} = \{D_a\}$. It is clear that for any cut vector Φ, $\Phi_T \gg \Phi \gg \Phi_\perp$.

Proposition 6. *(i)* $R_{\Phi_T,a} = \{D_{a1}, D_{a2}, \ldots, D_{an}\}$. *Therefore, for any* $a \in A$, $\{a\}$ *is a reduct of* R_{Φ_T}; *(ii)* $R_{\Phi_T,a} = R$. *Therefore,* $B \subseteq C$ *is a reduct of* $R_{\Phi\perp}$ *iff B is a reduct of R.*

Given a relation R and two cut vectors Φ and Ψ, if $\Phi \gg \Psi$ then, for any subsets $B, C \subseteq A$, $R_\Psi| = B \longrightarrow C$ and $R_\Phi| = B \longrightarrow C$ are not related; i.e., it is possible that $R_\Psi| = B \longrightarrow C$ and $R_\Phi| \neq B \longrightarrow C$; or $R_\Psi| \neq B \longrightarrow C$ and $R_\Phi| = B \longrightarrow C$. By Propositions 3 and 4, we have that there are R, Φ, and Ψ such that $\Phi \gg \Psi$, $R_\Psi| \neq B \longrightarrow C$, and $R_\Phi| = B \longrightarrow C$. Let $\Phi = \Phi_T$, such that $\Phi \gg \Psi$ and $R_\Psi| \neq B \longrightarrow C$. Because, for any $B, C \subseteq A$, $R_{\Phi_T}| = B \longrightarrow C$. We give the following example to show that there are R, Φ, and Ψ such that $\Phi \gg \Psi$, $R_\Psi| = B \longrightarrow C$, and $R_\Phi| \neq B \longrightarrow C$.

Example 7. Let $A = \{a_1, a_2\}$, $D_{a1} = \{1, 2, 3, 4, 5, 6\}$, and $D_{a2} = \{1, 2\}$. Let $R = \{x_1, \ldots, x_6\}$, which is described in Table hierarchies(a), Let $\Psi = (\{\{1, 2\}, \{3, 4\}, \{5, 6\}\}, \{\{1\}, \{2\}\})$, and $\Phi = (\{\{1, 2, 3, 4\}, \{5, 6\}\}, \{\{1\}, \{2\}\})$. Then, R_Ψ and R_Φ are represented in Tables 1(b) and 3(c).

We have that $R_\Psi| = a_1 \longrightarrow a_2$ and $R_\Phi| \neq a_1 \longrightarrow a_2$.

Definition 8. Given a relation R and two cut vectors Φ and Ψ, assume that $\Phi \gg \Psi$. Let $b, c \in A$ such that $R_\Psi| = b \longrightarrow c$,

Table 3

(a)

tuples	a_1	a_2
x_1	1	1
x_2	2	1
x_3	3	2
x_4	4	2
x_5	5	1
x_6	6	1

(b)

tuples	a_1	a_2
x_1	[1]	1
x_2	[1]	1
x_3	[3]	2
x_4	[3]	2
x_5	[5]	1
x_6	[5]	1

(c)

tuples	a_1	a_2
x_1	[1]	1
x_2	[1]	1
x_3	[1]	2
x_4	[1]	2
x_5	[5]	1
x_6	[5]	1

and f: $D_{b,\alpha_b} \longrightarrow D_{c,\alpha_c}$ witnesses that $R_\Psi| = \mathrm{b} \longrightarrow \mathrm{c}$, where $D_{b,\alpha_b} = \{[u]_{\alpha_b} : u \in D_b\}$.

We say that $R_\Psi| = \mathrm{b} \longrightarrow \mathrm{c}$ is compatible with Φ in R if for any $\mathrm{x} \in \mathrm{R}$, there is a $\mathrm{v} \in \mathrm{D_c}$ such that $\{[f([u]_\Psi)]_\Phi : u \in [x(b)]_\Phi\} = [v]_\Phi$. Let $\mathrm{B}, \mathrm{C} \subseteq \mathrm{A}$ such that $R_\Psi| = \mathrm{b} \longrightarrow \mathrm{c}$, and $f : \prod_{b\in B} D_{b,\alpha_b} \longrightarrow \prod_{c\in C} D_{c,\alpha_c}$ witness that $R_\Psi| = \mathrm{b} \longrightarrow \mathrm{c}$. We say that $R_\Psi| = \mathrm{b} \longrightarrow \mathrm{c}$ is compatible with Φ in R if, for any $\mathrm{x} \in \mathrm{R}$, there is a vector ($\mathrm{v_c} \in \mathrm{D_c}$: $\mathrm{c} \in \mathrm{C}$) such that

$$\left\{\left[f\left(\prod_{b\in B}[u_b]_\Psi\right)\right]_\Phi : u \in [x(b)]_\Phi, b \in B\right) = \prod_{c\in C}[v_c]_{\alpha_c} \qquad (*)$$

Proposition 9. *Given a relation R and two cut vectors Φ and Ψ, if $\Phi \gg \Psi$ then for any subsets* B, C $\subseteq$ A, $R_\Psi| = \mathrm{B} \longrightarrow \mathrm{C}$ *implies* $R_\Phi| = \mathrm{B} \longrightarrow \mathrm{C}$ *iff* $R_\Psi| = \mathrm{B} \longrightarrow \mathrm{C}$ *is compatible with* Φ *in R.*

Proof. ($\Longleftarrow$) Assume that $R_\Psi| = \mathrm{B} \longrightarrow \mathrm{C}$ is compatible with Φ in R, and $R_\Phi| = \mathrm{B} \longrightarrow \mathrm{C}$. Then, for any x, y $\in$ R, if, for every b $\in$ B, $[x(b)]_\Psi = [y(b)]_\Psi$, then, for every c $\in$ C, $[x(c)]_\Psi = [y(c)]_\Psi$. Assume that for every b $\in$ B, $[x(b)]_\Phi = [y(b)]_\Phi$, There are two cases.

Case 1. If for every b $\in$ B, $[x(b)]_\Psi = [y(b)]_\Psi$, then by the assumption that $R_\Psi| = \mathrm{B} \longrightarrow \mathrm{C}$, for every c $\in$ C, $[x(c)]_\Psi = [y(c)]_\Psi$, so $[x(c)]_\Phi = [y(c)]_\Phi$.

Case 2. If there is a b $\in$ B such that $[x(b)]_\Psi \neq [y(b)]_\Psi$ then by ($*$), for every c $\in$ C, $[x(c)]_\Phi = [y(c)]_\Phi = [v_c]_\Phi = \mathrm{ss}[v_c]_{\alpha_c}$.

($\Longrightarrow$) Assume that $R_\Psi| = \mathrm{B} \longrightarrow \mathrm{C}$ is not compatible with Φ in R. Then, there is an x $\in$ R such that for any ($\mathrm{v_c} \in \mathrm{D_c}$: c $\in$ C), ($*$) does not hold. Then, as in Example 7, we can construct R, Φ, and Ψ such that $\Phi \gg \Psi$, $R_\Psi| = \mathrm{B} \longrightarrow \mathrm{C}$ implies $R_\Phi| \neq \mathrm{B} \longrightarrow \mathrm{C}$. □

4. Experimental Analysis

4.1. Data Availability. We use the experimental data for heart disease dataset in UCI database, which can be downloaded from the website http://archive.ics.uci.edu/ml/datasets/Heart+Disease. Concept lattice tools ConExp1.3 and lattice miner can be downloaded from the websites: https://sourceforge.net/projects/conexp/ and https://sourceforge.net/projects/lattice-miner/.

4.2. Data Preparation. The heart disease dataset in UCI database contains 303 objects and 14 available attributes. The main purpose of this paper is to analyze the connections between association rules in an original relation and ones in rough relations. To both verify the validity of the method and reduce the computational complexity, we select 24 objects and 5 attributes, as shown in Table 4. The 5 attributes are age: age in years; trestbps: resting blood pressure (in mm Hg on admission to the hospital); restecg: resting electrocardiographic results: Value 0: normal; Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV); Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria; thalach: maximum heart rate achieved; cp: chest pain type: value 1: typical angina; value 2: atypical angina; value 3: nonanginal pain; value 4: asymptomatic.

Table 4 can be converted to a binary relation K = (D, T, I), where D is the set of 24 objects, T= {age, trestbps, restecg, thalach, chest pain type}. The elements of T are abbreviated to *a, d, g, h* and *n*, respectively. "*a*" and "*d*" belong to type 2 in Section 2.3. "*g*" and "*n*" belong to type 1 in Section 2.3. "*h*" belongs to type 3 in Section 2.3, which is necessary to use the FKM clustering.

4.3. Experimental Procedure and Rule Acquisition. We made two groups experiments. The first experiment firstly analyzed hierarchies of attribute domains and then generalized to different levels of attribute values, in order to control the size of the concept lattice. The second experiment used fuzzy attribute values to control the size of the concept lattice.

The First Experiment. Using the concept lattice tool ConExp1.3, which can be downloaded from the website https://sourceforge.net/projects/conexp/, we extract association rules from context. The general form is "<N>P=[C] =><N′ >C′", where N is the number of objects satisfying the premise, P is a precondition, C is the confidence of

Table 4: Heart disease set (partly).

No. of patients	age	trestbps	restecg	thalach	chest pain type
1	63	145	2	150	0
2	67	160	2	108	2
3	67	120	2	129	1
4	37	130	0	187	0
5	58	132	2	173	3
6	60	130	2	132	4
7	40	110	2	114	3
8	71	160	0	162	0
9	67	125	0	163	3
10	66	120	2	151	0
11	34	118	2	174	0
12	63	150	2	154	4
13	55	160	2	145	4
14	64	120	2	96	3
15	51	140	0	173	1
16	58	100	2	122	0
17	70	160	0	112	3
18	53	142	2	111	0
19	57	152	0	88	1
20	56	132	2	105	1
21	55	180	1	117	2
22	76	140	1	116	0
23	55	128	1	130	3
24	58	114	1	140	4

Table 5: Association rules from concept lattice in Figure 1.

No.	Rules
1	<3> age2 =[100%]=><3> xueya1
2	< 4 > xueya3 =[75%]=><3> age6
3	<9> age5 =[67%]=><6> xueya1
4	<6> xueya2 =[50%]=><3> age5
5	<6> age6 =[50%]=><3> xueya3
6	<6> age6 =[50%]=><3> xueya1
7	<13> xueya1 =[46%]=><6> age5

Table 6: Association rules from concept lattice in Figure 2.

No.	Rules
1	<3> age2 =[100%]=><3> xueya1
2	<5> age4 =[80%]=><4> xueya2
3	<9> age5 =[67%]=><6> xueya1
4	<6> age6 =[50%]=><3> xueya2
5	<6> age6 =[50%]=><3> xueya1
6	<13> xueya1 =[46%]=><6> age5

association rules, N' is the number of objects meeting the premise, and, in conclusion, and C' is the conclusion. For example, the second association rule in Table 5, i.e., <4> xueya3 =[75%]=><3> age6, meaning that there are 4 objects satisfying the premise for the second level of blood pressure and three objects among them also meet the elderly early old age, and its confidence is 75%.

(1) The age values and blood pressure values are divided into 8 and 4 categories, respectively. Thus, we obtain a relation, called A8BP4. The concept lattice of A8BP4 is shown in Figure 1, from which we obtain some association rules, as shown in Table 5.

(2) The age values and blood pressure values are divided into 8 and 2 categories, respectively. Thus, we obtain a relation, called A8BP2. The concept lattice of A8BP2 is shown in Figure 2. From Figure 2, we obtain some association rules, as shown in Table 6.

(3) The age values and blood pressure values are divided into 3 and 4 categories, respectively. Thus, we obtain a relation, called A3BP4. The concept lattice of A3BP4 is shown in Figure 3. From Figure 3, we obtain some association rules, as shown in Table 7.

(4) The age values and blood pressure values are divided into 3 and 2 categories, respectively. Thus, we obtain a relation, called A3BP2. The concept lattice of A3BP2 is shown in Figure 4. From Figure 4, we obtain some association rules, as shown in Table 8.

For analyzing the concept lattices, we followed the methods of Lei et al. 2016 [16]. Table 9 describes the number of concepts, edges, height, width, and rules in Figures 1–4.

From the description above, we have the following results: (1) the higher the value levels are, the smaller the complexity

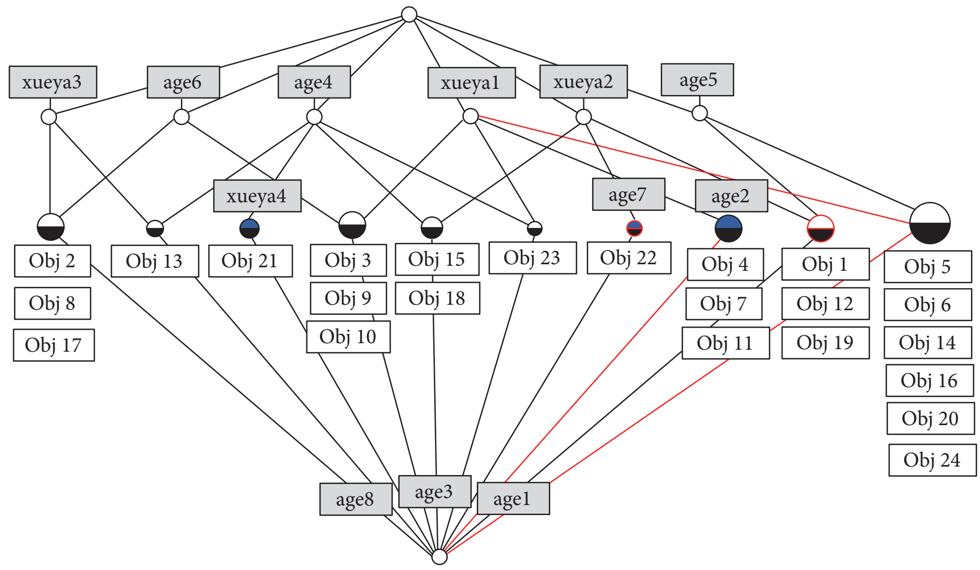

Figure 1: The concept lattice of A8BP4.

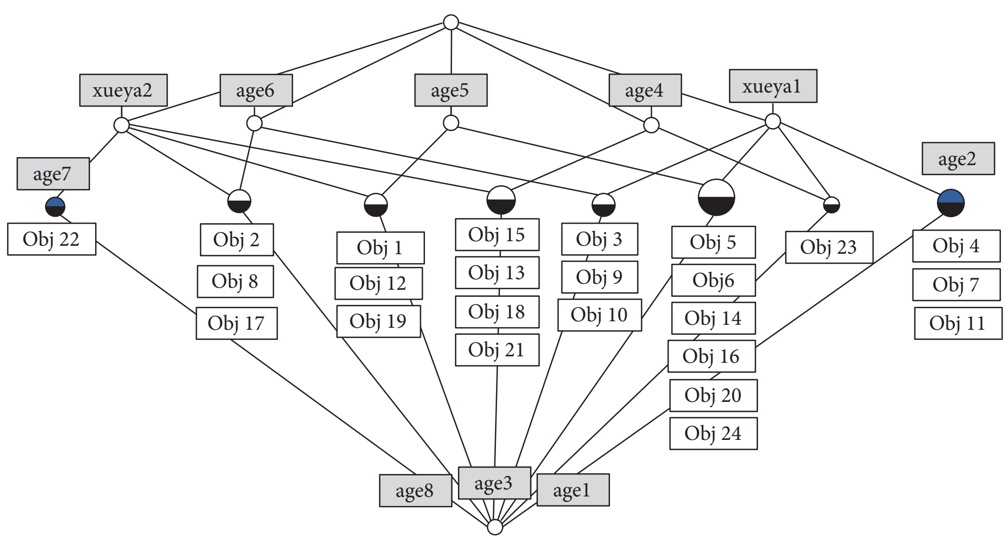

Figure 2: The concept lattice of A8BP2.

Table 7: Association rules from concept lattice in Figure 3.

No.	Rules
1	<3> age1 =[100%]=><3> xueya1
2	<6> xueya2 =[83%]=><5> age2
3	<4> xueya3 =[75%]=><3> age3
4	<13> xueya1 =[54%]=><7> age2

Table 8: Association rules from concept lattice in Figure 4.

No.	Rules
1	<3> age1 =[100%]=><3> xueya1
2	<11> xueya2 =[64%]=><7> age2
3	<7> age3 =[57%]=><4> xueya2
4	<13> xueya1 =[54%]=><7> age2
5	<14> age2 =[50%]=><7> xueya2

of constructing concept lattice is, and the less the association rules are generated; (2) association rules often vary according to the level of abstraction of attribute values. The finer the granularity of value abstraction is, the more the general rules are, and the more the detailed rules are; and (3) the method can reduce relation, control the size of the concept lattice, and satisfy the purpose of different users. In addition, there is a certain relationship among the association rules, as shown in Table 10.

In Table 10, there are some rules which can be obtained at some value level, but not at the other level. There are some rules of implication when the fuzzy processing to different

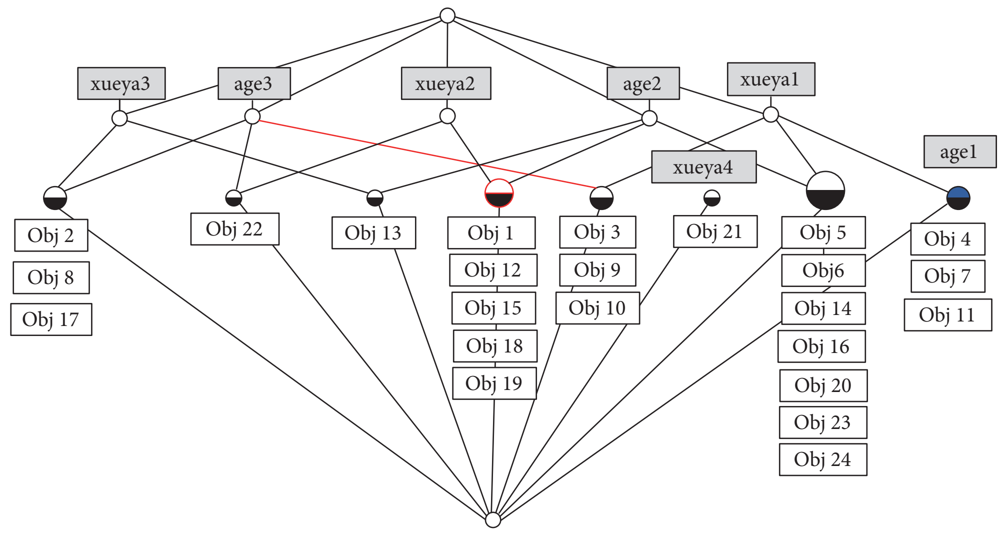

Figure 3: The concept lattice of A3BP4.

Table 9: Comparison of concept lattice from Figures 1 to 4.

	concepts	edges	height	width	rules
Figure 1	18	33	3	[10, 15]	39
Figure 2	15	27	3	[8, 12]	29
Figure 3	15	27	3	[8, 12]	25
Figure 4	11	18	3	[5, 8]	15

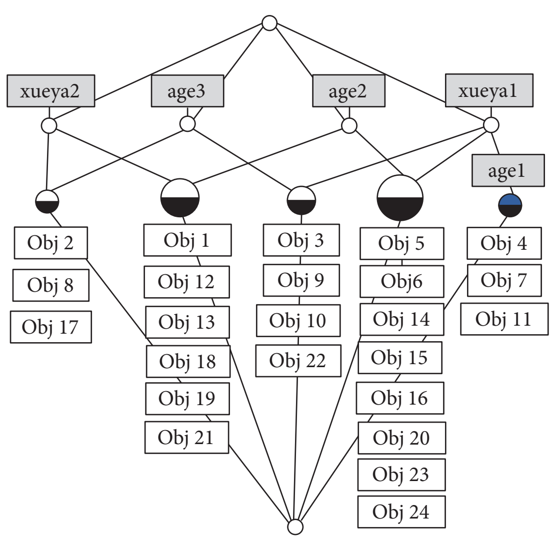

Figure 4: The concept lattice of A3BP2.

levels. For example, rule 3 contains rule 1; rule 4 contains rule 2.

The Second Experiment. This experiment is used to reduce the size of the context and control the number of concepts and improve the quality of the concept. By using the FKM algorithm, we can obtain some fuzzy values of attributes and hence obtain some rough relations from an original relation. Generally, the most common is to classify the age values into three categories and the blood pressure values into two categories. Therefore, Table 4 is converted to a binary relation, as shown in Table 11. The attributes are as follows: a1: young people, a2: middle-aged, a3: elderly, d1: hypertensive, d2: normotensive, g0: normal, g1: having ST-T wave abnormality, g2: probable or definite left ventricular hypertrophy by Estes' criteria, h1: maximum heart rate of low, h2: maximum heart rate of medium, and h3: maximum heart rate of high value, and n0-n4 are heart disease severity, 0 indicates normal, and 1 to 4 indicates serious degree, respectively.

By using lattice miner 1.4, which can be downloaded from the following website: https://sourceforge.net/projects/lattice-miner/, we obtain the concept lattice of Table 11, as shown in Figure 5. It contains 113 concepts and 283 edges, and its height is 6. We define the minimum support and the minimum confidence as 50% and 75%, respectively. Thus, we extract more than 900 rules, as shown in Table 12 (partly).

We select rules with larger values of support and confidence, as shown in Table 13.

To visualize these rules, we use column chart to show the support and confidence of the rules. The support and confidence of the rules in Table 13 are illustrated in Figure 6.

We can get a lot of useful information from Table 13. For example, rule 6 illustrates that the elderly have the larger probability of high blood pressure, and rule 3 illustrates that the elderly maximum heart rate is small. If we put a few rules together, we will obtain more valuable knowledge. For example, we can get a3 □ d1 from rules 6 and 12. From rules

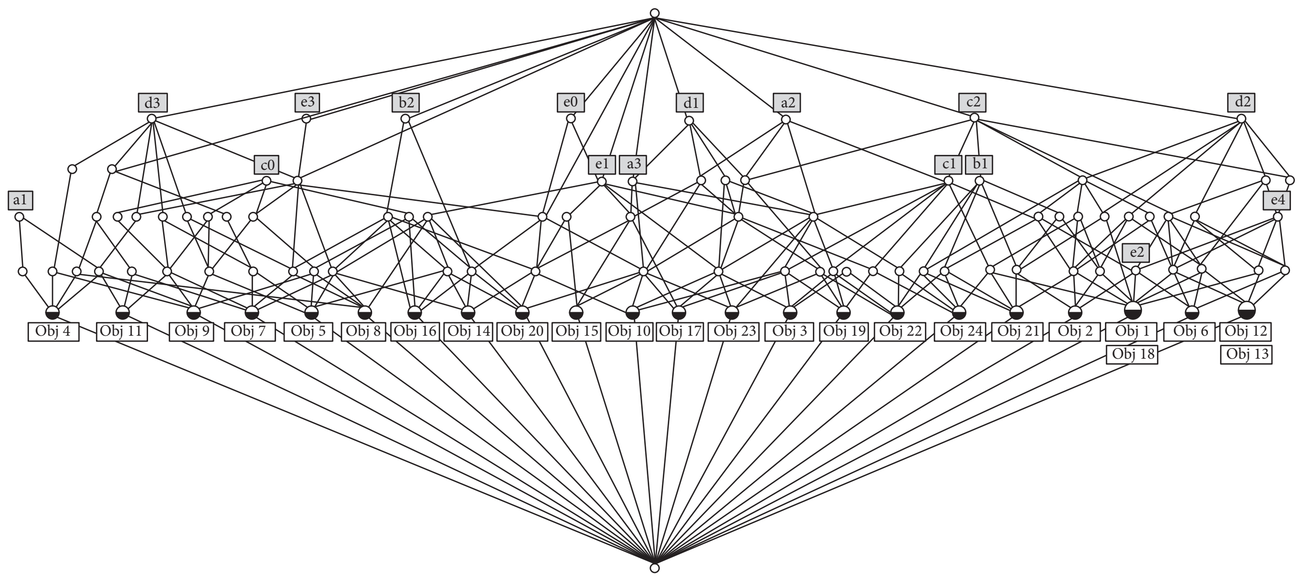

Figure 5: The concept lattice of Table 11.

Table 10: Part of the rules and instructions from Tables 5 to 8.

No.	Rules
1	<4> xueya3 =[75%]=><3> age6
2	<6> age6 =[50%]=><3> xueya2
3	<4> xueya3 =[75%]=><3> age3
4	<7> age3 =[57%]=><4> xueya2

Table 11: The binary relation from Table 4.

	a1	a2	a3	d1	d2	g0	g1	g2	h1	h2	h3	n0	n1	n2	n3	n4
1		+		+				+		+		+				
2			+	+				+	+					+		
3			+		+			+	+				+			
4	+				+	+					+	+				
5		+			+			+			+				+	
6		+			+			+		+						+
7	+				+			+	+						+	
8			+	+		+					+	+				
9			+		+	+					+				+	
10			+		+			+		+		+				
11	+				+			+			+	+				
12		+		+				+		+						+
13		+		+				+		+						+
14		+			+			+	+						+	
15		+		+		+					+		+			
16		+			+			+			+	+				
17			+	+		+				+					+	
18		+		+				+		+		+				
19		+		+		+			+				+			
20		+			+			+			+		+			
21		+		+			+		+					+		
22			+	+			+		+			+				
23		+			+		+		+						+	
24		+			+		+			+						+

TABLE 12: The part of rules from Figure 5.

No.	Rules	Support	Confidence
R1	a3⟹g1	88.11%	98.88%
R2	a3⟹n4	85.47%	95.92%
R3	a3⟹h1	85.47%	95.92%
R4	a3⟹n2	79.2%	88.88%
R5	a3⟹n3	79.2%	88.88%
R6	a3⟹d1	63.03%	70.74%
R7	h1⟹g1	77.55%	99.57%
R8	a1⟹g1	92.73%	98.59%
R9	a1⟹n4	90.09%	95.78%
R10	a1⟹n2	82.17%	87.36%
R11	a1⟹n3	83.16%	88.42%
R12	d1⟹a3	63.03%	92.71%
R13	d1⟹h1	53.13%	78.15%
R14	a3∧g1⟹n4	84.81%	96.25%
R15	h1∧n2⟹a3	64.68%	92.89%
R16	g1∧n4⟹a1	89.1%	94.07%
R17	h1∧n3⟹g1	71.94%	99.54%
R18	a1∧a3⟹g1	82.17%	98.8%
R19	a3∧n1⟹g1	70.95%	98.62%
R20	a1∧h1⟹g1	72.27%	99.72%
R21	g1∧h1∧n2⟹n4	66.99%	96.66%
R22	a1∧g1⟹n4	89.1%	96.08%
R23	g1∧h1∧n2⟹a3	64.35%	92.85%
R24	g1∧n2∧n3∧n4⟹a3	65.67%	91.28%

TABLE 13: The part of rules of higher support and confidence.

No.	Rules	Support	Confidence
R1	a3⟹g1	88.11%	98.88%
R2	a3⟹n4	85.47%	95.92%
R3	a3∧n1⟹g1	70.95%	98.62%
R4	a3⟹n3	79.2%	88.88%
R5	g1∧h1∧n2⟹a3	64.35%	92.85%
R6	a1⟹g1	92.73%	98.59%
R7	a1⟹n4	90.09%	95.78%
R8	a1⟹n2	82.17%	87.36%
R9	a1⟹n3	83.16%	88.42%
R10	d1⟹a3	63.03%	92.71%
R11	d1⟹h1	53.13%	78.15%
R12	g1∧n2∧n3∧n4⟹a3	65.67%	91.28%

(2, 4, and 5) and rules (9, 10, and 11), we have the following result: in the current data, the young people and elderly have little difference in heart disease.

5. Conclusions

In order to process natural texts for storing mobile generated data, we firstly extract some formal objects and attributes using NLP, secondly translate the texts into relations, and thirdly process the relations using FCA. In this paper, we mainly discuss the third step. In order to reduce the number of association rules, we propose a method based concept lattice and attribute analysis. This paper establishes the connection between the functional dependencies in an original relation R and corresponding rough relations, proposes the method for extracting reducts in R, and demonstrates the implementation of proposed method on an application in data mining of associative rules. By using rough-values of attributes, we can control the number of concepts and hence improve the quality of the concept. Our experiments show that the method is feasible and effective, which can be applied directly to regular mobile data such as spatial locations and

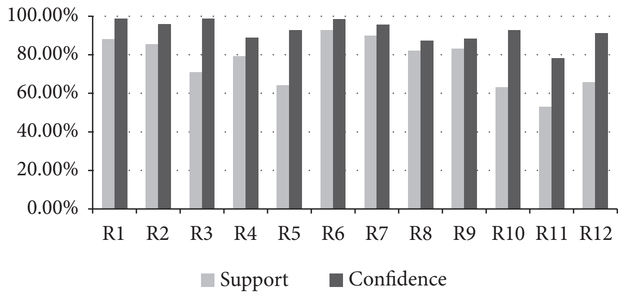

Figure 6: Column chart of support and confidence for rules in Table 13.

healthcare data. In the mobile computing, associative rules can provide potential and useful information for mobile clients.

In the mobile environment, there are the following interesting problems: (1) how to automatically translate natural texts into relations; (2) how to analyze those relations with columns having null values or more complex information; and (3) how to precisely capture mobile users' interests, in order to automatically provide corresponding recommended services.

Data Availability

We use the experimental data for heart disease dataset in UCI database, which can be downloaded from the website http://archive.ics.uci.edu/ml/datasets/Heart+Disease. Concept lattice tools can be downloaded from the following websites: https://sourceforge.net/projects/lattice-miner/ or https://sourceforge.net/projects/conexp/. If readers are interested in our results, they can contact us by email.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is partly supported by the Natural Science Foundation of China under Grants nos. 61572284 and 61502272 and the Promotive Research Fund for Excellent Young and Middle-Aged Scientists of Shandong Province under Grants nos. BS2014DX004 and BS2014DX005.

References

[1] T. Hao, X. Pan, Z. Gu, Y. Qu, and H. Weng, "A pattern learning-based method for temporal expression extraction and normalization from multi-lingual heterogeneous clinical texts," *BMC Medical Informatics and Decision Making*, vol. 18, 2018.

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

[3] J. Poelmans, D. I. Ignatov, S. O. Kuznetsov, and G. Dedene, "Formal concept analysis in knowledge processing: A survey on applications," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6538–6560, 2013.

[4] P. K. Singh, "Interval-Valued Neutrosophic Graph Representation of Concept Lattice and Its (α, β, γ)-Decomposition," *Arabian Journal for Science and Engineering*, vol. 43, no. 2, pp. 723–740, 2018.

[5] P. K. Singh, C. Aswani Kumar, and J. Li, "Knowledge representation using interval-valued fuzzy formal concept lattice," *Soft Computing*, vol. 20, no. 4, pp. 1485–1502, 2016.

[6] C. A. Kumar, "Fuzzy clustering-based formal concept analysis for association rules mining," *Applied Artificial Intelligence*, vol. 26, no. 3, pp. 274–301, 2012.

[7] K. Raza, "Formal concept analysis for knowledge discovery from biological data," *International Journal of Data Mining and Bioinformatics*, vol. 18, no. 4, pp. 281–300, 2017.

[8] X. Tu, Y. Wang, M. Zhang, and J. Wu, "Using Formal Concept Analysis to Identify Negative Correlations in Gene Expression Data," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 13, no. 2, pp. 380–391, 2016.

[9] P. K. Singh, A. K. Cherukuri, and J. Li, "Concepts reduction in formal concept analysis with fuzzy setting using Shannon entropy," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 1, pp. 179–189, 2017.

[10] S. M. Dias and N. J. Vieira, "Concept lattices reduction: Definition, analysis and classification," *Expert Systems with Applications*, vol. 42, no. 20, pp. 7084–7097, 2015.

[11] C. Aswani Kumar, S. M. Dias, and N. J. Vieira, "Knowledge reduction in formal contexts using non-negative matrix factorization," *Mathematics and Computers in Simulation*, vol. 109, pp. 46–63, 2015.

[12] Y. Lei, Y. Sui, and B. Cao, "Formal concept analysis in hybrid relational databases," *International Review on Computers and Software*, vol. 7, no. 6, pp. 2904–2910, 2012.

[13] J. Y. Yu and L. Gan, "FCA application in wireless sensor network," *Journal of Xinyang Agricultural College*, vol. 16, no. 3, pp. 123–126, 2006.

[14] F. Fkih and M. N. Omri, "IRAFCA: an O(n) information retrieval algorithm based on formal concept analysis," *Knowledge and Information Systems*, vol. 48, no. 2, pp. 465–491, 2016.

[15] T. Qian, L. Wei, and J. Qi, "Decomposition methods of formal contexts to construct concept lattices," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 1, pp. 95–108, 2017.

[16] V. Torra and S. Miyamoto, "A definition for I-fuzzy partitions," *Soft Computing*, vol. 15, no. 2, pp. 363–369, 2011.

[17] Y. Djouadi, B. Alouane, and H. Prade, "Fuzzy clustering for finding fuzzy partitions of many-valued attribute domains in a concept analysis perspective," in *Proceedings of the Joint 2009 International Fuzzy Systems Association World Congress, IFSA 2009 and 2009 European Society of Fuzzy Logic and Technology Conference, EUSFLAT 2009*, pp. 420–425, Portugal, July 2009.

[18] C. Aswani Kumar and S. Srinivas, "Concept lattice reduction using fuzzy K-Means clustering," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2696–2704, 2010.

[19] Y. Lei, J. Tian, and F. Jiang, "Two FCA-based methods for extracting concepts and corresponding concept lattices from hybrid relations," *International Journal of Simulation: Systems, Science and Technology*, vol. 17, no. 27, pp. 35.1-35.12, 2016.