UDC 004.8
Original Scientific Paper

Use of Concept Lattices for Data Tables with Different Types of Attributes

Peter Butka *peter.butka@tuke.sk*
Technical University of Košice
Faculty of Electrical Engineering and Informatics, Department of Cybernetics and Artificial Intelligence

Jozef Pócs *pocs@saske.sk*
Mathematical Institute, Slovak Academy of Sciences

Jana Pócsová *jana.pocsova@tuke.sk*
Technical University of Košice
BERG Faculty, Institute of Control and Informatization of Production Processes

Abstract

In this paper we describe the application of Formal Concept Analysis (FCA) for analysis of data tables with different types of attributes. FCA represents one of the conceptual data mining methods. The main limitation of FCA in classical case is the exclusive usage of binary attributes. More complex attributes then should be converted into binary tables. In our approach, called Generalized One-Sided Concept Lattices, we provide a method which deal with different types of attributes (e.g., ordinal, nominal, etc.) within one data table. Therefore, this method allows to create same FCA-based output in form of concept lattice with the precise many-valued attributes and the same interpretation of concept hierarchy as in the classical FCA, without the need for specific unified preprocessing of attribute values.

Keywords: formal concept analysis, concept lattices, data mining, fuzzy logic

1. Introduction

The large amount of available data and the growing needs for their analysis brings up the new challenges to the area of data mining. It is an emerging field where the need for more effective and understandable methods and algorithms for data analysts is evident. The common methods for analysis and adequate software are sometimes too complex and are able only to work with the limited set of attributes types. Moreover, most of the classical data mining solutions have limitations according to the understanding of data, which can limit the completeness of analysis and make it difficult.

On the other hand, conceptual models of data focus on their meaning and are capable of dealing with both unstructured and structured data. One of the conceptual data mining method is called Formal Concept Analysis (FCA, [5]), which is a theory of data analysis for identification of conceptual structures among data sets. It is also known as a theory of concept lattices based on the notion of formal context, which is represented by the binary relation between the set of objects and the set of attributes. FCA constructs objects-attributes pairs known as the formal concepts, which together with the hierarchical structure of them ordered by the generalization and specialization forms a concept lattice. It is a type of concept hierarchy, where each node represents a subset of objects (extent) with the corresponding set of attributes (intent). FCA has been found useful in data/text mining, knowledge discovery, information retrieval, business intelligence, as well as in other areas related to machine learning and artificial intelligence.

One of the problems usually related to the applications of classical FCA framework is in exclusive usage of binary data tables. It means that classical FCA approach provides crisp case, where object-attribute model is based on the binary relation (object has/has-not the attribute). In practice, there are natural examples of object-attribute models for which relationship between objects and attributes are represented by fuzzy relations. Generally, there are two possibilities how to deal with such issue. First option is scaling method, where each complex attribute is decomposed into appropriate number of binary attributes. Second option is to consider various fuzzy generalizations of classical FCA. We mention work of Bělohlávek [1], Georgescu and Popescu [6], Krajči [10], Popescu [14], Medina, Ojeda-Aciego, Ruiz-Calviño [11], and also an approach of Pócs generalizing all approaches based on Galois connections [13]. A survey and comparison of some existing approaches to fuzzy concept lattices is presented in [2].

A special case of fuzzy FCA is so-called one-sided concept lattice, where usually objects are considered as a crisp subsets and attributes obtain fuzzy values. The main advantage of such approaches is in combination of object clusters interpretation as in classical FCA, with fuzzy attributes for analysis of their non-binary attributes. From existing one-sided approaches we mention papers of Krajči [9], Yahia and Jaoua [3], Jaoua and Elloumi [8]. All recently known approaches allows only one type of structure for truth degrees. However, it is reasonable that in some data mining problems we have to consider object-attribute models with different truth value structures for their attributes (different types of attributes).

Our main aim is to introduce one-sided fuzzy approach for processing of non-homogeneous set of attributes within data tables applicable in data mining or similar domains. It means that this approach can be applied for different types of attributes, e.g., qualitative attributes with possible values 0 and 1, quantitative attributes from some real-valued interval, ordinal attributes, etc. The possibility of this approach was presented in [13]. The main definitions and proofs regarding the generalization of one-sided concept lattices based on the Galois connections were introduced in [4], some of the mathematical basics (necessary for the purposes of this paper) will be presented here. The processing of non-homogeneous data tables (i.e., data tables with different types of attributes) is the main difference in usage of FCA within the standard data mining solutions. The main advantage to them is direct usage of data models with different types of attributes (not only binary) without necessary conversion of attributes to set of binary attributes. Also, it is possible (instead of previous fuzzy FCA approaches) to work with the different types of attributes in one context using the proposed algorithm for creation of so-called generalized one-sided concept lattices. And finally, without the need for specific scaling or discretization method, it is possible to define any attribute from the raw data exactly in the form, as it was measured/extracted. It means that if data analyst has own understanding of truth values structure for his attribute, our approach allows him to work with the same structure during the whole process of analysis without the need for changing of his semantic model (i.e., how he understands the model in his mind) of attribute, which can be quite complex lattice-based structure (e.g., modeled directly for particular problem).

The paper is organized as follows. In section 2 we will describe the details regarding the method of generalized one-sided concept lattices and its algorithm. Use of different types of attributes in classic approaches by scaling to binary tables and benefits of our approach are described in section 3, together with the illustrative example, which shows the usage of our approach.

2. Generalized One-sided Concept Lattices

The main idea of fuzzifications of classical FCA is the usage of graded truth. In classical logic, each proposition is either true or false, hence classical logic is bivalent. In fuzzy logic, to each proposition there is assigned a truth degree from some scale L of truth degrees. The structure L of truth degrees is partially ordered and contains the smallest and the greatest element. If to the propositions ϕ and ψ are assigned truth degrees $\parallel \phi \parallel = a$ and $\parallel \psi \parallel = b$, then $a \leq b$ means that ϕ is considered less true than ψ. In object-attribute models the typical propositions are of the form “object has attribute in degree a". The structures of truth degrees commonly used in various modifications of fuzzy logic are real unit interval $[0,1]$, Boolean algebras, MV-algebras or more generally residuated lattices. All this structures are equipped with binary operations simulating implication and the logical connective and, but the important fact is that they form a complete lattice according to the partial order defined on them. In order to introduce the notion of the generalized one-sided concept lattices as a fuzzy generalization of FCA we assumed the only one minimal condition, i.e., the structures of truth degrees form complete lattices.

First, we recall some basic notions about complete lattices and Galois connections, which stay behind the theory of concept lattices. Further, we describe generalized one-sided concept lattices (GOSCL) and we give an algorithm for creation of GOSCL.

2.1 Mathematical preliminaries

In this subsection we briefly describe algebraic framework for GOSCL and we give a basic overview of the algebraic notions needed for our purposes.

As we already mentioned, theory of concept lattices is build within the framework of lattice theory, hence we recall some basic notions. We will use the standard terminology and the notation as in [7]. In the sequel, we will assume that the reader is familiar with the notion of partially ordered set. Let $(P,\leq)$ be a partially ordered set and $H\subseteq P$ be an arbitrary subset. An element $a\in P$ is said to be the *least upper bound* or *supremum* of H, if a is the upper bound of the subset H ($h\leq a$ for all $h\in H$) and a is the least of all elements majorizing H ($a\leq x$ for any upper bound x of H). We shall write $a=\sup H$ or $a=\bigvee H$. The concepts of the *greatest lower bound* or *infimum* is similarly defined and it will be denoted by $\inf H$ or $\bigwedge H$.

A partially ordered set $(L,\leq)$ is a *lattice* if $\sup\{a,b\}$ and $\inf\{a,b\}$ exist for all $a,b\in L$. A lattice L is called *complete* if $\bigvee H$ and $\bigwedge H$ exist for any subset $H\subseteq L$. Obviously, each finite lattice is complete. Note that any complete lattice contains the greatest element $1_L=\sup L=\inf\emptyset$ and the smallest element $0_L=\inf L=\sup\emptyset$. In what follows we will denote the class of all complete lattices by CL.

If L_i for $i\in I$ is a family of lattices the *direct product* $\prod_{i\in I}L_i$ is defined as the set of all functions

$$f: I\to\bigcup_{i\in I}L_i \tag{1}$$

such that $f(i)\in L_i$ for all $i\in I$ with the “componentwise" order, i.e, $f\leq g$ if $f(i)\leq g(i)$ for all $i\in I$. If $L_i=L$ for all $i\in I$ we get a direct power L^I. The direct product of lattices forms complete lattice if and only if all members of the family are complete lattices (see [7]). The straightforward computations show that the lattice operations in the direct product $\prod_{i\in I}L_i$ of complete lattices are calculated componentwise, i.e., for any subset $\{f_j : j\in J\}\subseteq\prod_{i\in I}L_i$ we obtain

$$(\bigvee_{j\in J}f_j)(i)=\bigvee_{j\in J}f_j(i) \text{ and } (\bigwedge_{j\in J}f_j)(i)=\bigwedge_{j\in J}f_j(i), \tag{2}$$

where this equalities hold for each index $i\in I$.

Let L be a complete lattice and $U\neq\emptyset$ be a set. L-sets (L-fuzzy subsets) in universe U are defined as elements of $L^U=\prod_{u\in U}L$. In practice the most common example are fuzzy subsets, where values are obtained from real unit interval $[0,1]$. Hence, in the sequel for a given family L_u, $u\in U$ one can consider about members of $\prod_{u\in U}L_u$ as a generalization of the notion of fuzzy subsets. In this case, any element $u\in U$ can obtain the values from the corresponding lattice L_u.

Crucial role in the mathematical theory of fuzzy concept lattices play special pairs of mappings between complete lattices, commonly known as Galois connections. Hence, we provide necessary details regarding Galois connections and related topics.

Let $(P, \leq)$ and $(Q, \leq)$ be an ordered sets and let $\varphi : P \to Q$ and $\psi : Q \to P$ be maps between these ordered sets. Such a pair (φ, ψ) of mappings is called a *Galois connection* between the ordered sets if the following condition is fulfilled:

$$p \leq \psi(q) \quad \text{if and only if} \quad \varphi(p) \geq q. \tag{3}$$

Galois connections between complete lattices are closely related to the notion of *closure operator* and *closure system*. Let L be a complete lattice. By a closure operator in L we understand a mapping $c_L : L \to L$ satisfying:

(a) $x \leq c_L(x)$ for all $x \in L$,

(b) $c_L(x_1) \leq c_L(x_2)$ for $x_1 \leq x_2$,

(c) $c_L(c_L(x)) = c_L(x)$ for all $x \in L$, (i.e., c_L is idempotent).

A subset X of a complete lattice L is called a closure system in L if X is closed under arbitrary meets. We note, that this condition guarantee that $(X, \leq)$ is a complete lattice, in which the infima are the same as in L, but the suprema in X may not coincide with those from L. It was shown in [15] that each closure system uniquely determines closure operator and similarly each closure operator uniquely determines closure system.

From the result of Ore [12] one obtain that any Galois connection between complete lattices L and M induces closure operators on L and M, respectively. Moreover, the corresponding closure systems are dually isomorphic. On the other side, each pair of dually isomorphic closure systems uniquely determine Galois connection.

As an example, we now describe Galois connections between power sets, which are the cornerstones of the classical FCA (see [5]).

Let (B, A, I) be a formal context, i.e., $B, A \neq \emptyset$ and $I \subseteq B \times A$ be a binary relation between B and A. There is defined a pair of mappings $^{\uparrow} : 2^B \to 2^A$ and $^{\downarrow} : 2^A \to 2^B$ as follows:

$$X^{\uparrow} = \{y \in A : (x, y) \in I \text{ for all } x \in X\}, \tag{4}$$

$$Y^{\downarrow} = \{x \in B : (x, y) \in I \text{ for all } y \in Y\}. \tag{5}$$

Note that for any set S symbol 2^S denotes the power set of the set S, i.e., the set of all subsets of S. As it can be proved, this pair of mappings forms Galois connection between 2^B and 2^A. On the other hand, any Galois connection between 2^B and 2^A can be obtained this way, thus this approach provides the most general method of creating Galois connections between power sets. The mentioned universality is one of the key feature of the classical FCA.

The properties of Galois connections allow us to construct complete lattices (the notion Galois lattices is also common in the literature), which are used as a framework for general fuzzy concept lattices. Formally, let (φ, ψ) be a Galois connection between complete lattices L and M. Denote by $\mathcal{G}_{\varphi,\psi}$ a subset of $L \times M$ consisting of all pairs (x, y) with $\varphi(x) = y$ and $\psi(y) = x$. Define a partial order on $\mathcal{G}_{\varphi,\psi}$ as follows:

$$(x_1, y_1) \leq (x_2, y_2) \quad \text{iff} \quad x_1 \leq x_2 \text{ and } y_1 \geq y_2. \tag{6}$$

Let (φ, ψ) be a Galois connection between complete lattices L and M. Then $(\mathcal{G}_{\varphi,\psi}, \leq)$ forms a complete lattice, where

$$\bigwedge_{i \in I}(x_i, y_i) = \Big(\bigwedge_{i \in I} x_i, \varphi\big(\psi\big(\bigvee_{i \in I} y_i\big)\big)\Big) \tag{7}$$

and

$$\bigvee_{i\in I}(x_i, y_i) = \Big(\psi(\varphi(\bigvee_{i\in I} x_i)), \bigwedge_{i\in I} y_i\Big). \tag{8}$$

for each family $(x_i, y_i)_{i\in I}$ of elements from $\mathcal{G}_{\varphi,\psi}$.

Let us remark that lattices of the form $\mathcal{G}_{\varphi,\psi}$ are considered as fuzzy concept lattices. In the special case if $L = 2^B$, $M = 2^A$ and the Galois connection $(\uparrow, \downarrow)$ is determined by binary relation $I \subseteq B \times A$, the resulting lattice $\mathcal{G}_{\uparrow,\downarrow}$ is usually denoted by $\mathfrak{B}(B, A, I)$ and it is called concept lattice determined by formal context (B, A, I).

If one of the lattices L or M is equal to the power set 2^B of some non-empty set B, we will refer to $\mathcal{G}_{\varphi,\psi}$ as one-sided concept lattices.

2.2 Theory of Generalized One-Sided Concept Lattices

As we mentioned, one-sided concept lattices are defined via Galois connection between 2^B (set of all subsets of a given set of objects) and arbitrary complete lattice. In homogeneous cases, i.e., if each attribute has assigned the same lattice representing the structure of truth values, this complete lattice was considered as L^A (L-fuzzy sets over the universe of the set of attributes A). The information about the relationship between objects and attributes is expressed by L-binary relation $R : B \times A \to L$ and its interpretation is given by: object b has attribute a in degree $R(b, a) \in L$.

In order to obtain a non-homogeneous generalization of one-sided approach for handling object-attribute models with different truth value structures for their attributes, it was introduced the notion of formal one-sided context, which differs only little from that commonly used (see [4]).

A 4-tuple $\big(B, A, \mathcal{L}, R\big)$ is said to be a *generalized one-sided formal context* if the following conditions are fulfilled:

(1) B is a non-empty set of objects and A is a non-empty set of attributes.

(2) $\mathcal{L} : A \to \mathsf{CL}$ is a mapping from the set of attributes to the class of all complete lattices. Hence, for any attribute a, $\mathcal{L}(a)$ denotes the complete lattice, which represents structure of truth values for attribute a.

(3) R is generalized incidence relation, i.e., $R(b, a) \in \mathcal{L}(a)$ for all $b \in B$ and $a \in A$. Thus, $R(b, a)$ represents a degree from the structure $\mathcal{L}(a)$ in which the element $b \in B$ has the attribute a.

Now we provide a basic results about generalized one-sided concept lattices.

Let $\big(B, A, \mathcal{L}, R\big)$ be a generalized one-sided formal context. Then we define a pair of mapping $\uparrow: 2^B \to \prod_{a\in A} \mathcal{L}(a)$ and $\downarrow: \prod_{a\in A} \mathcal{L}(a) \to 2^B$ as follows:

$$\uparrow \big(X\big)(a) = \bigwedge_{b\in X} R(b, a), \tag{9}$$

$$\downarrow (g) = \{b \in B : \text{ for each } a \in A,\ g(a) \le R(b, a)\}. \tag{10}$$

The pair $(\uparrow, \downarrow)$ defined by (9) and (10) forms a Galois connection between 2^B and $\prod_{a\in A} \mathcal{L}(a)$.

Consequently, for a given formal context $\big(B, A, \mathcal{L}, R\big)$ denote $\mathcal{C}\big(B, A, \mathcal{L}, R\big)$ the set of all pairs (X, g), where $X \subseteq B$, $g \in \prod_{a\in A} \mathcal{L}(a)$, satisfying

$$\uparrow \big(X\big) = g \ \text{ and } \ \downarrow (g) = X. \tag{11}$$

In this case, the set X is usually referred as *extent* and g as *intent* of the concept (X, g). Further we define partial order on $\mathcal{C}\big(B, A, \mathcal{L}, R\big)$ as follows:

$$(X_1, g_1) \leq (X_2, g_2) \quad \text{iff} \quad X_1 \subseteq X_2 \text{ iff } g_1 \geq g_2. \tag{12}$$

Let $\big(B, A, \mathcal{L}, R\big)$ be a generalized one-sided formal context. According to the equations (7) and (8) from previous subsection, the set $\mathcal{C}\big(B, A, \mathcal{L}, R\big)$ with the partial order defined above forms a complete lattice, where

$$\bigwedge_{i \in I} (X_i, g_i) = \Big(\bigcap_{i \in I} X_i, \uparrow\downarrow \big(\bigvee_{i \in I} g_i\big)\Big) \tag{13}$$

and

$$\bigvee_{i \in I} (X_i, g_i) = \Big(\downarrow\uparrow \big(\bigcup_{i \in I} X_i\big), \bigwedge_{i \in I} g_i\Big) \tag{14}$$

for each family $(X_i, g_i)_{i \in I}$ of elements from $\mathcal{C}\big(B, A, \mathcal{L}, R\big)$. This lattice is called *generalized one-sided concept lattice*.

It was proved in [4] that any Galois connection between 2^B and $\prod_{a \in A} \mathcal{L}(a)$ can be obtained by suitable generalized one-sided formal context, hence GOSCL have the same universality property as classical FCA. As a consequence of this fact, one easily obtain that GOSCL approach contain any existing approach to the one-sided concept lattices defined via Galois connections as a special case.

2.3 Incremental Algorithm for Creation of GOSCL

At the end of this section we provide an incremental algorithm for creation of generalized one-sided concept lattice. The main idea of the presented algorithm is to create the set of all intents corresponding to the Galois connection $(\uparrow, \downarrow)$.

Let $\big(B, A, \mathcal{L}, R\big)$ be a generalized one–sided formal context. For $b \in B$ put $R(b)$ an element of $\prod_{a \in A} \mathcal{L}(a)$ such that $R(b)(a) = R(b, a)$, i.e., $R(b)$ represents b-th row in data table R. Let 1_L denote the greatest element of $L = \prod_{a \in A} \mathcal{L}(a)$, i.e., $1_L(a) = 1_{\mathcal{L}(a)}$ for all $a \in A$.

Algorithm

```
Input:  generalized context (B, A, 𝓛, R)
begin
   create lattice L := ∏_{a∈A} 𝓛(a)
   C := {1_L}, C ⊆ L, C - set of all intents
   while(B ≠ ∅)
   {
      choose b ∈ B
      C* := C
      for each c ∈ C*
         C := C ∪ {c ∧ R(b)}
      B := B ∖ {b}
   }
   for each c ∈ C
      𝒞(B, A, 𝓛, R) := 𝒞(B, A, 𝓛, R) ∪ {(↓ (c), c)}
end
Output:  set of all concepts 𝒞(B, A, 𝓛, R)
```

Correctness of the algorithm yields from the following facts. Evidently, C is the smallest closure system in L containing $\{R(b) : b \in B\}$. Since $R(b) = \uparrow (\{b\})$, we obtain $C \subseteq \uparrow \big(2^B\big)$.

Conversely, if $g = \uparrow(X) \in \uparrow(2^B)$, then $g = \bigwedge_{b \in X} \uparrow(\{b\}) = \bigwedge_{b \in X} R(b) \in C$. Hence $C = \uparrow(2^B)$.

Let us remark that algorithm step for creation of the lattice $L := \prod_{a \in A} \mathcal{L}(a)$ can be done in various ways and it is up to programmer. For example, it is not necessary to store all elements of $\prod_{a \in A} \mathcal{L}(a)$, but it is sufficient to store only particular lattices $\mathcal{L}(a)$, since lattice operations in L are calculated component-wise.

3. Illustrative Example

In this section we present the illustrative example, which describes the usage of GOSCL for application of concept lattices to the tables with different types of attributes, together with the comparison of our approach and standard scaling. The method of scaling, described in monography [5], is simply the representation of complete lattices using the classical (crisp) concept lattices. In our case we will show the representation of some well-known types of scaling (nominal and ordinal), and also the representation of one attribute with the scale based on the general lattice.

The presented illustrative example is very simple due to fact that we only want to theoretically show the main difference our generalized approach and classical case (where complex attributes are preprocessed into sets of binary attributes using scaling and discretization), as well as the benefits of its usage. As we already mentioned, most of the fuzzy approaches cannot be used for data tables with different types of attributes. In next example we will show that GOSCL can work with data tables containing different lattice-based types of attributes (nominal, ordinal, general lattice, etc.) without the need for specific preprocessing to binary case. As we will see, the main benefits of our approach are in better interpretability of attributes and created concept lattice (due to fact that attributes are described as defined, not by the combination of many binary attributes), as well as in the possibility to work with more complex attributes without some specific preprocessing. We just want to emphasis that it is only an illustrative theoretical example with fictive data inputs and attributes.

Nominal scale is in general represented using the lattices of type M_n, i.e., consists of n-mutually incomparable elements with the smallest and the greatest element, respectively. This kind of scale is appropriate for attributes in data tables, with mutually incomparable values like names, types, etc. As example in our case we have lattice M_3, which represents three incomparable values, e.g., as a car species like Fiat, Ford and BMW. The visualization and representation of this nominal scale (attribute) is presented on Fig.1 and in Table 1 respectively. The table also shows that for correct representation of such scale we need three binary attributes, or n binary attributes in general.

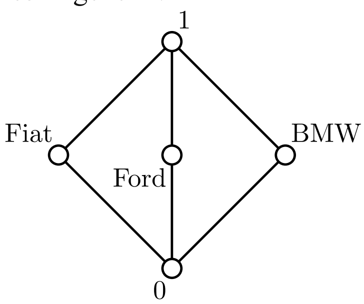

Figure 1: Nominal scale

	fiat	ford	bmw
Fiat	×		
Ford		×	
BMW			×

Table 1: Corresponding formal context

Representation of this scale by binary table means, that if an object has the value of this attribute equal to Fiat, then corresponding row of the table should be applied for resulting formal context in classical FCA.

The second commonly used attribute type is known as ordinal scale. In this case, the representation of such type is chain, i.e., totally ordered set where $x \leq y$ or $y \leq x$, for all elements

x, y. The usage of such scale is applicable for the attributes with the ordered values like numerical values, rankings, etc. As example in our case we have chain with four elements as some values expressing the driving levels, e.g., Beginner, Advanced, Expert and Professional. The visualization and representation of this ordinal scale (attribute) is presented on Fig.2 and in Table 2 respectively. For correct representation of such scale in classical FCA framework four binary attributes are needed, or in general it is needed to use m binary attributes for any ordinal scale of length m. Similarly, the corresponding row should be used in formal context for classical FCA.

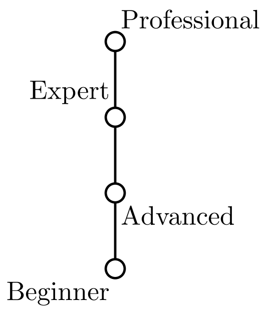

Figure 2: Ordinal scale

	`beg`	`adv`	`exp`	`pro`
Beginner	×			
Advanced	×	×		
Expert	×	×	×	
Professional	×	×	×	×

Table 2: Corresponding formal context

The third presented attribute is based on general lattice scale. While we are presenting the simple example, in general it is possible to define very complex attributes, e.g., interordinal, biordinal, multiordinal, etc. The usage of such scales is applicable for the attributes which are usually specific for some analysis. As example in our case we have a lattice with five elements as some values of car insurance types like Basic, Medium, Ext1, Ext2 and Full. The visualization and representation of this general scale (attribute) is presented on Fig.3 and in Table 3 respectively. For correct representation of such scale in classical FCA framework five binary attributes are needed. It can be shown, that in general case the number of binary attributes needed for the representation of general lattice scale with m values is from interval with the lower bound $\log_2 m$ and the upper bound m. As it can be seen, for larger and complicated lattices, the interpretation of the inputs based on the set of m binary attributes becomes difficult, especially for the cases without some specific patterns (like was in nominal or ordinal case).

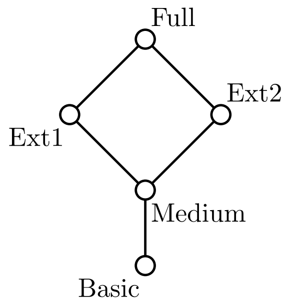

Figure 3: General lattice scale

	`bas`	`med`	`ext1`	`ext2`	`full`
Basic	×				
Medium	×	×			
Ext1	×	×	×		
Ext2	×	×		×	
Full	×	×	×	×	×

Table 3: Corresponding formal context

Numerical attributes (e.g., defined as interval of reals) are in crisp case usually discretized into ordinal attribute, which is then scaled into set of binary attributes (as it was shown in example of ordinal attribute representation). As it is not different usage according to crisp case, we do not need to describe it in illustrative example. Our approach is able to work also directly with the numerical attribute due to fact that it is only a specific ordinal scale for current examples in data tables. It can be seen as ordinal scale for ordered set of attribute values available in data table during the incremental runs of GOSCL algorithm. This is the reason why example of ordinal scale is sufficient for the comparison of classical FCA and our approach in case of numerical attribute. The main benefit for practical applications is that numerical attribute is used directly (without the preprocessing based on the discretization and ordinal scale).

Now we provide the illustrative example of the context with many-valued attributes. Consider data table based on the values of previously introduced scales with the set of eight objects representing some imaginary information about the particular cases of car insurance contracts. Further, there are three attributes A1, A2, A3, representing type of car, driving experiences, and type of insurance product, respectively. The particular values of attributes are defined on Fig.1, Fig.2 and Fig.3. The corresponding data table is depicted in Tab.4. As one can easily see, this data table also represents generalized one-sided formal context, as it was defined in section 2.

	A1	A2	A3
c1	BMW	Professional	Full
c2	Ford	Advanced	Full
c3	BMW	Professional	Ext1
c4	Fiat	Advanced	Medium
c5	BMW	Advanced	Full
c6	Ford	Advanced	Ext1
c7	Fiat	Beginner	Medium
c8	BMW	Advanced	Ext1

Table 4: Example of generalized one-sided formal context

For the usage of classical FCA framework, converting and preprocessing of this data table is needed. One of the possible approaches is the usage of scaling method. In the case of attributes in presented example scaling is described by the corresponding binary tables. Hence, this scaling process results in input table for classical FCA framework presented in Tab.5.

	fiat	ford	bmw	beg	adv	exp	pro	bas	med	ext1	ext2	full
c1			×	×	×	×	×	×	×	×	×	×
c2		×		×	×			×	×	×	×	×
c3			×	×	×	×	×	×	×	×		
c4	×			×	×			×	×			
c5			×	×	×			×	×	×	×	×
c6		×		×	×			×	×	×		
c7	×			×				×	×			
c8			×	×	×			×	×	×		

Table 5: Classical formal context using binary scaling of our example

After the application of classical FCA on converted tables the resulted concepts are described by plenty of binary attributes (12 in our example) and it can be difficult to interpret resulting concepts as a object-attribute pairs. On the other side, the application of GOSCL approach is possible directly for generalized one-sided formal context described in Tab.4. In this case object-attribute pairs consist of subsets of objects and generalized fuzzy subsets of attributes, i.e., in our case intent is formed by a triple of attribute values. Therefore, the interpretation of resulting generalized one-sided concept lattice can be easier and straightforward, due to directly applicable semantics and interpretation of attributes contained in concepts.

The application of GOSCL approach for table Tab.4 gives the concept lattice shown on Fig.4 (abbreviations of some values names were used for better reading). Every concept is represented by the rectangle with the lists of corresponding objects (extent, at the bottom of the concept description) and values of attributes (intent, at the top of the concept description). Lines between the concepts show relationships between the concepts in form of concept hierarchy, from the most general concept (top of the concept lattice) to the most specific concept (bottom of the concept lattice).

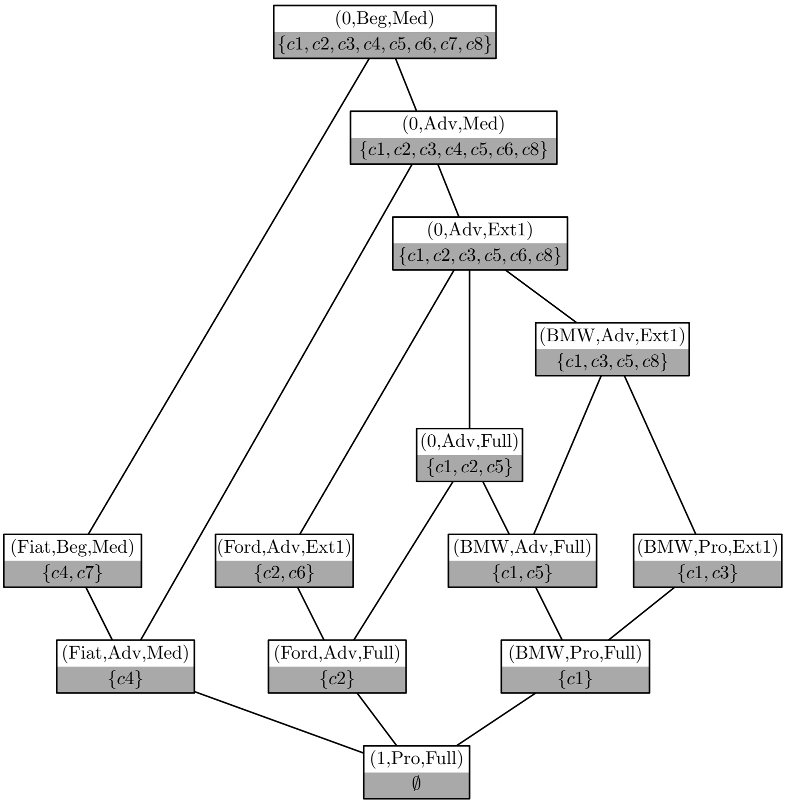

Figure 4: Generalized one-sided concept lattice

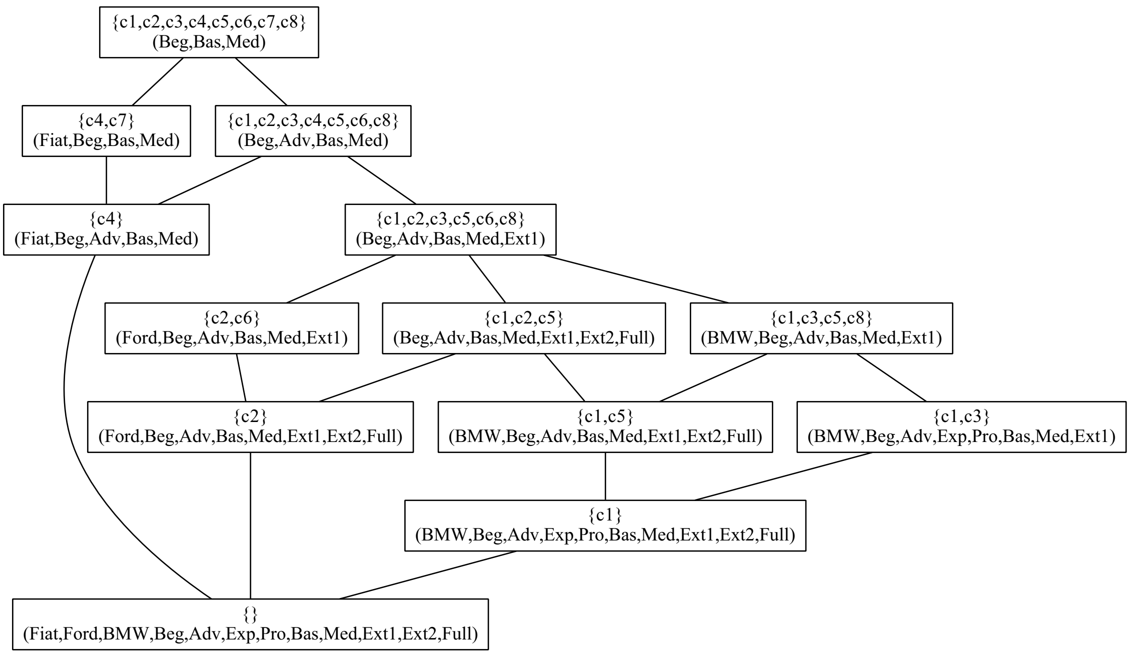

Figure 5: Classical concept lattice defined by Table 5

For comparison, the classical concept lattice involving new introduced binary attributes obtained using pair of operators (4) and (5) applied on Tab.5 is shown on Fig.5. The better interpretation of concepts can be easily seen from the fact that every concept (in classical case) should be described by the set of more binary attributes (in our case twelve attributes), which are grouped in sequences related to the concrete truth value of the particular attribute.

If there are many and more complex attributes, it can be very difficult for interpretation of the results within the classical FCA framework. For example, for attribute with the general truth value structure with 10 different values, the number of necessary binary attributes is bounded between $\log_2 10$ and the upper bound 10 (it depends on the structure of such attribute lattice). It means that our approach also provide space reduction for representation of the attributes (and input data), e.g., for 10 different values we need minimum of 4 binary attributes (in best case). If data analyst works with the classical FCA for 10 complex attributes, it is at least 40 binary attributes as minimum (it is usually more in practice). Therefore, data analyst should combine results for at least 40 binary attributes in order to interpret real values of original 10 attributes in created concept lattice.

In our small illustrative example we had 12 binary attributes for only 3 original attributes. The difference can be seen when you imagine that every concept have 12 binary values in its intent (instead of 3 concrete values), and you have to convert them into original values using table for every presented example of scale. Moreover, it is necessary (for data analyst) to prepare correct scales and their converting tables (what can be quite non-trivial for some complex general attribute lattice) into suitable representation of complex attributes in binary representation. Our approach allows data analyst to work directly with the attributes without specific preprocessing and conversion of attributes into set of binary attributes, and it helps him to interpret the results with the attributes values as they were originally defined. Moreover, it can be done for different types of attributes within the one input data table.

4. Conclusion

In this paper we have presented generalized one-sided concept lattices based on the Galois connections, which are applicable for object-attributes models with different types of attributes. This approach extends the possibility of classical FCA to work with different attributes without the need for their scaling and converting to binary tables. The resulted generalized one-sided concept lattice leads to better and straightforward interpretation of object-attribute pairs contained in concepts.

Acknowledgments

This work was supported by the Slovak Research and Development Agency under contracts APVV-0035-10 and APVV-0208-10.

References

[1] Bělohlávek, R. Lattices of Fixed Points of Fuzzy Galois Connections. *Mathematical Logic Quarterly*, 47(1), pp. 111-116, 2001.

[2] Bělohlávek, R; Vychodil, V. What is a fuzzy concept lattice? *Concept Lattices and their Applications (CLA 2005)*, pp. 34-45, Olomouc, Czech Republic, 2005.

[3] Ben Yahia, S; Jaoua, A. Discovering knowledge from fuzzy concept lattice. *Data Mining and Computational Intelligence*, pp. 167-190, Physica-Verlag, Heidelberg, Germany, 2001.

[4] Butka, P; Pócs, J. Generalization of one-sided concept lattices. *Computing and Informatics*, forthcoming.

[5] Ganter, B; Wille, R. *Formal concept analysis: Mathematical foundations*. Springer, Berlin, 1999.

[6] Georgescu, G; Popescu, A. Non-dual fuzzy connections. *Archive for Mathematical Logic*, 43, pp. 1009-1039, 2004.

[7] Grätzer, G. *Lattice Theory: Foundation*. Springer, Basel, 2011.

[8] Jaoua, A; Elloumi, S. Galois connection, formal concepts and Galois lattice in real relations: application in a real classifier. *The Journal of Systems and Software*, 60, pp. 149-163, 2002.

[9] Krajči, S. Cluster based efficient generation of fuzzy concepts. *Neural Network World*, 13(5), pp. 521-530, 2003.

[10] Krajči, S. A generalized concept lattice. *Logic Journal of the IGPL*, 13(5), pp. 543-550, 2005.

[11] Medina, J; Ojeda-Aciego, M; Ruiz-Calviño, J. Formal concept analysis via multi-adjoint concept lattices. *Fuzzy Sets and Systems*, 160, pp. 130-144, 2009.

[12] Ore, O. Galois Connexions. *Transactions of the American Mathematical Society*, 55, pp. 493-513, 1944.

[13] Pócs, J. Note on generating fuzzy concept lattices via Galois connections. *Information Sciences*, 185(1), pp. 128-136, 2012.

[14] Popescu, A. A general approach to fuzzy concepts. *Mathematical Logic Quarterly*, 50(3), pp. 265-280, 2004.

[15] Ward, M. The closure operators of a lattice. *Annals of Mathematics*, 43(2), pp. 191-196, 1942.