From Data to Pattern Structures: Near Set Approach

Marcin Wolski[1] and Anna Gomolińska[2]

[1] Maria Curie-Skłodowska University, Department of Logic and Cognitive Science,
Maria Curie-Skłodowska Sq. 4, 20-031 Lublin, Poland
marcin.wolski@umcs.lublin.pl
[2] University of Białystok, Faculty of Mathematics and Informatics,
Konstantego Ciołkowskiego 1M, 15-245 Białystok, Poland
anna.gom@math.uwb.edu.pl

Abstract. Pattern structures were introduced by Ganter and Kuznetsov in the framework of formal concept analysis (FCA). Usually these structures are regarded as a means of direct analysis of objects with complex descriptions. In FCA a pattern structure is *given* (in some sense a priori to the analysis) rather than *built* (a posteriori) from data. Of course, each data table/formal context induces a pattern structure; but it brings no new bits of information (and in that sense is trivial). In the present paper we would like to discuss two methods of generating (non-trivial) pattern structures from (simple) data. However, our study is done in the conceptual framework of rough set and near set theories (instead of FCA). The main reason is that both methods are based on the near set methodology.

1 Introduction

Pattern structures were originally introduced by Ganter and Kuznetsov [6] in the framework of formal concept analysis (FCA) [5, 24]. The initial structure of FCA is a formal context (G, M, I), which consists of a set of objects G, a set of properties M, and a relation $I \subseteq G \times M$, where $(g, m) \in I$ reads as *the object* g *has the property* m. A description of an object g is then given by the set of its properties. Pattern structures were introduced as a tool allowing one to directly analyse objects whose descriptions are given in some complex form, e.g., in the form of graphs, rather than in terms of simple properties. Thus, in this new frame, a given object g may be associated with some arbitrary structure s (instead of a set of properties) serving as a representation of (information about) g. However, it is additionally assumed that the set D of these representations has to form a meet semi-lattice. The main rationale is that the lattice meet operation $\sqcap$ would express the similarity (or nearness) between descriptions/representations. This enriched formal context, called a *pattern structure*, has a form of $(G, (D, \sqcap), \delta)$, where $\delta : G \to D$ maps objects to descriptions (sometimes also called patterns). In FCA a pattern structure is *given* rather than *built* from (simple) data. Of course, every formal context (G, M, I) is a pattern structure $(G, (\mathcal{P}M, \cap), I)$, where $\mathcal{P}M$ is the powerset of M and δ assigns to each object g its description $\{m \in M : (g, m) \in I\} \subseteq M$. But $(G, (\mathcal{P}M, \cap), I)$ brings no new bit of information when compared to (G, M, I), and in that sense is trivial. The main goal of our paper is to define methods of building non-trivial pattern structures from simple data. However, our study is done in the conceptual framework of rough set and near set theories (instead of FCA). The main reason is that both methods are based on (inspired by) the near set methodology.

Near set theory was introduced by Peters in [11, 12] as the result of collaborative work with Pawlak, Skowron, and Stepaniuk (see [13]). Originally, it was (mathematically) articulated within the conceptual framework of rough set theory, introduced by Pawlak in the early 1980s [18] (roughly in the very same time when Wille introduced FCA [23]). Therefore near sets share a lot of bits of conceptualisation together with rough sets. Another natural context of construing near sets is the spatial/topological nearness of sets. This topic may be traced back to the address by Riesz at the International Congress of Mathematicians in Rome in 1908 [22]. The most fundamental concepts were introduced by Čech in the years 1936-1939 (published in [2]), and Efremovič in 1933 (published in [3]). The study of formal connections between spatially near sets and descriptively near sets may be found in papers by Peters and Naimpally [9, 14]. In the framework of near sets we deal with objects via their descriptions (thus, in an indirect way) and two sets are near, provided that they share the same description (rather than the same object). However, the main emphasis is put upon descriptive nearness, rather than descriptions. In the present paper we would like to shift this emphasis along the lines of FCA and discuss (patterns of) descriptions and their nearness (similarity) given by means of pattern structures. To this end we are going to apply our previous results about modal relationships between rough set theory and near set theory [25], and explicitly define pattern structures within the rough and near theoretic conceptual framework. We offer two different methods of generating pattern structures; the first one is designed for symbolic attribute values, the second one is suitable for numerical values. Interestingly, we regard symbolic and numerical values as semantically different, and therefore we define two different methods of their processing. Pattern structures built in these ways add a lot of flexibility to the original theories; significantly, they allow one to analyse objects, set approximations, and nearness of sets at different levels of resolution. Thus, to some extent the system may be adapted to the context of application or environment.

2 Mathematics of Data

When we think about data in the context of computer science, we usually bear in mind a data table. This object may be mathematically defined/described in a number of different ways, e.g., [5, 15–18, 21, 24]. In this paper we start with a definition taken from rough set theory [16, 17, 19, 18] and stick to this conceptualisation through out the whole paper. Thus, we are going to discuss both formal concept analysis (FCA) and near set theory within the rough set framing (notation); specially, we change the original FCA notation used in Introduction. Additionally, we restrict our attention only to the specific fragments/parts of these theories which are relevant to our research; so, we do not offer a proper introduction to them. Yet, we shall try to make the paper self-contained.

Definition 1 (Information System [15, 18]). *A quadruple* $\mathcal{I} = (U, Att, Val, f)$ *is called an* information system*, where:*

- *U is a nonempty finite set of objects,*
- *Att is a nonempty finite set of attributes,*
- *$Val = \bigcup_{A \in Att} Val_A$, where Val_A is the value–domain of the attribute A,*
- *$f : U \times Att \to Val$ is an information function, such that for all $A \in Att$ and $x \in U$ it holds that $f(x, A) \in Val_A$.*

If f is a partial function, then the information system $\mathcal{I}$ is called incomplete. *If the codomain of f is the powerset of Val, then the system is called* multivalued, approximate, *or* nondeterministic.

The difference between multivalued, approximate, and nondeterministic systems is semantic. The first interpretation is *the object x has all values from V_A for the attribute A* (multivalued systems), the second reads as *the object x has a single value from the set V_A for the attribute A* (approximate systems), and finally *for the object x and the attribute A the set V_A provides some possible values* (nondeterministic systems) [15].

In what follows we shall confine our attention to the complete information systems. Every (sub)set of attributes $\mathcal{A} \subseteq Att$ induces an *approximation space*, which is a pair $(U, E_\mathcal{A})$, where the relation $E_\mathcal{A}$ is defined by

$$E_\mathcal{A} = \{(x, y) : f(x, A) = f(y, A) \text{ for all } A \in \mathcal{A}\}.$$

Definition 2 (Lower and Upper Approximations [20]). *A pair (U, E), where E is an equivalence relation, is called an* approximation space. *Define:*

$$Low_E(X) = \{x \in U : [x]_E \subseteq X\},$$

$$Upp_E(X) = \{x \in U : [x]_E \cap X \neq \emptyset\}.$$

$Low_E(X)$ is called the lower approximation of X, whereas $Upp_E(X)$ is called the upper approximation of X.

In order to simplify the notation, we shall write $Low_\mathcal{A}(X)$ and $Upp_\mathcal{A}(X)$ for $Low_{E_\mathcal{A}}(X)$ and $Upp_{E_\mathcal{A}}(X)$, respectively. In the case when $\mathcal{A} = Att$, we shall leave E without any subscript.

Definition 3 (Formal Context [5, 24]). *A formal context is a triple (U, Att, R), where U is a set of objects, Att a set of properties and $R \subseteq U \times Att$ is a binary relation, where $(x, A) \in R$ reads as* the object x has the property A.

There is some technical problem when dealing with complete information systems and formal contexts; namely, the correct formalisation of properties (predicates) as attributes (the other way is easy). Let us consider chess and a predicate *King*. A given object/piece may or may not be a *King*. Since the information function is defined for all pieces, we could take *no* and *yes* as the (only) values of *King* regarded as an attribute. However, the attribute *King* is substantially different than, e.g., *colour* restricted to two values *black* and *white*. Starting from *colour*, one can derive two properties, e.g., $colour = white$ and $colour = black$. Usually, we do not want to have two *King*-based properties ($King = yes$, $King = no$). In this settings, it seems that *King* should be construed as a single valued attribute (with some dummy value), but then the function f would need to be partial in order to cover objects such as, e.g., bishops. In this very sense, formal contexts are more general than complete information systems.

Definition 4 (Derivation Operators [5,24]). *For a formal context $\mathcal{C} = (U, Att, R)$, we define:*

$$R'(X) = \{A \in Att : (x, A) \in R, \text{ for all } x \in X\}$$

$$R'(\mathcal{A}) = \{x \in U : (x, A) \in R, \text{ for all } A \in \mathcal{A}\},$$

for all $X \subseteq U$ and $\mathcal{A} \subseteq Att$.

We have just defined the classical part of FCA. In order to deal with objects having complex descriptions Ganter and Kuznetsov introduced pattern structures.

Definition 5 (Pattern Structure [6]). *A pattern structure is a triple $(U, (D, \sqcap), \delta)$, where U is a set of objects, $(D, \sqcap)$ is a complete meet-semilattice of descriptions and $\delta : U \to D$ maps an object to a description.*

The derivation operators for a pattern structure are defined as follows:

$$X^{\square} = \bigsqcap_{x \in X} \delta(x) \quad \text{for all } X \subseteq U,$$

$$d^{\square} = \{x \in U : d \sqsubseteq \delta(x)\} \quad \text{for all } d \in D.$$

Given a set of objects X, the operator $\square$ returns the description $X^{\square}$ which is common to all objects in X. And given a description d, $\square$ returns the set $d^{\square}$ of all objects whose description subsumes d. The partial order $\sqsubseteq$ on D is defined as usual: $c \sqsubseteq d$ iff $c \sqcap d = c$.

As noted in the introduction, each formal context $\mathcal{C} = (U, Att, R)$ induces a pattern structure $(U, (\mathcal{P}Att, \cap), \delta)$, where

$$\delta(x) = \{A \in Att : (x, A) \in R\}.$$

For $a, b \in D$, the *pattern implication* $a \Rightarrow b$ holds if $a^{\square} \sqsubseteq b^{\square}$.

As already said, the concept of spatial nearness may be traced back to 1908. An introduction to spatially near sets is given by Naimpally [8]. The role of nearness in topology, proximity spaces, and uniform spaces is discussed in great detail in [14].

Definition 6 (Topology). *Let U be a set. A* topology *in U is family τ of subsets such that:*

- *each union of members of τ is a member of τ;*
- *each finite intersection of members of τ is also a member of τ;*
- *U and $\emptyset$ are members of τ.*

If τ is closed under arbitrary intersections, then τ is called an Alexandrov topology. *A couple (U, τ) is called a* topological space*; members of τ are called* open sets. *A set $X \subseteq U$ is* closed *if $U \setminus X \in \tau$.*

As usual, the smallest closed set containing X is denoted by $Cl(X)$, and the largest open set contained in X is denoted by $Int(X)$. By the standard abuse of notation the corresponding set operators are denoted by the same *names*: $Cl : \mathcal{P}U \to \mathcal{P}U$ sending X to $Cl(X)$ is called a *closure*, whereas an operator $Int : \mathcal{P}U \to \mathcal{P}U$ sending X to $Int(X)$ is called an *interior*.

Definition 7 (Spatial Nearness). *A spatial nearness relation* ρ *(called a* discrete *proximity) induced by* (U, τ) *is defined by*

$$\rho = \{(X, Y) \in \mathcal{P}U \times \mathcal{P}U : Cl(X) \cap Cl(Y) \neq \emptyset\} .$$

A pair (U, ρ) *is called a* proximity space*.*

If $(X, Y) \in \rho$, then we shall also write $X \rho Y$. The discrete proximity defined above is only one among many proximity relations discussed in topology. Actually, the theory of nearness spaces [1, 4] is very rich and we discuss here a very small fragment, which is relevant to our study.

As an example, let us consider an approximation space (U, E). The quotient space (a family of all equivalence classes of E) is denoted by U/E. If for $X \subseteq U$ it holds that $X = \bigcup \mathcal{Y}$, for some $\mathcal{Y} \subseteq U/E$, then X is called *definable* in (U, E). The following proposition belongs to the *folklore* of rough sets.

Proposition 1. *Let be given an approximation space* (U, E) *and the collection* σ *of all definable sets. Then* (U, σ) *is an Alexandrov topological space, whose closure operator is an upper approximation operator. Therefore, for the induced proximity space* (U, ρ)*, it holds that:*

$$\rho = \{(X, Y) : X, Y \subseteq U \quad \& \quad \overline{X} \cap \overline{Y} \neq \emptyset\}.$$

The next step in unpacking the richness of the concept of nearness relation ρ is the introduction of nearness collections ξ:

$$\xi(X) = \{Y \in U : X \rho Y\} .$$

These collections lead to very rich mathematical structures: *nearness spaces* [4] and the category **Near** [1].

As already said, we restrict our conceptual framework to rough set theory, therefore we introduce near set theoretic structures in an altered form.

Definition 8 (Perceptual System). A perceptual system *is a pair* $(U, \mathbb{F})$*, where* U *is a non-empty finite set of perceptual objects and* $\mathbb{F}$ *is a finite sequence of probe functions* $\phi_i : U \rightarrow Val_i$*.*

The probe functions describe physical features of objects and usually are regarded as sensors. Originally each function ϕ_i was defined by Peters [11, 12] as the real valued function. However, we allow probe functions to take any values. Thus, given an information system (U, Att, Val, f), if we order the set of attributes Att, then we may build a perceptual system $(U, \mathbb{F})$, where $\phi_i(x) = f(x, A_i)$. Because $\mathbb{F}$ is a sequence, it allows us to assign to each object $x \in U$ a feature vector $\Phi(x)$ over $\mathbb{F}$, i.e., a vector $(\phi_1(x), \phi_2(x), \ldots, \phi_n(x))$ of feature values that describe the object x. For a set $X \subseteq U$ let us define:

$$\mathcal{Q}(X) = \{\Phi(x) : x \in X\} .$$

Definition 9 (Descriptive Nearness). *A set* X *is* descriptively near *to* Y *in* $(U, \mathbb{F})$*, denoted by* $X \rho_\Phi Y$*, iff* $\mathcal{Q}(X) \cap \mathcal{Q}(Y) \neq \emptyset$*.*

It is worth noting that descriptively near sets can be spatially far sets. However, as noted in [25], we may convert descriptive nearness into its spatial counterpart. Firstly, each perceptual system $(U, \mathbb{F})$ gives rise to an approximation space (U, E), where $(x, y) \in E$ iff $\{x\}\ \rho_{\Phi}\ \{y\}$. Thus, secondly, we also have its Alexandrov topological space (U, σ) and the discrete proximity ρ_σ.

Proposition 2 ([25]). *A set X is descriptively near to Y in $(U, \mathbb{F})$ iff X is spatially near to Y in the corresponding topological space (U, σ).*

Let us now consider a nearness collection of X defined by $(U, \mathbb{F})$:

$$\xi_\sigma(X) = \{Y \in U : X\ \rho_\sigma\ Y\}\,.$$

It is worthy to emphasise the following similarity between upper approximations and nearness collections.

Proposition 3 ([25]). *For a perceptual system $(U, \mathbb{F})$ and its approximation space (U, E) it holds that:*

$$\overline{X} = \{x : \boxed{[x]_E \cap X \neq \emptyset}\} = \bigcup\{[x]_E : \boxed{[x]_E \cap X \neq \emptyset}\}, \tag{1}$$

$$\xi_\sigma(\{x\}) = \{X \subseteq U : \boxed{[x]_E \cap X \neq \emptyset}\}. \tag{2}$$

This proposition establishes a connection between rough set and near set perspectives on points, sets, and a granulation of U (in terms of equivalence classes). In the rough set framework we examine how points are related to a given set X via their minimal neighbourhoods. More precisely, we collect equivalence classes having nonempty intersection with a given set. In the near set approach we examine how sets are related to a given point via the same granulation. That is, we collect sets having nonempty intersection with a given equivalence class. This observation allows us to define new pattern structures induced by information/perceptual systems.

3 Nearness Pattern Structures

In the present section we introduce pattern structures induced by perceptual systems (derived from complete information systems). The main idea is to (somehow) employ Proposition 3. Vaguely speaking, we are going to take an object x, and for every attribute collect all "sets" (extensions of attribute values) having nonempty intersection with some specific equivalence class generated by x. So, we are going to define a kind of nearness collections, which can be informally described as:

$$\xi_i(\{x\}) = \{\text{extension of } a : a \in Val_{A_i} \ \&\ (\text{extension of } a) \cap [x]_{E_i} \neq \emptyset\},$$

where E_i is an equivalence relation, which corresponds to ξ_i.

Going into details, as earlier, we start with an information system (U, Att, Val, f) and convert it into a perceptual system $(U, \mathbb{F})$: $\phi_i(x) = f(x, A_i)$. The set $\mathcal{Q}(U)$ consists of all feature vectors/descriptions induced by our perceptual system. Following

Ganter and Kuznetsov [6], in order to convert $\mathcal{Q}(U)$ into a pattern structure, we need to provide this set with a lattice meet operation $\sqcap$. Let us recall that in FCA there is a special tool, called a *scale* or *scaling*, which allows one to convert an information system (U, Att, Val, f) into a formal context. Formally, a scale S for an attribute $A \in Att$ is a formal context $S = (Val_A, Val_A, R_A)$ having Val_A as both the set of objects and the set of attributes. Since each attribute may have its own scale, we actually work with a family of scales: $\mathcal{S} = \{(Val_A, Val_A, R_A) : A \in Att\}$. It is also assumed that for every A and its scale $S \in \mathcal{S}$ the identity relation $\{(v, v) : v \in Val_A\}$ is included in R_A. After scaling each pair attribute-value (A, v) is regarded as a separate attribute of the new context $C_{\mathcal{I}}^{S} = (U, \{(A, v)\}_{A \in Att,\ v \in Val_A}, R)$, where R is defined by

$$R = \{(x, (A, v)) : v \in f_s(x, A)\},$$

$$f_s(x, A) = \{v_i \in Val_A : f(x, A) = v \ \&\ (v, v_i) \in R_A\}.$$

It is easy to observe that every information system $\mathcal{I} = (U, Att, Val, f)$ may also be converted – by the application of a family of scales $\mathcal{S} = \{(Val_A, Val_A, R_A) : A \in Att\}$ – into a multivalued, approximate, or nondeterministic information system $\mathcal{I}_S = (U, Att, Val, f_s)$. A *nominal scale* S for A is the identity relation over Val_A. Let us now consider a multivalued information system $\mathcal{I}_S = (U, Att, Val, f_s)$ induced by $\mathcal{I}$ and a family $\mathcal{S}$ of nominal scales only. Then – instead of $\mathcal{Q}(U)$ – we may take $\mathcal{Q}_S(U)$ derived from $\mathcal{I}_S$. It is easy to observe that $\mathcal{Q}_S(U)$ consists of feature vectors of the form $\Phi(x) = (\{\phi_1(x)\}, \{\phi_2(x)\}, \ldots, \{\phi_n(x)\})$, for $x \in U$. Thus, $\mathcal{Q}_S(U)$ may be provided with natural lattice operations defined in terms of the standard set-theoretic meet and join operations:

$$\Phi(x) \sqcap \Phi(y) = (\{\phi_1(x)\} \cap \{\phi_1(y)\}, \{\phi_2(x)\} \cap \{\phi_2(y)\}, \ldots, \{\phi_n(x)\} \cap \{\phi_n(y)\})$$

$$\Phi(x) \sqcup \Phi(y) = (\{\phi_1(x)\} \cup \{\phi_1(y)\}, \{\phi_2(x)\} \cup \{\phi_2(y)\}, \ldots, \{\phi_n(x)\} \cup \{\phi_n(y)\})$$

The (complete) lattice generated by $\mathcal{Q}_S(U)$ will be denoted by $\mathcal{L}_{\mathbb{F}} = (\mathcal{Q}, \sqcap, \sqcup)$.

What we still lack to obtain a pattern structure is the map $\delta : U \to \mathcal{Q}$ assigning to each object $x \in U$ some (non-trivial) description/pattern. As already said, following Proposition 3, we would like to somehow employ the concept of nearness collections. The plan of building δ is as follows. Firstly, with each probe function ϕ_i we associate an equivalence relation $E_{Att \setminus \{A_i\}}$, where $Att \setminus \{A_i\}$ denotes set-theoretic difference of Att and $\{A_i\}$, determined by the corresponding attribute A_i. In order to simplify the notation let us denote it by E_i. Secondly, we take an object x, its equivalence class $[x]_{E_i}$, and then for the attribute A_i and the object x we collect all these values from V_{A_i}, which have nonempty intersection (by means of their extensions) with the class $[x]_{E_i}$.

$$\xi_i(\{x\}) = \{a \in Val_{A_i} : |a| \cap [x]_{E_i} \neq \emptyset\},$$

$$|a| = \{x \in U : \phi_i(x) = a\}, \text{ for } a \in Val_{A_i},$$

$$E_i = \{(x, y) : f(x, A_j) = f(y, A_j) \text{ for all } A_j \in \mathcal{A} \text{ such that } j \neq i\}.$$

Of course, $\phi_i(x)$ (that is, $f(x, A_i)$) must be included in the set $\xi_i(\{x\})$; or, in terms of the multivalued system obtained by scaling process, $f_s(x, A_i) \subseteq \xi_i(\{x\})$. It means that objects must not have $\emptyset$ as the value in their descriptions. Before we further formalise these ideas, let us clarify them by discussion of a simple information system of fruits (which is a modified version of data table discussed by Kuznetsov[3]).

$$U = \{apple, grapefruit, kiwi, plum, mango\},$$
$$Att = \{\textbf{colour}, \textbf{firm}, \textbf{smooth}, \textbf{form}\}$$
$$\mathbb{F} = (\textbf{colour}, \textbf{firm}, \textbf{smooth}, \textbf{form})$$

fruit	**colour**	**firm**	**smooth**	**form**
apple	*yellow*	*no*	*yes*	*round*
grapefruit	*yellow*	*no*	*no*	*round*
kiwi	*green*	*no*	*no*	*oval*
plum	*blue*	*no*	*yes*	*oval*
mango	*green*	*no*	*yes*	*oval*

(a)

fruit	**colour**	**firm**	**smooth**	**form**
apple	$\{yellow\}$	$\{no\}$	$\{yes\}$	$\{round\}$
grapefruit	$\{yellow\}$	$\{no\}$	$\{no\}$	$\{round\}$
kiwi	$\{green\}$	$\{no\}$	$\{no\}$	$\{oval\}$
plum	$\{blue\}$	$\{no\}$	$\{yes\}$	$\{oval\}$
mango	$\{green\}$	$\{no\}$	$\{yes\}$	$\{oval\}$

(b)

Fig. 1. (a) represents an information system of fruits $\mathcal{I}$; (b) represents the result of nominal scaling

As depicted in Fig. 1 (a), a database of fruits $\mathcal{I} = (U, Att, Val, f)$ is given. Firstly, by nominal scales $\mathcal{S}$, we convert it into a multivalued information system $\mathcal{I}_S$ (Fig. 1 (b)). Then we regard $\mathcal{I}_S$ as the perceptual system and compute $\mathcal{Q}_S(U)$. As easily noted, it includes, e.g.,

$$\Phi(kiwi) = (\{green\}, \{no\}, \{no\}, \{oval\}),$$
$$\Phi(mango) = (\{green\}, \{no\}, \{yes\}, \{oval\}).$$

The next step is to take $\mathcal{Q}_S(U)$ as the generator of $\mathcal{L}_{\mathbb{F}} = (\mathcal{Q}, \sqcap, \sqcup)$; thus, $\mathcal{Q}$ includes, e.g.,

$$\Phi(kiwi) \sqcap \Phi(mango) = (\{green\}, \{no\}, \emptyset, \{oval\}),$$
$$\Phi(kiwi) \sqcup \Phi(mango) = (\{green\}, \{no\}, \{no, yes\}, \{oval\}),$$
$$\top = (\{green, yellow, blue\}, \{no, yes\}, \{no, yes\}, \{oval, round\})$$
$$\bot = (\emptyset, \{no\}, \emptyset, \emptyset),$$

where $\top$ and $\bot$ denote the top and bottom of $\mathcal{L}_{\mathbb{F}}$, respectively. Now we can define the function $\delta_{\mathcal{I}} : U \to \mathcal{Q}$.

$$\delta_{\mathcal{I}}(x) = (\xi_1(\{x\}), \xi_2(\{x\}), \ldots, \xi_n(\{x\})).$$

Let us compute δ for, e.g., $mango$. For the attribute **colour**, we obtain $[mango]_{E_1} = \{mango, plum\}$; since the extension of $blue$, which is $\{plum\}$, has nonempty intersection with $[mango]_{E_1}$, we would like to add $blue$ to the the set **colour**$(mango)$; that is, $\xi_1(\{mango\}) = \{blue, yellow\}$. For the attribute **firm** we have $[mango]_{E_2} =$

[3] The lecture presented at 11th International Conference, ICFCA 2013, Dresden, Germany, May 21-24, 2013.

$\{mango\}$. Only $no \in Val_{\mathbf{firm}}$ has nonempty intersection (of its extension) with the class $[mango]_{E_2}$, therefore $\xi_2(\{mango\}) = \{no\}$. Next, for the attribute **smooth** we obtain $[mango]_{E_3} = \{mango, kiwi\}$. Thus $\xi_3(\{mango\}) = \{yes, no\}$. Finally, for **form**, we have $[mango]_{E_4} = \{mango\}$ and $\xi_4(\{mango\}) = \{oval\}$. In this way we have obtained:

$$\delta_{\mathcal{I}}(mango) = (\{yellow, blue\}, \{no\}, \{yes, no\}, \{oval\}).$$

Interestingly, the pattern structure $(U, (\mathcal{Q}, \sqcap), \delta_{\mathcal{I}})$ may be seen as a nondeterministic transformation of $\mathcal{I}$. Just define $(U, \mathbb{F}_\delta)$, where $\phi_i^\delta(x) = \xi_i(x)$ for every $\phi_i^\delta \in \mathbb{F}_\delta$. Continuing our example we may convert the fruit data table as depicted by Fig. 2.

fruit	**colour**	**firm**	**smooth**	**form**
$apple$	$\{yellow\}$	$\{no\}$	$\{yes, no\}$	$\{round\}$
$grapefruit$	$\{yellow\}$	$\{no\}$	$\{yes, no\}$	$\{round\}$
$kiwi$	$\{green\}$	$\{no\}$	$\{yes, no\}$	$\{oval\}$
$plum$	$\{blue, green\}$	$\{no\}$	$\{yes\}$	$\{oval\}$
$mango$	$\{blue, green\}$	$\{no\}$	$\{yes, no\}$	$\{oval\}$

Fig. 2. A nondeterministic version of $\mathcal{I}$ via the pattern structure $(U, (\mathcal{Q}, \sqcap), \delta_{\mathcal{I}})$

As said earlier, scales allow us to convert information systems into multivalued information systems. But $(U, \mathbb{F}_\delta)$ obtained from our fruit data set cannot be derived from $\mathcal{I}$ by any scale. E.g., because $green$ is related to $blue$, for any scale S we must obtain $\{green, blue\}$ for $kiwi$, but we have only $\{green\}$. Interestingly, it seems that $(U, \mathbb{F}_\delta)$ should be interpreted as an approximate information system, rather than as a multivalued information system (as in the case of systems obtained by scaling). We would like to regard this approximate information system as a *flexible* version of the original one. However, the relevance of this approach to data analysis yet need to be tested. In contrast to the standard information systems, approximate information systems are equipped with three important relationships between the elements of the universe.

Definition 10 (Informational Relations [15, 18]). *Let (U, Att, Val, f) be a multivalued information system; then one can define:*

informational indiscerniblity: $x\ Ind\ y$ iff $f(x, A) = f(y, A)$,
informational connectivity (similarity): $x\ Sim\ y$ iff $f(x, A) \cap f(y, A) \neq \emptyset$,
informational inclusion: $x\ Incl\ y$ iff $f(x, A) \subseteq f(y, A)$,

for all $A \in Att$ and $x, y \in U$.

Having three relations at disposal adds a lot of flexibility to data analysis. Since Ind is an equivalence relation, we can easily define the lower and upper approximation operators induced Ind. However, we can also generalise E to any reflexive relation $P \subseteq U \times U$ (e.g. Sim or $Incl$) and obtain generalised approximation operators. Let $[x]_P = \{y \in U : (x, y) \in P\}$ and define:

$$Low_P(X) = \{x \in U : [x]_P \subseteq X\},$$

$$Upp_P(X) = \{x \in U : [x]_P \cap X \neq \emptyset\}.$$

Since $E \subseteq Ind \subseteq Incl \subseteq Sim$, multivalued information systems allow us to introduce *zooming*. Let $\mathcal{I}$ be our fruit data table. Then the approximation space $(U, E_{\mathcal{I}})$ induced by $\mathcal{I}$ gives the discrete topological space. So, at the highest resolution $apple$ is viewed as $Upp_{E_{\mathcal{I}}}(\{apple\}) = \{apple\}$. At the next level we obtain $Upp_{Ind}(\{apple\}) = \{apple, grapefruit\}$. Further zooming out does not change anything. Let us now consider $kiwi$: $Upp_{E_{\mathcal{I}}}(\{kiwi\}) = \{kiwi\}$; $Upp_{Ind}(\{kiwi\}) = \{kiwi\}$. Let us zoom out once again: $Upp_{Incl}(\{kiwi\}) = \{kiwi, mango\}$. And finally: $Upp_{Incl}(\{kiwi\}) = \{kiwi, plum, mango\}$.

Apart from set approximations, another important task in rough set theory is to compute decision rules. In the case of approximate information systems obtained by pattern structures, we replace standard decision rules by pattern implications. If D is a decision (binary) attribute, then a pattern implication $P \to D$ would hold, if $P^{\square} \subseteq |D|$. Our hypothesis (to be tested in the future) is that pattern implications will be more stable during the evolution/extension (by new objects) of the original information system.

So far we have considered only symbolic values. Usually, numeric values are preprocessed and then represented in the form which is further processed in the same way as symbolic values. However, our idea is to regard them as semantically different, and to define a different form of pattern structures for numeric values. For example, let us consider an attribute **price**. What would $\xi_{\textbf{price}}(apple) = \{2.0, 2.5, 3.1\}$ mean?

In order to answer this question let us come back to the original concept of perceptual system $(U, \mathbb{F})$, where all probe functions are real valued.

Definition 11 (Perceptual Tolerance Relation [11, 12]). *Let $\langle U, \mathbb{F}\rangle$ be a perceptual system (where $\phi_i : U \to \mathbb{R}$, for every $\phi_i \in \mathbb{F}$) and let $\varepsilon \in \mathbb{R}$. For every $\mathcal{B} \subseteq \mathbb{F}$ the* perceptual tolerance relation *$T_{\mathcal{B},\varepsilon}$ is defined as follows:*

$$T_{\mathcal{B},\epsilon} = \{(x, y) \in U \times U : \ (|\phi_i(x) - \phi_i(y)| \leq \varepsilon) \ \textit{for all}\ \phi_i \in \mathcal{B}\}.$$

For notational convenience, when $\mathcal{B} = \mathbb{F}$, this relation is denoted by T_ε.

This definition suggests us a new transformation of the original system. To make the paper notationally consistent, let us assume that we start with a complete information system $\mathcal{I} = (U, Att, Val, f)$, such that $Val_A \subseteq \mathbb{R}$, for every $A \in Att$. Now we define a new information system $\mathcal{I}_\varepsilon = (U, Att, Val, f_\varepsilon)$, where

$$f_\varepsilon(x, A) = [f(x, A) - \varepsilon, f(x, A) + \varepsilon].$$

Thus, f_ε sends an object x and an attribute A to an interval of real numbers. As in the previous case of symbolic values, firstly we regard $\mathcal{I}_\varepsilon$ as the perceptual system $(U, \mathbb{F}_\varepsilon)$ (i.e. Att is a sequence and $\phi^i_\varepsilon = f_\varepsilon(x, A_i)$), secondly we take the set $\mathcal{Q}_\varepsilon(U)$ of all feature vectors over $\mathcal{I}_\varepsilon$, and finally we build a complete meet-semi lattice.

The natural pattern structure for real intervals suggested by Kuznetsov – e.g., [7] – is defined as $(\mathbf{I}, \sqcap)$, where $\mathbf{I} = \{[a, b] : a, b \in \mathbb{R}\}$, and

$$[a_1, b_1] \sqcap [a_2, b_2] = [min(a_1, a_2), max(b_1, b_2)].$$

Thus, let us generate a complete meet-semi lattice $\mathcal{L}_\varepsilon = (\mathcal{Q}_\varepsilon, \sqcap)$ from $\mathcal{Q}_\varepsilon(U)$ and $\sqcap$ (by abuse of notation we use the same symbol for both operators). Of course, the full definition is as follows:

$$\Phi_\varepsilon(x) \sqcap \Phi_\varepsilon(y) = (\phi^1_\varepsilon(x) \sqcap \phi^1_\varepsilon(y), \phi^2_\varepsilon(x) \sqcap \phi^2_\varepsilon(y), \ldots, \phi^n_\varepsilon(x) \sqcap \phi^n_\varepsilon(y)).$$

Then, for any real valued attribute A_i, we would have:

$$\xi^i_\varepsilon(\{x\}) = \sqcap_{y \in [x]_{E_i}} f_\varepsilon(y, A_i) \in \mathbf{I}.$$

However, the equivalence relation E_i seems to be too strong, therefore we would like to replace it by the perceptual tolerance T^i_ε:

$$T^i_\epsilon = \{(x, y) \in U \times U : |f(x, A_j) - f(y, A_j)| \leq \varepsilon) \text{ for all } A_j \in Att \text{ such that } j \neq i\}.$$

So, the final definition δ_ε is:

$$\delta_\varepsilon(x) = (\xi^1_\varepsilon(\{x\}), \xi^2_\varepsilon(\{x\}), \ldots, \xi^n_\varepsilon(\{x\})) \in \mathcal{Q}_\varepsilon \text{ and } \xi^i_\varepsilon(\{x\}) = \sqcap_{y \in [x]_{T^i_\varepsilon}} f_\varepsilon(y, A_i) \in \mathbf{I}.$$

In this way we obtain a *perceptual pattern structure* $(U, (\mathcal{Q}_\varepsilon, \sqcap), \delta_\varepsilon)$ induced by a perceptual system $(U, \mathcal{F}_\varepsilon)$ of (U, Att, Val, f), where $Val_A \subseteq \mathbb{R}$, for every $A \in Att$. The elements of $\mathcal{Q}_\varepsilon$ are called *perceptual patterns*. Now we can translate the informational relations into the realm of perceptual patterns:

$$[a_1, b_1] \cap [a_2, b_2] = [max(a_1, a_2), min(b_1, b_2)].$$

Of course, any $[a, b]$, such that $a > b$, denotes the empty interval [ ]. Two patterns P_1 and P_2 are indiscernible iff $\mathbf{proj}_i(P_1) = \mathbf{proj}_i(P_2)$, for all $i \leq n$, where $\mathbf{proj}_i(P)$ denotes the standard ith projection of a given n-dimensional vector P. Two patterns P_1 and P_2 are *similar* iff $\mathbf{proj}_i(P_1) \cap \mathbf{proj}_i(P_2) \neq [\,]$, for all $i \leq n$.

Definition 12 (Structural Nearness of Sets). *Let be given a complete information system $\mathcal{I} = (U, Att, Val, f)$, such that $Val_A \subseteq \mathbb{R}$, for every $A \in Att$. Then two sets $X, Y \subseteq U$ are structurally near iff there exist $x \in X$ and $y \in Y$, such that the patterns $\delta_\varepsilon(x)$ and $\delta_\varepsilon(y)$ are similar.*

In near set theory the main task is to find sets which are near to each other. We suggest to test the idea of finding sets which are structurally near.

4 Conclusions

In the paper we adopted and adapted pattern structures [6] defined in the framework of formal concept analysis [5, 23, 24] to the conceptual framing of rough set theory [15–18, 20] and near set theory [11–13]. In contrast to FCA, the pattern structures introduced in the paper were computed directly from data; they were regarded as *a posteriori* structures rather than *a priori*, as in the case of FCA.

Pattern structures allowed us to convert a given information system into an approximate information system [15]. From a technical standpoint, this conversion leads to coverings of value sets V_A, for each attribute A, and can be compared to symbolic value grouping introduced by Nguyen *et al.* [10]. However, we simply computed these coverings, whereas the main objective of [10] was to find optimal partitions. Moreover, our goal was to define flexible descriptions (patterns), whereas Nguyen *et al.* aimed at decreasing the number of attribute values. The detailed study of relationships between these two approaches would be a topic for another research.

References

1. Adámek, J., Herrlich, H., Strecker, G.: Abstract and Concrete Categories. Wiley-Interscience Publ., London (1990)
2. Čech, E.: Topological Spaces. John Wiley & Sons Ltd., London (1966)
3. Efremovič, V.: The geometry of proximity I. Mat. Sb. 31 (in Russian) MR 14, 1106, 189–200 (1951)
4. Herrlich, H.: A concept of nearness. Gen. Top. & Appl. 4, 191–212 (1974)
5. Ganter B, Wille R. Formal Concept Analysis. Mathematical Foundations. Springer Verlag (1999)
6. Ganter, B., Kuznetsov, S.O.: Pattern structures and their projections. In Delugach, H., Stumme, G. (eds.) Conceptual Structures: Broadening the Base SE - 10, LNCS 2120, 129–142 (2001)
7. Kuznetsov, S.O.: Pattern structures for analyzing complex data. In Sakai, H., Chakraborty, M. et al. (eds.) Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Conference proceedings RSFDGrC 2009, LNCS 5908 (2009)
8. Naimpally, S.: Proximity approach to general topology. Lakehead University (1974)
9. Naimpally, S., Peters, J.: Topology with Applications. Topological Spaces via Near and Far. World Scientific, Singapore (2013)
10. H.S. Nguyen, H.S., Nguyen, S.H.: Pattern extraction from data. Fund. Inf. 34, 129–144 (1998)
11. Peters, J.: Near sets. Special theory about nearness of objects. Fundam. Inf. 75(1-4), 407–433 (2007)
12. Peters, J.: Near sets. General theory about nearness of objects. Applied Mathematical Sciences 1(53), 2609–2029 (2007)
13. Peters, J., Skowron, A., Stepaniuk, J.: Nearness in approximation spaces. In: Proc. Concurrency, Specification and Programming (CS&P 2006). Humboldt Universitat, 435–445 (2006)
14. Peters, J., Naimpally, S.: Applications of near sets. Amer. Math. Soc. Notices 59(4), 536–542 (2012) DOI: http://dx.doi.org/10.1090/noti817.
15. Orłowska E, Pawlak Z.: Representation of nondeterministic information. Theoretical Computer Science 29, 27–39 (1984)
16. Pawlak, Z.: Classification of Objects by Means of Attributes. Institute of Computer Science, Polish Academy of Sciences PAS 429 (1981)
17. Pawlak, Z.: Rough sets. Int. J. Computer and Information Sci. 11, 341–356 (1982)
18. Pawlak Z. Systemy informacyjne. Podstawy teoretyczne. Warszawa, WNT (1983)
19. Pawlak, Z.: Rough sets and decision tables, LNCS 208, 186–196 (1985)
20. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publisher (1991)
21. Pratt, V.: Chu spaces and their interpretation as concurrent objects. In van Leeuwen, J. (Ed.), Computer Science Today. Recent Trends and Developments, LNCS 1000, 392-405 (1995)
22. Riesz, F.: Stetigkeitsbegriff und abstrakte mengenlehre. IV Congresso Internazionale dei Matematici II, 18–24 (1908)
23. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In Rival, I. (Ed.) Ordered sets. Reidel, Dordrecht-Boston, 445–470 (1982)
24. Wille R.: Concept lattices and conceptual knowledge systems. Computers & Mathematics with Applications 23, 493–515 (1992)
25. Wolski, M., Gomolińska, A.: Rough and Near: Modal History of Two Theories. In: Polkowski, L., et al. (eds) Rough Sets - International Joint Conference, IJCRS 2017, Olsztyn, Poland, July 3-7, 2017, Proceedings, Part I, LNCS 10313, 109–122 (2017)