



Knowledge representation and processing with formal concept analysis

Sergei O. Kuznetsov^{1*} and Jonas Poelmans^{1,2}

During the last three decades, formal concept analysis (FCA) became a well-known formalism in data analysis and knowledge discovery because of its usefulness in important domains of knowledge discovery in databases (KDD) such as ontology engineering, association rule mining, machine learning, as well as relation to other established theories for representing knowledge processing, like description logics, conceptual graphs, and rough sets. In early days, FCA was sometimes misconceived as a static crisp hardly scalable formalism for binary data tables. In this paper, we will try to show that FCA actually provides support for processing large dynamical complex (may be uncertain) data augmented with additional knowledge. © 2013 Wiley Periodicals, Inc.

How to cite this article:

WIREs Data Mining Knowl Discov 2013, 3: 200–215 doi: 10.1002/widm.1088

INTRODUCTION

The name formal concept analysis (FCA) was coined by Rudolf Wille, who started his research project of ‘restructuring lattice theory’ in late 1970s at Technical University of Darmstadt.¹ FCA takes its philosophical roots in Logique de Port-Royale of Arnauld and Nicole,² and its mathematical roots in the work by Birkhoff,³ Ore,⁴ and other on Galois connections and lattices of closed sets, and in the early work on applying lattice-theoretical ideas in information science, like it was done by Barbut and Monjardet.⁵ In early days, FCA was sometimes misconceived as a static crisp hardly scalable formalism for binary data tables. In this paper, we will try to show that FCA actually provides support for processing large dynamical complex (may be uncertain) data augmented with additional knowledge. During the last three decades, FCA-based models of knowledge representation, discovery,⁶ and processing were intensively designed and used in many research and

industrial projects all over the world, see the reviews by Carpineto and Romano⁷ and Poelmans et al.⁸ on FCA applications in information retrieval, the review by Priss⁹ on applications of FCA in information retrieval, knowledge discovery, artificial intelligence in general, the overview of FCA software by Tilley,¹⁰ the overview by Tilley et al.¹¹ on FCA applications in software engineering, the survey by Lakhal and Stumme¹² of FCA-based applications in data mining, the analytical review by Doerfel et al.¹³ with bibliometric analysis of publications on FCA, and study of the FCA research community. At the same time, many important connections to other established directions of knowledge discovery and processing were established: description logics (DLs),¹⁴ concept graphs,¹⁵ mining association rules,¹⁶ rough set theory (RST),¹⁷ biclustering and multimodal clustering,¹⁸ and others.

The remainder of this paper is organized as follows. In section *FCA: Main Definitions*, we introduce the main notions and definitions of FCA. In section *FCA Algorithms and Scalability*, we discuss scalability problems and algorithms of FCA. In section *Relationships of FCA to Models of Knowledge Representation and Processing*, we consider connections of FCA to other research areas of knowledge representation and processing. In section *Extensions of FCA*, we give an overview of several ‘extensions’ of

The authors have declared no conflicts of interest in relation to this article.

*Correspondence to: skuznetsov@hse.ru

¹National Research University Higher School of Economics, Moscow, Russia

²KU Leuven, Faculty of Business and Economics, Leuven, Belgium

DOI: 10.1002/widm.1088

TABLE 1 | Example of a Formal Context

	Browsing	Mining	Software	Web Services	FCA	Information Retrieval
Paper 1	X	X	X		X	
Paper 2			X		X	X
Paper 3		X		X	X	
Paper 4	X		X		X	
Paper 5				X	X	X

the main model. *Conclusions* section concludes the paper.

FCA: MAIN DEFINITIONS

In what follows, we will keep to the notation of the book,¹⁹ which is by now the main reference for the mathematical foundations of FCA.

The underlying notion of FCA is that of a *formal context*, which is a triple of sets (G, M, I) , where $I \subseteq G \times M$ is a binary relation. This triple can be represented by a cross-table with the set of rows G (called objects), columns M (called attributes), and crosses representing relation I , see an example in Table 1, where objects are papers, attributes as terms, and incidence relation shows how terms occur in papers.

The notion of a formal concept is central to FCA. The way FCA looks at concepts is in line with the international standard ISO 704 that formulates the following definition: ‘A concept is considered to be a unit of thought constituted of two parts: its extent and its intent’. The extent consists of all objects belonging to the concept, whereas the intent comprises all attributes shared by those objects. Let us illustrate the notion of concept of a formal context using the data in Table 1. For a set of objects $O \subseteq G$, the set of common attributes can be defined by:

$$A = O' = \{m \in M \mid (o, m) \in I \text{ for all } o \in O\}$$

Take the attributes that describe paper 4 in Table 1, for instance. By collecting all papers of this context that share these attributes, we get to a set $O \subseteq G$ consisting of papers 1 and 4. This set O of objects is related to the set A consisting of the attributes ‘browsing,’ ‘software,’ and ‘FCA’.

$$O = A' = \{o \in G \mid (o, m) \in I \text{ for all } m \in A\}$$

That is, O is the set of all objects sharing all attributes of A , and A is the set of all attributes that

are valid descriptions for all the objects contained in O . Each such pair (O, A) is called a formal concept (or concept) of the given context. The set $A = O'$ is called the *intent*, whereas $O = A'$ is called the *extent* of the concept (O, A) .

There is a natural hierarchical ordering relation between the concepts of a given context that is called the subconcept–superconcept relation.

$$(O_1, A_1) \leq (O_2, A_2) \Leftrightarrow (O_1 \subseteq O_2 \wedge A_2 \subseteq A_1)$$

The ordered set of all concepts makes a complete lattice called the concept lattice of the context, that is, every subset of concepts has infimum (meet) and supremum (join) w.r.t. \leq . The line diagram of the concept lattice of the context from Table 1 is given in Figure 1. The nodes of this diagram represent the formal concepts. The shaded boxes (upward) linked to a node represent the attributes used to name the concept. The nonshaded boxes (downward) linked to the node represent the objects used to name the concept. The information contained in the formal context of Table 1 can be derived from the diagram in Figure 1 by applying the following reading rule: An object ‘g’ is described by an attribute ‘m’ if and only if there is an ascending path from the node named by ‘g’ to the node named by ‘m’. For example, paper 1 is described by the attributes ‘browsing,’ ‘software,’ ‘mining,’ and ‘FCA’.

Another important notion of FCA is that of *attribute implication*. For subsets $A, B \subseteq M$, one has $A \rightarrow B$ if $A' \subseteq B'$. Implications obey Armstrong rules, which are valid for functional dependencies in relational databases.²⁰ Moreover, there is a two-way reduction between implications and functional dependencies. A minimal subset of implications from which all the rest can be deduced by means of Armstrong rules is called an implication base. Most well-known bases are Duquenne-Guigues or stem base,²¹ which is cardinality minimal, and proper premise base.¹⁹ Implications can be read from the lattice diagram

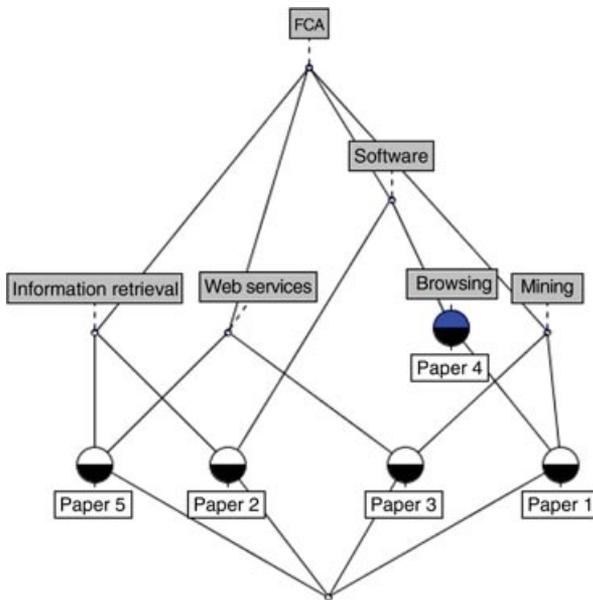


FIGURE 1 | Line diagram corresponding to the context from Table 1.

too, implications with singleton premises of the form $a \rightarrow B$ are easily seen: the concept (a', a'') lies below the concept (B', B'') . The first and one of the most well-known knowledge discovery FCA-based procedure is Attribute Exploration,¹⁹ at each step of which an implication base for the current context representing domain data is generated and offered to a domain expert, who either accepts all implications of the base (in which case the procedure terminates) or gives a counterexample to the implication he finds wrong. Data can be given in a more general way by many-valued contexts (relational tables), in which case they are reduced to binary contexts by means of conceptual scaling.¹⁹ Each type of scaling (nominal, ordinal, interordinal, dichotomic, etc.) is given by a scaling context.

FCA ALGORITHMS AND SCALABILITY

Scalability is a real issue for FCA, as the number of formal concepts can be exponential in the input context and counting them is #P-complete,²² however, all concepts can be constructed with polynomial delay, see an overview of many FCA-based algorithms in Ref 23. The sizes of implication bases, even the size of the minimal (stem or Duquenne-Guigues) implication base can also be exponential, counting the size of the stem base being #P-hard,²⁴ see more results on algorithmic complexity in Refs 25 and 26. That is why, efficiently han-

dling large and complex datasets became a challenge for FCA-based research. Besides designing more efficient practical algorithms, for example, Refs 27–34, the following approaches to reducing complexity are popular:

- Taking interesting small subsets of attributes and objects. Building nested line diagrams for zooming in data by taking subsets of attributes.¹⁹ Here, instead of one large diagram one obtains a smaller (depending on the size of the attribute subset) ‘outer’ diagram, where vertices correspond to ‘inner’ diagrams, where one can zoom in.
- Krajca et al.³⁵ present an algorithm selecting attributes while computing formal concepts. Dias and Vieira³⁶ propose an approach for replacing groups of similar objects by prototypical ones. Ganter and Kuznetsov³⁷ describe how scaled many-valued contexts of FCA may make feature selection easier. Snasel et al.³⁸ use matrix reduction to select attributes. Belohlavek and Vychodil^{39,40} propose an approach for factoring contexts, thus representing the set of original attributes by a smaller set of factors.
- ‘Concept weeding’⁹ by introducing interestingness measures of concepts. The simplest and most natural one is given by constraints on sizes of intents and extents proposed in Refs 22 and 41, which became very popular under the name of *iceberg lattices* because of their use in mining frequent association rules.^{42,43} Ventos and Soldano⁴⁴ show that iceberg lattices obtained by pruning concepts with small extents make a special class of *alpha-Galois lattice*. Jay et al.⁴⁵ used iceberg lattices in combination with *concept stability* for analyzing social communities. Concept stability measures the ratio of extent (intent) subsets that generate the same intent (extent), thus showing concept invariance to noise in data.^{46,41,47} *Concept independence*⁴⁸ is a similar index showing the sparseness of objects and attributes outside the concept extent and intent that have same attributes and objects, respectively. *Concept probability*⁴⁸ is the sample probability of a concept in a randomly generated context with the same number of units. Babin and Kuznetsov⁴⁹ propose probabilistic methods for approximating stability.
- Lazy classification, where generation of numerous classification rules, is avoided.^{50,51}

RELATIONSHIPS OF FCA TO MODELS OF KNOWLEDGE REPRESENTATION AND PROCESSING

FCA and DLs

DLs¹⁴ provide a structured algorithmically decidable language for representing knowledge in a certain domain. DLs are fragments of the first-order logic where decidability is attained due to syntactic constraints, in particular, due to restricted quantification. The set of underlying entities of DLs, so-called concepts (each concept is given as a set of domain objects) is closed w.r.t. concept intersection. Seen from FCA perspective, the quantifier-free fragment of DL is just the lattice of extents of a given context. Because of this relationship between DL and FCA,⁵² two main research directions are natural:

- Enriching FCA by using means of DL languages, for example, theory-driven scaling by combining DLs with attribute exploration.^{53–55}
- Using FCA for improving the potential of DLs in knowledge representation, using attribute exploration for computing the subsumption hierarchy of all conjunctions of a set of DL concepts.⁵⁶ Using FCA for bottom-up construction of DL knowledge bases.⁵⁶ Relational exploration,⁵⁷ where DLs are used to specify FCA attributes, and FCA approaches are used to refine DL knowledge bases. Using complex formulae in a DL language to replace the atomic attributes in FCA.^{25,58}

FCA and Ontology Engineering

Because of their properties, concept lattices are natural representations of domain taxonomies and meronomies with line diagrams being natural visualization tools. These properties were used in the models of ontology engineering,⁵⁹ merging,⁶⁰ and alignment,⁶¹ as the taxonomic ('be more general than') relation makes the backbone of an applied domain ontology. Constructing ontologies with FCA means is often very fruitful in text mining.^{62,63} For those who are used to see taxonomies as tree-like structures, the concept diagram can be easily transformed in this form by cutting (unordered) cycles and introducing concept duplicates. The problem of too numerous concepts in the ontology is solved by merging attributes, selecting most interesting concepts, or clustering the concept lattice (see the previous section on scalability). For example, in Refs 64 and 65, the structure of an epistemic community of researchers

(objects) described by their papers (attributes) is represented by a taxonomy given by the most stable concepts (concepts with the highest stability index) of the respective concept lattice.

Association Rule Mining

Mining association rules is one of the main models in data mining.¹⁶ An association rule is just a statement about conditional sample probability (called confidence) of an event w.r.t. another one, together with the statement of the joint sample probability of the two events (called support), where both events are described in terms of attribute sets. In FCA terms, for two subsets of attributes (called itemsets in data mining) Y_1 and $Y_2 \subseteq M$, the association rule $Y_1 \rightarrow Y_2$ has support $\frac{|(Y_1 \cup Y_2)'|}{|G|}$ and confidence $\frac{|(Y_1 \cup Y_2)'|}{|Y_1'|}$. The rule $Y_1 \rightarrow Y_2$ is called frequent if $\text{supp}(Y_1 \rightarrow Y_2) \geq \text{minsupp}$ for some threshold minsupp . In FCA, association rules were known under the name of partial implications. In the paper by Luxemburger⁶⁶, it was shown that a concise representation of association rules of a context can be given by the edges of the diagram (i.e., by covering relation) of the concept lattice, or even by the spanning tree of the graph of the covering relation, where for each association rule from this concise representation,^{67–69} the premise is given by the minimal generator of the upper concept intent, and the conclusion is given by the intent of the lower concept. The set of all frequent association rules is then represented by the upper part (order filter) of the lattice called iceberg lattice.¹² This representation gives much more efficient algorithms than standard Apriori, see for example, Ref 70.

FCA and Machine Learning

A model of learning from positive and negative examples (making the training set) of a target class w describes the way one constructs a generalized description of (some) positive examples that would cover just a 'small' number of negative examples. Several machine learning methods are naturally described in FCA terms. In the last few decades, several FCA-based learning approaches were proposed. Nguifo and Njiwoua⁷¹ proposed IGLUE, an algorithm combining lattice-based and instance-based learning techniques. Fu et al.⁷² give an experimental comparison of FCA-based classification algorithms, such as GRAND,⁷³ LEGAL,⁷⁴ GALOIS,⁷⁵ RULEARNER,⁷⁶ CIBLE, and CLNN and CLNB.⁷⁷ JSM hypotheses in FCA terms^{22,41,78} are positive intents (i.e., concept intents of the context given by positive examples) not contained in negative example intents,

symmetrically for negative hypotheses. Hypotheses can be used as classifiers for objects from the test set. Ganter and Kuznetsov⁷⁹ give a simple FCA-based description of version spaces and disjunctive version spaces. Kuznetsov²⁴ gives an overview of several learning models in terms of FCA: learning decision trees, learning with graph descriptions and pattern structures in general, and so on. Ricordeau⁸⁰ used FCA to generalize policies in reinforcement learning by grouping similar states using their descriptions. Maddouri⁸¹ proposes a machine learning method for incremental concept formation. Rudolph⁸² proposes to use FCA to design a neural network architecture in case some partial information about the networks desired behavior is already known and can be stated in the form of implications on the feature set. The author first translates implications in a formal context and then translates the context into a three-layered feedforward network, which computes closed sets of attributes. Tsope et al.⁸³ propose classification algorithms, where concept lattices are used to determine the architecture of a neural network. First a join semi-lattice of relevant concepts (w.r.t. some natural constraints) is built, then the join semi-lattice is translated into a topology of the neural network and the initial set of connection weights. Finally, the network is trained.

Belohlavek et al.⁸⁴ describe induction of decision trees in FCA terms. First, the categorical attributes are scaled to logical attributes. Then, a modified version of the NextNeighbor algorithm⁸⁵ is used to build a reduced concept lattice. Decision trees correspond to parts of lattice diagrams. The authors of Ref 86 use Boolean factor analysis³⁹ to transform the attribute space to improve the results of machine learning and, in particular, decision tree induction. Visani et al.⁸⁷ propose an approach of navigating in the concept lattice for supervised classification, and applied it to noisy symbol recognition.

FCA, Biclustering, and Multidimensional Clustering

There are several types of *biclusters* (coclusters) known in the literature⁸⁸: biclusters of equal values, similar values, coherent values, the commonality of them being the existence of inclusion-maximal set of objects described by inclusion-maximal set of attributes with some special pattern of behavior. Clustering objects based on sets of attributes taking similar values dates back to the work⁸⁹ and was called biclustering in Ref 18. Attention to biclustering approaches started to grow from the beginning of 2000s with the growth of the need to analyze similarities in

gene expression data⁹⁰ and design of recommender systems.⁹¹ Given an object-attribute numerical data table (many-valued context in terms of FCA), the goal of biclustering is to group together some objects having similar values of some attributes. For example, in gene expression data, it is known that genes (objects) may share a common behavior for a subset of biological situations (attributes) only: one should accordingly produce local patterns to characterize biological processes, the latter should possibly overlap, as a gene may be involved in several processes. The same remark applies for recommender systems, where one is interested in local patterns characterizing groups of users that strongly share almost the same tastes for a subset of items.^{91,92}

A bicluster in a binary object-attribute data table is a pair (A, B) consisting of an inclusion-maximal set of objects A and an inclusion-maximal set of attributes B such that almost all objects from A have almost all attributes from B and vice versa. Of course, formal concepts can be considered as 'rigid' biclusters where all objects have all attributes and vice versa. Hence, it is not surprising that some bicluster definitions coming from practice⁹³ are just definitions of a formal concept.⁹⁴

A bicluster of similar values in a numerical object-attribute data-table is usually defined⁹⁵⁻⁹⁷ as a pair consisting of an inclusion-maximal set of objects and an inclusion-maximal set of attributes having similar values for the objects. Such a pair can be represented as an inclusion-maximal rectangle in the numerical table, modulo rows and columns permutations. In Ref 94, it was shown that biclusters of similar values correspond to triconcepts of a triadic context,⁹⁸ where the third dimension is given by a scale that represents numerical attribute values by binary attributes. This fact can be generalized to n -dimensional case, where n -dimensional clusters of similar values in n -dimensional data are represented by $n + 1$ -dimensional concepts. This reduction allows one to use standard definitions and algorithms from *multidimensional concept analysis*^{97,99} for computing multidimensional clusters.

FCA and RST

RST, introduced in Ref 17 is a mathematical theory to deal with uncertainty and imperfect knowledge. In RST, the data are given by a set of objects (called universe U) described by many-valued attributes. Objects having same attribute values are called indiscernible (similar) in view of the available information about them. By modeling indiscernibility as an equivalence relation, $E \subseteq U \times U$ one can partition a finite

universe of objects into pairwise disjoint subsets denoted by U/E . The partition provides a granulated view of the universe. An equivalence class is considered as a whole, instead of many individuals. For an object $x \in U$, the equivalence class containing x is given by $[x]_E = \{y \in U | xEy\}$. The empty set, equivalence classes, and unions of equivalence classes form a system of definable subsets under discernibility. It is a σ -algebra $\sigma(U/E) \subseteq 2^U$ with basis U/E . All subsets not in the system are consequently approximated through definable sets. Various definitions of rough set approximations have been proposed including the subsystem-based, granule-based, and element-based formulation.¹⁰⁰ In this section, we introduce the subsystem-based formulation. In an approximation space $\text{apr} = (U, E)$, a pair of approximation operators $(\underline{\cdot}, \overline{\cdot}) : 2^U \rightarrow 2^U$ is defined. The lower approximation

$$\underline{A} = \cup \{X | X \in \sigma(U/E), X \subseteq A\}$$

is the greatest definable set contained in A , and the upper approximation

$$\overline{A} = \cup \{X | X \in \sigma(U/E), A \subseteq X\}$$

is the union of all definable sets not disjoint with A . Any set of all indiscernible (similar) objects is called an elementary set (neighborhood) and forms a basic granule (atom) of knowledge about the universe. Any union of elementary sets is a crisp (precise) set, otherwise the set is rough (imprecise, vague). Each rough set has boundary-line cases, that is, objects that cannot be classified with certainty as either members of the set or its complement. Crisp sets have no boundary-line elements at all. Boundary-line cases cannot be properly classified by employing the available knowledge.

Vague concepts (in contrast to precise concepts) cannot be characterized in terms of information about their elements. In other words, given an arbitrary subset $A \subseteq U$ of the universe of objects, it may not be the extent of a formal concept. This subset can be seen as an undefinable set of objects and can be approximated by definable sets of objects, namely extents of formal concepts. Any vague concept is replaced by a pair of precise concepts, called the lower and upper approximation of the vague concept. The lower approximation consists of all objects that surely belong to the concept and the upper approximation contains all objects that possibly belong to the concept. The difference between the lower and upper approximation constitutes the boundary region of the vague concept.

Many efforts have been made to combine FCA and RST.¹⁰⁰ This combination is typically referred to as Rough Formal Concept Analysis (RFCA), see

Refs 101 and 102. In RFCA, $\sigma(U/E)$ is replaced by lattice L and definable sets of objects by extents of formal concepts. The extents of the resulting two concepts are the approximations of A . For a set of objects $A \subseteq U$, its lower approximation is defined as

$$\underline{l}(A) = (\cup \{X | (X, Y) \in L, X \subseteq A\})'$$

and its upper approximation is defined by

$$\overline{l}(A) = (\cup \{X | (X, Y) \in L, A \subseteq X\})'$$

The lower approximation of a set of objects A is the extent of $(\underline{l}(A), (\underline{l}(A))')$ and the upper approximation is the extent of the formal concept $(\overline{l}(A), (\overline{l}(A))')$. The concept $(\underline{l}(A), (\underline{l}(A))')$ is the supremum of concepts where extents are subsets of A and $(\overline{l}(A), (\overline{l}(A))')$ is the infimum of those concepts where extents are superset of A . For the other RST formulations, a similar description can be given for the combination of FCA and RST. Ganter et al.¹⁰³ use a generalization of the indiscernibility relation, which is considered as a quasi-order of equivalence given by $g \leq h \Leftrightarrow g' \subseteq h'$ for $g, h \in G$. This generalization allows for defining version spaces and hypotheses in terms of RST. Ganter and Kuznetsov³⁷ generalized the classical rough set approach by replacing lower and upper approximations with arbitrary kernel and closure operators respectively. Lattices of rough set abstractions were described as P-products. Meschke¹⁰⁴ further investigates the role of robust elements, the possible existence of suitable negation operators, and the structure of corresponding lattices, restricts the view for large contexts to a subcontext without losing implicational knowledge about the selected objects and attributes. Ganter and Meschke¹⁰⁵ use a slightly different approach based on FCA to mine the Infobright dataset containing so-called rough tables where objects are combined together in data packs. Their approach makes FCA useful for analyzing extremely large data.

EXTENSIONS OF FCA

In practice, data are often more complex than those given by formal contexts. A first approach to handle complex data in FCA was conceptual scaling,¹⁰⁶ where complex data are turned into binary contexts by using scales. Since the beginning of FCA, many authors attempted to generalize its definitions to more complex data representations, such as graphs,^{46,107–109} intervals,^{46,94,107} logical formulas,^{110–112} and so on. Pattern structures⁷⁸ give

a general approach for extending FCA to objects with partially ordered descriptions. Several approaches allow for analyzing arbitrary relations between objects: *power context families* (PCFs),¹¹³ *relational concept analysis* (RCA),^{114,115} and *logical concept analysis* (LCA).¹¹⁶ *Triadic concept analysis* was introduced by Lehmann and Wille^{98,117} to analyze three-dimensional data. Fuzzy FCA^{118–120} and rough FCA¹⁰⁵ were developed to work with uncertain data and approximations.

FCA-Based Models of Knowledge Processing with Relations on Objects

PCFs were proposed by Wille¹¹³ as a method to transform conceptual graphs¹⁵ into a family of formal contexts. A PCF $K = (K_n)_{n \in \mathbb{N}^0}$ is a family of formal contexts $K_k = (G_k, M_k, I_k)$ such that $G_k \subseteq (G_0)^k$ for $k = 1, 2, \dots$. The formal contexts K_k with $k \geq 1$ are called relational contexts. Prediger¹²¹ presented a logic for defining concept graphs as syntactical constructs with a semantics in PCFs, introduced relational scaling to transform a many-valued context (relational table) into a PCF. The concept graphs of a PCF were shown to form a lattice, which can be described as a subdirect product of specific intervals of the concept lattices of the PCF. Groh and Eklund¹²² discussed various algorithms for creating relational PCFs from conceptual graphs. Dau and Klinger^{123,124} developed a logical theory based on concept graphs and subsumption relation on them. Wille¹²⁵ proposed ‘Boolean Judgment Logic’ and the ‘negating inversion’ to define the negation of formal judgments (defined as valid propositions) represented by concept graphs of PCFs. Wille¹²⁶ showed that the implicational theory of implicational concept graphs of PCFs is equivalent to the theory of attribute implications of formal contexts. In other words, implications can be seen as special judgments, namely as specific concept graphs of PCFs. Rudolph⁵⁷ used binary PCFs to define an extensional semantic for concept descriptions expressed in $\mathcal{FL}\mathcal{E}$ DL.

RCA was proposed in Ref 114 to study relations between objects of the context (given by separate tables) by FCA means, introducing new attributes representing these relations. An RCA-based approach is implemented in Galicia platform.¹¹⁵ The main applications of RCA are in software refactoring¹²⁷ and ontology engineering.^{54,115}

Complex Object Descriptions

Pattern structures⁷⁸ generalize some models extending FCA definitions to partially ordered descrip-

tions, such as graphs, numeric intervals, and logical formula. Let G be a set of objects, let (D, Π) be a meet-semi-lattice (of potential object descriptions) and let $\delta: G \rightarrow D$ be a mapping. Then, $(G, \underline{D}, \delta)$ with $\underline{D} = (D, \Pi)$ is called a pattern structure, and the set $\delta(G) := \{\delta(g) | g \in G\}$ generates a complete subsemi-lattice (D_δ, Π) of (D, Π) . Thus, each $X \subseteq \delta(G)$ has an infimum ΠX in (D, Π) and (D_δ, Π) is the set of these infima. Semilattice (D_δ, Π) has both lower and upper bounds, respectively 0 and 1. Elements of D are called patterns and are ordered by subsumption relation \sqsubseteq : given $c, d \in D$ one has $c \sqsubseteq d \Leftrightarrow c \Pi d = c$. A pattern structure $(G, \underline{D}, \delta)$ gives rise to the following derivation operators $(\cdot)^\square$:

$$A^\square = \prod_{g \in A} \delta(g) \text{ for } A \subseteq G, g \in A$$

$$d^\square = \{g \in G | d \sqsubseteq \delta(g)\} \text{ for } d \in D$$

These operators form a Galois connection between the powerset of G and (D, \sqsubseteq) . Pattern concepts of $(G, \underline{D}, \delta)$ are pairs of the form (A, d) $A \subseteq G, d \in D$ such that $A^\square = d$ and $A = d^\square$. For a pattern concept (A, d) , the component d is called a pattern intent, it is a description of the set of all objects in A , called pattern extent. If (A, d) is a pattern concept, adding any element to A changes d through $(\cdot)^\square$ and equivalently taking $e \sqsupseteq d$ changes A . Like in case of formal contexts, for a pattern structure $(G, \underline{D}, \delta)$, a pattern $d \in D$ is called closed if $d^\square = d$, and a set of objects $A \subseteq G$ is called closed if $A^{\square\square} = A$. Obviously, pattern extents and intents are closed. A seeming limitation of pattern structures to descriptions in semi-lattice is easily waived: one can start from descriptions in an arbitrary order (P, \leq) taking as (D, Π) the lattice of all order ideals of (P, \leq) .

Since 1980s, pattern structures on sets of graphs^{107,108,128} and vectors of intervals^{94,107} were used for designing models of machine learning and data mining with applications in bioinformatics. Kuznetsov et al.¹²⁹ summarize applications of pattern structures in the study of biological activities of chemical compounds and show the relationship to methods of graph mining based on closed graphs. Kuznetsov¹²⁹ further discusses the analysis of complex data with pattern structures, stressing that pattern structures are computationally more efficient and provide better visualization of results than FCA methods based on scaling. Kaytoue et al.⁹⁴ applied pattern structures to mine gene expression data given as a many-valued context and therefore had to be scaled

to a binary context. The authors found the interval pattern structures offering a more concise representation, better scalability, and better readability of results than interordinal scaling. Assaghir et al.¹³⁰ did an exploratory study on induction of decision trees from numerical data by using interval pattern structures, proposing an algorithm for generating decision trees from minimal hypotheses. Minimal hypotheses correspond to optimal splits in decision trees. Pfaltz¹³¹ extends FCA to deal with numerical values, by deriving logical implications containing ordinal inequalities as atoms, such as $y \leq 11$. This extension employs the fact that orderings are antimatroid closure spaces. Valverde-Albacete and Peláez-Moreno¹³² introduced a generalization of FCA, where values of the incidence matrices are elements of a semi-ring, this model was used by Gonzalez Calabozo et al.¹³³ in the analysis of gene expression data.

A similar approach to pattern structures is that of LCA, which started as an extension of FCA to descriptions based on first order formulas.^{110,116} Ferré and Ridoux¹¹⁰ give an introduction to logical information systems (LIS) based on LCA. LIS are mostly suited for information retrieval and navigation in taxonomies based on complex descriptions, like DNA sequences and collection of images. An approach similar to that of LCA was proposed in Refs 111 and 112, with existentially quantified conjunctions of first-order literals as object descriptions. Another similar approach was developed in Ref 134, where a combination of FCA with first-order logic was used to learn ontological rules. Descriptions given by sequences of states are natural models of time. Temporal concept analysis (TCA) is based on FCA and addresses the problem of conceptually representing discrete temporal phenomena.^{135,136} Conceptual time system is given by situations and states describing the observations of objects at several points of time.

Conceptual Knowledge Processing with Background Knowledge

Many authors emphasized the need for taking into account background knowledge in FCA. In Ref 137, the author describes how attribute exploration (interactive generation of implicational dependencies of a domain) can be attained when not only data in the context form is given, but some knowledge in the form of implications. Belohlavek and Vychodil¹³⁸ present an approach for modeling background knowledge that represents user's priorities regarding attributes and their relative importance. Only those concepts that are compatible with user's priorities are considered relevant and are extracted from the data. In Ref 139,

FCA is used in combination with a tag context to formally incorporate important kinds of background knowledge. The results are generalized contingency structures and tagged contingency structures, which can be used for data summarization in epidemiology. In Ref 138, a context is complemented with a probability logic, which makes use of statistical and propositional probability inference. The authors introduce a new type of concept called 'previously unknown and potentially useful' and formalize KDD as a process to find such concepts. In its classical form, FCA considers attributes as a nonordered set. When attributes of the context are partially ordered to form taxonomy, conceptual scaling allows the taxonomy to be taken into account by producing a context completed with all attributes deduced from the taxonomy. Cellier et al.¹⁴¹ propose an algorithm for learning concept-based rules under existing taxonomy.

Triadic and n -Adic FCA

Triadic concept analysis (TCA) was proposed by Lehmann and Wille¹¹⁷ and Wille⁹⁸ to extend FCA to three-way data. A triadic context is defined as a quadruple (G, M, B, Y) where G (objects), M (attributes), and B (conditions) are sets and Y is a ternary relation between G , M , and B : $Y \subseteq G \times M \times B$. A triple $(g, m, b) \in Y$ means object g has attribute m under condition b . In the last decade, the analysis of triadic and n -adic data is gaining more interest because of the increasing popularity of social resource sharing systems like folksonomies, where users can assign tags to resources. A triadic concept is a triple (A_1, A_2, A_3) with $A_1 \subseteq G$ (extent), $A_2 \subseteq M$ (intent), and $A_3 \subseteq B$ (modus) such that for $X_1 \subseteq G, X_2 \subseteq M$ and $X_3 \subseteq B$ with $X_1 \times X_2 \times X_3 \subseteq Y$; the containments $A_1 \subseteq X_1, A_2 \subseteq X_2$ and $A_3 \subseteq X_3$ always imply $(A_1, A_2, A_3) = (X_1, X_2, X_3)$. Triadic concepts consist of three sets, namely objects, attributes, and conditions under which objects may possess certain attributes. Stumme¹⁴² discusses how traditional line diagrams of standard dyadic concept lattices can be used for exploring triadic data. The author showcases how it can be used for navigating through the Federal Office for Information Security IT baseline Protection Manual. Jäschke et al. introduced the TRIAS algorithm for mining all frequent closed itemsets from three-dimensional data and applied it to the popular bibsonomy (users-tags-papers) dataset in Ref 143. TRIAS basically relies on extraction of closed two sets from two binary relations. Belohlavek et al.⁴⁰ extended TCA to deal with fuzzy attributes instead of traditional Boolean data tables. Alternatives to TRIAS include the algorithms RSM

and CubeMiner, which were introduced in Ref 144 for mining closed three sets from ternary relation. In contrast to RSM and TRIAS, CubeMiner works directly on the ternary relation generalizing the closeness checking of Besson et al.¹⁴⁵ to the triadic case. Another approach was proposed in Ref 146 for mining temporal gene expression data. The authors extracted maximal triclusters satisfying certain homogeneity criteria from these gene expression tables. Mining triadic implications from Boolean triadic contexts was discussed in Refs 147 and 148. Ref 149 generalized these triadic implications to a fuzzy setting. Missaoui and Kwuida¹⁵⁰ discuss the different types of methods for mining triadic association rules from ternary data. Ignatov et al.¹⁵¹ proposed concept-based triclustering as an extension of concept-based biclustering. Tentative triclusters are taken from all possible combinations of inclusion-maximal closed sets of objects, attributes, and modi. Kaytoue et al.⁹⁴ apply the methods of TCA for analyzing gene expression data given by a many-valued numerical context, which is interordinally scaled to triadic (binary) context, where triconcepts correspond to maximal biclusters of numerical values in the initial many-valued numerical context. Several authors generalized bi- and triclustering to the n -dimensional clustering. Trabelsi et al.¹⁵² propose novel methods for improving scalability of frequent triconcepts mining from Folksonomies. Voutsadakis⁹⁹ extended triadic concept analysis to n -dimensional case.

Fuzzy FCA

Fuzzy concept lattices were first introduced in Ref 153; however, their theory had its limitations as they did not use residuated lattices and merely introduced some basic notions. Belohlavek and Polandt further developed the basic notions of FCA with fuzzy attributes independently, so there are two seminal works¹⁵⁴ and ¹¹⁸ on this topic. The paper¹⁵⁴ focuses on studying similarity in fuzzy concept lattices rather than introducing basic notions. The structure of truth degrees L endowed by logical connectives is represented by a complete residuated lattice $L = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1 \rangle$ such that:

- $\langle L, \wedge, \vee, 0, 1 \rangle$ is a complete lattice with 0 and 1 being the least and greatest element of L
- $(L, \otimes, 1)$ is a commutative monoid, that is, \otimes is commutative, associative, and $a \otimes 1 = 1 \otimes a = a$ for each $a \in L$
- and L satisfy the adjointness property $a \otimes b \leq c \Leftrightarrow a \leq b \rightarrow c$; for each $a, b, c \in L$

- and \rightarrow are (truth functions of) ‘fuzzy conjunction’ and ‘fuzzy implication’
- By L^U (or L^U), we denote the collection of all fuzzy sets in a universe U , that is, mappings A of U to L . For $A \in L^U$ and $a \in L$, a set ${}^a A \rightarrow \{u \in U \mid A(u) \geq a\}$ is called an a -cut of A .

In Refs 155 and 156, the author studies fuzzy closure operators, which are, along with fuzzy Galois connections, the basic structures behind fuzzy concept lattices. These results have been later used many times in different papers on the topic. Belohlavek and Vychodil¹⁵⁷ give an overview of recent developments concerning attribute implications in a fuzzy setting.

The following definition of a formal fuzzy concept lattice was introduced in^{119,158}. A fuzzy formal context (L -context) is a triple $\langle X, Y, I \rangle$, where X is the set of objects, Y is the set of attributes and $I : X \times Y \rightarrow L$ is a fuzzy relation (L -relation) between X and Y . A truth degree $I(x, y) \in L$ is assigned to each pair (x, y) , where $x \in X, y \in Y$ and L is the set of values of a complete residuated lattice L . The element $I(x, y)$ is interpreted as the degree to which attribute y applies to object x . Fuzzy sets $A \in L^X$ and $B \in L^Y$ are mapped to fuzzy sets $A^\uparrow \in L^Y, B^\downarrow \in L^X$ according to Ref 119.

$$A^\uparrow(y) = \bigwedge_{x \in X} (A(x) \rightarrow I(x, y))$$

$$B^\downarrow(x) = \bigwedge_{y \in Y} (B(y) \rightarrow I(x, y))$$

for $y \in Y$ and $x \in X$. A formal fuzzy concept $\langle A, B \rangle$ consists of a fuzzy set A of objects (extent of the concept) and a fuzzy set B of attributes (intent of the concept) such that $A^\uparrow = B$ and $B^\downarrow = A$. The set of all formal fuzzy concepts is $\mathcal{B}(X, Y, I) = \{ \langle A, B \rangle \mid A^\uparrow = B, B^\downarrow = A \}$. We also define \leq , which models the subconcept–superconcept hierarchy in $\mathcal{B}(X, Y, I)$: $\langle A_1, B_1 \rangle \leq \langle A_2, B_2 \rangle \Leftrightarrow A_1 \subseteq A_2 \Leftrightarrow B_2 \subseteq B_1$ for $\langle A_1, B_1 \rangle, \langle A_2, B_2 \rangle \in \mathcal{B}(X, Y, I)$. $(\mathcal{B}(X, Y, I), \leq)$, that is, $\mathcal{B}(X, Y, I)$ equipped with relation \leq is a complete lattice w.r.t. \leq . Other related formalisms, such as profuzzy concepts,¹⁵⁹ multiadjoint concept lattices,¹⁶⁰ and their relations to fuzzy setting are discussed, for example, in Ref 161.

CONCLUSIONS

During the last three decades, FCA became a well-known formalism in data analysis and knowledge discovery because of its usefulness in such important KDD domains as ontology engineering, association rule mining, machine learning, as well as relation to other established theories for representing knowledge

processing, like DLs, conceptual graphs, and rough sets.

FCA has been used effectively in many domains for gaining actionable intelligence from large amounts of information: web mining (e.g., Ref 162), text mining, bioinformatics, medicine (e.g., cancer treatment) and chemistry (e.g., predicting biological activity of chemical compounds; Ref 163), information retrieval (e.g., Ref 164), ontology engineering, and merging. There are numerous FCA-based knowledge processing models based on extensions to complex, uncertain, and many-dimensional data.

Currently, there are several well-known open source FCA-based software tools, such as ConExp,¹⁶⁵ Conexp-clj (<http://daniel.kxpq.de/math/conexp-clj/>), Galicia,¹⁶⁶ Tockit (Framework for Conceptual Knowledge Processing <http://www.tockit.org>), ToscanaJ,¹⁶⁷ FCAStone,¹⁶⁸ Lattice Miner,¹⁶⁹ OpenFCA,¹⁷⁰ and FCART.¹⁷¹ These tools are mostly Java-based (except for FCART), cross-platform, rather easy to use, and they do not need to be installed. Universal integrated environment for knowledge discovery based on FCA was partially created in the Tockit project and in FCART.

REFERENCES

1. Wille R. Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival I, ed. *Ordered Sets*. Dordrecht, The Netherlands/Boston, MA: Reidel; 1982, 445–470.
2. Arnauld A, Nicole P. *La logique ou l'art de penser*; 1683.
3. Birkhoff G. *Lattice Theory*. 3rd ed. Providence, RI: American Mathematical Society Coll. Publ. 25; 1973.
4. Ore O. Galois Connexion. *Trans Am Math Soc* 1944, 55:493–513.
5. Barbut M, Monjardet B. *Ordre et classification, algèbre et combinatoire*. Paris: Hachette; 1970.
6. Hereth J. *Relational Scaling and Databases ICCS*. Heidelberg, Germany: Springer; 2002, 62–76.
7. Carpineto C, Romano G. *Concept Data Analysis: Theory and Applications*. Chichester, West Sussex, England: John Wiley & Sons; 2004.
8. Poelmans J, Elzinga P, Viaene S, Dedene G, Kuznetsov S. Text mining scientific papers: a survey on FCA-based information retrieval research. *Industrial Conference on Data Mining*. Berlin, Germany; 2012.
9. Priss U. Formal concept analysis in information science. In: Blaise C, ed. *Annual Review of Information Science and Technology. ASIST*. Chichester, West Sussex: John Wiley & Sons; 2006, 40.
10. Tilley T. Tool support for FCA. In: Eklund P, ed. *Proceedings of the 2nd ICFCA, Lect Notes Artif Intell 2961*; February 23–26; Sydney, Australia. Berlin, Heidelberg: Springer; 2004, 104–111.
11. Tilley T, Cole R, Becker P, Eklund P. A survey of formal concept analysis support for software engineering activities. In: Ganter B, Godin R, eds. *Formal Concept Analysis, LNAI 3626*; February 14–18; Lens, France. Berlin, Heidelberg: Springer; 2005, 250–271.
12. Lakhil L, Stumme G. Efficient mining of association rules based on formal concept analysis. In: Ganter B, Godin R, eds. *Formal Concept Analysis, LNAI 3626*; February 14–18; Lens, France. Berlin, Heidelberg: Springer; 2005, 180–195.
13. Doerfel S, Jäschke R, Stumme G. Publication analysis of the FCA community. *ICFCA, Lect Notes Comput Sci 7278*; May 06–10; Leuven, Belgium. Berlin, Heidelberg: Springer; 2012, 77–95.
14. Baader F, Calvanese D, McGuinness D, Nardi D, Patel-Schneider PF, eds. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press; 2003.
15. Sowa JF. *Conceptual Structures: Information Processing in Mind and Machine*. Reading, MA: Addison-Wesley; 1984.
16. Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. *ACM SIGMOD Rec* 1993, 22:207–216.
17. Pawlak Z. Rough sets. *Int J Comput Inf Sci* 1982, 11:341–356.
18. Mirkin B. *Mathematical Classification and Clustering*. Boston, MA/Dordrecht, The Netherlands: Kluwer Academic Publisher; 1996.
19. Ganter B, Wille R. *Formal Concept Analysis. Mathematical Foundations*. Berlin, Heidelberg: Springer; 1999.
20. Maier D. *The Theory of Relational Databases*. Rockville: Computer Science Press; 1983.
21. Guigues J-L, Duquenne V. Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Math Sci Hum* 1986, 95:5–18.
22. Kuznetsov SO. Interpretation on graphs and complexity characteristics of a search for specific patterns. *Autom Document Math Ling* 1989, 24:27–35.
23. Kuznetsov SO, Obiedkov SA. Comparing performance of algorithms for generating concept lattices. *J Exp Theor Artif Intell* 2002, 14:189–216.
24. Kuznetsov SO. (2004) Machine learning and formal concept analysis. In: Eklund P, ed. *Proceedings of*

- the 2nd International Conference on Formal Concept Analysis, ICFCA'04, Lect Notes Artif Intell 2961*; February 23–26; Sydney, Australia. Berlin, Heidelberg Springer; 2004, 287–312.
25. Distel F. An approach to exploring description logic knowledge bases. In: Kwuida L, Sertkaya B, eds. *Proceedings of the 8th International Conference on Formal Concept Analysis, (ICFCA 2010), Vol. 5986 of Lecture Notes in Artificial Intelligence*; March 15–18; Agadir, Morocco. Berlin, Heidelberg: Springer; 2010, 209–224.
 26. Babin MA, Kuznetsov SO. Computing premises of minimal cover of functional dependencies is intractable. *Discr Appl Math*; 2013, 161:742–749.
 27. Ganter B. *Two Basic Algorithms in Concept Analysis, FB4-Preprint No. 831*. TH-Darmstadt; 1984.
 28. Bordat J-P. Calcul pratique du treillis de Galois d'une correspondance. *Math Sci Hum* 1986, 96:31–47.
 29. Kuznetsov SO. A fast algorithm for computing all intersections of objects in a finite semi-lattice. *Autom Document Math Ling* 1993, 27:11–21.
 30. Van der Merwe D, Obiedkov SA, Kourie DG. AddIntent: a new incremental algorithm for constructing concept lattices. *ICFCA 2004*, 2964:372–385
 31. Berry A, Bordat J-P, Sigayret A. A local approach to concept generation. *Ann Math Artif Intell* 2007, 49:117–136.
 32. Valtchev P, Missaoui R, Godin R. A framework for incremental generation of closed itemsets. *Discr Appl Math* 2008, 156:924–949.
 33. Krajca P, Vychodil V. Distributed algorithm for computing formal concepts using map-reduce framework. *IDA*; 2009, 333–344
 34. Outrata J, Vychodil V. Fast algorithm for computing fixpoints of Galois connections induced by object-attribute relational data. *Inf Sci* 2012, 185:114–127.
 35. Krajca P, Outrata J, Vychodil V. Computing formal concepts by attribute sorting. *Fundam Inform* 2012, 115:395–417.
 36. Dias SM, Vieira N. Reducing the size of concept lattices: the JBOS approach. *CLA 2010*, 2010:80–91.
 37. Ganter B, Kuznetsov SO. *Scale Coarsening as Feature Selection*. In: Medina R, Obiedkov SA, eds. *ICFCA, Lect Notes Artif Intell 4933*; February 25–28; Montreal, Canada. Berlin, Heidelberg: Springer; 2008, 217–228.
 38. Snasel V, Polovincak M, Abdulla HMD, Horak Z. On concept lattices and implication bases from reduced contexts. *ICCS Suppl*; 2008, 83–90.
 39. Belohlavek R. and Vychodil V. *Factorizing Three-Way Binary Data with Triadic Formal Concepts*. KES'10. Cardiff, UK/Berlin, Heidelberg: Springer; 2010, 471–480.
 40. Belohlavek R, Vychodil V. Discovery of optimal factors in binary data via a novel method of matrix decomposition. *J Comput Syst Sci* 2010, 76:3–20.
 41. Kuznetsov SO. Mathematical aspects of concept analysis. *J Math Sci* 1996, 80:1654–1698.
 42. Pasquier N, Bastide Y, Taouil R, Lakhal L. Efficient mining of association rules using closed itemset lattices. *Inf Syst* 1999, 24:25–46.
 43. Bastide Y, Pasquier N, Taouil R, Stumme G, Lakhal L. Mining minimal non-redundant association rules using frequent closed itemsets. *Comput Logic* 2000, 2000:972–986.
 44. Ventos V, Soldano H. Alpha galois lattices: an overview. In: Ganter B, Godin R, eds., *ICFCA, Lect Notes Artif Intell 3403*; February 14–18; Lens, France. Berlin, Heidelberg: Springer; 2005, 299–314.
 45. Jay N, Kohler F, Napoli A. Analysis of social communities with iceberg and stability-based concept lattices. In: Medina R, Obiedkov SA, eds. *ICFCA, Lect Notes Artif Intell 4933*; February 25–28; Montreal, Canada. Berlin, Heidelberg: Springer; 2008, 258–272.
 46. Kuznetsov SO. Stability as an estimate of the degree of substantiation of hypotheses derived on the basis of operational similarity. *Autom Document Math Ling* 1990, 12:21–29.
 47. Kuznetsov SO. On stability of a formal concept. *Ann Math Artif Intell* 2007, 49:101–115.
 48. Klimushkin M, Obiedkov SA, Roth C. *Approaches to the selection of relevant concepts in the case of noisy data*. *ICFCA*; March 15–18; Agadir, Morocco. Berlin, Heidelberg: Springer; 2010, 255–266.
 49. Babin MA, Kuznetsov SO. Approximating concept stability. *Proceedings of the International Conference on Formal Concept Analysis (ICFCA 2012), Lect Notes Artif Intell 7278*; 2012, 7–15.
 50. Kuznetsov SO. Fitting pattern structures to Knowledge Discovery in Big Data. In: Cellier P, Distel F, Ganter B, eds. *ICFCA 2013, Lect Notes Artif Intell 7880*. Berlin, Heidelberg: Springer; 2013, 254–266.
 51. Kuznetsov SO, Revenko AV. Finding errors in data tables: an FCA-based approach. *Ann Math Artif Intell* 2013.
 52. Sertkaya B. A survey on how description logic ontologies benefit from FCA. *Proceedings of CLA*; 2010, 2–21.
 53. Prediger S, Wille R. The lattice of concept graphs of a relationally scaled context. *Proc ICCS 1999, LNAI 1640*; July 12–15; Blacksburg, VA. Berlin, Heidelberg: Springer; 1999, 401–414.
 54. Rouane-Hacene M, Huchard M, Napoli A, Valtchev P. A proposal for combining formal concept analysis and description logics for mining relational data. *ICFCA*; February 12–16; Clermont-Ferrand, France. Berlin, Heidelberg: Springer; 2007, 51–65.

55. Baader F, Ganter B, Sertkaya B, Sattler U. Completing description logic knowledge bases using formal concept analysis. In: Eklund P, eds. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2007)*; 2007, 230–235.
56. Baader F, Sertkaya B. Applying formal concept analysis to description logics. In: Eklund P, eds. *Proceedings of the 2nd International Conference on Formal Concept Analysis (ICFCA 2004)*, Vol. 2961 of LNCIS. Sydney, Australia: Springer; 2004, 261–286.
57. Rudolph S. Exploring relational structures via FLE. In: Wolff KE, Pfeiffer HD, Delugach HS, eds. *ICCS, LNAI 3127*; July 19–23; Huntsville, AL. Berlin, Heidelberg: Springer; 2004, 196–212.
58. Baader F, Distel F. Exploring finite models in the description logic ELgfp. In: Ferre S, Rudolph S, eds. *Proceeding of the 7th International Conference on Formal Concept Analysis (ICFCA 2009)*, LNAI 5548; 2009, 146–161.
59. Maio CD, Fenza G, Gaeta M, Loia V, Orciuoli F, Senatore S. RSS-based e-learning recommendations exploiting fuzzy FCA for knowledge modeling. *Appl Soft Comput Elsevier* 2012, 12:113–124.
60. Stumme G, Maedche A. FCA-Merge: bottom-up merging of ontologies. *Proceeding of the IJCAI-2001* 2001, 17:225–234.
61. Euzenat J, Valtchev P. Similarity-based ontology alignment in OWL-lite. *Proceedings of 16th European Conference on Artificial Intelligence (ECAI)*; 2004, 333–337.
62. Hotho A, Staab S, Stumme G. Ontologies improve text document clustering. *Proceedings of the IEEE International Conference on Data Mining (ICDM)*; 2003, 541–544.
63. Cimiano A, Hotho A., Staab S. Learning concept hierarchies from text corpora using formal concept analysis. *J Artif Intell Res* 2005, 24:305–339.
64. Kuznetsov SO, Obiedkov SA, Roth C. Reducing the representation complexity of lattice-based taxonomies. In: Priss U, Polovina S, Hill R, eds. *Proceedings of the ICCS 2007*, LNAI 4604; July 22–27; Sheffield, UK. Berlin, Heidelberg: Springer; 2007, 241–254.
65. Roth C, Obiedkov S, Kourie DG. On succinct representation of knowledge community taxonomies with formal concept analysis. *Int J Found Comput Sci* 2008, 19:383–404.
66. Luxenburger M. Implications partielles dans un contexte. *Math Inf Sci Hum* 1991, 29:35–55.
67. Ben Yahia S, Hamrouni T, Nguifo EM. Frequent closed itemset based algorithms: a thorough structural and analytical survey. *ACM SIGKDD Expl Newslett* 2006, 8:93–104.
68. Boulicaut J-F. Condensed representations for data mining. In: Wang J, ed. *Encyclopedia of Data Warehousing and Mining*. Hershey, PA: Idea Group Reference; 2005, 207–211.
69. Gasmi G, Yahia SB, Nguifo EM, Slimani Y. IGB: a new informative generic base of association rules. *PAKDD*; 2005, 81–90.
70. Szathmary L, Napoli A, Kuznetsov SO. ZART: a multifunctional itemset mining algorithm. *CLA*; 2007.
71. Nguifo ME, Njiwoua P. IGLUE: A lattice-based constructive induction system. *Intell Data Anal* 2001, 5:73–91.
72. Fu H, Fu H, Njiwoua P, Nguifo EM. A comparative study of FCA-based supervised classification algorithms. *ICFCA 2004*, 2004:313–320.
73. Oosthuizen G. *The application of concept lattices to machine learning*. Technical Report CSTR 94/01. Pretoria, South Africa: Department of Computer Science, University of Pretoria; 1994.
74. Liquiere M, Mephu Nguifo E. *LEGAL: LEarning with GALois Lattice*. In: *Actes des Journees Francaises sur l'Apprentissage (JFA)*. Lannion, France; 1990, 93–113.
75. Carpineto C, Romano G. Galois: an order-theoretic approach to conceptual clustering. *Proceedings of 10th International Conference on Machine Learning (ICML'93)*. Amherst, MA: Morgan Kaufmann; 1993, 33–40.
76. Sahami M. Learning classification rules using lattices. In: Crete H, Lavrac GN, Wrobel S, eds. *Proceedings of ECML'95*; Heraklion, Crete, Greece. Berlin, Heidelberg: Springer 1995, 343–346.
77. Xie Z, Hsu W, Liu Z, Lee M. Concept lattice based composite classifiers for high predictability. *J Exp Theor Artif Intell* 2002, 14:143–156.
78. Ganter B, Kuznetsov SO. Pattern structures and their projections. In: Stumme G, Delugach H, eds. *Proceedings of 9th International Conference on Conceptual Structures, ICCS'01*. LNAI, 2120; July 30–August 3, 2001; Stanford University, CA. Berlin, Heidelberg: Springer 2001, 129–142.
79. Ganter B, Kuznetsov SO. Hypotheses and version spaces. In: de Moor A, Lex W, Ganter B, eds. *ICCS. LNAI 2746*. July 21–25; Dresden, Germany, Berlin, Heidelberg: Springer; 2003, 83–95.
80. Ricordeau M. Q-Concept-learning: generalization with concept lattice representation in reinforcement learning. *ICTAI* 2003, 2003:316–323.
81. Maddouri M. Towards a machine learning approach based on incremental concept formation. *Intell Data Anal* 2004, 8:267–280.
82. Rudolph S. Using FCA for encoding closure operators into neural networks. In: Priss U, Polovina S, Hill R, eds. *Proceedings of the 15th International Conference on Conceptual Structures: Knowledge Architectures for Smart Applications (ICCS '07)*. Berlin, Heidelberg: Springer-Verlag; 2007, 321–332.

83. Tsope N, Mephu Nguifo E, Tindo G. CLANN: concept-lattices-based artificial neural networks. In: Diatta J, Eklund P, Liquière M, eds. *Proceedings of Fifth International Conference on Concept Lattices and Applications (CLA 2007)*. Montpellier, France: University of Montpellier; 2007, 157–168.
84. Belohlavek R, De Baets B, Outrata J, Vychodil V. Inducing decision trees via concept lattices. *Int J Gen Syst* 2009, 38:455–467.
85. Lindig C. *Fast concept analysis, Working with Conceptual Structures—Contributions to ICCS 2000*; August 14–18; Darmstadt, Germany. Aachen, Germany: Shaker Verlag; 2000, 152–161.
86. Outrata J. Boolean factor analysis for data preprocessing in machine learning. *Proceedings of the 2010 Ninth International Conference on Machine Learning and Applications (ICMLA '10)*. Washington, DC: IEEE Computer Society; 2010, 899–902.
87. Visani M, Bertet K, Ogier J-M. NAVIGALA: an original symbol classifier based on navigation through a galois lattice. *Int J Pattern Recogn Artif Intell* 2011; 25:449–473
88. Madeira S, Oliveira A. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Trans Comput Biol Bioinform* 2004, 1:24–45.
89. Hartigan JA. Direct clustering of a data matrix. *J Am Stat Assoc* 1972, 67:123–129.
90. Cheng Y, Church G. Biclustering of expression data. In: Altman R, Bailey TL, Bourne P, Gribskov M, Lengauer T, Shindyalov IN, Ten Eyck LF, Weissig H, eds. *Proceeding of the 8th International Conference on Intelligent Systems for Molecular Biology (ISBM)*; August 16–23, 2000, La Jolla, CA. Menlo Park, CA: AAAI Press 2000, 93–103.
91. Adomavicius C, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans Knowl Data Eng* 2005, 17:734–749.
92. Ignatov DI, Kuznetsov SO, Poelmans J. Concept-based biclustering for internet advertisement. In: Vreeken J, Ling C, Zaki MJ, Siebes A, Yu JX, Goethals B, Webb GI, Wu X, eds. *ICDM Workshops IEEE Computer Society*. Brussels, Belgium; 2012, 123–130.
93. Prelic S, Zimmermann BP, Wille A, Buhlmann P, Gruissem W, Hennig L, Thiele L, Zitzler E. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics* 2006, 22:1122–1129.
94. Kaytoue M, Kuznetsov SO, Macko J, Napoli A. *Biclustering Meets Formal Concept Analysis, Annals of Mathematics and Artificial Intelligence*, 2013.
95. Pensa RG, Leschi C, Besson J, Boulicaut J-F. Assessment of discretization techniques for relevant pattern discovery from gene expression data. In: Zaki MJ, Morishita S, Rigoutsos I, eds. *Proceedings of the 4th ACM SIGKDD Workshop on Data Mining in Bioinformatics (BIODDD 2004)*; August 22nd; Seattle, WA: ACM Press; 2004, 24–30.
96. Besson J, Robardet C, Raedt LD, Boulicaut J-F. Mining bi-sets in numerical data. In: Dzeroski S, Struyf J, eds. *KDID, LNCS 4747*. Berlin, Heidelberg: Springer; 2007, 11–23.
97. Cerf L, Besson J, Robardet C, Boulicaut J-F. Closed patterns meet n-ary relations. *Trans Knowl Discov Data* 2009, 3.
98. Wille R. The basic theorem of triadic concept analysis. *Order* 1995, 12:149–158.
99. Voutsadakis G. Polyadic concept analysis. *Order* 2002, 19:295–304.
100. Yao YY. A comparative study of formal concept analysis and rough set theory in data analysis. In: Tsumoto S, Slowinski R, Komorowski HJ, Grzymala-Busse JW, eds. *RSCTC, LNAI 3066*; June 1–5; Uppsala, Sweden. Berlin, Heidelberg: Springer; 2004, 59–68.
101. Kent RE. Rough concept analysis: a synthesis of rough sets and formal concept analysis. *Fundam Inform* 1996, 27:169–181.
102. Saquer J, Deogun JS. Concept approximations based on rough sets and similarity measures. *Int. J Appl Math Comput Sci* 2001, 11:655–674.
103. Ganter B. Lattices of rough set abstractions as P-products. In: Medina R, Obiedkov SA, eds. *ICFCA, LNAI 4933*; February 25–28; Montreal, Canada. Berlin, Heidelberg: Springer; 2008, 199–216.
104. Meschke C. Approximations in concept lattices. In: Kwuida L, Sertkaya B, eds. *Proceedings of the 8th International Conference on Formal Concept Analysis (ICFCA'10)*. Berlin, Heidelberg: Springer; 2010, 104–123.
105. Ganter B, Meschke C. A formal concept analysis approach to rough data tables. *Trans Rough Sets* 2011, 14:37–36.
106. Ganter B, Wille R. Conceptual scaling. In: Roberts F, ed. *Applications of Combinatorics and Graph Theory to the Biological and Social Sciences*. Berlin, Heidelberg: Springer; 1989, 139–167.
107. Kuznetsov SO. JSM-method as a machine learning method. *Itogi Nauki i Tekhniki, ser. Informatika* 1991, 15:17–50 (in Russian).
108. Kuznetsov SO. Learning of simple conceptual graphs from positive and negative examples. In: Zytkov JM, Rauch J, eds. *Proceedings of the PKDD 1999, LNAI 1704*; September 15–18; Prague, Czech Republic. Berlin, Heidelberg: Springer; 1999, 384–391.
109. Liquiere M, Sallantin J. Structural machine learning with galois lattice and graphs. *Proceedings of 15th International Conference on Machine Learning*; July 24–27. Madison, WI: Morgan Kaufmann; 1998.

110. Ferré S, Ridoux O. Introduction to logical information systems. *Inf Process Manage* 2004, 40:383–419.
111. Chaudron L, Maille N. From students to approximately reasoning agents. In: Skowron A, Komorowski J, eds. *Proceedings of ECAI'98 Workshop on Synthesis of Intelligent Agent Systems from Experimental Data*. Brighton, UK; 1998.
112. Chaudron L, Maille N, Boyer M. The cube lattice model and its applications. *Appl Artif Intell* 2003, 17:207–242.
113. Wille R. Conceptual graphs and formal concept analysis. *ICCS*; August 3–8; Seattle, WA. Berlin, Heidelberg: Springer; 1997, 290–303.
114. Huchard M, Roume C, Valtchev P. When concepts point at other concepts: the case of UML diagram reconstruction. In: Duquenne V, Ganter B, Liquière M, Mephu E, Stumme Nguifo et G, eds. *Proceeding of FCAKDD 2002, Advances in Formal Concept Analysis for Knowledge Discovery in Databases, International Workshop ECAI 2002*. Lyon: University of Lyon; 2002, 32–43.
115. Huchard M, Rouane-Hacene M, Roume C, Valtchev P. Relational concept discovery in structured datasets. *Ann Math Artif Intell* 2007, 49:39–76.
116. Ferré S, Ridoux O, (2000) A logical generalization of formal concept analysis. *ICCS*; August 14–18; Darmstadt, Germany. Berlin, Heidelberg: Springer; 2000, 371–384.
117. Lehmann F, Wille R. A triadic approach to formal concept analysis. *ICCS*; August 14–18; Santa Cruz, CA. Berlin, Heidelberg: Springer; 1995, 32–43.
118. Pollandt S. *Fuzzy Begriffe*. Berlin, Heidelberg: Springer; 1997.
119. Belohlavek R. Fuzzy Galois connections. *Math Logic Q* 1999, 45:497–504.
120. Georgescu G, Popescu A. Concept lattices and similarity in non-commutative fuzzy logic. *Fundam Inform* 2002, 55:23–54.
121. Prediger S. Nested concept graphs and triadic power context families. *ICCS*. Berlin, Heidelberg: Springer; 2000, 249–262.
122. Groh B, Eklund PW. Algorithms for creating relational power context families from conceptual graphs. *ICCS*; July 12–15; Blacksburg, VA. Berlin, Heidelberg: Springer; 1999, 389–400.
123. Dau F. *The Logic System of Concept Graphs with Negation: And its Relationship to Predicate Logic. Lecture Notes in Artificial Intelligence*. Berlin, Heidelberg: Springer; 2004, 2892.
124. Dau, F, Klinger J. From formal concept analysis to contextual logic. In: Stumme G, ed. *Formal Concept Analysis: The State of the Art. First International Conference on Formal Concept Analysis. Darmstadt, Germany, 2003, Proceedings of the LNCS 3626*. Heidelberg-Berlin: Springer; 2005, 81–100.
125. Wille R. *Boolean Judgment Logic. ICCS*; July 30–August 3; Stanford University, CA. Berlin, Heidelberg: Springer; 2001, 115–128.
126. Wille R. *Implicational Concept Graphs. ICCS*; July 19–23; Huntsville, AL. Berlin, Heidelberg: Springer; 2004, 52–61.
127. Dolques X, Huchard M, Nebut C, Reitz P. Fixing generalization defects in UML use case diagrams. *Fundam Inform* 2012, 115:327–356.
128. Ganter B, Grigoriev PA, Kuznetsov SO, Samokhin MV. Concept-based data mining with scaled labeled graphs. In: Wolff KE, Pfeiffer HD, Delugach HS, eds. *ICCS, LNAI 3127*; July 19–23; Huntsville, AL. Berlin, Heidelberg: Springer; 2004, 94–108.
129. Kuznetsov SO. Pattern Structures for Analyzing Complex Data. In: Sakai H, Chakraborty MK, Hasanien AE, Slezak D, Zhu W, eds. *Proceeding of the 12th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing (RSFD-GrC'09), LNAI 5908*; December 15–18; Delhi, India. Berlin, Heidelberg: Springer; 2009, 33–44.
130. Assaghir Z, Napoli A, Kaytoue M, Dubois D, Prade H. Numerical information fusion: lattice of answers with supporting arguments. *ICTAI*; 2011, 621–628.
131. Pfaltz JL. Representing numeric values in concept lattices. In: Diatta J, Eklund P, Liquière M, eds. *Proceedings of the CLA 2007*. LIRMM & University of Montpellier II.
132. Valverde-Albacete FJ, Peláez-Moreno C. Extending conceptualisation modes for generalised formal concept analysis. *Inform Sci Elsevier* 2011, 181:1888–1909.
133. Gonzalez Calabozo JM, Peláez-Moreno C, Valverde-Albacete FJ. Gene expression array exploration using K-formal concept analysis. *ICFCA*; May 2–6; Nicosia, Cyprus. Berlin, Heidelberg: Springer; 2011, 119–134.
134. Bain M. Inductive construction of ontologies from formal concept analysis. *Aust Conf Artif Intell* 2003, 88–99.
135. Wolff KE. States, transitions and life tracks in temporal concept analysis. In: Ganter B, et al., eds. *Formal Concept Analysis, LNAI 3626*. Berlin, Heidelberg: Springer; 2005, 127–148.
136. Wolff KE. Temporal relational semantic systems. In: Croitoru M, Ferre S, Lukose D, eds., *Conceptual Structures: From Information to Intelligence. ICCS 2010. LNAI 6208*. Heidelberg: Springer; 2010, 165–180.
137. Ganter B. Attribute exploration with background knowledge. *Theor Comput Sci* 1996, 217:215–233.
138. Belohlavek R, Vychodil V. Formal concept analysis with background knowledge: attribute priorities. *IEEE Trans Syst Man Cybern* 2009, 39.

139. Pogel A, Ozonoff D. Contingency structures and concept analysis. In: Medina R, Obiedkov SA, eds. *ICFCA, LNAI 4933*; February 25–28; Montreal, Canada. Berlin, Heidelberg: Springer; 2008, 305–320.
140. Deogun J, Jiang L, Xie Y, Raghavan V. Probability logic modeling of knowledge discovery in databases. In: Zhong N, Ras ZW, Tsumoto S, Suzuki E, eds. *ISMIS, LNAI 2871*; October 28–31; Maebashi City, Japan. Berlin, Heidelberg: Springer; 2003, 402–407.
141. Cellier P, Ferré S, Ridoux O, Ducassé M. A parameterized algorithm to explore formal contexts with a taxonomy. *Int J Found Comput Sci* 2008, 19:319–343.
142. Stumme G. A finite state model for on-line analytical processing in triadic contexts. In: Ganter B, Godin R, eds. *ICFCA, LNCS 3403*; February 14–18; Lens, France. Berlin, Heidelberg: Springer; 2005, 315–328.
143. Jäschke R, Hotho A, Schmitz C, Ganter B, Stumme G. Discovering shared conceptualizations in folksonomies. *Web Seman* 2008, 6:38–53.
144. Ji L, Tan K-L, Tung AKH. Mining frequent closed cubes in 3D datasets, in VLDB '06. *VLDB Endowment*; 2006, 811–822.
145. Besson J, Robardet C, Boulicaut J-F, Rome S. Constraint-based concept mining and its application to microarray data analysis. *Intell Data Anal* 2005, 9:59–82.
146. Zhao L, Zaki MJ. TriCluster: an effective algorithm for mining coherent clusters in 3D microarray data. *ACM Press SIGMOD International Conference on Management of Data (ACM SIGMOD'05)*; June 13–16; Baltimore, Maryland: ACM Press 2005, 694–705.
147. Biedermann K. How triadic diagrams represent conceptual structures. *ICCS*; 1997:304–317.
148. Ganter B, Obiedkov SA. Implications in triadic formal contexts. *ICCS* 2004, 186–195.
149. Glodeanu CV. Fuzzy-valued triadic implications. In: Napoli A, Vychodil V, eds. *CLA 2011 INRIA Nancy-Grand Est and LORIA*; 2011, 159–173.
150. Missaoui R. and Kwuida L. Mining triadic association rules from ternary relations. In: Valtchev P, Jäschke R, eds. *ICFCA, Vol. 6628 of LNCS*; May 2–6; Nicosia, Cyprus. Berlin, Heidelberg: Springer; 2011, 204–218.
151. Ignatov DI, Kuznetsov SO, Poelmans J, Zhukov LE. Can triconcepts become triclusters? *Int J Gen Syst* 2013.
152. Trabelsi C, Jelassi N, Ben Yahia S. Scalable mining of frequent tri-concepts from folksonomies. *PAKDD*; 2012, 231–242.
153. Burusco A, Fuentes-Gonzalez R. The study of L-fuzzy concept lattice. *Mathw Soft Comput* 1994, 3:209–218.
154. Belohlavek R. Fuzzy concepts and conceptual structures: induced similarities. *Joint Conference on Information Sciences '98 Proceedings, Vol. I*. Durham, NC; 1998, 179–182.
155. Belohlavek R. Fuzzy closure operators. *J Math Anal Appl* 2001, 262:473–489.
156. Belohlavek R. *Fuzzy Relational Systems: Foundations and Principles*. New York: Kluwer; 2002.
157. Belohlavek R, Vychodil V. Attribute implications in a fuzzy setting. In: Missaoui R, Schmid J, eds. *ICFCA, LNAI 3874*; February 13–17. Berlin, Heidelberg: Springer; 2006, 45–60.
158. Belohlavek R. Concept lattices and order in fuzzy logic. *Ann Pure Appl Logic* 2004, 128:277–298.
159. Kridlo O, Krajci S. Proto-fuzzy concepts, their retrieval and usage radim belohlavek. In: Kuznetsov SO, ed. Olomouc: CLA Palacky University; 2008, 83–95.
160. Medina J, Ojeda-Aciego M, Ruiz-Calviño J. Formal concept analysis via multi-adjoint concept lattices. *Fuzzy Sets Syst* 2009, 160:130–144.
161. Pankratieva VV, Kuznetsov SO. Relations between proto-fuzzy concepts, crisply generated fuzzy concepts, and interval pattern structures. *Fundam Inform* 2012, 115:265–277.
162. Cuvelier E, Aufaure MA. A buzz and E-reputation monitoring tool for twitter based on galois lattices. *ICCS*; 2011, 91–103.
163. Blinova VG, Dobrynin DA, Finn VK, Kuznetsov SO, Pankratova ES. Toxicology analysis by means of the JSM-method. *Bioinformatics* 2003, 19:1201–1207.
164. Godin R, Missaoui R. An incremental concept formation approach for learning from databases. *Theor Comput Sci* 1994, 133:387–419.
165. Yevtushenko SA. System of data analysis “Concept Explorer”. *Proceedings of the 7th National Conference on Artificial Intelligence KII-2000*. 2000, 127–134 (in Russian).
166. Valtchev P, Grosser D, Roume C, Mohamed Rouane Hacene. *GALICIA: an open platform for lattices, in Using Conceptual Structures: Contributions to the 11th Intl. Conference on Conceptual Structures (ICCS'03)*; July 21–25; Dresden, Germany, Shaker Verlag; 2003, 241–254.
167. Becker P, Hereth J, Stumme G. *ToscanaJ: An Open Source Tool for Qualitative Data Analysis, Proc. Workshop FCAKDD of the 15th European Conference on Artificial Intelligence (ECAI 2002)*; Lyon, France; 2002.
168. Priss U. FcaStone–FCA file format conversion and interoperability software. *Conceptual Structures Tool Interoperability Workshop (CS-TIW)*; 2008.
169. Lahcen B, Kwuida L. Lattice miner: a tool for concept lattice construction and exploration. In: Boumedjout L, Valtchev P, Kwuida L, Sertkaya B, eds. *Supplementary Proceedings of the International Conference on*

- Formal Concept Analysis (ICFCA'10)*; March 15–18. Agadir, Morocco: University of Quebec 2010.
170. Borza PV, Sabou O, Sacarea C. OpenFCA, an open source formal concept analysis toolbox. *Proceedings of IEEE International Conference on Automation Quality and Testing Robotics (AQTR)*; 28–30 May. Cluj Napoca, Romania: IEEE Press 2010, 1–5.
171. Neznanov AA, Kuznetsov SO. Information retrieval and knowledge discovery with FCART. *Proceedings of the Formal Concept Analysis Meets Information Retrieval, Moscow*; 2013.