Rokia Missaoui
Léonard Kwuida
Talel Abdessalem *Editors*

Complex Data Analytics with Formal Concept Analysis

Springer

Rokia Missaoui • Léonard Kwuida
Talel Abdessalem
Editors

Complex Data Analytics with Formal Concept Analysis

Springer

Editors
Rokia Missaoui
University of Quebec in Outaouais
Gatineau, QC, Canada

Talel Abdessalem
Place Marguerite Perey
Télécom-Paris, Institut Polytechnique
Palaiseau, France

Léonard Kwuida
Business School
Bern University of Applied Sciences
Bern, Switzerland

ISBN 978-3-030-93277-0 ISBN 978-3-030-93278-7 (eBook)
https://doi.org/10.1007/978-3-030-93278-7

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Chapter 9
Formal Methods in FCA and Big Data

Domingo López-Rodríguez, Emilio Muñoz-Velasco, and Manuel Ojeda-Aciego

9.1 Introduction

The term *Big Data* generally refers to massive quantities of data exceeding the typical processing and computing capacity of conventional databases and data analysis techniques. Due to its particular features, the *Big Data* problem requires the design of *ad hoc* tools and methods to analyze and extract patterns from large-scale data. Increased data storage capabilities and processing power, together with the availability of massive volumes of data constitute the cause of the recent rise of Big Data. Organizations have more data available than they can process since, in general, their computing resources and technologies are limited and not adapted to the large-scale processing inherent in Big Data. In addition to the obvious massive data volume, Big Data has associated other specific qualities, often referred to as the five Vs: Volume, Variety, Velocity, Veracity and Value (the worth of the information extracted from data) [36, 39].

Machine learning is a subdomain of computer science used to analyze data, automating the construction of analytical models. The purpose of machine learning algorithms is to learn from existing data without the need to explicitly program an analytical model. The trained models learn from preceding data and calculations and aim to produce certain and replicable decisions and outcomes.

Machine learning has an extensive variety of applications in fields such as artificial intelligence, optimal control, statistics, information theory, optimization theory, and many other disciplines of mathematics, engineering, and science [66].

The essence of Big Data Analytics is mining and extracting *patterns* for decision-making, prediction and other types of inference, from massive data. A significant principle is that the extracted patterns should be meaningful and provide some understanding about the analyzed data. As machine learning models are used in critical and sensitive areas like medicine, the criminal justice system, and financial

D. López-Rodríguez (✉) • E. Muñoz-Velasco • M. Ojeda-Aciego
Universidad de Málaga, Departamento Matemática Aplicada, Málaga, Spain
e-mail: dominlopez@uma.es; ejmunoz@uma.es; aciego@uma.es

R. Missaoui et al. (eds.), *Complex Data Analytics with Formal Concept Analysis*,
https://doi.org/10.1007/978-3-030-93278-7_9

markets, the inability of humans to understand these patterns becomes problematic [22, 46].

In this sense, logic-based approaches are perfect candidates to help make machine learning models *understandable*, be it through the hybridization with other techniques [71], or as the core of an expert system [25], just to mention two interesting cases.

Formal Concept Analysis (FCA) is a formal framework, built on lattice theory and Galois connections, allowing to mathematically formalize the notion of "concept" (a general idea that corresponds to some type of entity and that can be characterized by some essential features of the class). The mechanisms used to extract these concepts from a dataset in FCA allow us to hierarchically organize them in the so-called "concept lattice". An important aspect to emphasize is that the concept lattice captures *all* the implicit knowledge that can be deduced from a formal context. Another way to extract knowledge from a context is in the form of implications. In essence, an "attribute implication" is an expression of type $A \to B$ where A and B are sets of attributes/properties, and we say that this is fulfilled if every object that has the attributes of A has those of B as well. The use of implicational systems allows to handle all this implicit knowledge (knowledge that is tacit in the experiences, but not codified nor formalized [65]) and to reason over it. We believe that this logic-based approach is more suitable to provide understandable answers and, hence help avoid the lack of interpretability and explainability of the results.

Explainability and interpretability are often used interchangeably. Although they are very closely related, a greater comprehension of the Big Data problems can be gained if their differences are well-understood.

There is no mathematical definition of interpretability. An approximation could be [56]: "Interpretability is the degree to which a human can understand the cause of a decision". Another one is: "Interpretability is the degree to which a human can consistently predict the model's result" [47]. A higher interpretability of a model means an easier understanding of why certain decisions or predictions are made. Interpretability is about the capability of predicting the outcome given a change in the input data or in the algorithm parameters. That is, it is the ability to understand the relationship between the input provided and the outcome given by the model.

Explainability, meanwhile, is the ability to explain the internal mechanics of a system in human terms. The difference with interpretability is very subtle, and this makes the two terms frequently interchanged. However, one can say, in other words, that interpretability is the ability to understand the influence of the mechanics, and explainability is the ability to describe its mechanics.

Formal logical methods, such as the management of implicational systems stated above, are, contrary to the statistical techniques already in use, highly interpretable and explainable, what makes them more suitable for reasoning and extracting/representing knowledge.

The main issue is to what extent these logic-based methods can be applied to a Big Data framework. The scalability of FCA methods and algorithms depends on, and is constrained by, the complexity of the problems to be solved, namely: the identification of concepts and construction of the concept lattice from a *massive* context, the

computation of a *canonical basis* of implications and the efficient computation of closures with respect to an implication system.

Here, the term *basis* is used for a set of implications, which, besides being sound and complete, satisfies some minimality condition among all equivalent sets of implications (defining the same *closure* system); thus, there may be different types of bases. The *canonical* basis (also called *stem* basis or Duquenne-Guigues' basis [41]) is a basis of minimal cardinality.

It has been proved [7, 33] that, unless P = NP, there is no polynomial delay algorithm to enumerate the implications of the canonical basis, both in the lectic and in the reverse order. Let us note that, even in the simplest cases, the canonical basis can have exponential size [51]. More precisely, the problems of enumerating all pseudo-intents in a formal context (premises of implications in the canonical basis) and computing the lectically largest pseudo-intent belongs to the coNP-complete complexity class [7]. Even if the enumeration of pseudo-intents can be done without an explicit ordering, this result is still true, as it was proved by comparing this problem with that of enumerating minimal transversals in a hypergraph [34, 45]. Thus, somehow taming the complexity issues by designing alternative approaches or increasing the threshold up to which the complexity is still tractable is a major open question.

Therefore, most efforts are focused on two (overlapping) strategies: the development of algorithms which, in the average case, have a short running time and low complexity, and the definition of probabilistic and approximated logic approaches, which may capture essential knowledge with high probability and reduced computational effort. These two strategies seek to build better techniques to explore large datasets.

The objective of this work is to present a survey of the theoretical and technical foundations of some trends of FCA with respect to the issues stated above. We have explored and collected formal and technical developments regarding three pillars:

- The efficient construction of the concept lattice associated to a formal context by the use of simplification procedures.
- The definition and computation of bases of implications different from the canonical basis, satisfying other optimality conditions, and of logic rules and operators that may lead to more efficient inference systems.
- The definition of *probably approximately correct implication bases*, as a way to capture most of the knowledge contained in a dataset with efficient and scalable algorithms.

The remainder of this work is structured as follows: in Sect. 9.2, we present techniques for context and concept lattice reduction, including methods to simplify the structure and decrease the practical complexity of the problem. In Sect. 9.3, logic tools are described in order to operate on implication sets and define optimal bases which may have an important impact on the performance of inference systems. In Sect. 9.4, minimal generators are presented as a means to encapsulate a compressed representation of the knowledge present in a formal context. Finally, Sect. 9.5 presents the foundations for approximated implication bases and efficient algorithms that

can be used to compute them. We finish with Sect. 9.6, where we present some conclusions and potential future trends in the application of formal methods to Big Data.

9.2 Context and Concept Lattice Reduction Methods

As stated before, the complexity of (the algebraic and logic tools related to) FCA can be considered one of the most outstanding problems when trying to make effective use of FCA in Big Data situations. Thus, some previous steps may be taken in order to decrease the computational efforts needed to build the concept lattice and to extract the implicational basis, actually reducing the complexity of the context or of the associated concept lattice, both in terms of magnitude (size of object and attributes sets) and of inter-relationships, while relevant information is kept. This set of techniques is known as *concept lattice reduction methods*.

In [31], the authors make a classification of concept lattice reduction methods into three categories: redundant information removal, concept lattice simplification and attribute selection.

Redundant Information Removal

The aim of redundant information removal techniques is, given the formal context $\mathbb{K}$, to build a formal context $\mathbb{K}'$ whose concept lattice has the same structure as that of $\mathbb{K}$.

Definition 9.1 For a given formal context $\mathbb{K} = (G, M, I)$, an object $g \in G$ (or an attribute $m \in M$ or an incidence $i \in I$, resp.) is said to be *redundant information* if its removal builds a formal context $\mathbb{K}' = (G', M', I')$ with $G' = G \setminus \{g\}$ (resp., $M' = M \setminus \{m\}$, or $I' = I \setminus \{i\}$) whose concept lattice is isomorphic to that of $\mathbb{K}$.

Note that redundant information is a term coined from the purely algebraic point of view, but not from the point of view of the application domain. This means that not all the redundant information in a context may be considered not relevant, since the relevance of an object or attribute is defined by the application context.

The *clarification* of a context [38] consists in replacing a set of objects $\{g_i\} \subseteq G$ with exactly the same attributes by a representative object (that is, substitute all g_i by a single object $g \in [g_i]$, where $[g_i]$ denotes the equivalence class of the objects with the same attributes as g_i) and making the analogous substitution in attributes. This way, the new formal context $\mathbb{K}' = (G', M', I|_{G' \times M'})$, isomorphic to $\mathbb{K}$, is obtained by removing duplicated rows and columns of the formal context.

Depending on the background knowledge, it may be advisable not to merge two objects with the same attributes. In the application domain, they may still be considered different from one another and changing a concept extent can influence

the quality measures [52] computed from it. Thus, it is recommended to proceed carefully when performing *clarification* in some contexts.

Other research lines [73] aim to analyse the information in the incidence table in terms of the partitions induced on the sets of objects and attributes by some functions of single attributes and objects of the context, obtaining the so-called Formal Equivalence Analysis (FEA). Rather than looking on the effect of these partitions on set representation, as done in Rough Sets Theory [63], the emphasis is put on making explicit the information in the context.

Another kind of reduction is the removal of attributes representable by a combination of other attributes [38]. The former are called *reducible* attributes:

Definition 9.2 Given a formal context $\mathbb{K} = (G, M, I)$, an attribute $m \in M$ is called *reducible* if there exists $M' \subseteq M$ with $m \notin M'$ such that the extent of m coincides with the extent of M'. In other words, m is reducible if it is contained in the closure of M'.

Analogously, an object $g \in G$ is *reducible* if there is a $G' \subseteq G$ with $g \notin G'$ such that the intents of g and G' coincide. In the dual context $\tilde{\mathbb{K}} = (M, G, \tilde{I})$ ($m\tilde{I}g \iff gIm$), the closure of g and that of G' (as attribute sets) are equal.

The result of processing a formal context by using clarification and the removal of reducible attributes gives the *standard context*, and it is isomorphic to the original context [38]. The complexity of constructing the standard context using the clarification and reduction methods is $O(|G||M|^2)$, hence polynomial.

These *clarification* and *reduction* methods are only a sample of the techniques which aim to minimize the number of attributes in the context while maintaining the isomorphic correspondence between the original concept lattice and the associated to the reduced context. In general, these techniques focus on computing the possible *reducts* of a context:

Definition 9.3 [48] Given a context $\mathbb{K} = (G, M, I)$, a set $M' \subsetneq M$ is called *consistent* if the concept lattice associated to $(G, M', I|_{G \times M'})$ is isomorphic to that of $\mathbb{K}$. A *reduct* is a minimal consistent set of attributes, that is, X is a reduct if X is consistent and no $X' \subsetneq X$ is consistent.

An equivalent definition of *reduct* is that it is a maximal subset of M without reducible attributes [49].

Although the *clarification* and *reduction* methods have polynomial complexity (in the number of attributes), the general problem of computing all reducts of a formal context has an exponential computational cost, since the number of possible subsets to explore is exponential.

In order to propose a mechanism to compute a minimal set of attributes, in [76], the concept of discernibility matrix is introduced:

Definition 9.4 Given a formal context $\mathbb{K}$, the *discernibility* between concepts (A, B) and (C, D) is defined by the symmetric difference between sets B and D:

$$\mathrm{dis}\,((A, B), (C, D)) = (B \setminus D) \cup (D \setminus B)$$

The *discernibility matrix* associated to the context is a matrix indexed by the concepts, whose entries are the corresponding pairwise discernibilities. The set of non-empty elements of the discernibility matrix is represented by $\Lambda_{\mathbb{K}}$.

Using a discernibility matrix, all the reducts can be obtained, by the application of the following result:

Theorem 9.1 *Given a context* $\mathbb{K} = (G, M, I)$ *and* $\Lambda_{\mathbb{K}}$ *the set of non-empty elements of its discernibility matrix, then* $D \subseteq M$ *is consistent if and only if* $D \cap H \neq \varnothing$ *for all* $H \in \Lambda_{\mathbb{K}}$.

This theorem shows that to find a reduct of a formal context is to find the minimal subset D of attributes which verify the mentioned restriction.

In the same work [76], the authors classify the attributes as *absolutely necessary*, *relatively necessary* or *absolutely unnecessary* to reflect whether they are necessary, contingent or superfluous.

Definition 9.5 Given a formal context $\mathbb{K} = (G, M, I)$, an attribute is called absolutely necessary if it is present in all reducts; it is said to be relatively necessary if it is present in at least one, but not in all minimal consistent sets; and finally, an attribute is unnecessary if it is not in any reduct.

However, the computation of discernibility the matrix requires a high computational effort, since all pairs of concepts are checked for its construction.

A more adequate refinement is proposed in [67], where the authors determine that many of the discernibility sets computed following the strategy in [76] do not actually contribute to finding the reducts. It was proved that the only discernibility sets that need to be computed are those related to adjacent concepts in the concept lattice, that is, to pairs of concept-superconcept. This number of computations corresponds to the number of edges in the lattice, and is clearly lower than the total number of concept pairs. However, both discernibility matrix methods need to compute all formal concepts beforehand, which requires an exponential time in the worst-case.

In [48], a comparison between the traditional clarification and reduction method with that of the discernibility matrix is presented. It is proved that the sets of attributes that are merged in the clarification and reduction steps are exactly minimal non-empty discernibility sets, therefore with the clarification and reduction we can obtain the same result as with methods of attribute reduction based on the discernibility matrix. The relevance of this result can be better understood if we see that the complexity of the clarification and reduction method is $O(|G||M|^2)$ (polynomial) and the discernibility matrix requires $O(|M|^2|\mathscr{L}|))$, where $\mathscr{L}$ is the set of concepts, which can be of cardinality up to $2^{\min\{|G|,|M|\}}$, so actually the computation of the discernibility matrix is exponential in the worst-case.

A further improvement is presented in [49], where a modified algorithm is introduced that only computes the minimal discernibility sets, allowing for polynomial time complexity, in contrast to the exponential complexity mentioned above. In addition, it is stated that if the consistent elimination of all unnecessary computations of discernibility sets is pursued, the resulting method is just cosmetically different

from clarification and reduction. In conclusion, the methods based on the computation of the discernibility matrix cannot become more efficient than those based on clarification and reduction of the context.

It is worth noting that the classification into three categories (necessary, contingent, superfluous) is independent of the following frameworks: (usual) concept lattices, property-oriented concept lattices and object-oriented concept lattices [55]. Thus, only one of these three types of lattices needs to be considered when computing the reducts, and the algorithms developed, such as [15, 74], can be applied to any of them.

A different reduction strategy is studied in [44]. It is focused on the incidence relationship, considering the influence of eliminating a single incidence from the formal context in the complexity of building a the new reduced concept lattice.

Simplification of the Structure

Other reduction techniques apply an abstraction of the concept lattice, looking for a high-level overview that preserves only the essential aspects. This type of techniques are grouped as *simplification methods*. In general, these approaches try to build a simplified context or lattice, by grouping similar objects, attributes or concepts.

Another line of research is the use of matrix factorization techniques which allow to decompose the formal context into a simpler representation. Most works [23, 24, 37] in this line use the technique of singular value decomposition (SVD), which allows to project a high-dimensional matrix into one of lower dimensionality. The SVD is used to embed objects into an Euclidean space where similarity measures can be defined, such as the cosine similarity between the vector representation of objects [23], or induce equivalence classes of objects or attributes.

Another application of matrix factorization, is to find key factors which could represent, exactly or approximately, an underlying structure in data. The linear combination of these factors is a lossy approximation of the original formal context. Since the number of factors is usually much lower than the cardinality of the set of objects, they can be considered as representative rows in the formal context, achieving a significant reduction and preserving (with a little loss) all the information contained in the context. Among matrix factorization techniques for context reduction, we can find non-negative matrix factorization [50] and binary matrix factorization [13].

In other works, the matrix factorization is induced not from the formal context, but from the concept lattice itself [11, 62], and the aim of those methods is to get the best representation of the knowledge as a set of factors, which are then called *concept factors* [13].

Selection of Attributes and Concepts

The underlying idea is the application of some criterion which quantifies the importance of attributes and concepts, in order to keep a subset of those with the highest

relevance. The importance criterion is application-dependent and, thus, varies between different domains.

There are two different approaches in the selection of attributes and concepts: select relevant items *a posteriori*, after the computation of the whole concept lattice, or make an *a priori* analysis of a relevance measure and then build the *reduced* concept lattice with pre-chosen constraints.

As commented above, the first strategy makes use of the complete concept lattice to infer the importance of attributes, objects and concepts.

A simple notion of relevance may be related to the cardinality of the intension or extension of a concept, but more refined definitions include the assignment of weights [10] to each attribute, and then selecting formal concepts considered relevant. In this case, the relevance of a concept is calculated as the average relevance of the attributes in its intension [75]. In [10], it is also proposed the use of minimal generators of concepts, instead of their intension, to evaluate the importance of a formal concept.

A comprehensive review of interestingness measures of concepts is presented in [52]. In that work, the measures considered are compared regarding aspects such as efficiency of computation and applicability to noisy data.

From the algorithmic perspective, a major issue in this strategy is that concepts must be computed before verifying whether their relevance is above a predefined threshold, due to the inherent complexity of enumerating all formal concepts.

As a solution to this problem, another research line attempts to pre-select attributes based on a relevance measure (thus, used *a priori*) and therefore define constraints that the computed concepts must fulfill.

The relationship between the notions of frequent formal concept (a concept whose support is above a predefined threshold) and of (the analogous) frequent itemsets in transaction databases was exploited in [72] to introduce the Titanic algorithm for the construction of *iceberg lattices*. An *iceberg* lattice consists of frequent itemsets associated to a predefined minimum support, and is therefore more efficient to build than the complete concept lattice.

Another method to reduce the complexity of concept and attribute selection is presented in [14]. In that work, the authors build a general framework for concept selection, in which the relevance criterion is represented as a closure operator. The relevant concepts are the fixpoints of the associated closure operator and they form a complete $\vee$-sublattice of the original concept lattice. Several constraints related to the cardinality of the extent and to the presence or absence of different attributes are also studied. The proposed method allows to compute the reduced lattice and extract minimal bases of attribute dependencies as well, with lower computational effort (polynomial delay complexity), since there is no need to compute all concepts or pseudo-intents beforehand.

A more recent approach [42] introduces a method for attribute selection in formal contexts based on the notion of attribute relevance: according to that definition, an attribute is relevant if and only if it is irreducible in the context. This allows to define a *relative relevance function* which captures both the order structure in the concept lattice and the distribution of objects.

The relative relevance function is not computationally feasible (it has a high complexity), so the problem is approximated using ideas from information theory, defining the *Shannon object information entropy of a formal context* and its *object information entropy*. It is experimentally tested and proved that the use of the entropy measures is appropriated in contexts with many attributes and reflects the relative relevance of attributes properly.

As a consequence, in a Big Data problem, this type of mechanisms based on the pre-definition of a relevance measure (probably based on information theory, due to its computational efficiency) which allows to restrict the computation of concepts to only the ones with a higher relevance, presents great potential of use, and can therefore help develop optimized methods to mine the knowledge in a large formal context.

9.3 Improved Management of Implications

The computation of closures is one of the core steps when reasoning from an implication system. It requires to apply repeatedly the implications in the system (usually the *canonical* basis) until getting to a fixpoint of the closure operator, which means it has exponential complexity.

The main reason that forces to apply several times all the implications in the system in order to compute a closure (causing the high computational complexity) is given by the application of the *transitivity* axiom in the logic [4]:

$$[\texttt{Tran}]\ \frac{A \to B,\ B \to C}{A \to C}$$

The transitivity rule somehow reflects the cut rule in other logical systems and, hence, is not suitable for automation. Even works defining equivalent axiom systems [5, 43] do not arrive at an appropriate way of handling its inherent complexity in an efficient manner.

As a consequence, there has been traditionally a necessity of finding efficient computational methods by modifying the axiom system (to avoid the computationally expensive transitivity `[Tran]`) and designing new inference and reasoning methods.

Another way to reduce the computational cost of computing closures is to define some modified implicational systems, equivalent to the Duquenne-Guigues basis [41], but with a simpler structure that could be exploited in the calculation of closures. Although Duquenne-Guigues basis has minimum cardinality, there are other parameters which can be used to define alternative minimality conditions (e.g. *directness* [17, 18]) which, in turn, might have better computational properties.

Simplification Logic

The investigation on defining axiom systems equivalent to Armstrong's rules, but removing the transitivity axiom, led to the idea of the Simplification Logic.

Simplification Logic $\mathrm{SL_{FD}}$ [27], and its fuzzy counterpart FASL [9], define a logic equivalent to Armstrong's Axioms that avoids the use of transitivity and is guided by the idea of simplifying the set of implications by removing redundancies, which can be defined in the following terms:

Definition 9.6 Let $\mathbb{K} = (G, M, I)$ be a formal context and $\Gamma = \{A \to B : A, B \subseteq M\}$ be an implicational system.

- An implication φ is *superfluous* in Γ if φ can be inferred from $\Gamma \setminus \{\varphi\}$.
- $\varphi = X \to Y$ is *l-redundant* in Γ if there exists $\emptyset \neq Z \subseteq X$ such that φ can be inferred from $(\Gamma \setminus \{\varphi\}) \cup \{(X \setminus Z) \to Y\}$.
- $\varphi = X \to Y$ is *r-redundant* in Γ if there exists $\emptyset \neq Z \subseteq Y$ such that φ can be inferred from $(\Gamma \setminus \{\varphi\}) \cup \{X \to (Y \setminus Z)\}$.

We say that Γ has redundancy if it has a superfluous, l-redundant or r-redundant element.

$\mathrm{SL_{FD}}$ provides new substitution operators which allow the natural design of automated deduction methods and new substitution rules which can be used bottom-up and top-down to get equivalent sets of implications, but without redundancy.

Definition 9.7 The $\mathrm{SL_{FD}}$ system has one axiom:

$$[\texttt{Ax}]\ \frac{}{X \to Y} \quad \text{if } Y \subseteq X$$

and three inference rules (fragmentation, composition and substitution):

$$[\texttt{Frag}]\ \frac{X \to Y}{X \to Y'} \quad \text{if } Y' \subseteq Y$$

$$[\texttt{Comp}]\ \frac{X \to Y, U \to V}{X \cup U \to Y \cup V}$$

$$[\texttt{Subst}]\ \frac{X \to Y, U \to V}{U \setminus Y \to V \setminus Y} \quad \text{if } X \subseteq U, X \cap Y = \varnothing$$

The corresponding substitution operators associated to the [Subst] rule are given below.

Definition 9.8 Given an implication $X \to Y$:

- The *substitution operator* associated to $X \to Y$ is defined as:

$$\Phi_{X \to Y}(U \to V) = \begin{cases} U \setminus Y \to V \setminus Y & \text{if } X \subseteq U, X \cap Y = \varnothing \\ U \to V & \text{otherwise} \end{cases}$$

- The *right-substitution operator* associated to $X \to Y$ is defined as:

$$\Phi^{\mathrm{r}}_{X\to Y}(U \to V) = \begin{cases} U \to V \smallsetminus Y & \text{if } X \not\subseteq U, X \cap Y = \varnothing, X \subseteq U \cup V \\ U \to V & \text{otherwise} \end{cases}$$

The extension to the fuzzy setting was presented in [9]. In that case, the axiomatic system used the following inference rule (simplification)

$$[\texttt{Sim}]\ \frac{X \to Y, U \to V}{X \cup (U \smallsetminus Y) \to V}$$

instead of `[Subst]`.

The application of these operators generates a simpler implicational system, removing superfluous and l- and r-redundant implications.

The use of a *simplified* implication system can have a computational impact when making inference on *massive amounts of data*, since the number of operations to be performed decrease as redundancies are removed. For instance, this logic has proved to be useful for automated reasoning with implications [25, 28, 29, 59, 60]. However, the overall complexity of computing a closure remains exponential, in the worst-case scenario, as is in the case of the Duquenne-Guigues basis.

Direct Bases

One reason for the high complexity in computing closures is the need to make several applications of a whole implicational system until getting to a fixpoint, due to the *transitivity* axiom. It is reasonable to investigate the definition of implicational systems equivalent to a given one, such that closures can be computed in a single pass over the implication set [17, 18, 26, 69, 70].

This property for these systems is called *directness* [17, 18], and can be introduced formally as follows: suppose an implicational system Γ, over the set of attributes M, and define the operator $\pi_\Gamma : 2^M \to 2^M$ as $\pi_\Gamma(X) = X \cup \{b \in B | A \to B \in \Gamma \text{ for some } A \subseteq X\}$.

This function π_Γ is isotone and extensive and, therefore, for all $X \in 2^M$, the chain $X, \pi_\Gamma(X), \pi^2_\Gamma(X), \pi^3_\Gamma(X), \ldots$ reaches a fixpoint, and the closure of the set X coincides with this fixpoint. For specific implicational systems, π_Γ is idempotent, which means that the fixpoint is reached in the first iteration, i.e. with a single traversal of the implicational system [26].

Thus, inference with this type of bases can be done in a time complexity which is linear with respect to the number of the implications in the basis.

If, to the condition of a system Γ being direct, one adds that both its cardinality $|\Gamma|$ (number of implications in it) and its size [53] ($\|\Gamma\| = \sum_{A\to B\in\Gamma}(|A| + |B|)$) are minimum among all equivalent systems, an important reduction of the complexity of computing closures may be achieved.

Definition 9.9 A direct implicational system Γ is said to be a *direct-optimal* basis if, for any direct implicational system Γ', equivalent to Γ, one has $\|\Gamma\| \leq \|\Gamma'\|$.

It is proved in [17] that, for any implicational system Γ, there exists a unique direct-optimal basis $\Gamma_{\text{do}} \equiv \Gamma$.

Direct-optimal bases combine the directness and optimality properties. On the one hand, directness ensures that the computation of the closure may be done in just one traversal of the implication set; on the other hand, due to its minimal size provided by the optimality, the number of visited implications is reduced to the minimum. Due to these features, it is desirable to design methods to transform an arbitrary set of implications into its equivalent direct-optimal basis. Thus, the problem of building a direct-optimal basis is one of the outstanding problems in FCA.

Several schemes have been proposed to compute the direct-optimal basis [17, 70], by studying the possibility of using unitary implications. In those cases, although the output implicational system is of higher cardinality (since all *conclusions* are unitary), its computation is more efficient.

In the general (n-ary) case, in [69], a simple algorithm to compute such a direct-optimal basis is presented, based on the sequential application of the rules of the previous Simplification Logic SL_{FD}. A new rule can be obtained which improves the *overlap* rule `Ovl` from [18]

$$[\texttt{Ovl}]\ \frac{A \to B, C \to D}{A \cup (C \smallsetminus B) \to D} \quad \text{if } B \cap C \neq \varnothing$$

the new rule, called *strong simplification*, can be formally derived from SL_{FD} and is used iteratively in conjunction with the *(right-)substitution operator*, leading to simpler implications.

$$[\texttt{sSimp}]\ \frac{A \to B, C \to D}{A \cup (C \smallsetminus B) \to D \smallsetminus (A \cup B)} \quad \text{if } B \cap C \neq \varnothing$$

Its associated *strong-simplification operator* is the core of the method to compute the unique direct-optimal basis.

The complexity of the calculation of such direct-optimal basis is still exponential in the number of implications in the original implication system in the worst-case. One may argue that, in this case, there may be no gain in a practical situation. However, in practice [69], it is asymptotically faster (i.e., when the number of implications or attributes is increased) with respect to the previous most efficient algorithm [70].

In addition, note that in this case the computation of closures is reduced to a single pass over a (larger in size) implicational system. Thus, when reasoning in a Big Data context, the use of direct-optimal bases may be more efficient. Since the computationally expensive construction of such a basis can be done *offline*, this overhead does not impact the end user. Then, the main benefit is that the computation of closures using this basis can be done in the time constraints currently imposed in Big Data settings.

Ordered-Direct Bases

Given a set of indexed implications $\Gamma = \{A_i \to B_i\}$, $1 \leq i \leq n$, Adaricheva et al. [1] introduced an alternative approach to directness, named *ordered-directness*, in terms of the *ordered iteration operator*, which is defined by $\rho_\Gamma = \pi_{\Gamma_n} \circ \ldots \pi_{\Gamma_1}$, where $\Gamma_i = \{A_i \to B_i\}$ for all i.

Analogously to the π_Γ operator, ρ_Γ has the same computational cost and it is isotone and extensive. In addition, one can get that, for all $X \subseteq M$, it holds $\pi_\Gamma(X) \subseteq \rho_\Gamma(X)$, what can lead to a faster convergence to a fixpoint of the chain $X, \rho_\Gamma(X), \rho^2_\Gamma(X), \ldots$

Definition 9.10 [1] An implicational system Γ is said to be *ordered direct* if ρ_Γ is idempotent, that is, if $\rho_\Gamma(X)$ coincides with the closure of X with respect to Γ, for all $X \subseteq M$.

In this case, it must be noted that the directness of an implicational system implies its ordered-directness, but the converse is not always true [1]. In the same work, the authors propose a new kind of bases, called the D-bases, by using the ρ_Γ operator.

Definition 9.11 [1] Let Γ be a reduced implicational system (that is, $A \cap B = \varnothing$ for all $A \to B \in \Gamma$), and let us denote by A^+_Γ the closure of the set A with respect to Γ. The D-basis for Γ is the pair $\langle \Gamma_a, \Gamma_n \rangle$, where:

- $\Gamma_a = \{x \to y \mid y \in M, x \in \{y\}^+_\Gamma, x \neq y\}$
- $\Gamma_n = \{X \to x \mid X \subseteq M, x \in M, X \text{ is a minimal proper cover of } x\}$

Here, the notion of a set $X \subseteq M$ being a *proper cover* of $y \in M$ with respect to the implicational system Γ means that $y \in X^+_\Gamma \smallsetminus \left(\bigcup_{x \in X} \{x\}^+_\Gamma \right)$, i.e., the closure of X contains y, but no single element $x \in X$ has y in its closure. This implies that, evidently, $y \notin X$.

A *proper cover* $X \subseteq M$ of $y \in M$ is said to be *minimal* if, for all other proper cover Z of y, with $Z \subseteq \bigcup_{x \in X} \{x\}^+_\Gamma$, we have $Z \subseteq X$.

Note that implications in Γ_a are atomic, that is, both premise and conclusion are singletons (atoms of M). Implications in Γ_n have unitary conclusion but n-ary premise.

The next result states that D-bases are actually a subclass of *ordered-direct* bases.

Theorem 9.2 ([1]) *Let Γ be an implicational system, with $\langle \Gamma_a, \Gamma_n \rangle$ its associated D-basis and let $\Gamma_D = \Gamma_a \cup \Gamma_n$, ordered in such a way that, in the ordered iteration operator ρ_{Γ_D}, atomic implications from Γ_a are checked before those of Γ_n. Then Γ_D is an ordered-direct basis equivalent to Γ.*

In addition, if Γ_{udo} is the unitary (relative to conclusions) direct-optimal basis equivalent to Γ, then $\Gamma_D \subseteq \Gamma_{udo}$.

D-bases belong to the family of bases whose implications have unitary conclusions. In the mentioned work [1], a method is proposed to extract Γ_D from any direct unit basis Γ_{udo} in polynomial time with respect to the size of Γ_{udo}, and taking only linear time (of the cardinality of the produced Γ_D) to put it into the order assumed in the previous theorem (atomic implications before n-ary ones).

A further improvement was made in [68], where the idea of aggregating a D-basis is presented. A D-basis is *aggregated* if its premises are pairwise disjoint. In that work, the uniqueness of aggregated D-bases is proved. By using the simplification logic, the authors propose the `fastD-basis` algorithm, whose input is an implication system, not necessarily a direct unit basis, and returns the unique aggregated D-basis equivalent to it. Also, the authors proved empirically a higher performance with respect to the previous technique [1].

Running the D-basis in one iteration is more efficient than running a shorter, but unordered, canonical basis, such as Duquenne-Guigues. There are examples demonstrating that the canonical basis cannot always be ordered [1].

As stated above, since the computation of an implication basis can be made *offline* in practical Big Data scenarios, the main goal in this research line is to obtain the most adequate presentation of an implication system which allows to perform the computation of closures fast enough to be used in a real-world application. A D-basis presents a promising step towards the development of a knowledge engine completely automated and applicable in real-world situations.

9.4 Minimal Generators to Represent Knowledge

A compact representation of closed sets in a closure system can have an important impact on the efficient evaluation and construction of implicational bases.

In this sense, minimal generators[8] (or *mingens*) constitute a key part of the closure structure, since they represent minimal sets in the underlying equivalence relation over subsets of the attribute set M.

Definition 9.12 Given a closed set $X \subseteq M$ and a set of implications Γ over M, a subset $Y \subseteq M$ is called a *generator* of X if $X = Y_\Gamma^+$, that is, if the closure of Y with respect to the implicational system Γ is equal to X.

Note that any other subset of X containing its generator Y is also a generator of X. As the set of attributes M is finite, the set of its generators can be characterized by looking for those with the minimality condition.

Definition 9.13 Let M be a finite set of attributes and Γ an implicational system on M. $X \subseteq M$ is called a *minimal generator* (*mingen*) if, for all proper subsets $Y \subsetneq X$, it holds that $Y_\Gamma^+ \subsetneq X_\Gamma^+$.

For a given closed set $C \subseteq M$, let us denote as $\text{mg}(C)$ the set its minimal generators, therefore $\text{mg} : 2^M \to 2^{2^M}$.

Minimal generators [8] were introduced in various fields under various names: minimal keys in the database field [54], irreducible gaps [41], minimal blockers [64] and 0-free itemsets [21].

The importance of minimal generators is that they store a compact representation of the knowledge stored in a context. Minimal generators favor the principle of *minimum description length*, that is, the best hypothesis for a given dataset is the one leading to the best compression of the data [40].

A remarkable contribution on the relevance of *mingens* is that they can be used to build the *iceberg* lattice [72] using the TITANIC algorithm, a smart procedure that is able to take advantage of minimal generators (or *key sets* as they are called in that work) to generate the concept lattice of minimal generators.

This kind of lattices has been successfully used in several applications, such as the analysis of large databases, and extracting implications and mining association rules [32]. This is specially useful when dealing with *large* datasets, since the algorithm is able to retrieve *mingens* with high support.

Also, from this set of minimal generators, we can rebuild all the information that may be inferred from the context, in the form of an implicational basis. Due to this compact representation, the derived bases may allow for a better performance of the reasoning methods.

In [35, 61], some methods for computing the set of *mingens* have been proposed. In parallel to these methods, in [57, 58] minimal generators are used to compute implication bases (on contexts with positive and negative attributes) whose premises are minimal generators.

In all the previous works, the context was considered as the input of the problem, that is, the set of minimal generators (and of closed sets) was inferred directly from the dataset. A more logic-oriented approach was used in [30], where the presented method allowed to derive the set of *mingens* from an implication basis, using the previously mentioned Simplification Logic, instead of using the context as starting point. This complete and precise specification allows for faster managing the semantics of the information contained in the dataset.

Another technique was developed in [28], where the `MinGen` algorithm was designed to build a search tree space (of the implications) that can be traversed (using inference rules) to find all the minimal generators. This shape of the search space limits its execution for medium-sized problems, because of the overwhelming requirements of the sequential `MinGen` algorithm.

In order to get this algorithm working in medium-sized problems, in [16] the authors present an efficient reduction of the search space technique to improve the performance of the enumeration of *mingens*. The new method was designed to fit the Map-Reduce architecture, and thus parallel computation makes it possible to deal with large datasets. As the authors state in their work, "the empirical study proves the very significant improvement achieved w.r.t the original sequential version. The parallel methods to compute minimal generators can make really usable these methods in practical applications."

As commented before, *minimal generators* can be used to rebuild an implication basis. To this end, the notion of labelled set of items is introduced:

Definition 9.14 [28] A *labelled set of items* (LSI) is a collection $\Phi = \{\langle A_i, B_i\rangle\}$, where $A_i \subseteq M$ and $B_i \subseteq 2^{A_i}$ such that if $X, Y \in B_i$ with $X \subseteq Y$ then $X = Y$.

Particularly, for an arbitrary set of implications Γ, we can consider the LSI $\Phi = \{\langle C, \mathrm{mg}(C)\rangle \mid C \text{ is a closed set under } \Gamma\}$ of special interest, since it can be used [29] as input to a procedure aimed to extract a so-called *left-minimal* basis, defined as follows.

Definition 9.15 An implicational system Γ is a *left-minimal* basis if there is no $A \to B \in \Gamma$ and $A' \subsetneq A$ such that $\Gamma \setminus \{A \to B\} \cup \{A' \to B\}$ is equivalent to Γ. In addition, Γ is *direct* if, for all $A \to B \in \Gamma$, one has that $A \cup B$ is closed with respect to Γ.

In other words, a *left-minimal direct* basis is a set of implications where premises are minimal generators and the corresponding conclusions are their associated closed sets. Such a *left-minimal direct* basis has the minimality property of canonical bases, and the characteristics of the implications described before: minimal information in the left hand side and a fast computation of closures.

In order to compute the basis, the following result, given in [29], states that two aggregation operators can be used iteratively.

Theorem 9.3 *Let $T = \{(M_i, A_i \to B_i)\ldots\}$ be a set of pairs with minimal generators and implications obtained from minimal generators and closed sets. The exhaustive application of the two following aggregation rules:*

- *If $A \subseteq C$, then $\{A \to B, C \to D\} \equiv \{A \to B, B \cup C \to D \setminus B\}$.*
- *If $A \subseteq C \subseteq A \cup B$, then $\{A \to B, C \to D\} \equiv \{A \to B \cup D\}$.*

produces a left-minimal direct basis.

Using this result, Cordero et al. propose in [29] an extension of the algorithm designed to compute the Duquenne-Guigues basis [12, 41], in order to obtain the left-minimal basis. This algorithm runs in polynomial time on the length of the LSI used as input. The first step is to compute an implication set from the LSI of the minimal generators (as the one described in the previous theorem) and, then, apply the aggregation rules to manipulate the implications and obtain a left-minimal basis.

9.5 Probably Approximately Correct Implication Bases

Current algorithms to find implication bases have an enormous overhead since they have to find *all* closed sets as a necessary step in their execution. This has led to another research line in which the exactness of the implication basis to be found is not considered as fundamental, that is, it is allowed to have non-exact or non-complete but representative and informative bases.

Besides this difficulty in computing implication bases, there is another more practical reason to relax the constraint of having exact implications: real-world

datasets are generally noisy, containing errors and inaccuracies, therefore computing exact implication bases from such datasets may be useless or even a nonsense. In this case, a rather pragmatic approach is to consider implications as strong association rules and then use highly optimized association rule algorithms [2], but the number of resulting implications increases even more.

Some works [6, 19, 20] have studied approximations of the exact implication basis, taking into account that such bases should have a controllable error, otherwise they become useless. Their proposal is, instead of calculating large exact bases, to compute *approximately correct bases* that could capture most of the implicational theory of a given dataset (or at least the most essential parts) but are easier to compute.

It can be said [20] that a set Ω of implications is an *approximately correct basis* of the formal context $\mathbb{K}$ if most closed sets of Ω are closed in $\mathbb{K}$ and vice versa. The formal idea behind this intuition on *approximate* bases is to define a measure of *proximity* between sets of implications in terms of their closed sets. Thus, the key to build approximate bases is to define a *distance* between closure operators.

In [6], one can find initial results on approximate bases and some experimental evaluations, and a set of implications Ω is defined as an *approximation* of another set Γ if the closure operators of both coincide on most subsets of the attribute set M. This measure can be defined in terms of the cardinality of $\{S \subseteq M : \Omega(S) \neq \Gamma(S)\}$, being $\Omega(S)$ and $\Gamma(S)$ the closures of S with respect to both sets of implications. This can be understood from the application point of view, since that definition ensures that, in most cases, operating with Γ and with Ω will provide the same closures.

This work has been extended to the idea of building *probably approximately correct bases* (PAC bases) in [20], which are approximately correct with high probability. This notion gains strength, since PAC bases can be computed in polynomial time. In this new approach, the concept of *approximation* is slightly different from the one in [6]. Ω is an approximation of Γ if and only if the number of closed sets in which Ω and Γ differ is small. More precisely, they define an *approximately correct basis* as follows.

Definition 9.16 [20] Let M be a finite set and let $\mathbb{K} = (G, M, I)$ be a formal context. A set of implications Ω is a *approximately correct basis* for $\mathbb{K}$ with accuracy $\varepsilon > 0$ if

$$d(\Omega, \mathbb{K}) := \frac{|\mathrm{cl}(\Omega) \triangle \mathrm{cl}(\mathbb{K})|}{2^{|M|}} < \varepsilon$$

where $\mathrm{cl}(\Omega)$ and $\mathrm{cl}(\mathbb{K})$ are the sets of closed sets of Ω and $\mathbb{K}$, respectively, and $S_1 \triangle S_2$ represents the set symmetric difference of S_1 and S_2. $d(\Omega, \mathbb{K})$ is called the *Horn distance* between Ω and $\mathbb{K}$.

And, from this definition, they build the idea of a PAC basis as follows.

Definition 9.17 [20] Let M be a finite set and let $\mathbb{K} = (G, M, I)$ be a formal context, let $\mathrm{Imp}(M)$ be the set of all possible implications between the elements of M, and let $\mathscr{O} = (\mathscr{W}, \mathscr{E}, Pr)$ be a probability space. A random variable $\Omega : \mathscr{O} \to 2^{\mathrm{Imp}(M)}$ is

called a *probably approximately correct basis* (PAC basis) of $\mathbb{K}$ with accuracy $\varepsilon > 0$ and confidence $\delta > 0$ if $Pr(d(\Omega, \mathbb{K}) > \varepsilon) < \delta$.

The computational efficiency of this approach comes from the fact that it is a modified version of the Horn algorithm [3], making use of *membership* and *equivalence oracles* (playing the role of *domain experts*), which allows to compute PAC bases in polynomial time in size of M, the output Ω, as well as $\frac{1}{\varepsilon}$ and $\frac{1}{\delta}$ (ε and δ are inputs to the algorithm), provided that the invocations of the oracles are counted as single steps [20]. In this work, the authors test the usability of PAC bases in real-world situations, comparing several measures of the practical quality of the approximation: the Horn distance between the canonical basis and the approximating bases, and the usual precision and recall measures, defined as follows:

Definition 9.18 [20] Let M be a finite set, let $\mathbb{K} = (G, M, I)$ be a formal context and let Γ be the canonical basis of $\mathbb{K}$. The *precision* and *recall* of a basis Ω, are defined as:

$$\text{prec}(\mathbb{K}, \Omega) = \frac{|\{(A \to B) \in \Omega : \Gamma \models A \to B\}|}{|\Omega|}$$

$$\text{recall}(\mathbb{K}, \Omega) = \frac{|\{(A \to B) \in \Gamma : \Omega \models A \to B\}|}{|\Gamma|}$$

From this definition, *precision* measures the fraction of valid implications in the approximating basis, and *recall* measures the fraction of valid implications in the canonical basis Γ that follow semantically from the approximating basis Ω.

It has been found [20] that increasing the value of ε in the algorithm always leads to a considerable increase in the Horn distance, meaning that the PAC basis deviates more and more from the canonical basis. In addition, the theoretical upper bound ε for the Horn distance between Ω and the canonical basis is never realized in their experiments, meaning that the obtained PAC basis is indeed much closer to the canonical basis than what the algorithm is theoretically designed to obtain. Lastly, for small values of ε (the choice of δ seems to have little impact in the experimental results), both precision and recall are very high, i.e., close to one, what means that the algorithm is able to retrieve most of the canonical basis, and that most of the implications of Ω are valid.

The important result that PAC bases can be computed in output-polynomial time opens the way to decrease the long running times of the algorithms to compute *exact* implication bases. Thus, this is a promising line in the integration of FCA techniques into Big Data situations, as the applicability of the former to larger datasets become feasible.

9.6 Summary and Possible Future Trends

As stated above, there are important reasons which suggest the convenience of further developing logical methods for FCA if we are targeting big datasets: the main issues here are the explainability and interpretability, which are inherent to logic-based methods but are missing in the usual machine learning tools for Big Data.

In this work, we have surveyed some trends in FCA that could be of potential application in Big Data settings, many of them are focused on alleviating the high computational cost of current methods to build the concept lattice, to find a basis of implications and to reason with it.

First, we have considered some techniques to remove redundant information in a formal context, such as clarification and reduction, or the search for reducts through the computation of the discernibility matrix. The aim of these techniques is to build a simpler formal context, with exactly the same closure space as the original one, where computations are less expensive since the number of objects or attributes is reduced. The techniques based on the discernibility matrix are, in the best case, equivalent to clarification and reduction, but with a complexity that is exponential in the worst case. Thus, it depends on the particular application which of the two approaches is more suitable.

Other techniques, based on matrix factorization methods, study the problem of simplifying the structure of the formal context, preserving most of the information, but not all.

The selection of relevant attributes (and concepts) is another open problem, since the measure of relevance is application-dependent. In this sense, the Titanic algorithm is used to build *iceberg* lattices which consists only of concepts with support above a predefined threshold. Since it is more computationally efficient than computing the whole concept lattice, it is potentially applicable to real-world Big Data problems.

With respect to the ability to reason and make inference using implication systems, there is a blocking issue which makes the use of logic tools difficult in Big Data. The presentation of an implication system, together with the axiomatization of the logic (more precisely, the transitivity axiom), makes the computation of closures in Big Data an unsolved issue. We have collected some mechanisms which could potentially help reduce the computational overhead of using logic in Big Data. First, the Simplification Logic can be used to remove atomic redundancies in implication systems. The axiomatization of the Simplification Logic removes the need of the transitivity axiom, thus providing a mechanism to get simpler implication systems which could be traversed in a more efficient manner.

In this direction, other types of implication bases are defined. Particularly, *direct*, *ordered-direct* and D- bases are implication bases, equivalent to the canonical basis, which only require a single pass over the implication set to compute a closure. Since the computation of one of these implication bases can be made *offline* in practical Big Data scenarios, an adequate presentation of the implication system which allows to perform the calculation of closures with enough speed could be used in practical real-world situations.

Regarding the presentation of the knowledge, a compact representation of the closure system of a Big Data formal context can be useful. In this line, minimal generators are at the core of the closure space, and they have been used (e.g. in the Titanic algorithm) to compute simplified versions of the concept lattice, and to rebuild all information that can be inferred from the context.

The last line that has been explored in this work is the computation of *approximately correct* implication bases. Since in practical situations it may be neither advisable nor useful to compute a complete basis of *exact* implications, due either to the prohibiting computational cost or to the noise that might be present in data, it is more pragmatic to capture just *most* of the implication theory. This is the idea behind *approximately correct* bases, which tend to cover most of the closure space of the canonical basis, with valid implications.

This idea has been extended to present *probably approximately correct* bases, which are approximately correct *with high probability*. In this approach, the main strength is that, by using *oracles* in the role of *domain experts*, this kind of bases can be computed in polynomial time. Also, experimentally, PAC bases have presented a good practical quality, in terms of its similarity to the canonical basis.

Certainly, the computational complexity of many problems easily exceeds the tractability threshold (i.e. makes FCA unusable in real-world Big Data problems), hence it makes no sense to expect a complete logic-driven FCA approach; but, what about trying to hybridize techniques? One could wonder to develop, firstly, formal and logic-driven methods up to certain level and, then, applying machine learning techniques, making use of alternative data structures and parallelization techniques. It could be worth to further push the research line of using neural networks to implement the closure operators directly from the context.

Continuing with this line, further development of reduction methods for the contexts is essential, and here, one could consider, for instance, approaching the dimensionality reduction via principal components analysis (PCA) in terms of fuzzy computing instead of numerical PCA. Since PCA is a space decomposition technique, it may be interesting to study how it could be applied to decompose a closure space or a formal context to reduce the computational overhead of present methods. In this case, it is worth remarking that the precise computation of a *fuzzy* concept lattice is more complex than in the crisp case but, following the analogy with fuzzy control, which has proven to be successful to handle very complex systems, it could be interesting to approach FCA by computing in terms of linguistic variables.

Acknowledgements This work has been partially supported by the Spanish Ministry of Science, Innovation, and Universities (MCIU), the State Agency of Research (AEI), the Junta de Andalucía (JA), the Universidad de Málaga (UMA), and the European Social Fund (FEDER) through the research projects with reference PGC2018-095869-B-I00 (MCIU/AEI/FEDER, UE) and UMA2018-FEDERJA-001 (JA/UMA/FEDER, UE)

References

1. Adaricheva, K.V., Nation, J.B., Rand, R.: Ordered direct implicational basis of a finite closure system. Discrete Applied Mathematics **161**(6), 707–723 (2013)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: VLDB, pp. 487–499. Morgan Kaufmann (1994)
3. Angluin, D.: Queries and concept learning. Mach. Learn. **2**(4), 319–342 (1987)
4. Armstrong, W.W.: Dependency structures of data base relationships. In: J.L. Rosenfeld (ed.) Information Processing, Proceedings of the 6th IFIP Congress 1974, Stockholm, Sweden, August 5–10, 1974., pp. 580–583. North-Holland (1974)
5. Atzeni, P., Antonellis, V.D.: Relational Database Theory. Benjamin/Cummings (1993)
6. Babin, M.A.: Models, methods, and programs for generating relationships from a lattice of closed sets. Ph.D. thesis, Higher School of Economics, Moscow (2012)
7. Babin, M.A., Kuznetsov, S.O.: Computing premises of a minimal cover of functional dependencies is intractable. Discrete Applied Mathematics **161**(6), 742–749 (2013)
8. Bastide, Y., Pasquier, N., Taouil, R., Stumme, G., Lakhal, L.: Mining minimal non-redundant association rules using frequent closed itemsets. In: Computational Logic, *Lecture Notes in Computer Science*, vol. 1861, pp. 972–986. Springer (2000)
9. Belohlavek, R., Cordero, P., Enciso, M., Mora, Á., Vychodil, V.: Automated prover for attribute dependencies in data with grades. International Journal of Approximate Reasoning **70**, 51–67 (2016)
10. Belohlávek, R., Macko, J.: Selecting important concepts using weights. In: ICFCA, *Lecture Notes in Computer Science*, vol. 6628, pp. 65–80. Springer (2011)
11. Belohlávek, R., Outrata, J., Trnecka, M.: Factorizing boolean matrices using formal concepts and iterative usage of essential entries. Inf. Sci. **489**, 37–49 (2019)
12. Belohlávek, R., Vychodil, V.: Formal concept analysis with constraints by closure operators. In: ICCS, *Lecture Notes in Computer Science*, vol. 4068, pp. 131–143. Springer (2006)
13. Belohlávek, R., Vychodil, V.: Discovery of optimal factors in binary data via a novel method of matrix decomposition. J. Comput. Syst. Sci. **76**(1), 3–20 (2010)
14. Belohlávek, R., Vychodil, V.: Closure-based constraints in formal concept analysis. Discrete Applied Mathematics **161**(13–14), 1894–1911 (2013)
15. Benítez-Caballero, M.J., Medina, J., Ramírez-Poussa, E.: Attribute reduction in rough set theory and formal concept analysis. In: IJCRS (2), *Lecture Notes in Computer Science*, vol. 10314, pp. 513–525. Springer (2017)
16. Benito-Picazo, F., Cordero, P., Enciso, M., Mora, A.: Minimal generators, an affordable approach by means of massive computation. The Journal of Supercomputing **75**(3), 1350–1367 (2019)
17. Bertet, K., Monjardet, B.: The multiple facets of the canonical direct unit implicational basis. Theoretical Computer Science **411**(22–24), 2155–2166 (2010)
18. Bertet, K., Nebut, M.: Efficient algorithms on the moore family associated to an implicational system. Discrete Mathematics and Theoretical Computer Science **6**(315–338), 107 (2004)
19. Borchmann, D.: Learning terminological knowledge with high confidence from erroneous data. Ph.D. thesis, Saechsische Landesbibliothek-Staats-und Universitaetsbibliothek Dresden (2014)
20. Borchmann, D., Hanika, T., Obiedkov, S.: On the usability of probably approximately correct implication bases. In: ICFCA, *Lecture Notes in Computer Science*, vol. 10308, pp. 72–88. Springer (2017)
21. Boulicaut, J., Bykowski, A., Rigotti, C.: Free-sets: A condensed representation of boolean data for the approximation of frequency queries. Data Min. Knowl. Discov. **7**(1), 5–22 (2003)

22. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., Elhadad, N.: Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1721–1730. ACM (2015)
23. Cheung, K.S.K., Vogel, D.R.: Complexity reduction in lattice-based information retrieval. Inf. Retr. **8**(2), 285–299 (2005)
24. Codocedo, V., Taramasco, C., Astudillo, H.: Cheating to achieve formal concept analysis over a large formal context. In: CLA, *CEUR Workshop Proceedings*, vol. 959, pp. 349–362. CEUR-WS.org (2011)
25. Cordero, P., Enciso, M., López, D., Mora, A.: A conversational recommender system for diagnosis using fuzzy rules. Expert Systems with Applications p. 113449 (2020)
26. Cordero, P., Enciso, M., Mora, A.: Directness in fuzzy formal concept analysis. In: Proc. of the 17th Intl Conf on Information Processing and Management of Uncertainty in Knowledge-Bases IPMU'18, *Communications in Computer and Information Science*, vol. 853, pp. 585–595 (2018)
27. Cordero, P., Enciso, M., Mora, A., de Guzmán, I.P.: $\mathrm{SL}_{\mathrm{FD}}$ logic: Elimination of data redundancy in knowledge representation. In: Ibero-American Conference on Artificial Intelligence, pp. 141–150. Springer (2002)
28. Cordero, P., Enciso, M., Mora, A., Ojeda-Aciego, M.: Computing minimal generators from implications: a logic-guided approach. In: CLA, vol. 2012, pp. 187–198. Citeseer (2012)
29. Cordero, P., Enciso, M., Mora, A., Ojeda-Aciego, M.: Computing left-minimal direct basis of implications. In: CLA, vol. 2013, pp. 293–298 (2013)
30. Cordero, P., Enciso, M., Mora, A., Ojeda-Aciego, M., Rossi, C.: Knowledge discovery in social networks by using a logic-based treatment of implications. Knowl.-Based Syst. **87**, 16–25 (2015)
31. Dias, S.M., Vieira, N.J.: Concept lattices reduction: Definition, analysis and classification. Expert Syst. Appl. **42**(20), 7084–7097 (2015)
32. Dias, S.M., Z'arate, L.E., Vieira, N.J.: Using iceberg concept lattices and implications rules to extract knowledge from ANN. Intelligent Automation & Soft Computing **19**(3), 361–372 (2013)
33. Distel, F.: Hardness of enumerating pseudo-intents in the lectic order. In: International Conference on Formal Concept Analysis, pp. 124–137. Springer (2010)
34. Distel, F., Sertkaya, B.: On the complexity of enumerating pseudo-intents. Discrete Applied Mathematics **159**(6), 450–466 (2011)
35. Dong, G., Jiang, C., Pei, J., Li, J., Wong, L.: Mining succinct systems of minimal generators of formal concepts. In: DASFAA, *Lecture Notes in Computer Science*, vol. 3453, pp. 175–187. Springer (2005)
36. Dumbill, E.: What is big data? an introduction to the big data landscape. Strata 2012: Making Data Work (2012)
37. Gajdos, P., Moravec, P., Snásel, V.: Concept lattice generation by singular value decomposition. In: CLA, *CEUR Workshop Proceedings*, vol. 110. CEUR-WS.org (2004)
38. Ganter, B., Wille, R.: Formal Concept Analysis - Mathematical Foundations. Springer (1999)
39. Geerts, F., Missier, P., Paton, N.W.: Editorial: Special issue on improving the veracity and value of big data. J. Data and Information Quality **9**(3), 13:1–13:2 (2018)
40. Grünwald, P.D., Grunwald, A.: The minimum description length principle. MIT press (2007)
41. Guigues, J.L., Duquenne, V.: Familles minimales d'implications informatives résultant d'un tableau de données binaires. Mathématiques et Sciences humaines **95**, 5–18 (1986)
42. Hanika, T., Koyda, M., Stumme, G.: Relevant attributes in formal contexts. In: ICCS, *Lecture Notes in Computer Science*, vol. 11530, pp. 102–116. Springer (2019)
43. Ibaraki, T., Kogan, A., Makino, K.: Inferring minimal functional dependencies in horn and q-horn theories. Ann. Math. Artif. Intell. **38**(4), 233–255 (2003)

44. Kauer, M., Krupka, M.: Removing an incidence from a formal context. In: CLA, *CEUR Workshop Proceedings*, vol. 1252, pp. 195–206. CEUR-WS.org (2014)
45. Khardon, R.: Translating between horn representations and their characteristic models. Journal of Artificial Intelligence Research **3**, 349–372 (1995)
46. Kim, B.: Interactive and interpretable machine learning models for human machine collaboration. Ph.D. thesis, Massachusetts Institute of Technology (2015)
47. Kim, B., Khanna, R., Koyejo, O.O.: Examples are not enough, learn to criticize! criticism for interpretability. In: Advances in Neural Information Processing Systems, pp. 2280–2288 (2016)
48. Konecny, J.: On attribute reduction in concept lattices: Methods based on discernibility matrix are outperformed by basic clarification and reduction. Inf. Sci. **415**, 199–212 (2017)
49. Konecny, J., Krajca, P.: On attribute reduction in concept lattices: The polynomial time discernibility matrix-based method becomes the cr-method. Inf. Sci. **491**, 48–62 (2019)
50. Kumar, C.A., Dias, S.M., Vieira, N.J.: Knowledge reduction in formal contexts using non-negative matrix factorization. Mathematics and Computers in Simulation **109**, 46–63 (2015)
51. Kuznetsov, S.O.: On the intractability of computing the duquenne-guigues basis. J. UCS **10**(8), 927–933 (2004)
52. Kuznetsov, S.O., Makhalova, T.P.: On interestingness measures of formal concepts. Inf. Sci. **442–443**, 202–219 (2018)
53. Lorenzo, E.R., Cordero, P., Enciso, M., Bonilla, A.M.: Canonical dichotomous direct bases. Inf. Sci. **376**, 39–53 (2017)
54. Maier, D.: The Theory of Relational Databases. Computer Science Press (1983)
55. Medina, J.: Relating attribute reduction in formal, object-oriented and property-oriented concept lattices. Computers & Mathematics with Applications **64**(6), 1992–2002 (2012)
56. Miller, T.: Explanation in artificial intelligence: Insights from the social sciences. Artificial Intelligence **267**, 1–38 (2019)
57. Missaoui, R., Nourine, L., Renaud, Y.: An inference system for exhaustive generation of mixed and purely negative implications from purely positive ones. In: M. Kryszkiewicz, S.A. Obiedkov (eds.) Proceedings of the 7th International Conference on Concept Lattices and Their Applications, Sevilla, Spain, October 19–21, 2010, *CEUR Workshop Proceedings*, vol. 672, pp. 271–282. CEUR-WS.org (2010)
58. Missaoui, R., Nourine, L., Renaud, Y.: Computing implications with negation from a formal context. Fundam. Inform. **115**(4), 357–375 (2012)
59. Mora, A., Cordero, P., Enciso, M., Fortes, I., Aguilera, G.: Closure via functional dependence simplification. International Journal of Computer Mathematics **89**(4), 510–526 (2012)
60. Mora, A., Enciso, M., Cordero, P., de Guzmán, I.P.: An efficient preprocessing transformation for functional dependencies sets based on the substitution paradigm. In: Conference on Technology Transfer, pp. 136–146. Springer (2003)
61. Nishio, N., Mutoh, A., Inuzuka, N.: On computing minimal generators in multi-relational data mining with respect to θ-subsumption. In: ILP (Late Breaking Papers), *CEUR Workshop Proceedings*, vol. 975, pp. 50–55. CEUR-WS.org (2012)
62. Osicka, P., Trnecka, M.: Boolean matrix decomposition by formal concept sampling. In: CIKM, pp. 2243–2246. ACM (2017)
63. Pawlak, Z.: Rough sets. Int. J. Parallel Program. **11**(5), 341–356 (1982)
64. Pfaltz, J.L.: Incremental transformation of lattices: A key to effective knowledge discovery. In: ICGT, *Lecture Notes in Computer Science*, vol. 2505, pp. 351–362. Springer (2002)
65. Polanyi, M.: The Tacit Dimension. Doubleday, Garden City, NY (1966)
66. Portugal, I., Alencar, P.S.C., Cowan, D.D.: The use of machine learning algorithms in recommender systems: A systematic review. Expert Syst. Appl. **97**, 205–227 (2018)
67. Qi, J.J.: Attribute reduction in formal contexts based on a new discernibility matrix. Journal of applied mathematics and computing **30**(1–2), 305–314 (2009)

68. Rodríguez-Lorenzo, E., Adaricheva, K.V., Cordero, P., Enciso, M., Mora, A.: Formation of the d-basis from implicational systems using simplification logic. Int. J. General Systems **46**(5), 547–568 (2017)
69. Rodríguez-Lorenzo, E., Bertet, K., Cordero, P., Enciso, M., Mora, Á.: Direct-optimal basis computation by means of the fusion of simplification rules. Discrete Applied Mathematics **249**, 106–119 (2018)
70. Rodríguez-Lorenzo, E., Bertet, K., Cordero, P., Enciso, M., Mora, A., Ojeda-Aciego, M.: From implicational systems to direct-optimal bases: A logic-based approach. Applied Mathematics & Information Sciences. L **2**, 305–317 (2015)
71. Rudolph, S.: Using FCA for encoding closure operators into neural networks. Lecture Notes in Computer Science **4604**, 321–332 (2007)
72. Stumme, G., Taouil, R., Bastide, Y., Pasquier, N., Lakhal, L.: Computing iceberg concept lattices with Titanic. Data Knowl. Eng. **42**(2), 189–222 (2002)
73. Valverde-Albacete, F.J., Peláez-Moreno, C., Cordero, P., Ojeda-Aciego, M.: Formal equivalence analysis. In: 2019 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (EUSFLAT 2019). Atlantis Press (2019)
74. Wei, W., Wu, X., Liang, J., Cui, J., Sun, Y.: Discernibility matrix based incremental attribute reduction for dynamic data. Knowl.-Based Syst. **140**, 142–157 (2018)
75. Zhang, S., Guo, P., Zhang, J., Wang, X., Pedrycz, W.: A completeness analysis of frequent weighted concept lattices and their algebraic properties. Data Knowl. Eng. **81–82**, 104–117 (2012)
76. Zhang, W., Wei, L., Qi, J.: Attribute reduction theory and approach to concept lattice. Science in China Series F: Information Sciences **48**(6), 713–726 (2005)