HAL open science

Relational Concept Analysis: Mining Concept Lattices From Multi-Relational Data

Amine Mohamed Rouane Hacene, Marianne Huchard, Amedeo Napoli, Petko Valtchev

To cite this version:

Amine Mohamed Rouane Hacene, Marianne Huchard, Amedeo Napoli, Petko Valtchev. Relational Concept Analysis: Mining Concept Lattices From Multi-Relational Data. Annals of Mathematics and Artificial Intelligence, 2013, 67 (1), pp.81-108. 10.1007/s10472-012-9329-3 . lirmm-00816300

HAL Id: lirmm-00816300

https://hal-lirmm.ccsd.cnrs.fr/lirmm-00816300v1

Submitted on 22 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Noname manuscript No.
(will be inserted by the editor)

Relational Concept Analysis: Mining Concept Lattices From Multi-Relational Data

Mohamed Rouane-Hacene · Marianne Huchard · Amedeo Napoli · Petko Valtchev

Received: date / Accepted: date

Abstract The processing of complex data is admittedly among the major concerns of knowledge discovery from data (KDD). Indeed, a major part of the data worth analyzing is stored in relational databases and, since recently, on the Web of Data. This clearly underscores the need for Entity-Relationship and RDF compliant data mining (DM) tools. We are studying an approach to the underlying *multi-relational data mining* (MRDM) problem, which relies on formal concept analysis (FCA) as a framework for clustering and classification. Our *relational concept analysis* (RCA) extends FCA to the processing of multi-relational datasets, i.e., with multiple sorts of individuals, each provided with its own set of attributes, and relationships among those. Given such a dataset, RCA constructs a set of concept lattices, one per object sort, through an iterative analysis process that is bound towards a fixed-point. In doing that, it abstracts the links between objects into attributes akin to role restrictions from description logics (DLs). We address here key aspects of the iterative calculation such as evolution in data description along the iterations and process termination. We describe implementations of RCA and list applications to problems from software and knowledge engineering.

Keywords Formal concept analysis · relational data · relational concept analysis · concept lattices · knowledge representation · description logics

M. Rouane-Hacene
Dépt. Informatique, UQÀM, CP 8888, succ. CV, Montréal, Canada, H3C 3P8
E-mail: rouanehm@gmail.com

M. Huchard
LIRMM (CNRS – Université de Montpellier), 161 rue Ada, 34095 Montpellier Cedex 5 - France
E-mail: huchard@lirmm.fr

A. Napoli
LORIA (CNRS – INRIA – Université de Lorraine), B.P. 239, F-54506 Vandœuvre-lès-Nancy, France
E-mail: Amedeo.Napoli@loria.fr

P. Valtchev
Dépt. Informatique, UQÀM, CP 8888, succ. CV, Montréal, Canada, H3C 3P8
E-mail: valtchev.petko@uqam.ca

Mathematics Subject Classification (2000) MSC 06-A99 · 06-B99 · 68-R99

1 Introduction

Knowledge discovery from data (KDD) is the process of distilling useful facts from a dataset. A KDD task is a multi-step process in which, beside the central step of data mining (DM) which focuses on the mechanics of pure discovery, also covers the prior data preparation as well as the subsequent interpretation of mining results [13]. As a scientific discipline, primary concerns in DM are speed and scalability of methods and the versatility of the practical tools, then proper processing of structure in data, reflecting the available domain knowledge in the mining process, etc. Furthermore, KDD addresses the intelligibility of discovered patterns and integration of newly discovered knowledge with existing sources.

Among the current research topics in DM, the work presented here pertains to two major trends: (1) processing of data of ever increasing structural complexity and (2) exploration of domain knowledge, in particular, meta-data about the semantics of the data, to achieve more topical and precise analysis. For instance, alternatives to introduce structure into the basic DM data format – transactions with descriptors that are valued attributes or set of atomic items – have been made. On one hand, various topological structures on the set of items, e.g., sequences [2] or general graphs [36, 9], have been examined with the corresponding pattern languages. On the other hand, descriptions of patterns based on first-order logic (FOL), i.e., with variables for items, predicates and quantifiers, have been studied within the *relational learning and data mining* [12, 26] trend. Variants thereof comprise Datalog [10], description logics (DLs) [19], and even datasets corresponding to entire relational databases [11]. The latter approach, called *multi-relational data mining* (MRDM), is particularly appealing since a large proportion of the data worth analyzing is nowadays stored in relational databases. Recently, such data has been gradually pumped into the Web, where various datasets are published and interlinked with other published data using a graph-based format, the Resource Description Framework (RDF), and the encompassing Linked Data technology[1].

Domain knowledge, in turn, has typically been incorporated in the mining process as a taxonomy on items, thus giving rise to *generalized patterns* [32]. Further popular formats for expressing such knowledge are sets of logical rules and ground facts [26] (ch. 5), or, alternatively, a fully-blown domain ontology [1]. It is noteworthy that the availability of taxonomies or richer ontologies is independent to the presence of structure in the data. Within a multi-relational dataset, an important part of the background knowledge is the underlying data schema. Moreover, when mining the individuals of a particular relation, the links to individuals of other relations provide further knowledge about the context of the mined data.

Recently, we designed a MRDM framework, on top of the standard formal concept analysis (FCA) [15]. In FCA, the input data, a $[individuals \times attributes]$ cross-table or *context*, is transformed into a hierarchy of *all* conceptual abstractions combining

[1] www.w3.org/standards/semanticweb/data

sets of individuals with the sets of shared attributes. Through the unique structure of its output, FCA simultaneously addresses two DM tasks: (conceptual) clustering and pattern mining. Various attempts have been made at extending the FCA to data of complex structure: In [17, 20], the formal objects are graphs, whereas in [29, 14] they are described by logical formulae.

In our own approach, called *relational concept analysis* (RCA) [27], we focus on dataset compatible with the Entity-Relationship model or, alternatively, with the Resource Description Framework (RDF). In RCA, the input is made of several contexts, each corresponding to a specific type of individuals. Moreover, additional cross-tables describe $[individuals \times individuals]$ binary relations. Our analysis method extends the basic FCA mechanisms in many respects starting with the processing of multiple sorts of individuals. In doing that, the links between individuals are *propositionalized* – in a way inspired by DLs roles – and end up describing the output concepts much like the initially available attributes. Thus, new attributes are gradually inferred from links at several levels of depth. The resulting iterative analysis process deals smoothly with the circular links in the input data, a known source of difficulties for DM methods. In summary, RCA is an original framework for extracting conceptual knowledge from multi-relational data which successfully addresses problems such as dealing with arbitrary relations (incl. circular ones), effective construction and maintenance of the concept lattices (Hasse diagrams thereof) and intelligible representation of the extracted knowledge.

This paper extends previous work [16] with emphasis put on theoretical and algorithmic aspects of RCA including the dynamics of contexts, the concept formation principles, and the properties of the overall analysis process. It thus provides formal definitions for scaling operators that help propositionalize the initial relational descriptions of the formal objects. The iterative analysis process is then analytically expressed and its convergence demonstrated.

The remainder of the paper is organized as follows. First we recall basic notions from FCA (section 2). Then the basic RCA framework is described in section 3, whereas the iterative analysis process and the MULTI-FCA method are introduced in section 4. Related work is discussed in section 5 while a conclusion with future work ends the paper.

2 Formal concept analysis

FCA provides a framework for designing concepts and conceptual hierarchies related to a context [15]. Below we recall the fundamental notions of FCA and pinpoint some difficulties in processing relations in data.

2.1 Motivating example

Throughout the paper, we use a pharmacovigilance dataset as a running example. Pharmacovigilance as a discipline studies the adverse reactions to drugs (ADR) by analysing data collected by reporting systems and stored in case report databases [21].

A pharmacovigilance case database contains a collection of ADR reports, also known as *case reports*. Generally, a well-documented case report captures suspected and concomitant product therapy details, description of the ADR or disease experience, including set of signs or symptoms. Further information covers the patient characteristics, including demographic information (e.g., age, race, gender), relevant family history of diseases, baseline medical condition prior to product therapy, etc. Table 1 depicts a fragment of a hypothetical case report database representing AIDS patients and the anti-HIV drugs they have taken.

Patient	Age	Gender	Observed adverse drug reactions
Farley	63	Male	Oedema, Hives, Headache, Nausea, Heart failure, Hair loss
Lane	27	Female	Fatigue, Oedema, Hives, Hair loss, Bleeding
Shana	33	Female	Fatigue, Oedema, Hair loss
Trudy	41	Male	Fatigue, Breath disorder, Nausea, Heart failure, Bleeding, Vomiting

Table 1 A fragment of a case report database in pharmacovigilance showing AIDS patients with the observed adverse reactions that appeared after taking anti-HIV drugs.

Healthcare providers involved in the treatment of HIV patients examine the response to therapy in different categories of patients in order to successfully devise a new therapy. An analysis of the above table may yield combinations of several drugs with a single ADR, called *drug interactions*. *Drug protocols*, i.e., concomitant use of drugs that may cause several reactions, can be discovered as well. In our example, the known ADR of anti-HIV drugs are provided in Table 2 below. For instance, antiretroviral Dactinomycin and Isentress are known to cause {Breathdisorder, Fatigue, Hairloss} and {Diarrhoea, Headache, Nausea}, respectively. In the following section, we introduce FCA as a method for analyzing such data and drugs interactions.

2.2 FCA basics

FCA is a mathematical framework for designing conceptual descriptions from a set of individuals described by unary attributes (similar to unary predicates). Formally, a *formal one-valued context* $\mathcal{K}$ associates a set of objects (O) to a set of attributes (A) through an incidence relation $I \subseteq O \times A$. A sample context[2], called $\mathcal{K}_P$, is depicted on Table 2 where formal objects are anti-HIV drugs, and attributes their expected ADR such as Nausea, Headache, Hairloss, Rash, Liverdamage, etc. Further attributes represent the respective active agents behind the drugs.

A pair of derivation operators, both denoted $'$, are applied to objects and attributes. On objects, the operator $'$ is defined on $\wp(O) \rightarrow \wp(A)$ such as for $X \in \wp(O), X' = \{a \in A \mid oIa \text{ for all } o \in X\}$. On attributes, the operator $'$ is defined on $\wp(A) \rightarrow \wp(O)$ such as for $Y \in \wp(A), Y' = \{o \in O \mid oIa \text{ for all } a \in Y\}$. For instance, in the context

[2] Adapted from MEDEFFECT, the dataset of Canada pharmacovigilance database.

	Actinomycin	Efavirenz	Maraviroc	Raltegravir	Ritonavir	Tenofovir	Breathdisorder	Diarrhea	Fatigue	Hairloss	Headache	HeartFailure	LiverDamage	Nausea	Rash	Vomiting
Dactinomycin	×						×		×	×						
Isentress				×				×			×			×		
Kaletra					×			×	×		×		×	×	×	×
Selzentry			×									×			×	
Sustiva		×						×	×						×	×
Viread						×		×					×	×		×

Table 2 Context $\mathcal{K}_D$ of anti-HIV drugs involved into HAART therapy with the associated expected ADR.

of drugs $\mathcal{K}_D$ depicted on Table 2, $\{\texttt{Sustiva}, \texttt{Viread}\}' = \{\texttt{Diarrhea}, \texttt{Vomiting}\}$ and $\{\texttt{Nausea}, \texttt{Headache}\}' = \{\texttt{Isentress}, \texttt{Kaletra}\}$.

The $'$ operators induce a Galois connection [6] between $\wp(O)$ and $\wp(A)$, whence the compound operators $''$ are closure operators over $\wp(O)$ and $\wp(A)$. For instance, in $\mathcal{K}_D$ depicted on Table 2 $\{\texttt{HeartFailure}\}'' = \{\texttt{HeartFailure}, \texttt{Maraviroc}, \texttt{Rash}\}$ whereas $\{\texttt{Sustiva}, \texttt{Viread}\}'' = \{\texttt{Sustiva}, \texttt{Viread}, \texttt{Kaletra}\}$. Thus, $'$ operators induce two closure families $\mathcal{C}^o \subseteq \wp(O)$ and $\mathcal{C}^a \subseteq \wp(A)$ which, provided with set-inclusion order, form two complete lattices which are dually-isomorphic.

A pair (X, Y) of sets where $X \subseteq O$, $Y \subseteq A$, $X = Y'$, and $Y = X'$, is called a *(formal) concept* [37] with X is called *extent* and Y is called *intent*. For example, in the drug context $\mathcal{K}_D$, $(\{\texttt{Sustiva}, \texttt{Kaletra}\}, \{\texttt{Diarrhea}, \texttt{Fatigue}, \texttt{Rash}, \texttt{Vomiting}\})$ is a formal concept. The set $\mathcal{C}_\mathcal{K}$ of all concepts from $\mathcal{K}$ ordered by extent inclusion forms a complete lattice, $\mathcal{L}_\mathcal{K} = \langle \mathcal{C}_\mathcal{K}, \leq_\mathcal{K} \rangle$, which is termed the *concept lattice* [37] of the context (or the *Galois lattice* [6] of the binary relation I).

The lattice $\mathcal{L}_\mathcal{K}$ is represented by a Hasse diagram where nodes represent the concepts provided with the respective intent and extent, while segments represent the specialization relation (or subsumption) between concepts. It is noteworthy that the lattice represents in an exhaustive manner the structure sharing among objects of the context: two objects "join" each other in a non-trivial concept extent iff they share at least one attribute.

A labeling mode of the diagram known as *simplified* or *reduced* labeling is used, where each attribute (resp. object) appears once, in the maximal (resp. minimal) concept –in term of size of extent– having that attribute (resp. object). The concept lattice $\mathcal{L}_D$ of context $\mathcal{K}_D$ is drawn in Figure 1. The full extent (resp. intent) of a concept is the union of all objects (resp. attributes) whose labels are reachable starting from this concept and by taking downward paths (resp. upward). Thus, the concepts represent all maximum factorizations of the attributes between objects. For example, the formal concept $c_{\#5}$ represents the NRTI drug class grouping `Kaletra` and `Viread` that are known to cause severe liver problems, nausea, diarrhea and vomiting.

As many practical applications involve richer data descriptions, *many-valued contexts* were introduced. In such a context $\mathcal{K} = (O, A, V, J)$, an object o is described

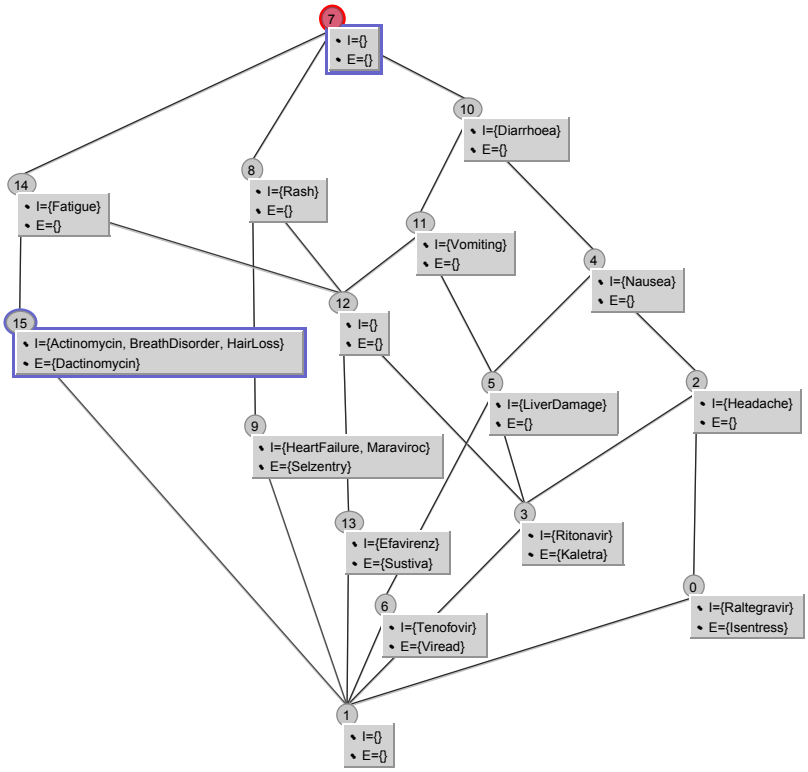

Fig. 1 Concept lattice $\mathcal{L}_D$ of the drug context given in Table 2. The reduced intents and extents are drawn on both sides of a node representing a concept.

by a set of attribute value pairs (a, v) where $v \in V$. Thus, J is a ternary relation $J \subseteq O \times A \times V$. Table 1 illustrates a many-valued context, called $\mathcal{K}_P$, encoding a set of AIDS patients and their features such as age, gender, and adverse reactions to anti-HIV drugs.

Before being processed using standard FCA algorithms, a many-valued context has to be transformed into a one-valued context, a transformation known as *scaling* [15]. Basically, the scaling encodes a many-valued attribute as a set of predicates ranging on attribute values. For example, the "`Age`" attribute, whose values are numerical, can be substituted by the attributes `Adult` (predicate `Age` $\leq$ `60`, ordinal scaling in [15]) and `Senior` (predicate `Age` $>$ `60`). Hence, the object `Farley` is assigned `Senior` while other objects are assigned `Adult`, as illustrated in Table 3. Similarly, `Gender` in Table 1 is *nominally* scaled with predicates `Female` and `Male`. The predicates on a many-valued attribute compose a *scale*. One-valued context $\mathcal{K}_P$ showing the scaling on patients is illustrated in Table 3 while the corresponding lattice is depicted in Figure 2. For example, the formal concept $c_{\#9}$ = ({`Farley`, `Trudy`}, {`Male`, `Nausea`, `HeartFailure`}) represents male patients having nausea and heart disorder.

	Senior	Adult	Male	Female	Bleeding	Breathdisorder	Fatigue	Hairloss	Headache	HeartFailure	Hives	Nausea	Oedema	Vomiting
Farley	×		×					×	×	×	×	×	×	
Lane		×		×	×		×	×			×		×	
Shana		×		×			×	×					×	
Trudy		×	×		×	×	×			×		×		×

Table 3 One-valued context $\mathcal{K}_P$ encoding AIDS patients with the observed ADR (discrepancies with Table 1 persist due to the small size of the patient sample).

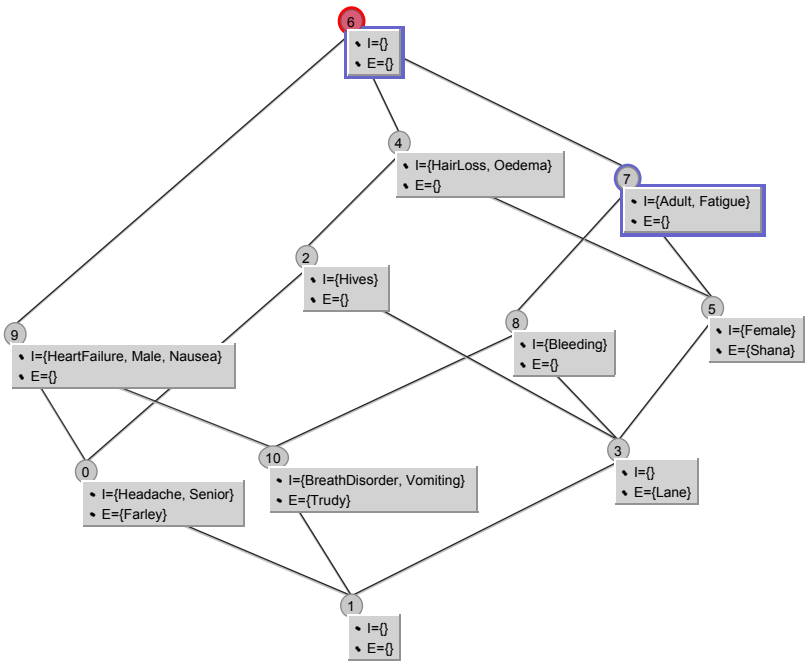

Fig. 2 Concept lattice $\mathcal{L}_P$ of context of AIDS patients given in Table 3.

3 Relational Concept Analysis

RCA was designed for the construction of formal concepts on top of multiple object sets described by both proper attributes and relations [16, 27]. Below we recall the formal background of RCA, including representational and analytical aspects.

3.1 Bringing relations to FCA

The ever increasing complexity of data to be analyzed has rapidly pushed the standard context-based conceptual methods to their limits. As an illustration, consider the lattices in Figure 1 and Figure 2 which represent all possible attribute sharing between objects in the respective contexts. They provide useful insights on what might be the classes of anti-HIV drugs and profiles of AIDS patients. Yet an integrated approach to the management of both populations of patients and anti-HIV drugs, i.e., for better targeted therapies, requires that profiles of patients are linked to the appropriate classes of drugs. Conversely, the drug class is more useful when it reflects the observed ADR depending on the profile of the target patients.

The mutual dependency between patients and anti-HIV drugs, triggers the introduction of links between the corresponding objects into the analysis process. This specific type of data, known as *relational* data as it connects objects among themselves, is not accessible for mainstream concept analysis.

Indeed, a thorough processing of relations between different sorts of objects challenges several core aspects of the FCA task, among them the use of a unique context, the static set of attributes in a context, the one-shot lattice construction process, etc. These limitations of core FCA motivate the design of specific methodologies for concept analysis of relational data. As an illustration of the target knowledge structures, consider the analysis of the pharmacovigilance dataset illustrated in Table 1 and Table 2. Using a tool that properly processes relational links between data items, the following remarkable fact could be gleaned: The combination of `Dactinomycin` and `Isentress` causes several **new** ADR that need to be investigated such as `Bleeding`, `Vomiting`, and `HeartFailure`, although both drugs share no direct ADR in Table 2. This fact is beyond the reach of existing statistics- and FCA-based analysis methods: Its discovery would require a multi-step process, involving human experts to connect different knowledge sources. With RCA-based analysis, this is a basic building block of the final result that could be explored in the design of new combined medications that cause minimal interactions or adverse reactions.

In what follows, we present our own approach to the above issue, which is called *relational concept analysis* (RCA), and provide some fundamental results on the method and on its computational mechanisms.

3.2 Input data format

In RCA, input data is organized as a pair made of a set of objects-to-attributes contexts $\mathbf{K} = \{\mathcal{K}_i\}_{i=1,\ldots,n}$ and a set of objects-to-objects binary relations $\mathbf{R} = \{r_k\}_{k=1,\ldots,m}$. Here, a relation $r \in \mathbf{R}$ links two object sets from two contexts, i.e., there are $i_1, i_2 \in \{1, \ldots, n\}$ (possibly $i_1 = i_2$) such that $r \subseteq O_{i_1} \times O_{i_2}$. Both contexts from $\mathbf{K}$ and relations from $\mathbf{R}$ are introduced as cross-tables[3]. The resulting structure, called a *relational context family* (RCF) can be likened to a relational database with a schema-like part comprising context, attribute and relation names, and an instance-like part

[3] Although this might be the result of scaling upon some of the attributes.

comprising the objects and the inter-object links that populate contexts and relations, respectively.

Definition 1 (Relational Context Family (RCF))
An RCF *is a pair* $(\mathbf{K}, \mathbf{R})$ *where:*

- $\mathbf{K} = \{\mathcal{K}_i\}_{i=1,\ldots,n}$ *is a set of contexts* $\mathcal{K}_i = (O_i, A_i, I_i)$ *and*
- $\mathbf{R} = \{r_k\}_{k=1,\ldots,m}$ *is a set of relations* r_k *where* $r_k \subseteq O_{i_1} \times O_{i_2}$ *for some* $i_1, i_2 \in \{1, \ldots, n\}$.

It is noteworthy that, in the above definition, all object sets O_i ($i \in \{1, \ldots, n\}$) are pair-wise *disjoint*. Moreover, O_{i_1} (*domain* of r_k) and O_{i_2} (*range* of r_k) are the object sets of the contexts $\mathcal{K}_{i_1}$ and $\mathcal{K}_{i_2}$, respectively.

takes	Dactinomycin	Isentress	Kaletra	Selzentry	Sustiva	Viread
Farley			×			×
Lane				×	×	
Shana			×			×
Trudy	×	×				

interacts with	Dactinomycin	Isentress	Kaletra	Selzentry	Sustiva	Viread
Dactinomycin						
Isentress						
Kaletra				×	×	
Selzentry			×			
Sustiva			×			×
Viread					×	

Table 4 Left: Binary relation `takes` linking AIDS patients to anti-HIV drugs. The relation `is taken by` (henceforth referred to as `itb`) is derived from `takes` ($\texttt{takes}^{-1}$).
Right: Binary relation `interacts with` (`iw`) that models interactions among drugs.

Table 4 depicts the `takes` and `is taken by` (`itb`) relations on patient and drug objects, as well as the relation `interacts with` (`iw`) that models antiretroviral interactions within the drug context. Drug-drug interactions occur when two or more drugs react with each other. This drug-drug interaction may cause patient to experience an unexpected adverse reaction. Relations `takes`, `itb` and `iw` together with the respective contexts $\mathcal{K}_D$ (Table 2) and $\mathcal{K}_P$ (Table 3) form the RCF of our running example.

For practical reasons, we shall process a relation $r \subseteq O_{i_1} \times O_{i_2}$ in the form of a *set-valued function* $r : O_{i_1} \rightarrow \wp(O_{i_2})$. Moreover, we introduce here some auxiliary functions to support relation-centric reasoning. First we define formally the *domain* and the *range* of a function from the RCF.

Definition 2 (The $dom(r)$ and $ran(r)$ functions for relations)
Let $(\mathbf{K}, \mathbf{R})$ *be an* RCF. *A pair of functions map relations in* $\mathbf{R}$ *to respective domain and range object sets from the object set family* $\mathbf{O} = \{O | \mathcal{K} = (O, A, I) \in \mathbf{K}\}$.

- *The domain function is* $dom : \mathbf{R} \rightarrow \mathbf{O}$ *where* $dom(r) = O_{i_1}$ ***iff*** *for all* $(x, y) \in r$, $x \in O_{i_1}$,

– *The range function is* $ran : \mathbf{R} \rightarrow \mathbf{O}$ *where* $ran(r) = O_{i_2}$ ***iff*** *for all* $(x, y) \in r$, $y \in O_{i_2}$.

Next, we provide a way to gather functions w.r.t. their domain, i.e., the context function rel.

Definition 3 (The $rel(\mathcal{K})$ function for contexts)
The family of relations originating at a given context is defined by:

– $rel : \mathbf{K} \rightarrow \wp(\mathbf{R})$ *where* $rel(\mathcal{K} = (O, A, I)) = \{r \in \mathbf{R} \mid dom(r) = O\}$.

Before addressing the key question of how inter-object links can be put into concept intents, some important conventions must be adopted. These are mainly about identifying the elements of an RCF along the analysis process. In fact, unlike core FCA, in RCA some of the manipulated entities evolve. Evolution is brought by the transformation of links into descriptors for formal objects: The sets A_i of attributes in the contexts of a RCF are extended by new members. Hence, strictly speaking, the resulting contexts are not the same constructs as the initial ones, just as a scaled context is not equivalent to its original many-valued counterpart. In contrast, respective object sets O_i remain unchanged, so they provide a basis for identifying the contexts: We consider the extended attribute sets to simply define a new version of the same contexts that share the O_i object sets. Thus, unless an explicit distinction is necessary, all of these versions will be denoted by $\mathcal{K}_i$.

Similarly, concepts $c = (X, Y)$ from the initial contexts of the RCF, are to be opposed to concepts from the respective extended versions of these contexts. As it will become clear in section 3.3.4, there is a significant continuity between the former and the latter, due to a correspondence based on extent preservation. Consequently, the concepts from different versions of a context having the same extent are here considered to be the subsequent versions of the same concept. They are therefore assigned the same identifier in all the respective lattices (see an example in section 3.3.4).

To sum up, we identify the contexts by their respective object sets and concepts by their extents. In the following, we use simple indexes for contexts and concepts as identifiers. These are capital letters and numbers, respectively.

3.3 Scaling upon relations

The question of how to factor in relations when constructing concept descriptions admits a variety of answers that fit our conceptual analysis goals to diverging degrees. In the following, we first motivate our particular way of turning links into attributes and then provide a formal expression of the corresponding constructs.

3.3.1 A naive approach

A straightforward, and somewhat naïve, way of reflecting object links would be to assimilate links to standard one-valued attributes. In other words, for each $\mathcal{K} = (O, A, I) \in \mathbf{K}$, we extend A with attributes $a_{r,\bar{o}}$ corresponding to the pairs made of a relation originating at the context, $r \in rel(\mathcal{K})$, and an object $\bar{o}$ from its targets,

	Senior	Adult	Male	Female	Bleeding	Breathdisorder	...	Vomiting	Dactinomycin	Isentress	Kaletra	Selzentry	Sustiva	Viread
Farley	×		×				...				×			×
Lane		×		×	×		...					×	×	
Shana		×		×			...				×			×
Trudy		×	×		×	×	...	×	×	×				

Table 5 One-valued context $\mathcal{K}_P$ encoding AIDS patients with the observed ADR and drugs taken.

i.e., $(o, \bar{o}) \in r$ for some $o \in O$. To illustrate this construct, observe the context in Table 5.

It should be noticed that to increase readability within the above table, we have skipped the name of the relation, `takes`, in the names of the new attributes. Thus, they are only identified by the target objects, i.e., the drugs taken by a patient. As it is the only relation that starts at the contexts of patients, this does not lead to any confusion. However, with more than one such relations going to the same range context, the object names are not enough to uniquely define the corresponding attributes.

The concept lattice corresponding to Table 5 is given in Figure 3. A quick examination of this lattice and the lattice in Figure 1 indicates that the former contains three extra concepts. In particular, the patient `Shana` is now located in a concept that is immediately above the bottom concept. This corresponds to the intuitive idea that the larger the attribute set of the context the (potentially) preciser the conceptual structure induced by them. While the new concepts constitute a clear gain with respect to Figure 1, the abstraction potential of the underlying attributes is rather limited[4]. Indeed, in order for two patients to share such an attribute, both of them must have taken the underlying drug. However, drugs, even without being completely identical, may have similar behaviour, e.g., in terms of ADR.

To translate the intake of similar drugs –instead of identical ones– a higher-level abstraction would be necessary that allows for some properties to be shared among the drugs taken by each patient from a given set X. Hence, a more purposeful processing of the links might seem to lay in the transfer of properties from the target objects to the source ones, i.e., here, from drugs to patients. Spelled differently, while the Table 5 is a mere join of Table 3 and the one in the left-hand-side of Table 4, one may imagine a more sophisticated procedure of bringing the attributes of the drug context into the patient contexts while using the `takes` relation as a filter. While such a solution has its merits, it is still insufficient for our purposes, in particular, since it would work poorly with: (i) chains of relations between a larger set of contexts (e.g., with a context of drug producers in our example), and (ii) relations of identical domain and range contexts (e.g., `interacts-with` for drugs).

[4] As an indication of the limits, compare the lattice here to the one drawn in Figure 5.

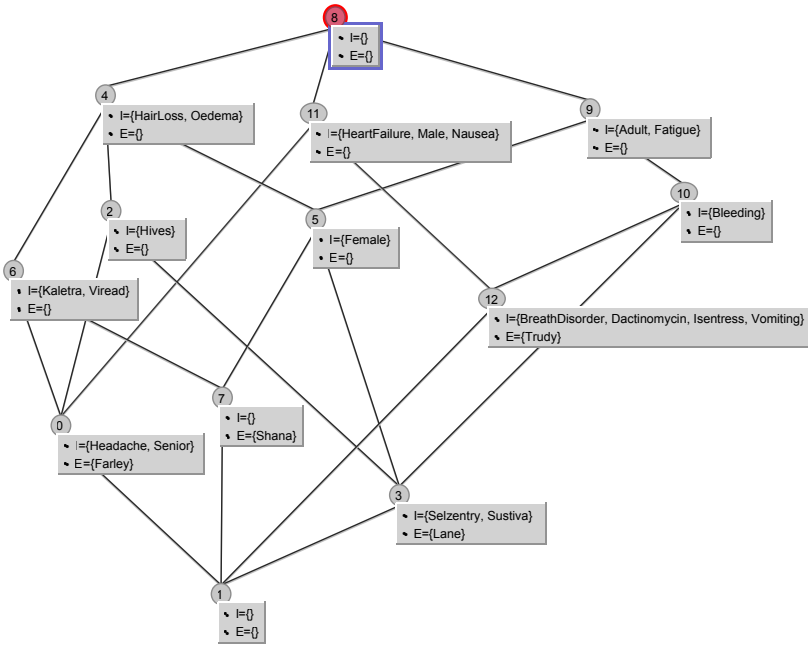

Fig. 3 The lattice of patients with drugs translated as one-valued attributes.

We therefore turn to the knowledge representation languages to look for ways of expressing links and find suitable abstractions that are both precise and shareable between the source objects of a relation. Such expressions are common in the logical representation formalisms as we discuss below.

3.3.2 Role restrictions, from DLs *to* RCA

Description logic-based formalisms [5] are nowadays the *de facto* standard for knowledge representation, especially in ontology engineering and on the semantic web. Such formalisms help to organize the knowledge about a domain by focusing on relevant domain abstractions, i.e., *concepts* and their relations called *roles*. The DLs offer a collection of constructors to express relational information on the schema level. Beside defining domains and ranges for roles, such constructors allow finer constraints on the outgoing links of data objects to be asserted such as the exact type of the target objects, the number of links per object, etc.

Technically speaking, relational information is modelled upon role restrictions from DLs [5]. In DLs, two types of knowledge are expressed at the schema level, *concepts* and *roles*. By properly restricting the roles of a given concept, one can easily define a sub-concept thereof. Typical constructs for role restriction include existential quantification ($\exists$R.C), universal quantification ($\forall$R.C), strict universal quantification ($\forall\exists$R.C), cardinality restriction ($\geq$n R), qualified cardinality restriction ($\geq$n R.C), etc. Here R stands for a role, i.e., the equivalent of a relation from an RCF, whereas C is a conceptual expression that might be either a name or a formula involving logical

connectors on sub-formula(s). Moreover, C works like a filter on the range of the role R: the role restriction as a whole is a predicate satisfied by the objects in the domain of R that are appropriately connected (depending on the restriction constructor) to objects from the extension (*aka* interpretation) of C.

Following strictly the DLs definitions of role restriction constructors, $r(o)$, the image by r of an object from $O_{i_1} = dom(r)$, must be compared to the extent of some concept c from $\mathcal{K}_{i_2} = (O_{i_2}, A_{i_2}, I_{i_2})$ where $O_{i_2} = ran(r)$. Some constructors can be easily envisaged in RCA:

- (i) *universal* - $r(o)$ is included in the extent of c,
- (ii) *existential* - $r(o)$ has a non-void intersection with the extent of c.

Further ones are discussed below. These constructors yield a set of new descriptors that apply to all objects from the range context of r and behave like standard formal attributes. Since the underlying encoding of links is kin to scaling of categorical attributes, it was named *relational scaling* and the constructors *scaling schemes*.

Such an approach is beneficial in many respects, in particular, it requires no new definitions of concepts nor lattices, hence all the results from FCA, both theoretical and algorithmic hold.

3.3.3 Scaling a relation from an RCF

From a mathematical point of view, the scaling of $\mathcal{K}_i$ along $r \in rel(\mathcal{K}_i)$ with $ran(r) = O_{i_2}$ and with respect to a lattice $\mathcal{L}_{i_2}$ extends A_i by adding a set of new attributes and completes I_i accordingly. In order to formalize this, we first introduce the auxiliary function ρ that maps each relation to a given scaling operator. It is defined as $\rho : \mathbf{R} \rightarrow \mathbf{Q}$ with $\mathbf{Q} = \{\exists, \forall, \forall\exists, \geq, \geq_q, \leq, \leq_q\}$. Now, each new attribute has the form "*q r:c*" where r is the relation (name), c stands for a concept from $\mathcal{L}_{i_2}$ and q is the relational constructor associated to r, i.e. $\rho(r)$. We chose the ":" separator instead of ".", the one used in DLs, in an analogy to standard type definitions for variables from object-oriented languages. For example, assume a $\exists$-based scaling on r, i.e., $\rho(r) = \exists$. The resulting operator $\mathbb{S}_{(r,\exists),\mathcal{L}_{i_2}}$, has the following effect: For each object o from $dom(r)$ and each concept c, o is incident to the attribute "$\exists r : c$" whenever $r(o)$ shares at least one object with the extent of c.

Definition 4 (The existential scaling operator)
Given $\mathcal{K} = (O, A, I)$ and $r \in rel(\mathcal{K})$, let i_r be such that $ran(r) = O_{i_r}$ where $\mathcal{K}_{i_r} = (O_{i_r}, A_{i_r}, I_{i_r})$. Let also $\mathcal{L}_{i_r}$ be the lattice of $\mathcal{K}_{i_r}$. The existential scaling operator $\mathbb{S}_{(r,\exists),\mathcal{L}_{i_r}}$ maps $\mathcal{K}$ into the derived context $\mathcal{K}^+ = (O^+, A^+, I^+)$, where:

- $O^+ = O$
- $A^+ = \{\exists r : c \mid c \in \mathcal{L}_{i_r}\}$, *where each* $\exists r : c$ *is a relational attribute*
- $I^+ = \{(o, \exists r : c) \mid o \in O, c \in \mathcal{L}_{i_r}, r(o) \cap Ext(c) \neq \emptyset\}$

In other terms, scaling a context along a relation consists in integrating this relation to the context in the form of one-valued attributes using a scaling operator. For instance, let us detail how the context of anti-HIV drugs $\mathcal{K}_D$ can be scaled along relation `itb` given in the left-hand side of Table 4 with respect to the lattice of patients depicted

in Figure 2. The anti-HIV drug object `Kaletra` –whose image upon the relation itb is $\{$`Farley`, `Shana`$\}$– will have the attributes $\exists$*itb:*c_i, with $i \in \{0, 2, 4, 5, 6, 7, 9\}$ in the lattice $\mathcal{L}_P$ (Figure 2), since extents of concepts c_0, c_2, c_4, c_5, c_6, c_7 and c_9 have a non-empty intersection with the image set itb (`Kaletra`). Table 6 provides the result of scaling existentially the drug context $\mathcal{K}_D$ upon itb.

	$\exists$ *itb:c0*	$\exists$ *itb:c2*	$\exists$ *itb:c3*	$\exists$ *itb:c4*	$\exists$ *itb:c5*	$\exists$ *itb:c6*	$\exists$ *itb:c7*	$\exists$ *itb:c8*	$\exists$ *itb:c9*	$\exists$ *itb:c10*
`Dactinomycin`						×	×	×	×	×
`Isentress`						×	×	×	×	×
`Kaletra`	×	×		×	×	×	×		×	
`Selzentry`		×	×	×	×	×	×	×		
`Sustiva`		×	×	×	×	×	×	×		
`Viread`	×	×		×	×	×	×		×	

Table 6 The existential scaling of the anti-HIV drug context $\mathcal{K}_D$ along the relation `itb` using the lattice $\mathcal{L}_P$ of AIDS patients (Figure 2). Observe that $\exists$ *itb:*c_1 is skipped as c_1 is the bottom concept whose extent is void.

Similarly, we define **the universal scaling operator** with respect to r and $\mathcal{L}_{i_r}$, denoted by $\mathbb{S}_{(r,\forall),\mathcal{L}_{i_r}}$. The difference is that instead of non-empty intersection, the object image $r(o)$ must be completely included in the extent of c in order for o to get the relational attribute $\forall r : c$:

$$I^+ = \{(o, \forall r : c) | o \in O, c \in \mathcal{L}_{i_r}, r(o) \subseteq Ext(c)\}.$$

For instance, universal scaling of relation itb using the concept lattice given in Figure 2 will assign relational attributes $\forall$*itb:*c_4 and $\forall$*itb:*c_6 to the drug object `Kaletra` since the respective extents of both concepts c_4 and c_6 comprise $\{$`Farley`, `Shana`$\}$, the image set of `Kaletra` for `itb`.

Yet the above definition presents on the well-known empty image problem: objects with no links for a relation r get all universally quantified attributes with r. To illustrate this, consider a universal scaling of the relation `iw` (see the right-hand side of Table 4) with the concept lattice of drugs in Figure 1. Now observe that both drug objects `Isentress` and `Dactinomycin` have empty-set image following the relation `iw`. Thus, a universal scaling of `iw` will assign every one of the resulting relational attributes to `Isentress` and `Dactinomycin`.

Clearly, the mere universal quantification is of little practical use. Therefore, we introduced a more constrained version called **strict universal scaling** which additionally requires that at least one link of r starts at the source object o. The scaling scheme, denoted $\mathbb{S}_{(r,\forall\exists),\mathcal{L}_{i_r}}$ under the above assumptions, comprises a slightly modified definition for the attributes $\forall\exists r : c$:

$$I^+ = \{(o, \forall\exists r : c) | o \in O, c \in \mathcal{L}_{i_r}, r(o) \subseteq Ext(c) \; and \; r(o) \neq \emptyset\}.$$

It is noteworthy that $\forall\exists r : c$ corresponds to the DLs expression $\forall R.C \sqcap \exists R$. Based on the role restrictions from DLs, further scaling operators are defined. Table 7

presents the current set of such operators together with their notation, the form of the generated relational attributes and the condition for assigning them to an object. We assume a relation r and a lattice $\mathcal{L}$ on its range set of objects.

Operator	Notation	Attribute form	Constraint in I^+
Universal (wide)	$\mathbb{S}_{(r,\forall),\mathcal{L}}$	$\forall\, r : c$	$r(o) \subseteq Ext(c)$
Existential	$\mathbb{S}_{(r,\exists),\mathcal{L}}$	$\exists\, r : c$	$r(o) \cap Ext(c) \neq \emptyset$
Universal strict	$\mathbb{S}_{(r,\forall\exists),\mathcal{L}}$	$\forall\exists\, r : c$	$r(o) \subseteq Ext(c)\ and\ r(o) \neq \emptyset$
Cardinality restriction (max)	$\mathbb{S}_{(r,\geq),\mathcal{L}}$	$\geq\ n\, r : \top_{\mathcal{L}}$	$\lvert r(o)\rvert \geq\ n$
Cardinality restriction (min)	$\mathbb{S}_{(r,\leq),\mathcal{L}}$	$\leq\ n\, r : \top_{\mathcal{L}}$	$\lvert r(o)\rvert \leq\ n$
Qualified card. restriction (max)	$\mathbb{S}_{(r,\geq_q),\mathcal{L}}$	$\geq\ n\, r : c$	$r(o) \subseteq Ext(c)\ and\ \lvert r(o)\rvert \geq\ n$
Qualified card. restriction (min)	$\mathbb{S}_{(r,\leq_q),\mathcal{L}}$	$\leq\ n\, r : c$	$r(o) \subseteq Ext(c)\ and\ \lvert r(o)\rvert \leq\ n$

Table 7 Operators for the relational scaling in RCA.

3.3.4 Scaling upon all outgoing relations of a context

In RCA, at each step, a context $\mathcal{K}$ is scaled upon all the relevant relations which we assume here to be all those that originate at the context, i.e., the respective set $rel(\mathcal{K})$. To formally express the context obtained by augmenting $\mathcal{K}$ with all the resulting relational attributes, what we call the *complete relational extension* of $\mathcal{K}$, one needs to factor in the available lattices on the respective range contexts of the relations in $rel(\mathcal{K})$. Let the set of lattices corresponding to the contexts in $\mathbf{K}$ be denoted simply by $\mathbf{L}$. Let also $rel(\mathcal{K}) = \{r_l\}_{l=1,\ldots,m_{\mathcal{K}}}$ and, for each r_l, let $\mathcal{L}_{i_l} \in \mathbf{L}$ be the lattice on $O_{i_l} = ran(r_l)$. Under these assumptions, the complete relational extension of $\mathcal{K}$ with respect to ρ and $\mathbf{L}$, denoted $\mathbb{E}_{\rho,\mathbf{L}}$, is defined as the *apposition* of $\mathcal{K}$ with the respective results of the scaling upon each r from $rel(\mathcal{K})$. The apposition operator on two contexts $\mathcal{K}_1$ and $\mathcal{K}_2$, as defined in [15], requires that these share the same set of objects whereas its result, denoted $\mathcal{K}_1|\mathcal{K}_2$, is a context over these objects where the attribute set and the incidence are obtained by union of the respective components of the argument contexts. Formally:

Definition 5 (Complete relational extension of a context)
Given a RCF $(\mathbf{K}, \mathbf{R})$*, with a set of lattices* $\mathbf{L}$*, a scaling constructor mapping* ρ*, and a context* $\mathcal{K} \in \mathbf{K}$ *with* $rel(\mathcal{K}) = \{r_l\}_{l=1,\ldots,m_{\mathcal{K}}}$*, the complete relational extension of* $\mathcal{K}$ *w.r.t.* ρ *and* $\mathbf{L}$ *is*

$$\mathbb{E}_{\rho,\mathbf{L}}(\mathcal{K}) = \mathcal{K} \mid \mathbb{S}_{(r_1,\rho(r_1)),\mathcal{L}_{i_1}}(\mathcal{K}) \mid \ \ldots \mid \mathbb{S}_{(r_{m_k},\rho(r_{m_k})),\mathcal{L}_{i_{m_k}}}(\mathcal{K})$$

For instance, consider the result of the complete extension of the drug context $\mathcal{K}_D^s = \mathbb{E}_{\rho,\mathbf{L}}(\mathcal{K}_D)$ where $\rho = \{$(`itb`,$\exists$), (`iw`,$\exists$), (`takes`,$\exists$)$\}$ and $\mathbf{L} = \{\mathcal{L}_P, \mathcal{L}_D\}$. The corresponding context $\mathcal{K}_D^s$ may be obtained by putting side-by-side the initial drug context $\mathcal{K}_D$ (Table 2), and the respective results of existentially scaling its relations `is-taken-by` and `interacts-with`. These are given in Table 6 and Table 8, respectively. Observe that in Table 8, the attribute $\exists$ *iw:*c_1 is missing from the table. In fact,

	$\exists$ *iw:c0*	$\exists$ *iw:c2*	$\exists$ *iw:c3*	$\exists$ *iw:c4*	$\exists$ *iw:c5*	$\exists$ *iw:c6*	$\exists$ *iw:c7*	$\exists$ *iw:c8*	$\exists$ *iw:c9*	$\exists$ *iw:c10*	$\exists$ *iw:c11*	$\exists$ *iw:c12*	$\exists$ *iw:c13*	$\exists$ *iw:c14*	$\exists$ *iw:c15*
`Dactinomycin`							×							×	×
`Isentress`	×	×		×			×			×					
`Kaletra`		×	×	×	×		×	×		×	×	×		×	
`Selzentry`							×	×	×						
`Sustiva`							×	×		×	×	×	×	×	
`Viread`				×	×	×	×			×	×				

Table 8 The result of an existential relational scaling applied to the anti-HIV drug context $\mathcal{K}_D$ upon the relation `iw` and using its own lattice $\mathcal{L}_D$ (Figure 1). The attribute $\exists$ *iw:*c_1 is skipped as c_1 is the bottom concept.

c_1 is the bottom concept and here it is of an empty extent, hence no object could get the attribute.

The lattice $\mathcal{L}_D^s$ corresponding to the above context is given in Figure 5. A first observation is that it is strictly larger than the initial lattice $\mathcal{L}_D$ in Figure 1 as no extent from the latter is missing in it. The intuitive explanation is that by adding some attributes to a context, one cannot destroy the extents from the initial context. Thus, it is by no means a coincidence, that concepts with the same identifier in both figures have the same extents (e.g., c_8 whose extent comprises `Selzentry`, `Sustiva` and `Kaletra`).

Let us elaborate on the relationship between an initial context and its extended version as well as between their respective lattices. Thus, let – under the hypotheses of Definition 5 – $\mathcal{K}^s = \mathbb{E}_{\rho,\mathbf{L}}(\mathcal{K})$ for some $\mathcal{K}$ from the RCF and consider the lattices $\mathcal{L}$ and $\mathcal{L}^s$. Since $\mathcal{K}^s$ is an apposed context of $\mathcal{K}$, it has a larger set of attributes yet *the same* set of objects. In [34], it was shown that each extent of $\mathcal{K}$ is also an extent of $\mathcal{K}^s$ which means the lattice $\mathcal{L}$ is "blended" in $\mathcal{L}^s$. Formally,

Property 1 (Lattices of apposed contexts, [34])
Given contexts $\mathcal{K}$, $\mathcal{K}_1$ and $\mathcal{K}_2$ such that $\mathcal{K} = \mathcal{K}_1|\mathcal{K}_2$, the respective lattices $\mathcal{L}$, $\mathcal{L}_1$ and $\mathcal{L}_2$ satisfy that for each concept $(X, Y) \in \mathcal{L}_i$ $(i = 1, 2)$, there is an equivalent concept $(X, Z) \in \mathcal{L}$ where $Y \subseteq Z$.

This basically means that by extending a context through relational scaling, its lattice expands while preserving its initial conceptual structure as a sub-order. This sub-order is witnessed by concept extents: whatever the extension of the context, its concept set will always comprise, for each concept from the initial context $\mathcal{K}$, a concept with the same extent, yet of a possibly larger intent. We therefore shall establish concept identities throughout evolving versions of the same initial context on the basis of their respective entities: two concepts from different versions whose extents coincide will be considered *two versions of the same concept*.

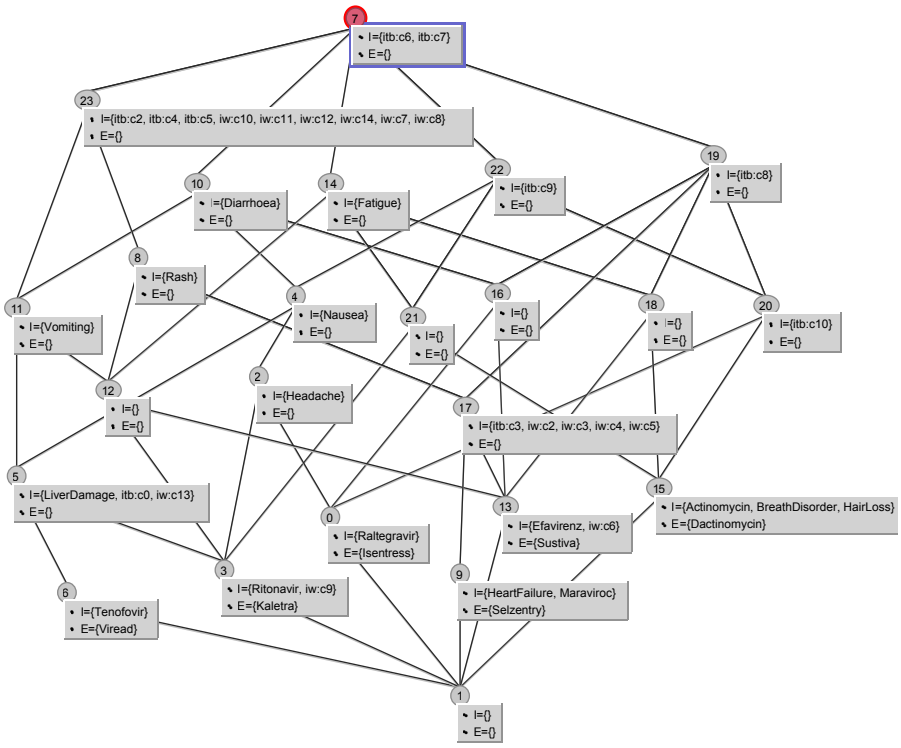

Fig. 4 The lattice $\mathcal{L}_D^s$ of the relationally extended `Drug` context $\mathcal{K}_D^s$ (quantifiers omitted in relational attributes).

3.3.5 Scaling the entire RCF

The above extension operator may now be expanded to reach over the entire set of contexts of an RCF. Here again, ρ and $\mathbf{L}$ are parameters of the multi-context operator on the set $\mathbf{K}$ which works as follows:

Definition 6 (Complete relational extension of an RCF)
Given a RCF $(\mathbf{K}, \mathbf{R})$ *whose context set is* $\mathbf{K} = \{\mathcal{K}_1, \ldots, \mathcal{K}_n\}$ *and whose set of lattices is* $\mathbf{L}$*, and a scaling constructor mapping* ρ*, the complete relational extension of* $\mathbf{K}$ *is a set of contexts defined as*

$$\mathbb{E}^*_{\rho,\mathbf{L}}(\mathbf{K}) = \{\mathbb{E}_{\rho,\mathbf{L}}(\mathcal{K}_1), \ldots, \mathbb{E}_{\rho,\mathbf{L}}(\mathcal{K}_n)\}.$$

For instance, the result of the above operator on our initial RCF unfolds whenever one adds to the above three tables representing the complementary parts of the `Drug` context, the initial `Patient` context in Table 3 and its respective image by existential relational scaling w.r.t. `takes` (Table 9).

Clearly, applying $\mathbb{E}^*$ to a context set yields a set of extended contexts where each individual context is an attribute-wise extension of its counterpart in the initial set. Then, to each of the extended contexts corresponds a potentially larger lattice. In our

	$\exists$ *takes:c0*	$\exists$ *takes:c2*	$\exists$ *takes:c3*	$\exists$ *takes:c4*	$\exists$ *takes:c5*	$\exists$ *takes:c6*	$\exists$ *takes:c7*	$\exists$ *takes:c8*	$\exists$ *takes:c9*	$\exists$ *takes:c10*	$\exists$ *takes:c11*	$\exists$ *takes:c12*	$\exists$ *takes:c13*	$\exists$ *takes:c14*	$\exists$ *takes:c15*
`Farley`		×	×	×	×	×	×	×		×	×	×		×	
`Lane`							×	×	×	×	×	×	×	×	
`Shana`		×	×	×	×	×	×	×		×	×	×		×	
`Trudy`	×	×		×			×			×				×	×

Table 9 The existential scaling of $\mathcal{K}_P$ upon `takes` and the lattice $\mathcal{L}_D$ (Figure 1).

example, the lattice of $\mathcal{K}_P^s$, the `Patient` context completely extended w.r.t. the initial lattice on drugs $\mathcal{L}_D$, is drawn in Figure 5.

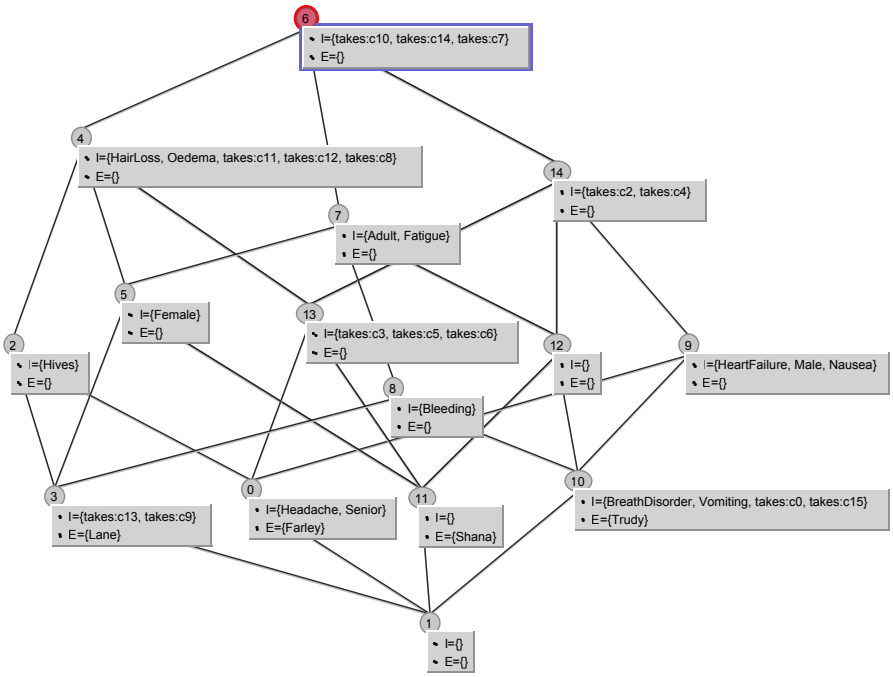

Fig. 5 The lattice $\mathcal{L}_P^s$ of the relationally extended `Patient` context $\mathcal{K}_P^s$ (quantifiers omitted).

Let us notice that the lattice in Figure 5 is substantially different from the one in Figure 3. For instance, the extent of concept c_{12} from Figure 5 groups `Shana` and `Trudy` while there is no such extent in Figure 3: the smallest extent comprising these patients is in c_9 and has also `Lane`. What distinguishes both former patients w.r.t. the latter one is that each of them took at least one drug from the extents of drug concepts c_2 and c_4 (see Figure 1). These concepts cover the drugs causing `Headache` and `Nausea`, respectively. Conversely, one could observe that any extent from Figure 3

is also present in Figure 5. We feel that the additional conceptual knowledge gained through relational scaling vindicates the technique as well as the entire RCA approach.

3.3.6 Refining the scales

The above Property 1 states that the lattices of the relationally extended contexts, albeit different conceptual structures, encompass all the conceptual information from the respective lattices of the initial contexts. Indeed, comparing $\mathcal{L}_D^s$ from Figure 4 to $\mathcal{L}_D$ (see Figure 1), one notices that eight new concepts have emerged as a result of the relational scaling, e.g., concept c_{17} that groups `Selzentry` and `Sustiva`. Now a natural question arises: Given that there might be additional concepts in some of the expanded lattices, i.e., concepts without a counterpart in the initial lattice, should not these be reused in a new round of scaling? In other terms, should one proceed by repeating the scaling step, this time with a more thorough conceptual information to use as scaling basis? In our example, knowing $\mathcal{L}_D^s$ could help make the scaling upon `takes` even more precise. Indeed, beside the attributes corresponding to (extents of) concepts from $\mathcal{L}_D$ (see Table 9), eight new attributes translating the aforementioned additional concepts are yielded by the scaling. These attributes are shown in Table 10.

	$\exists$ *takes:c16*	$\exists$ *takes:c17*	$\exists$ *takes:c18*	$\exists$ *takes:c19*	$\exists$ *takes:c20*	$\exists$ *takes:c21*	$\exists$ *takes:c22*	$\exists$ *takes:c23*
`Farley`						×	×	×
`Lane`	×	×	×	×				×
`Shana`						×	×	×
`Trudy`	×		×	×	×	×	×	

Table 10 The existential scaling of $\mathcal{K}_P$ upon `takes` and the lattice $\mathcal{L}_D^s$ (Figure 4). Only the differential part w.r.t. Table 9 is given.

Using the total set of attributes from Table 9 and from Table 10 clearly results in a further extension of the $\mathcal{K}_P$ that has a strictly larger set of attributes than $\mathcal{K}_P^s$, and hence, a potentially larger set of concepts. This amounts to inserting the previously discovered conceptual knowledge into the analysis process in order to refine its results (by refining the lattice-shaped scales).

Obviously, the process needs not stop at step two and could potentially go to an arbitrary number of such scaling/lattice construction iterations. Then, one needs to consider the termination problem for the iterative process that arises in this way: Could one guarantee that the process would converge into a well-defined result?

Further to the idea of iterating upon ever expanding contexts and lattices, we define a general method that constructs the ultimate result of the analysis task on an RCF.

4 Context dynamics and the iterative RCA method

4.1 Iterative context expansion

As stated previously, we capture the evolution of each context $\mathcal{K}_i \in \mathbf{K}$ in a sequence $\mathcal{K}_i^p$ whose zero member $\mathcal{K}_i^0 = (O_i^0, A_i^0, I_i^0)$ is the input context. From there on, each member of the sequence is obtained from the previous one by a complete relational expansion upon the relations r from $rel(\mathcal{K}_i)$ while using the respective constructors from ρ, i.e., $\rho(r)$, and the lattices of the p-th iteration in $\mathbf{L}^p$. The set of contexts in the p-th iteration may be recursively defined as follows:

Definition 7 *Given a* RCF $(\mathbf{K}, \mathbf{R})$ *and a mapping ρ, the vector of contexts at the step $p+1$ is*

$$\mathbf{K}^0 = \mathbf{K}$$

$$\mathbf{K}^{p+1} = \mathbb{E}^*_{\rho, L^p}(\mathbf{K}^p)$$

Now each of the contexts from the fix point member $\mathbf{K}^\infty$ is similarly defined:

Definition 8 *Given a* RCF $(\mathbf{K}, \mathbf{R})$*, a mapping ρ, and some $i \in \{1, \ldots, |\mathbf{K}|\}$ the i-th context at the step $p+1$ is the extension of the context of the same rank at step p:*

$$\mathcal{K}_i^0 = \mathcal{K}_i$$

$$\mathcal{K}_i^{p+1} = \mathbb{E}_{\rho, L^p}(\mathcal{K}_i^p)$$

A first observation is that the fix point members of the context sequence are well-defined. In other terms, for each $i \in \{1, \ldots, |\mathbf{K}|\}$ the context $\mathcal{K}_i^\infty$ is the fixpoint of the *non-decreasing sequence* $(\mathcal{K}_i^p)$.

Lemma 1 *Given a* RCF $(\mathbf{K}, \mathbf{R})$ *and a mapping ρ, for each $i \in \{1, \ldots, |\mathbf{K}|\}$ and $p \geq 0$, let $\mathcal{K}_i^p = (O_i^p, A_i^p, I_i^p)$ and $\mathcal{K}_i^{p+1} = (O_i^{p+1}, A_i^{p+1}, I_i^{p+1})$, then*

$$O_i^p = O_i^{p+1},\ A_i^p \subseteq A_i^{p+1} \text{ and } I_i^p \subseteq I_i^{p+1}.$$

To prove the above fact, we need an intermediate step that focuses on lattices:

Lemma 2 *Given a* RCF $(\mathbf{K}, \mathbf{R})$ *and a mapping ρ, for each $i \in \{1, \ldots, |\mathbf{K}|\}$ and $p \geq 0$*

$$\forall (X, Y) \in \mathcal{L}_i^p,\ \exists (X, Z) \in \mathcal{L}_i^{p+1}.$$

The fact that contexts can only grow or remain the same is readily proved by induction on p where the basic case is expressed as in Property 1.

This now allows us to assert that the relational attributes in $\mathcal{K}_i^p$ for some i and p chosen as above, are included in the relational attributes of $\mathcal{K}_i^{p+1}$. Indeed, as indicated at the end of section 3.2, we consider the concepts (X, Y) and (X, Z) from Lemma 2 to be identical, which, given that only the shared extent X is used in the scaling process, constitutes no real limitation.

To demonstrate that each context sequence converges, it is now enough to observe that for a lattice sequence $\mathcal{L}_i^p$, there is a natural upper bound represented by the

Boolean lattice[5] of O_i. And since the set of all extents from $\mathcal{L}_i^p$ does not decrease when p grows up to ∞ (as shown in Lemma 2), it necessarily converges. We can now state the convergence of the context sequence:

Theorem 1 *Given a* RCF $(\mathbf{K}, \mathbf{R})$ *and a mapping ρ, the sequence* $(\mathbf{K}^p)$ *converges toward a well-defined set of maximally extended contexts* $\mathbf{K}^\infty$.

To sum up, $\mathbf{K}^\infty$ is obtained from $\mathbf{K}^0$ through an iterative process of concept construction and attribute generation using the data in $\mathbf{R}$ and a user-provided set of (relation,constructor) pairs in ρ. Moreover, being a fix point of an expanding process, it is the closure of $\mathbf{K}^0$ for the implicit closure operator associated to the method.

4.2 The MULTI-FCA method

The MULTI-FCA method describes the step-wise construction of the fix point solution from the initial RCF. The logic of our analysis method is iterative one: Whenever the contexts of a RCF are extended, their corresponding lattices expand as well. Moreover, it represents a computational schema rather than a precise algorithm as many algorithmic choices are left to the analyst.

```
1: proc MULTI-FCA(
2: In: (K,R) an RCF, ρ a constructor mapping
3: Out: L array [1,...,n] of lattices)
4: p ← 0 ; halt ← false
5: for i from 1 to n do
6:   K_i^0 ← SCALE(K_i)
7:   L_i^0 ← BUILD-LATTICE(K_i^0)
8: while not halt do
9:   p = p + 1
10:   for i from 1 to n do
11:     K_i^p ← EXTEND-CONTEXT(K_i^{p-1}, ρ, L^{p-1})
12:     L_i^p ← UPDATE-LATTICE(K_i^p, L_i^{p-1})
13:   halt ← ⋀_{i=1,...,n} ISOMORPHIC(L_i^p, L_i^{p-1})
```

Algorithm 1: Producing a lattice for each context in an RCF.

Here we provide a detailed description of the algorithm.

At the initialization step (lines $4-7$), each context $\mathcal{K}_i^0$ is obtained from $\mathcal{K}_i$ by applying a conceptual scaling to the many-valued attributes in $\mathcal{K}_i$ using the primitive SCALE (line 6). The lattices $\mathcal{L}_i^0$ are constructed using the primitive BUILD-LATTICE (line 7). At the step p and for each relation $r_k \subseteq O_i \times Oj$, the lattice $\mathcal{L}_j^{p-1}$ of the range context is used to extend the domain context $\mathcal{K}_i^p$ using the primitive EXTEND-CONTEXT and then update the lattice of the domain context $\mathcal{L}_i^p$ using the primitive UPDATE-LATTICE (lines $8-13$). For both primitives of lattice construction and lattice update, the choice of the exact algorithms is free. The process ends whenever at

[5] Thus, in the worst case $\mathcal{L}_i^\infty$ may comprise all subsets of O_i as extents.

two subsequent steps all the pairs of corresponding lattices $\mathcal{L}_i^n$ and $\mathcal{L}_i^{n+1}$ are isomorphic (checked by the primitive ISOMORPHIC, line 13).

For example, the MULTI-FCA method applied to the RCF composed of the two contexts $\mathcal{K}_P$ and $\mathcal{K}_D$, and the relations `takes`, `itb` and `iw`, all of them scaled with an existential constructor, yields the final lattices $\mathcal{L}_P^2$ and $\mathcal{L}_D^2$, shown in Figure 7 and Figure 6, respectively. These have been obtained after two iterations.

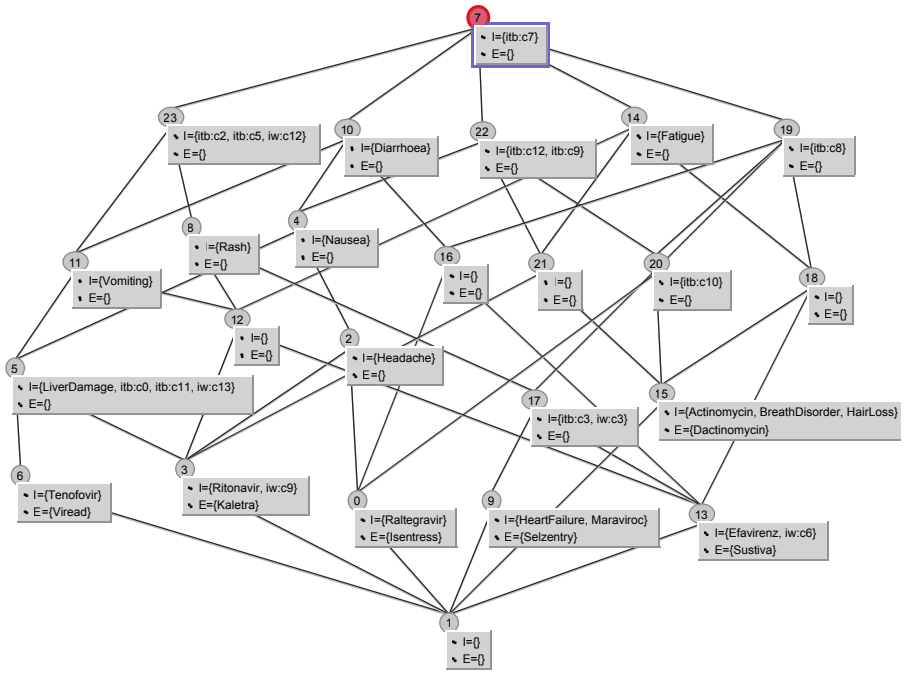

Fig. 6 The final lattice of drugs ($\mathcal{L}_D^\infty$). Quantifiers are omitted in relational attributes due to visualization limitation of GALICIA.

It is noteworthy that unlike concepts in Figure 5 and Figure 4, the ones of the final lattices have undergone an additional round of label reduction. The underlying simplification amounts to skip from the reduced intent of a concept relational attributes "$q\ r{:}c_1$" such that there is another attribute "$q\ r{:}c_2$" with $c_2 \leq c_1$. The intuition behind this reduction is that whatever the constructor q, for any object o from the domain of r, whenever o satisfies "$q\ r{:}c_2$", it will necessarily satisfy "$q\ r{:}c_1$" as well. For instance, the intent of patient concept c_{14} in Figure 5 comprises the `takes:c4` attribute that has been removed in the intent of the same concept in Figure 7. A simple check in Figure 1 reveals that the drug concept c_2 is a sub-concept of c_4. In actuality, this amounts to attribute-clarifying the parts of the fix-point context generated by scaling upon individual relations while keeping for each set of equivalent attributes the ones corresponding to minimal concepts from the relation range.

An important question focuses on the overall cost of the relational analysis of a dataset, especially given the iterative nature of the MULTI-FCA method. In fact,

in many cases there will be no necessity of iterating: A proper ordering of analysis tasks for individual contexts should help avoid it. To see this, one should imagine the "schema" of an RCF, i.e., the digraph made of contexts as vertexes and relations as directed edges. Now, provided this graph is a DAG (no circuits), a topological sort of the contexts would provide a total ordering that is compatible with the relation-induced dependencies among contexts. Thus, analysing the contexts according to this order will ensure a context $\mathcal{K}$ is only processed whenever all the lattices required for scaling the relations in $rel(\mathcal{K})$ have already been constructed up to their fixed point form. Yet in order to cover the entire spectrum of possible RCF, inclusive circular ones, we preserve the more general expression of MULTI-FCA that is bound toward a fix point.

4.3 Interpreting relational concepts

New abstractions emerge in a relational lattice characterizing the sharing of links between the objects (as already discussed in § 3.1). For instance, the drug concept $c_{\#20}$ in $\mathcal{L}_D^{\infty}$ (Figure 6, labeled ({},{$\exists\ itb : c10$}) does not belong to the initial lattice $\mathcal{L}_D^0$ (Figure 1). It is a new relational concept whose objects, namely `Dactinomycin` and `Isentress`, are described in a purely relational way indicating that both drugs were taken by patients who experienced common adverse reactions, e.g., breath disorder, heart failure, etc. (patient concept $c_{\#10}$ in $\mathcal{L}_P$).

In addition, some intents from $\mathcal{L}_D^0$ are extended by a relational part hence refining the description of the member objects. In this way, the concept $c_{\#5}$ in $\mathcal{L}_D^0$ (Figure 1) represents NRTI drug class that causes liver disorder. In the final lattice of drugs $\mathcal{L}_D^{\infty}$ (Figure 6), the intent of the same concept $c_{\#5}$, enriched with the relational attributes $\exists iw : c13$, $\exists itb : c0$ and $\exists itb : c11$ indicates that both NRTI drugs `Kaletra` and `Viread` interact with `Sustiva` and cause some unexpected adverse reactions including hair loss and oedema. More generally, the relational lattices of patients and drugs provided by RCA link profiles of patients (case reports) with classes of drugs.

4.4 Complexity issues

The MULTI-FCA method iterates over a set of tasks for individual contexts: lattice construction/maintenance, relational scaling, termination tests. Beside their individual complexity estimations, potential other factors include the number of iterations and the size of the set $\mathbf{K}$.

Lattice construction is a well-known *listing problem*. Unlike *decision problems* which focus on a single yes/no solution, listing problems enumerate all such solutions. As the number of such solutions may vary significantly among the instances of the problem, it is more appropriate to assess the average cost of producing a solution or, alternatively, to include the output size as a factor. The lowest known worst-case time complexity for the concept lattice construction is $O(n_c * n_o * n_a)$ where n_c, n_a and n_o are the number of the concepts in the lattice, the number of the attributes and the number of the objects in the context [18]. To get the overall cost for the lattice construction in MULTI-FCA, one could merely multiply by the iteration number.

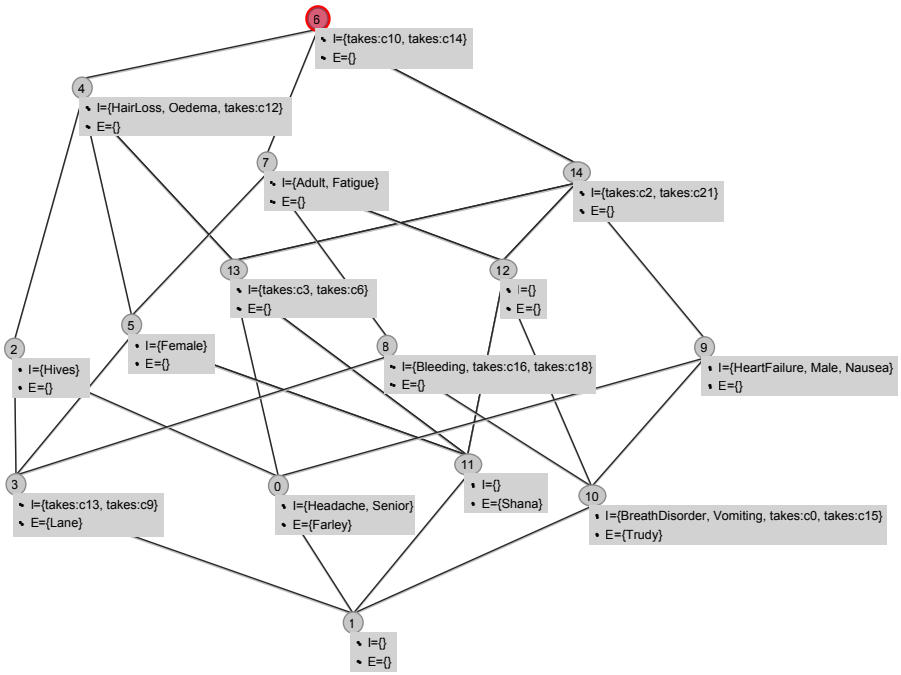

Fig. 7 The final lattice of patients ($\mathcal{L}_P^{\infty}$). Quantifiers are omitted in relational attributes because of visualization limitations of GALICIA.

Yet, if incremental construction is performed instead of batch one (recall algorithmic choices are left to the user), the construction of the fix point lattice of a context could be seen as a single task, split across the iterations. Its basic step is the incremental insertion of a single attribute into a context followed by the lattice update which, as argued in [35], amounts to a batch lattice construction at the same total cost of $O(n_c * n_o * n_a)$.

A subtle difference here is that in MULTI-FCA the fix point lattices are computed from the fix point contexts rather that the initial ones of the RCF. In a way, the fix point contexts are part of the RCA output and there is no simple way of estimating their sizes. We therefore rewrite the overall complexity of the lattice construction part (inclusive the initializing part, line 7 of Algorithm 1) as follows. Let the size of the largest lattice be n_{c_m} formal concepts, the maximal number of attributes in a fix point context be n_{a_m}, and the maximal number of objects in a context be n_{o_m}. The overall construction time is thus $O(n_{c_m} * n_{a_m} * n_{o_m})$. Notice that the above context number factor was skipped as it is expected to be modest (typically, less than 7). Furthermore, this computation offsets the necessity to factor in –hence to know– the number of iterations: only the set of attributes in the fix point context matters, the way their creation is split among the iterations is immaterial.

The cost factor due to scaling is assessed similarly: First, a concept $c = (X, Y)$ from the output contributes to attribute construction only at the iteration following its creation. With the incremental approach, at subsequent iterations the scaling would

ignore c. Its contribution to scaling effort depends on the number of relations r whose $ran(r)$ is object set behind X, the size of each $dom(r)$ and the cost of the incidence test for $qr : c$ and an object $o \in dom(r)$. The first one is small (in the low digits) and hence vanishes. The third factor is linear in the sizes of $r(o)$ and X and hence of $ran(r)$. This puts the complexity estimation of the scaling at $O(n_{c_m} * n_{o_m}^2)$.

Finally, the termination (line 13 of Algorithm 1) is efficiently tested by simply comparing the sizes of lattices on two subsequent iteration (due to the sub-order property). Thus, its cost factor could be neglected.

In summary, when lattice construction and scaling are performed by incremental algorithms, the overall time complexity function is in $O(n_{c_m} * n_{o_m} * (n_{a_m} + n_{o_m}))$.

The other important aspect of MULTI-FCA is its convergence, i.e., the number of steps till the fix point is reached. Intuitively, the process halts whenever there are no more attributes from the original sets A_i that can be propagated backwards along the chains of relational links, e.g., known ADR of a drug to the patient taking that drug. Thus, the maximal length of such a chain is an indicator of the iterations MULTI-FCA might need to converge. In fact, absent circuits in these chains, a proper ordering of the contexts enables a single step construction of the fix point. However, the calculation of the exact number in the general case remains a challenging problem. Indeed, with circularity in the links, the propagation could loop over a circuit for a number of iterations (that depends on the length of the circuit). In the extreme cases, i.e., with particularly unlucky configurations of the links in the dataset, the number of iterations is bound from above by the product of the length of some comparable circuits[6]. Fortunately, such cases are extremely rare: circuits, whenever present, tend to be of minimal size (two or three).

5 Related work

Our RCA approach follows several trends within the fields it contributes to. First, as an FCA-based framework, RCA relates to work on extending the standard input data format by integrating non atomic attributes in formal contexts. This trend dates back to the end of 90s, when studies on logical formulas as formal attributes were initiated, in particular, description logic expressions as in [23]. Simultaneously, the representation of relational information in FCA has been standardized in the form of a *power context family* (PCF) [24]. Compared to the multi-step mechanics of RCA, the core method here amounts to a static, i.e., single-step, scaling.

Independently, in [29], the *relational exploration* method has been defined that expands the basic method towards a full set of DLs constructors, inclusive universal and existential quantifiers for role restrictions. A similarly founded approach has been adopted in [14] within the *Logical Concept Analysis* trend. Indeed, despite some terminological differences the basic hypotheses and mechanisms remain essentially the same. Indeed, the relational descriptors in concept intents are borrowed from DLs whereas the main result relies on reasoning about syntax with role depth providing

[6] The precise function is the least common multiple (lcm) of the circuit lengths, hence the worst-of-the-worst case is with circuits of unique lengths corresponding to prime numbers.

the guidelines. The substantial difference with our approach is the relational format of data that is taken into account and the resulting relational lattices.

In [17, 20], the processing of formal objects with graph-based descriptions is examined. Notwithstanding the effective extension of the FCA machinery to a variety of data formats matching graphs, e.g., chemical compound models, the resulting methods are limited to output exclusively graph-based concept intents. Thus, they cannot be easily tuned to produce the entity-centered concept descriptions studied here.

Finally, RCA shares a name with the relational FCA approach towards analysis of lexical and semantic relationships between terms (synonymy, hyponymy, hyperonymy, etc.), as introduced in [25]. In this approach, relationships hold on entire concepts rather than on individual objects. In contrast, RCA processes relations that materialize as inter-object links and lifts them to the abstract level in the form of relational attributes that are generated by a scaling mechanism.

RCA framework has been implemented within the open source platforms GALICIA[7] and eRCA[8]. RCA has been successfully applied in software engineering and in ontology engineering. In software engineering, RCA is used in the analysis of UML artifacts [16, 3], detection and correction of design defects [22], model transformation learning from transformation examples [30], and Web service classification and composition [4]. In ontology engineering, RCA has been applied for the construction [8] and the refactoring [28, 31] of domain ontologies.

6 Conclusion and future work

RCA provides a relational framework that deals with datasets where object description is not restricted to one-valued contexts. The framework allows the derivation of potentially useful abstractions based on the inter-objects links. Using an iterative generalization process, these links are propagated on the level of concepts thus yielding to relational abstractions which describe in a generic way the links between objects. In comparison to other extensions of FCA, these links are considered at the launch of the concept formation process thanks to the original mechanism of relational scaling whose scales come from concept lattices. We have shown how the process converges and a description of the concrete method that implements it as well as an analytical expression of the obtained solution were proposed.

RCA approach has been implemented in the GALICIA and eRCA platforms and is currently operational for various applications. Scalability is an issue since the size of the lattices can grow rapidly due to the cross-fertilization of contexts. Algorithmic issues are our current primary concern as further progress can be realized with performances through carefully combined techniques for iterative lattice maintenance instead of plain construction.

At each step of the multi-FCA process, the concepts of a particular context, say $\mathcal{K}_i$, or its current relational extension $\mathcal{K}_i^p$, are translated into additional attributes involving the relations whose range is comprised in $\mathcal{K}_i$. The new attributes go to the contexts $\mathcal{K}_j$ that constitute the respective relation domains. There they get assigned

[7] http://www.iro.umontreal.ca/~galicia
[8] http://code.google.com/p/erca/

to the local objects o from O_j through the corresponding images $r(o)$. Therefore, a natural question to elucidate would be the exact dependency between object description and relational attribute, i.e., how do the former define the latter? Or, alternatively, what properties of the former are reflected in the latter? In other terms, one need to examine the correctness of the iterative method, i.e., is every relational attribute rooted in an element of the initial RCF, and its completeness, i.e., is every relation between objects reflected by at least one such relational attribute.

Another challenging research track lays in the combination of RCA with other FCA paradigms such as fuzzy FCA [7] and closed pattern mining [33].

References

1. Mehdi Adda, Petko Valtchev, Chabane Djeraba, and Rokia Missaoui. Toward recommendation based on ontology-powered web-usage mining. *IEEE Internet Computing*, 11(4):45–52, 2007.
2. Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In *Proceedings of the 11th Intl. Conf. on Data Engineering (ICDE'95), IEEE Computer Society.*, pages 3–14, 1995.
3. Gabriela Arévalo, Jean-Rémi Falleri, Marianne Huchard, and Clémentine Nebut. Building Abstractions in Class Models: Formal Concept Analysis in a Model-Driven Approach. In O. Nierstrasz, J. Whittle, D. Harel, and G. Reggio, editors, *Proc. of the 9th Intl. Conf. MoDELS*, volume 4199 of *LNCS*, pages 513–527. Springer, 2006.
4. Zeina Azmeh, Maha Driss, Fady Hamoui, Marianne Huchard, Naouel Moha, and Chouki Tibermacine. Selection of Composable Web Services Driven by User Requirements. In *ICWS*, pages 395–402. IEEE Computer Society, 2011.
5. Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, Cambridge, UK, 2003.
6. Marc Barbut and Bernard Monjardet. *Ordre et Classification: Algèbre et combinatoire*, volume 2. Hachette, 1970.
7. Radim Belohlavek. Concept lattices and order in fuzzy logic. *Annals of Pure and Applied Logic*, 128:277–298, 2004.
8. Rokia Bendaoud, Amedeo Napoli, and Yannick Toussaint. Formal Concept Analysis: A unified framework for building and refining ontologies. In A. Gangemi and J. Euzenat, editors, *Knowledge Engineering: Practice and Patterns, Proc. of the 16th Intl. Conf. EKAW*, LNCS 5268, pages 156–171, 2008.
9. Diane Cook and Lawrence Holder. *Mining Graph Data*. Wiley-Interscience, 2006.
10. Luc Dehaspe and Hannu Toivonen. Discovery of frequent datalog patterns. *Data Mining and Knowledge Discovery*, 3:7–36, 1999.
11. Sašo Džeroski. Multi-relational data mining: an introduction. *SIGKDD Explor. Newsl.*, 5:1–16, July 2003.
12. Sašo Džeroski and Nada Lavrac, editors. *Relational Data Mining*. Springer, 2001.

13. Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, 1996.
14. Sébastien Ferré, Olivier Ridoux, and Benjamin Sigonneau. Arbitrary Relations in Formal Concept Analysis and Logical Information Systems. In M.-L. Mugnier F. Dau and G. Stumme, editors, *Proc. of the 13th Intl. Conf. on Conceptual Structures (ICCS'05), Kassel, Germany*, volume 3596 of *LNCS*, pages 166–180. Springer, 2005.
15. Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis, Mathematical Foundations*. Springer-Verlag, 1999.
16. Marianne Huchard, Mohamed Rouane-Hacene, Cyril Roume, and Petko Valtchev. Relational concept discovery in structured datasets. *Annals of Mathematics and Artificial Intelligence*, 49(1-4):39–76, 2007.
17. Sergei O. Kuznetsov. Learning of Simple Conceptual Graphs from Positive and Negative Examples. In Jan M. Zytkow and Jan Rauch, editors, *Proc. of the Third European Conf. PKDD'99, Prague, Czech Republic*, volume 1704, pages 384–391, 1999.
18. Sergei O. Kuznetsov and Sergei A. Obiedkov. Comparing the performance of algorithms for generating concept lattices. *Journal of Experimental & Theoretical Artificial Intelligence*, 14(2-3):189–216, 2002.
19. Jens Lehmann and Pascal Hitzler. Concept learning in description logics using refinement operators. *Mach. Learn.*, 78:203–250, January 2010.
20. Michel Liquière and Jean Sallantin. Structural Machine Learning with Galois Lattice and Graphs. In *Proc. of the 15th Intl. Conf. on Machine Learning (ICML'98)*, pages 305–313, 1998.
21. Ronald D. Mann and Elizabeth B. Andrews, editors. *Pharmacovigilance*. John Wiley & Sons Ltd,, 2002.
22. Naouel Moha, Mohamed Rouane-Hacene, Petko Valtchev, and Yann-Gaël Guéhéneuc. Refactorings of Design Defects using Relational Concept Analysis. In R. Medina and S. Obiedkov, editors, *Proc. of the 6th Intl. Conf. on Formal Concept Analysis (ICFCA'08)*, volume 4933 of *LNCS*, pages 289–304. Springer-Verlag, 2008.
23. Susanne Prediger and Gerd Stumme. Theory-driven logical scaling. In *Proc. 6th Intl. Workshop Knowledge Representation Meets Databases, Heidelberg. CEUR Workshop Proc*, pages 46–49, 1999.
24. Susanne Prediger and Rudolf Wille. The Lattice of Concept Graphs of a Relationally Scaled Context. In *Proc. of the 7th Intl. Conf. on Conceptual Structures (ICCS'99)*, pages 401–414. Springer, 1999.
25. Uta Priss. Efficient implementation of semantic relations in lexical databases. *Computational Intelligence*, 15:79–87, 1999.
26. Luc De Raedt. *Logical and Relational Learning (Cognitive Technologies)*. Springer, first edition, 2008.
27. Mohamed Rouane-Hacene, Marianne Huchard, Amedeo Napoli, and Petko Valtchev. A proposal for combining formal concept analysis and description logics for mining relational data. In S. Kuznetsov and S. Schmidt, editors, *Proc. of the 5th Intl. Conf. on Formal Concept Analysis (ICFCA'07)*, volume 4390 of

LNCS, pages 51–65. Springer-Verlag, 2007.
28. Mohamed Rouane-Hacene, Petko Valtchev, and Roger Nkambou. Supporting Ontology Design through Large-Scale FCA-Based Ontology Restructuring. In *Proc. of the 19th International Conference on Conceptual Structures, ICCS 2011 - Conceptual Structures for Discovering Knowledge -, Derby, UK, July 25-29, 2011.*, pages 257–269, 2011.
29. Sebastian Rudolph. Exploring Relational Structures via FLE. In H.D. Pfeiffer K.E. Wolff and H.S. Delugach, editors, *Proc. of the 12th Intl. Conf. on Conceptual Structures, ICCS 2004, Huntsville (AL)*, volume 3127 of *LNAI*, pages 196–212. Springer, Berlin, 2004.
30. Hajer Saada, Xavier Dolques, Marianne Huchard, Clémentine Nebut, and Houari A. Sahraoui. Generation of operational transformation rules from examples of model transformations. In R. France, J. Kazmeier, R. Breu, and C. Atkinson, editors, *Proc. 15th Intl. Conf. MODELS'12, Innsbruck (AT)*, volume 7590 of *Lecture Notes in Computer Science*, pages 546–561. Springer, 2012.
31. Lian Shi, Yannick Toussaint, Amedeo Napoli, and Alexandre Blansché. Mining for reengineering: an application to semantic wikis using formal and relational concept analysis. In G. Antoniou, M. Grobelnik, E. Simperl, B. Parsia, D. Plexousakis, J. Pan, and P. De Leenheer, editors, *Proc. of the 8th Extended Semantic Web Conf. (ESWC'11)*, volume 6644 of *LNCS*, pages 421–435. Springer, 2011.
32. Ramakrishnan Srikant and Rakesh Agrawal. Mining generalized association rules. In *Proc. of the 21st Intl. Conf. on Very Large Databases (VLDB'95), Zurich (CH)*, September 1995.
33. Petko Valtchev, Rokia Missaoui, and Robert Godin. Formal Concept Analysis for Knowledge Discovery and Data Mining: The New Challenges. In P. Eklund, editor, *Proc. of the 2nd Intl. Conf. on Formal Concept Analysis (ICFCA'04), Sydney (AU)*, volume 2961 of *LNCS*, pages 352–371. Springer, 2004.
34. Petko Valtchev, Rokia Missaoui, and Pierre Lebrun. A partition-based approach towards building Galois (concept) lattices. *Discrete Mathematics*, 256(3):801–829, 2002.
35. Petko Valtchev, Mohamed Rouane-Hacene, and Rokia Missaoui. A generic scheme for the design of efficient on-line algorithms for lattices. In A. de Moor, W. Lex, and B. Ganter, editors, *Proc. of the 11th Intl. Conf. on Conceptual Structures (ICCS'03)*, volume 2746 of *LNCS*, pages 282–295. Springer, 2003.
36. Takashi Washio and Hiroshi Motoda. State of the art of graph-based data mining. *SIGKDD Explor. Newsl.*, 5(1):59–68, 2003.
37. Rudolf Wille. Restructuring the Lattice Theory: An Approach Based on Hierarchies of Concepts. In I. Rival, editor, *Ordered sets*, pages 445–470, Dordrecht-Boston, 1982. Reidel.