Annals of Mathematics and Artificial Intelligence (2022) 90:537–567
https://doi.org/10.1007/s10472-022-09790-6

Knowledge cores in large formal contexts

Tom Hanika[1] · **Johannes Hirth**[1]

Accepted: 24 February 2022 / Published online: 20 April 2022

Abstract

Knowledge computation tasks, such as computing a base of valid implications, are often infeasible for large data sets. This is in particular true when deriving canonical bases in formal concept analysis (FCA). Therefore, it is necessary to find techniques that on the one hand reduce the data set size, but on the other hand preserve enough structure to extract useful knowledge. Many successful methods are based on random processes to reduce the size of the investigated data set. This, however, makes them hardly interpretable with respect to the discovered knowledge. Other approaches restrict themselves to highly supported subsets and omit rare and (maybe) interesting patterns. An essentially different approach is used in network science, called k-cores. These cores are able to reflect rare patterns, as long as they are well connected within the data set. In this work, we study k-cores in the realm of FCA by exploiting the natural correspondence of bi-partite graphs and formal contexts. This structurally motivated approach leads to a comprehensible extraction of knowledge cores from large formal contexts.

Keywords k-cores · Bi-Partite graphs · Formal concept analysis · Lattices · Implications · Knowledge base

1 Introduction

Large (binary) relational data sets are a demanding challenge for contemporary knowledge discovery methods that use formal concept analysis [13]. This is due to the fact that many considered problems in this realm are computationally intractable, e.g., enumerating formal concepts, i.e., closed sets, or computing the canonical base [7, 21] of the underlying implicational theory. Moreover, knowledge bases of large data sets may be incomprehensible to human readers due to a large number of artifacts that arise from erroneous or infrequent facts. Different methods were developed to adapt FCA tools to the growth of data sets. Sophisticated algorithms employ filtering for data reduction. For example, formal concepts can be filtered by their support in the data set. This is done in Apriori-like techniques [29,

✉ Tom Hanika
tom.hanika@cs.uni-kassel.de

Johannes Hirth
hirth@cs.uni-kassel.de

[1] Knowledge & data engineering group, Department of Electrical Engineering and Computer Science, University of Kassel, Kassel, Germany

34]. More recent methods consider the minimum description length [10]. However, all these approaches are unable to cope with large relational data sets for two reasons: first, they cannot discover rare combinations of attributes that are (comparatively) highly supported in the data set; secondly, the computations require an infeasible amount of steps. Moreover, commonly employed random approaches fail to discover rare patterns, since low supported combinations are unlikely to be sampled. Other techniques, such as feature combination or object clustering [3, 5] lack in meaningfulness.

In general, there are two approaches to overcome the requirements of large data sets with respect to knowledge discovery. One line of research is to introduce novel knowledge features apart from closed sets and their related notions. This may lead to results that are not accessible through well studied knowledge procedures, e.g., from formal concept analysis. Another line of research develops data reduction procedures such as latent semantic analysis or unsupervised clustering of attributes [3, 5]. These, however, do often lead to unexplainable features.

Our approach is fundamentally different, as we translate a popular graph theoretic notion for data set reduction [1, 8, 14, 19, 20, 25], i.e., k-Cores by Seidman [27], to the realm of formal concept analysis. The inviolable constraint for our investigation is to maintain interpretability as well as explainability of knowledge with respect to the original data set. To this end we study theoretically as well as experimentally the impact of the core reduction process on the conceptual knowledge, i.e., closed sets and implications. Using this, we demonstrate a principle method to discover *interesting cores of knowledge* in large data sets. In detail, we give a formal overview of to be defined pq-cores and their reduction effects on conceptual structures and implicational theories. Furthermore, we provide specifications for choosing interesting cores in large relational data sets. We complement our findings by introducing core knowledge transformation algorithms. For a given data set and an initial pq-core they are able to provide a computationally efficient navigation process in the emerging knowledge structure of all pq-core. Finally, we argue that our methods are able to cope with arbitrary subsets of binary relational data.

The rest of our work is structured as follows. In Section 2 we first recollect common notations from formal concept analysis and introduce cores in formal contexts thereafter in Section 2.1. The related formal concept lattice and canonical base are investigated in Sections 3 and 3. This is followed by an extensive experimental study in Sections 5 and 6 which is concluded by a presentation of efficient algorithms for pq-cores in Section 7. After a discussion of related work in Section 8 we conclude with Section 9.

2 Formal concept analysis

Formal concept analysis (FCA) deals with binary relational data sets [13, 33]. These are represented in a *formal context* (G, M, I) where the finite sets G and M are called *objects* and *attributes*, respectively. The binary relation I between these sets is called *incidence*, where $(g, m) \in I$ is interpreted as "object g has attribute m". Two derivation operators emerge on the power sets of G and M: $\cdot' : \mathcal{P}(G) \to \mathcal{P}(M)$ where $A \mapsto A' := \{m \in M \mid \forall g \in A : (g, m) \in I\}$ and $\cdot' : \mathcal{P}(M) \to \mathcal{P}(G)$ dually. Composing the two operators leads to two *closure operators* (i.e., idempotent, monotone, and extensive maps) on $\mathcal{P}(G)$ and $\mathcal{P}(M)$. We investigate in this work *induced sub-contexts*, i.e., $\mathbb{S} = (H, N, J)$ with $H \subseteq G$, $N \subseteq M$, and $J = I \cap (H \times N)$, denoted by $\mathbb{S} \leq \mathbb{K}$. When multiple formal contexts are in play we often use the incidence relation for indicating a derivation, e.g., $\{g\}^I$ for a derivation of $g \in G$ in $\mathbb{K}$ and $\{g\}^J$ for a derivation of $g \in H$ in $\mathbb{S}$. A *formal* *concept* is a pair $(A, B) \in \mathcal{P}(G) \times \mathcal{P}(M)$ with $A' = B$ and $A = B'$. We call A the *extent* and B the *intent* of (A, B) and denote with $Ext(\mathbb{K})$ and $Int(\mathbb{K})$ the sets of all extents and intents respectively. The set of all formal concepts of $\mathbb{K}$ is denoted by $\mathfrak{B}(\mathbb{K})$. This set can be ordered by $\leq$ where $(A, B) \leq (C, D) :\Leftrightarrow A \subseteq C$ for $(A, B), (C, D) \in \mathfrak{B}(\mathbb{K})$. The ordered set of all formal concepts is denoted by $\underline{\mathfrak{B}}(\mathbb{K})$. The fundamental theorem of FCA states that $\underline{\mathfrak{B}}(\mathbb{K})$ is a (complete) lattice. Furthermore, we investigate *implications*, i.e., $A \to B$, where $A, B \subseteq M$. We say $A \to B$ is valid iff $A' \subseteq B'$. The set of all valid implications is denoted by $Th(\mathbb{K})$. Usually, one does work with a base of the theory, e.g., Duquenne–Guigues base [15] (*canonical base*), denoted by $\mathcal{C}_{\mathbb{K}}$. It can be computed using *pseudo-intents*, i.e., $P \subseteq M$ with $P \neq P''$ and $Q'' \subsetneq P$ holds for every pseudo-intent $Q \subsetneq P$. The recursive nature of this definition is by design. Despite being the minimal base of the implications from $Th(\mathbb{K})$, the set of all pseudo-intents can still be exponential in the size of the context [21].

2.1 Cores in formal contexts

Our theory on pq-cores is based on bipartite cores [1, Section 3.1]. We translated their approach to the realm of formal concept analysis, exploiting the natural correspondence between bipartite graphs and formal contexts. This results in the following definition.

Definition 1 Let $\mathbb{K} = (G, M, I), \mathbb{S} = (H, N, J)$ be formal contexts with $\mathbb{S} \leq \mathbb{K}$. We call $\mathbb{S}$ a *pq-core* of $\mathbb{K}$ for $p, q \in \mathbb{N}$, iff

1. $\mathbb{S}$ is *pq-dense*, i.e.,
$$\forall g \in H, \forall m \in N : |\{g\}^J| \geq p \wedge |\{m\}^J| \geq q,$$
2. $\mathbb{S}$ is *maximal*, i.e.,
$$\nexists \mathbb{O} \leq \mathbb{K} : \mathbb{O}\ pq\text{-dense} \wedge \mathbb{S} \neq \mathbb{O} \wedge \mathbb{S} \leq \mathbb{O}.$$

We denote this by $\mathbb{S} \leq_{p,q} \mathbb{K}$. In particular we call contexts $\mathbb{S}$ with $\mathbb{S} \leq_{0,q} \mathbb{K}$ an *attribute-core* and $\mathbb{S} \leq_{p,0} \mathbb{K}$ an *object-core*. We may note, that for any context $\mathbb{K}$ there exists a $p, q \in \mathbb{N}$ such that $(\emptyset, \emptyset, \emptyset)$ is the pq-core of $\mathbb{K}$. This is particular true for $p > |M|$ and $q > |G|$.

Proposition 1 (Uniqueness) *Let $\mathbb{K}$ be a formal context and $p, q \in \mathbb{N}$. Then there exists a unique $\mathbb{S} \leq \mathbb{K}$ with $\mathbb{S} \leq p, q\mathbb{K}$.*

Proof Let $\mathbb{S} = (H, N, J)$ and $\mathbb{T} = (U, V, L)$ be two different formal contexts with $\mathbb{S} \leq \mathbb{K}$ and $\mathbb{T} \leq \mathbb{K}$. Furthermore, for some $p, q \in \mathbb{N}$ we have that $\mathbb{S} \leq_{p,q} \mathbb{K}$ and $\mathbb{T} \leq_{p,q} \mathbb{K}$. Construct the context $\mathbb{D} = (H \cup U, N \cup V, J \cup L)$. Then it follows that

$$\forall g \in H \cup U, \forall m \in N \cup V : \left|\{g\}^{J \cup L}\right| \geq p \wedge \left|\{m\}^{J \cup L}\right| \geq q.$$

Hence, $\mathbb{D}$ is pq-dense and a $\mathbb{S}, \mathbb{T}$ are proper sub-contexts of $\mathbb{D}$. This contradicts the maximality of $\mathbb{S}$ and $\mathbb{T}$.

Given $p, q \in \mathbb{N}$ we can construct the pq-core by filtering the context's object and attribute sets until they satisfy the pq-core property. □

Based on this result we refer to $\mathbb{S} \leq_{pq} \mathbb{K}$ as *the* pq-core. We depict the formal context of an example pq-core in Fig. 1. On the left is the formal context of the prominent "Living beings and Water" example from Ganter & Wille [13] and on the right is the 4,3-core of it. We observe that the objects "Bean" and "Leech" as well as the attributes "suckles its

	1	2	3	4	5	6	7	8	9
1				×	×	×			×
2	×	×	×			×			
3	×	×		×		×		×	
4	×	×	×	×		×			
5	×		×			×			
6				×	×	×	×		
7			×	×	×	×	×		
8			×		×	×	×		

	1	2	3	4	5	6	7
2	×	×	×			×	
3	×	×		×		×	
4	×	×	×	×		×	
6				×	×	×	×
7			×	×	×	×	×
8			×		×	×	×

Attributes: 1. Can move around, 2. has limbs, 3. lives in water, 4. lives on land, 5. needs chlorophyll, 6. needs water, 7. one seed leafs, 8. suckles its offspring, 9. two seed leafs; **Objects**: 1. Bean, 2. Bream, 3. Dog, 4. Frog, 5. Leech, 6. Maize, 7. Reed, 8. Spike-weed

Fig. 1 *Living Beings and Water* context (l) and it's 4,3-core (r)

offspring" and "two seed leafs" are removed. Even though $|\{\text{Bean}\}'| \geq 4$ it is removed by a cascading effect triggered by the removal of the attribute "two seed leaves".

3 Concept lattices of *pq*-Cores

In this section we investigate the relation of the concept lattice for a pq-core to the concept lattice of the originating formal context. We investigate in particular the influence of the parameters p and q. The computation of the pq-core for some p, q can be understood as a sequential removal of objects and attributes in arbitrary order. Based on this observation we analyze the impact of object and attribute removal on concept lattices. To this end, we first take a look at a proposition about structural embeddings. For some $X \subseteq \mathfrak{B}(\mathbb{K})$ we use the notation $\bigvee X$ for the *supremum* of X in $\underline{\mathfrak{B}}(\mathbb{K})$ and $\bigwedge X$ for the *infimum* of X in $\underline{\mathfrak{B}}(\mathbb{K})$, cf. Ganter & Wille [13].

Proposition 2 ([13, Proposition 31 on page 98])
Let $\mathbb{K} = (G, M, I)$, $\mathbb{T} = (U, M, L)$, and $\mathbb{S} = (G, N, J)$, be formal contexts with $\mathbb{T} \leq \mathbb{K}$ and $\mathbb{S} \leq \mathbb{K}$. Then the mapping $\underline{\mathfrak{B}}(\mathbb{T}) \rightarrow \underline{\mathfrak{B}}(\mathbb{K})$ where (A, B) is mapped to the formal concept (B^I, B) is a $\bigvee$-preserving order-embedding of $\underline{\mathfrak{B}}(\mathbb{T})$ in $\underline{\mathfrak{B}}(\mathbb{K})$. Dually, the map $\underline{\mathfrak{B}}(\mathbb{S}) \rightarrow \underline{\mathfrak{B}}(\mathbb{K})$ with $(A, B) \mapsto (A, A^I)$ is a $\bigwedge$-preserving order embedding of $\underline{\mathfrak{B}}(\mathbb{S})$ in $\underline{\mathfrak{B}}(\mathbb{K})$.

For $\mathbb{K}$ we observe that Proposition 2 is not applicable since a pq-core has potentially a modified set of objects and attributes with respect to $\mathbb{K}$. Nonetheless, we can still exploit Proposition 2 in the following way. First, there exists an order-embedding of $\underline{\mathfrak{B}}$(H,M,I ∩ H × M) into $\underline{\mathfrak{B}}(\mathbb{K})$. Secondly, there is an order-embedding from $\underline{\mathfrak{B}}(\mathbb{S})$ into $\underline{\mathfrak{B}}$(H,M,I ∩ H × M). Hence, it is easy to see that the composition of the two maps results in an order-embedding from $\underline{\mathfrak{B}}(\mathbb{S})$ into $\underline{\mathfrak{B}}(\mathbb{K})$. However, suprema and infima are not necessarily preserved. Nonetheless, the existence of the order-embedding does in particular imply that a significant amount of structural (conceptual) information is preserved by the pq-core with respect to the lattice $\underline{\mathfrak{B}}(\mathbb{K})$ and $p, q \in \mathbb{N}$.

In the following we want to investigate more thoroughly how concepts change when objects/attributes are deleted or added. We start with recalling a fact from [13, p. 99] which is related to [13, Proposition 30 on p. 98]. It describes how attribute closures alter when attributes are removed.

Lemma 1 *Let $\mathbb{K} = (G, M, I), \mathbb{S} = (G, N, J)$ be two formal contexts with $\mathbb{S} \leq \mathbb{K}$ and $B \subseteq N$, then:*

i) $B^{JJ} = B^{II} \cap N,$
ii) $B^{II} = B^{JJ} \cup (B^{II} \setminus N).$

Proof Initially, we note that $B^I = B^J$, since $\mathbb{S} \leq \mathbb{K}$ implies $J = I \cap G \times N$.

$i)$ $\subseteq$: From $B^I = B^J$ we can infer $B^{JJ} \subseteq B^{II}$. Furthermore, since $B^{JJ} \subseteq N$ it follows that $B^{JJ} \subseteq B^{II} \cap N$. $\supseteq$: Based on our initial observation we can deduce that for $n \in B^{II} \cap N$ we find $n \in B^{JJ}$. Based on i) we can write ii) as $(B^{II} \cap N) \cup (B^{II} \setminus N)$ which equals B^{II}. □

Proposition 3 (Deleting Attributes) *Let $\mathbb{K} = (G, M, I)$ and $\mathbb{S} = (G, N, J)$ be formal contexts with $\mathbb{S} \leq \mathbb{K}$. Then, $Int(\mathbb{S}) = \{B \cap N | B \in Int(\mathbb{K})\}$.*

Proof $\subseteq$: For $B \in Int(\mathbb{S})$ $B^{II} \in Int(\mathbb{K})$ with $B^{II} \cap N = B$ due to i) in Lemma 1. $\supseteq$: Since $J = I \cap G \times N$ it holds for $B \in Int(\mathbb{K})$ that $B^{IJ} = B \cap N$. Furthermore, $B^{IJ} \in Int(\mathbb{S})$ since B^{IJ} is the object derivation of $B^I \subseteq G$ in $\mathbb{S}$. Thus, we find $B \cap N \in Int(\mathbb{S})$. □

The extent of an $B \in Int(\mathbb{S})$ is equal to the extent of B^{II}, since for $B \subseteq N$ $B^J = B^I = (B^{II})^I$. Note that B^{II} is the inclusion minimal set in $Int(\mathbb{K})$ whose intersection with N equals B due to the monotony of closure operators. Furthermore, for all supersets D of B^{II} in $Int(\mathbb{K})$ it holds $D^I \subseteq (B^{II})^I$. Thus, the following equality holds $B^J = \bigcup\{D^J | D \in Int(\mathbb{K}) : D \cap N = B\}$ and is useful for algorithms as seen in the later part of this work.

Proposition 4 (Adding Attributes) *Let $\mathbb{K} = (G, M, I)$ and $\mathbb{S} = (G, N, J)$ be formal contexts where $\mathbb{S} \leq \mathbb{K}$ is true. Then,*

i) $\forall B \in Int(\mathbb{K}) \setminus Int(\mathbb{S}) : B \setminus N \neq \emptyset,$
ii) $\forall B \in Int(\mathbb{S}) \setminus Int(\mathbb{K}) : \exists D \in Int(\mathbb{S}) \setminus Int(\mathbb{S})$ *with* $D \cap N = B,$

Proof Follows from Lemma 1 ii). Based on Proposition 3 there is a $D {\in} Int(\mathbb{K})$ with
$D \cap N = B$, i.e., $D = B^{II}$. Since $B \notin Int(\mathbb{K})$ the set $B^{II} \setminus N$ is not empty and thus $D \in Int(\mathbb{K}) \setminus Int(\mathbb{S})$. □

Based on the insights so far we may draw a proposition that will drive our to be proposed pq-core-algorithm. It will employ an identity: For $\mathbb{K}t = (G, M, I)$ and $\mathbb{S} = (G, N, J)$ it holds that $Int(\mathbb{K}) = (Int(\mathbb{S}) \cup (Int(\mathbb{K}) \setminus Int(\mathbb{S}))) \setminus (Int(\mathbb{S}) \setminus Int(\mathbb{K}))$.

Proposition 5 *Let $\mathbb{K} = (G, M, I)$ and $\mathbb{S} = (G, N, J)$ with $\mathbb{S} \leq \mathbb{K}$. We can enumerate the elements of the symmetric difference*

$$\triangle(Int(\mathbb{K}), Int(\mathbb{S})) := (Int(\mathbb{K}) \setminus \ Int(\mathbb{S})) \cup (Int(\mathbb{S}) \setminus Int(\mathbb{K}))$$

in

$$O\left(|(Int(\mathbb{K}) \setminus Int(\mathbb{S}))| \cdot (|G|^2 \cdot |M| + |G| \cdot |N|)\right).$$

Proof We use the well-known `next_closure` algorithm [11]. For a given context, this algorithm enumerates the set of all concepts in $O(|G|^2 \cdot |M|)$ time per concept. Let $\leq_M$ be a total order on M such that $\forall m \in M \setminus N \ \forall n \in N : m \leq_M n$. Our enumeration algorithm starts with N, which is the largest closure in $Int(\mathbb{S})$. All concepts enumerated by `next_closure` have at least one element of $M \setminus N$. This follows from the lectic order on M induced by $\leq_M$. We know from Proposition 4 i) that the enumerated set is equal to $Int(\mathbb{K}) \setminus Int(\mathbb{S})$. The so far not enumerated elements from $\triangle(Int(\mathbb{K}), Int(\mathbb{S}))$ are $Int(\mathbb{S}) \setminus Int(\mathbb{K})$. Due to Proposition 4 ii) we know that they can be determined by intersecting elements from $Int(\mathbb{K}) \setminus Int(\mathbb{S})$ with N and applying the closure operator of $\mathbb{S}$, which takes $O(|G| \cdot |N|)$ time. □

Since we want to explain the relation of pq-core lattices to the concept lattice of the original lattice we may state the following.

Corollary 1 *Let* $\mathbb{K} = (G, M, I)$ *and* $\mathbb{S} = (G, N, J)$ *with* $\mathbb{S} \leq \mathbb{K}$. *One can enumerate the elements of the symmetric difference* $\triangle(\mathfrak{B}(\mathbb{S}), \mathfrak{B}(\mathbb{K}))$ *in time*

$$O\left(|(Int(\mathbb{K}) \setminus Int(\mathbb{S}))| \cdot (|G|^2 \cdot |M| + |G| \cdot |N|)\right).$$

The enumeration problem from Corollary 1 is up to one attribute-derivation similar to Proposition 5.

We also see that all results in this section about attribute operations can be translated to object operations through duality. After the theoretical consideration on the impact of adding/removing attributes to formal contexts we now want to look into the dependence of pq-cores to removing objects.

Proposition 6 (Object Cores) *For two formal contexts* $\mathbb{K}$ *and* $\mathbb{S}$ *with* $\mathbb{S} \leq_{p,0} \mathbb{K}$ *and* $\mathcal{F} := \{B \in Int(\mathbb{K}) \mid |B| \geq p\}$ *the equality*

$$\{\bigcap \mathcal{X} \mid \mathcal{X} \subseteq \mathcal{F}\} = Int(\mathbb{S}) \quad \textit{holds.}$$

Proof $\subseteq$: Since $\mathbb{S}$ is p, 0-core of $\mathbb{K}$ we have that $\forall B \subseteq M : |B| \geq p \Rightarrow B^{II} = B^{JJ}$. Hence, $\forall X \in \mathcal{F} : X^{II} = X^{JJ} \in Int(\mathbb{S})$. Since $Int(\mathbb{S})$ is closed under intersection [13] we find that for all $\mathcal{X} \subseteq \mathcal{F} : \bigcap \mathcal{X} \in Int(\mathbb{S})$. $\supseteq$: Assume $\exists B \in Int(\mathbb{S})$ with $B \neq \bigcap \mathcal{X}$ for all $\mathcal{X} \subseteq \mathcal{F}$. Therefore, by construction of $\mathcal{F}$ we know that $|B| < p$, since $Int(\mathbb{S}) \subseteq Int(\mathbb{K})$[13, dual of Proposition 31]. Without loss of generality B is *meet-irreducible* in $Int(\mathbb{S})$, i.e., there is no $\mathcal{Y} \subseteq \mathcal{F} : \bigcap \mathcal{Y} = B$. We may note that meet-irreducible among intents means join-irreducible among the respective concepts (B^J, B) due to the dual order. Thus, there exists an object g of the formal context $\mathbb{S} = (H, N, J)$ with $g^J = B$, since meet-irreducibles intents are known to be among object concepts. This contradicts $|\{g\}^J| \geq p$. □

This proof employs meet-irreducible intents of $Int(\mathbb{K})$. Computing those can be achieved in time polynomial in the size of $\mathbb{K}$ using join-irreducible concepts and the down-arrow relation [13, Definition 25]. For $g \in G$ and $m \in M$ is $g \swarrow m :\Longleftrightarrow (g, m) \notin I$ and for all $h \in G : g' \subsetneq h' \implies (h, m) \in I$. Based on this definition, join-irreducible concepts are (g'', g') for which there exists am $m \in M$ with $g \swarrow m$. Alternatively, we can employ the cover relation among concepts in $\underline{\mathfrak{B}}(\mathbb{K})$. In this relation the meet-irreducible elements of $\mathfrak{B}(\mathbb{K})$ are the concepts with exactly one lower neighbor. The above impacts the computation of pq-core concept lattices.

Remark 1 For $\mathbb{S} = (H, N, J)$ and $\mathbb{K} = (G, M, I)$ with $\mathbb{S} \leq_{p,q} \mathbb{K}$ it holds that $\mathbb{S} \leq_{p,0} (G, N, I \cap G \times N)$.

Using this remark we find a useful correspondence between the concept lattices of a context, its induced sub-contexts and, in particular, its cores. For any two pq-cores $\mathbb{S} \leq_{p,q} \mathbb{K}$ and $\mathbb{T} \leq_{\hat{p},\hat{q}} \mathbb{K}$ we are able to efficiently compute (Corollary 1) the difference in their concept lattices, i.e., $\triangle(\underline{\mathfrak{B}}(\mathbb{S}), \underline{\mathfrak{B}}(\mathbb{T}))$. This difference, we claim, enables *navigating* the order structure of all pq-cores for a given formal context $\mathbb{K}$, see Fig. 2.

Given formal context $\mathbb{K}$, the set $\mathcal{K} := \{\mathbb{S} \leq \mathbb{K}\}$ constitutes a complete lattice. One can see this using the map $\mathcal{P}(G) \times \mathcal{P}(M) \to \mathcal{K}$, $(H, N) \mapsto (H, N, I \cap (H \times N))$, which is an order isomorphism from the lattice $\mathcal{P}(G) \times \mathcal{P}(M)$ to $\mathcal{K}$. Hence, for two arbitrary induced sub-contexts $\mathbb{S} = (H, N, J)$ and $\mathbb{T} = (U, V, L)$ of $\mathbb{K} = (G, M, I)$ one may compute $\mathfrak{B}(\bigvee\{\mathbb{S}, \mathbb{T}\})$ and $\mathfrak{B}(\bigwedge\{\mathbb{S}, \mathbb{T}\})$ in order to infer $\mathfrak{B}(\mathbb{T})$ efficiently using $\mathfrak{B}(\mathbb{S})$, or vice versa. The set of all pq-cores is contained in $\mathcal{K}$, however, it does not constitute a lattice. To see this a counter example is presented in Fig. 3.

3.1 A Small Case Study

We apply our notion for pq-cores on a particularly small example, the *Forum Romanum* (FR) context (cf. Ganter & Wille [13, Figure 1.16]), in order to study the applicability to real world data sets. The data set consists of monuments on the Forum Romanum (objects) and their star ratings by different travel guides (attributes). In Fig. 4 we depicted the concept lattice for FR and indicated by the red dashed lines the 2,4-core of FR.

At least all concepts between the red lines remain after the core reduction. In detail, the parameter $p = 2$ results in removing all objects that have a derivation of size less than two, as indicated by the upper horizontal dashed line. We understand (acc. to Proposition 6) that in this process all *join-irreducible* concepts (A, B), i.e., $\nexists \mathcal{F} \subseteq \mathfrak{B}(\text{FR}) : \bigvee \mathcal{F} = (A, B)$, above the $p = 2$ threshold are removed. For example, the concepts above the horizontal red dashed line having the shorthand notation labels *B**, *GB**, and *P** are join-irreducible and therefore removed. Their attributes are then contained by those lower concepts that are in cover relation to the removed concepts. In contrast, the concept with shorthand label *M** is

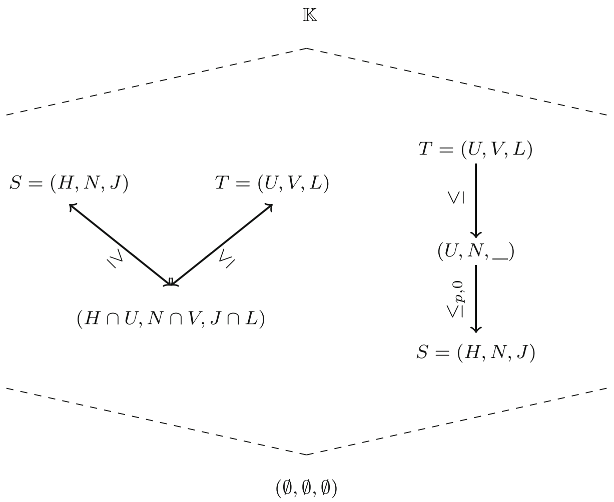


Fig. 2 Principle approach for analyzing multiple pq-cores from a formal context $\mathbb{K}$ (left) and their order/lattice relation (right)

	a_1	a_2	a_3	a_4	a_5	b_1	b_2	b_3	b_4	b_5	c_1
a_1	×	×	×	×	×						×
a_2	×	×	×								
b_1						×	×	×			
b_2						×	×		×		
b_3						×	×			×	
c_1				×	×						

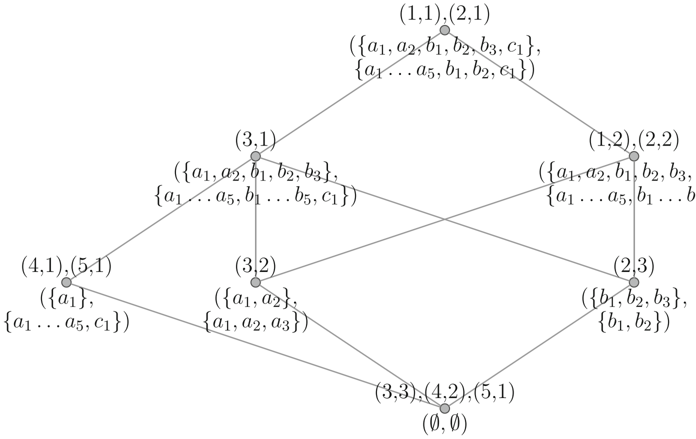

(1,1),(2,1)
$(\{a_1, a_2, b_1, b_2, b_3, c_1\},$
$\{a_1 \ldots a_5, b_1, b_2, c_1\})$
(3,1)
$(\{a_1, a_2, b_1, b_2, b_3\},$
$\{a_1 \ldots a_5, b_1 \ldots b_5, c_1\})$
(1,2),(2,2)
$(\{a_1, a_2, b_1, b_2, b_3, c_1\},$
$\{a_1 \ldots a_5, b_1 \ldots b_5\})$
(4,1),(5,1)
$(\{a_1\},$
$\{a_1 \ldots a_5, c_1\})$
(3,2)
$(\{a_1, a_2\},$
$\{a_1, a_2, a_3\})$
(2,3)
$(\{b_1, b_2, b_3\},$
$\{b_1, b_2\})$
(3,3),(4,2),(5,1)
$(\emptyset, \emptyset)$

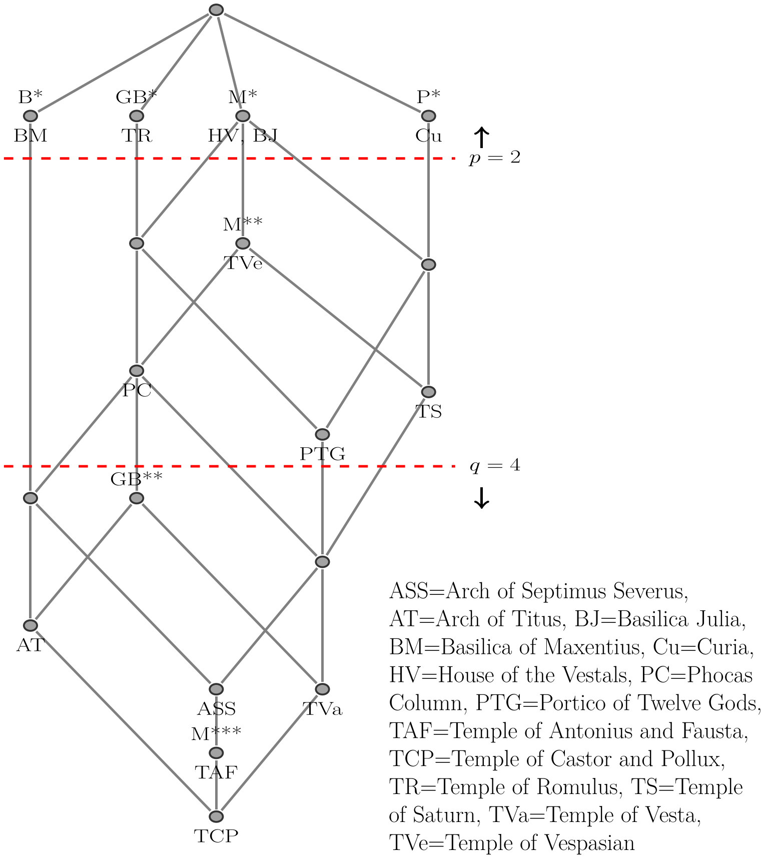
Fig. 3 An example context (upper) and the order relation of all pq-cores (lower). Each node in the order diagram represents a pq-core with its p, q values written above the node

join-reducible and is therefore closed after the removal of objects. The removal of attributes results in dual observations, i.e., *meet-irreducible* concepts are removed.

4 Implications of *pq*-Cores

Any pq-core allows for computing its canonical base of valid implications. The question at hand is, to what extent can the rules found be applied to the original data set? We start with investigating the impact of object set manipulations on implications. Consider the following two formal contexts $\mathbb{K} = (G, M, I)$ and $\mathbb{S} = (H, M, J)$ with $\mathbb{S} \leq \mathbb{K}$. By removing objects, i.e., there are objects present in G that are missing in H, we possibly remove unique counterexamples $g \in G$. Hence, implications previously invalid in $\mathbb{K}$, e.g., $A \to B$ with $A, B \subseteq M$ and $A^I \not\subseteq B^I$, become valid in $\mathbb{S}$, i.e., $A^J \setminus \{g\} \subseteq B^J$ Therefore, new valid implications may emerge in $\mathbb{S}$. On the other hand, valid implications in $\mathbb{K}$ cannot be disproved by removing objects. Thus, $Th(\mathbb{K}) \subseteq Th(\mathbb{S})$. Cores with $p \in N$ and $q = 0$ are of particular interest to us due to Remark 1. For those, i.e., $\mathbb{S} \leq_{p,0} \mathbb{K}$, we find that all valid implications $A \to B$ in $Th(\mathbb{S}) \setminus Th(\mathbb{K})$ have $|A < p$, since in this core we only remove objects $g \in G$ with $|\{g\}'| < p$. Hence, these are only able to refute implications with premise $|A| < p$. For the special case of $\mathbb{S} \leq_{0,q} \mathbb{K}$ we can deduce that $Th(\mathbb{S}) \subseteq Th(\mathbb{K})$.

There are two essential notions when discussing implications in data sets, *confidence* and *support*. Given $A, B \subseteq M$, the support of an implication $A \to B$ in $\mathbb{K}$ is defined by



ASS=Arch of Septimus Severus, AT=Arch of Titus, BJ=Basilica Julia, BM=Basilica of Maxentius, Cu=Curia, HV=House of the Vestals, PC=Phocas Column, PTG=Portico of Twelve Gods, TAF=Temple of Antonius and Fausta, TCP=Temple of Castor and Pollux, TR=Temple of Romulus, TS=Temple of Saturn, TVa=Temple of Vesta, TVe=Temple of Vespasian

Fig. 4 The 2,4-core of the concept lattice is indicated by the red lines. All objects present in the shorthand notation above the $p = 2$ barrier are removed as well as all attributes below $p = 4$ line are removed

$\operatorname{sup}_{\mathbb{K}}(A \to B) := {}^{|(A\cup B)'|}/_{|G|}$ and the confidence of $A \to B$ in $\mathbb{K}$ by $\operatorname{conf}_{\mathbb{K}}(A \to B) := {}^{|(A\cup B)'|}/_{|A'|}$. We may note that only implications with confidence one are considered valid in FCA and therefore included in $Th(\mathbb{K})$.

Proposition 7 (Core Implications) *Let $\mathbb{K}, \mathbb{S}$ be formal contexts, with $\mathbb{S} \leq_{p,q} \mathbb{K}$ where $\mathbb{K} = (G, M, I)$ and $\mathbb{S} = (H, N, J)$. For all $A \to B \in Th(\mathbb{S})$ it holds that*

i) ${}^{|H|}/_{|G|} \cdot \operatorname{sup}_{\mathbb{S}}(A \to B) \leq \operatorname{sup}_{\mathbb{K}}(A \to B)$,
ii) $\operatorname{sup}_{\mathbb{K}}(A \to B) \leq {}^{|H|}/_{|G|} \cdot \operatorname{sup}_{\mathbb{S}}(A \to B) + {}^{|G\setminus H|}/_{|G|}$,
iii) $\operatorname{conf}_{\mathbb{K}}(A \to B) \geq {}^{|(A\cup B)^J|}/_{|A^J|+|G\setminus H|}$,
iv) $|A| \geq p \implies \operatorname{conf}_{\mathbb{K}}(A \to B) = 1$,
v) $|A \cup B| \geq p \implies \operatorname{sup}_{\mathbb{K}}(A \to B) = {}^{|H|}/_{|G|} \cdot \operatorname{sup}_{\mathbb{S}}(A \to B)$.

Proof i) Since $J \subseteq I$ we can infer that $|A^J| \leq |A^I|$ and that ${}^{|H|}/_{|G|} \cdot \operatorname{sup}_{\mathbb{S}}(A) \leq \operatorname{sup}_{\mathbb{K}}(A)$. ii) With the same argument as in i) we can find $|A^I| \leq |A^J| + |G \setminus H|$, from which one can deduce the statement.
Using i) and ii), which would be the best-case / worst-case for supports, since all additional objects are counter examples for $A \to B$, we find $\operatorname{conf}_{\mathbb{K}}(A \to B) = {}^{|(A\cup B)^I|}/_{|A^I|}$ is greater

than or equal to $|H|/|G| \cdot \sup_{\mathbb{S}}(A \to B)$ divided by $|H|/|G| \cdot \sup_{\mathbb{S}}(A) + |G \setminus H|/|G|$. This can be simplified to $\operatorname{conf}_{\mathbb{K}}(A \to B) \geq |(A \cup B)^J|/|A^J|+|G\setminus H|$.
For $|A| \geq p$ we have $A^I = A^J$ by definition of pq-cores and also $(A \cup B)^J = (A \cup B)^I$. Together with the definition of confidence we obtain the statement.
With $|A \cup B| \geq p$ we see that $|(A \cup B)^I| = |(A \cup B)^J|$, which results in special case of i). □

Note that i), ii), and iii) are also valid for arbitrary sub-contexts. Based on this result we study minimal representations of implicational theories, i.e., the canonical base of $Th(\mathbb{K})$. So the next logical question is on how to (partially) derive the canonical base for some formal context $\mathbb{K}$ using one of its pq-cores. However, this endeavor is so far not understood, with the exception of simple cases. For example, when computing the canonical base of $(G, N_1 \dot{\cup} N_2, J_2 \dot{\cup} J_2)$ using the bases of (G, N_1, J_1), (G, N_2, I_2), there is a simple solution [32].

Another important base for a set of implications is its *canonical direct base* [6, 13] (CDB), i.e., a complete, sound and iteration-free base. Such a set of implications for a formal context $\mathbb{K} = (G, M, I)$ is constituted by the set of *proper premises*, i.e., sets $A \subseteq M$ where $A^{II} \setminus (A \cup \bigcup_{B \subsetneq A} B^{II}) \neq \emptyset$ does hold, cf. [12].

Proposition 8 (Induced Contexts CDB) *Let* $\mathbb{K} = (G, M, I)$, $\mathbb{S} = (G, N, J)$ *be two formal contexts with* $\mathbb{S} \leq \mathbb{K}$ *and let* $\mathcal{L}_p(\mathbb{S})$, $\mathcal{L}_p(\mathbb{K})$ *be their canonical direct bases, then*

$$\mathcal{L}_p(\mathbb{S}) \subseteq \mathcal{L}_p(\mathbb{K}).$$

Proof Let $A \subseteq N$ be a proper premise of $\mathbb{S}$. Hence, we know by definition that $A^{JJ} \setminus (A \cup \bigcup_{B \subset A} B^{JJ}) \neq \emptyset$. Following, there is an $n \in A^{JJ} \subseteq N$ with $n \notin A$ and $n \notin B^{JJ}$ for all $B \subsetneq A$. With Lemma 1, we find that forall $B \subset A$ we have $B^{II} = B^{JJ} \cup (B^{II} \setminus N)$. Therefore, we find that $n \notin B^{II}$. From this we can conclude that $n \in A^{II} \setminus (A \cup \bigcup_{B \subset A} B^{II})$ which is therefore not empty. □

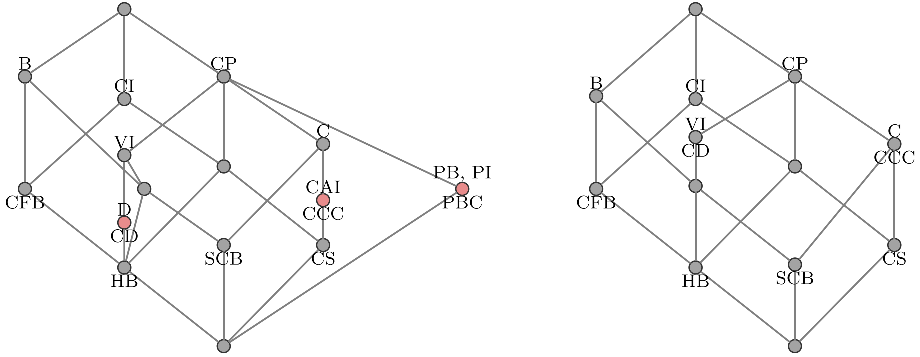
4.1 A Small Case Study on *pq*-core Implications

We want to motivate the applicability of our theoretical findings with a case study on a small real-world data set, the *Ben and Jerry's* context $\mathbb{K}_{\mathrm{BJ}}$ (Fig. 5, left) and its 2,3-core $\mathbb{S}_{\mathrm{BJ}}$ (Fig. 5, right). This context contains seven ice cream flavors of the Ben and Jerry's brand (objects) and nine ingredients (attributes). The incidence indicates which ingredients are included in an ice cream flavor. The 2,3-core consists of six flavors and five ingredients. We also depicted their concept lattices (Fig. 5). The original context $\mathbb{K}_{\mathrm{BJ}}$ has sixteen concepts and the pq-core $\mathbb{S}_{\mathrm{BJ}}$ has thirteen. In $\mathbb{S}_{\mathrm{BJ}}$ we note the absence of one object (*PBC*), four attributes (*PB, PI, CI, D*) and almost a third of the incidence pairs of I_{BJ}. However, we observe that the concept lattices of $\mathbb{K}_{\mathrm{BJ}}, \mathbb{S}_{\mathrm{BJ}}$ are very similar in terms of their structure.

Using the findings of the previous section, we can partially analyze the implicational structure of the original context $\mathbb{K}_{\mathrm{BJ}}$ through the use of $\mathbb{S}_{\mathrm{BJ}}$. We do this in terms of the canonical direct basis. The basis of $\mathbb{K}_{\mathrm{BJ}}$ contains thirty-two proper implications and the basis of the $\mathbb{S}_{\mathrm{BJ}}$ contains six. Again, this is surprising given the similarity of the concept lattices. The set of all proper implications of $\mathbb{S}_{\mathrm{BJ}}$ is:

	B	PB	PI	C	CAI	CI	CP	D	VI
CCC				×	×		×		
CFB	×					×			
CD							×	×	×
HB	×					×	×	×	×
CS				×	×	×	×		
PBC		×	×				×		
SCB	×			×			×		×

	B	C	CI	CP	VI
CCC		×		×	
CFB	×		×		
CD				×	×
HB	×		×	×	×
CS		×	×	×	
SCB	×	×		×	×



Attributes: B=Brownie, PB=peanut butter, PI=peanut ice, C=caramel, CAI=caramel ice, CI=choco ice, CP=choco pieces, D=dough, VI=vanilla ice; **Objects**: CCC=Caramel Chew Chew, CFB=Chocolate FudgeBrownie, CD=Cookie Dough, HB=Half Baked, KS=Karamel Sutra, PBC=Peanut Butter Cup, SCB=Salted Caramel Brownie

Fig. 5 Ben and Jerry's context (left) and its 2,3-core (right)

1) Brownie, Choco P. → Vanilla Ice
2) Brownie, Caramel → Vanilla Ice
3) Caramel → Choco P.
4) Vanilla Ice → Choco P.
5) Vanilla Ice, Caramel → Brownie
6) Vanilla Ice, Choco Ice → Brownie

We note that all implications are supported in $\mathbb{S}_{\mathrm{BJ}}$ and thus in $\mathbb{K}_{\mathrm{BJ}}$. In detail, the support of 1) in $\mathbb{S}_{\mathrm{BJ}}$ is 33% and using Proposition 7 v) we can infer that the support of 1) in $\mathbb{K}_{\mathrm{BJ}}$ is of 28%. As for the confidence of 1), we can use Proposition 7 iv) to deduce that 1) is also an implication of $\mathbb{K}_{\mathrm{BJ}}$, i.e., has confidence 1. For implication 4) we find that its support in $\mathbb{S}_{\mathrm{BJ}}$ is 50% and in $\mathbb{K}_{\mathrm{BJ}}$ 42% (Proposition 7 v)). The confidence of implication 4) can be estimated to be at least 75% (Proposition 7 iii)).

Based on these results, we suggest that the canonical direct base of a pq-core is useful for a meaningful investigation of the implicational structure of large contexts.

5 Experimental study

To support our observations from the last sections, we conducted an experimental study on larger real-world data sets. The most pressing question is to identify particularly interesting cores of a given formal context. A commonly used technique to assess the interestingness of k-cores in networks is to investigate the number of connected components depending on the core parameter k. A well-known observation is that the number of connected components increases the greater k is. Parameters that are considered interesting are those around the steepest rate of increase in the number of components. Also often considered are changes of some valuation function, such as the size of the largest connected component or some network statistical property. We will adapt the former idea and analyze the component structures.

Table 1 Numerical description of data sets. We included the number of non-empty pq-cores as well as the number of formal concepts

Name	$\|G\|$	$\|M\|$	$\|\mathfrak{B}(\mathbb{K})\|$	# pq-cores	density
Water	8	9	19	20	0.47
Romanum	14	7	19	34	0.45
Spices	56	37	421	136	0.23
Knives	159	108	1061	1072	0.11
Mushroom	8124	119	238710	80136	0.22
Wiki44k	45021	101	21923	$\approx$ 98000	0.05

5.0.1 Data Sets

We conduct our investigation on five data sets of different sizes and domains.
Living beings in Water is the well known FCA data set [13, Figure 1.1]. It consists of living beings as objects and their properties as attributes. **Forum Romanum** as already used in Section 3.1, is also taken from [13]. It is made of places of interest as objects and their ratings in different tour guides as attributes. **Spices** is created by the authors. The objects are dishes and the attributes are Spices to be used for these dishes. The incidence relation is extracted from a Spices planer [23]. **Mushroom** is an often used classification data set provided by UCI [9]. The objects are Mushrooms and the non-binary attributes are common Mushroom properties. Those were scaled using a nominal scale. **The Pocket Knives** data set was self-created by the authors through crawling the Victorinox AG website[1] in April 2019. The context contains all pocket knives as objects and their features as attributes. **Wiki44k** was created in an experimental study [18] on finding implications in Wikidata. It is a scaled context drawn from the most dense part of the Wikidata knowledge graph.

All presented data sets are available in the FCA software `conexp-clj` [16] through GitHub.[2] We collected their numerical properties in Table 1.

5.0.2 Interesting *pq*-cores

For all data sets we applied different combinations of parameters p and q and evaluated to what extent this leads to interesting pq-cores using the steepest increase method. For this we regarded all non-empty pq-cores as bipartite graphs and counted the resulting connected components. We observed that no data set has a pq-core with more than one connected component. This is surprising since constructing a formal context falling apart into multiple connected components for some p and q is easy. This might indicate that real-world data sets do not exhibit this property. However, we acknowledge that the number of considered data sets is comparatively low. Nonetheless, this observation might be attributed to the following fact: in all data sets there is a small number of objects with high support, i.e., many attributes, covering in union all attributes and having at least pairwise one attribute in common. These objects are contained in all pq-cores. Hence, we need to adapt the idea of components to the realm of formal contexts differently. For this we consider the context size distribution among all pq-cores. In this distribution we may characterize sub-contexts that are removed

[1] https://www.victorinox.com

[2] https://github.com/tomhanika/conexp-clj

while computing a pq-core as structural components. This is in contrast to the classical component analysis for graph k-cores. Using those we define interesting pq-cores as those where a further increase of p or q would result in a high increase in the size of the removed structural component. In our experiments we find that there are many such critical p and q for the investigated data sets. To narrow this set we propose the following pragmatic selection criteria due to computational limitations: 1. The size of a selected core should be in the range of computational feasibility (with respect to the to be employed analysis procedures). 2. The parameters p and q of a selected core should differ in magnitudes, i.e., either $p \ll q$ or $p \gg q$. The interpretation of either criterion depends on the particular data analysis application. For example, if one is more interested in keeping a larger attribute domain then one should choose an interesting core with low q and high p. Analogously one might want to keep more objects. Furthermore, the second criterion may only be suited for larger contexts with many pq-cores available.

This being said we want to propose a different approach for characterizing interesting pq-cores. In contrast to solely considering a pq-core $\mathbb{S} \leq_{p,q} \mathbb{K}$ of some context $\mathbb{K}$ one might look into the concept lattice that is created by this pq-core, i.e., $\mathfrak{B}(\mathbb{S})$. With this approach the size of the resulting concept lattice could be a criterion to select pq-core. The motivation for this is that we rather select a pq-core depending on the entailed conceptual knowledge than purely on contextual size. This approach is computationally costly since we need to compute a large number of concept lattices. However, relying on Proposition 5, Proposition 6 and Remark 1 we may ease this cost significantly. Analogously we propose selection criteria: 1. The diagram of a selected core lattice should be human readable, (e.g., the number of concepts should be in a human feasible range) 2. The parameters p and q of a selected core lattice should differ in magnitudes, i.e., either $p \ll q$ or $p \gg q$. Again, the concrete employment of either criterion depends on the particular data analysis application. For example, we find a lattice with more than thirty concepts too large for human comprehension, even if drawn with sophisticated drawing algorithms. Hence, we will consider this number for the rest of this work as a bound. In addition to that, we apply the criterion of $p \ll q$ or $p \gg q$ only on larger context. On a final note in this section, we consider the special cases of object- and attribute cores not to be interesting. They remove attributes or objects simply by their object/attribute support and do not represent an interesting sub-structure.

5.0.3 Experiment: Water

We analyze the *living beings and water* context Fig. 1 and present our core analysis in Fig. 6. For this we computed the size of all core concept lattices. A first observation is that interesting cores, with respect to our just introduced notion of interestingness, are the 4,3-, 3,4- and 2,4-core. We suspect that they include important knowledge. Increasing the core parameters more would lead to an (almost) empty concept lattice. From this list of interesting pq-cores we present the lattice diagram of 4,3 in Fig. 6. This lattice contains thirteen formal concepts in contrast to the nineteen in the original concept lattice. The 4,3-core captures a significant portion of knowledge from the original domain, however, only six out of eight objects and seven out of nine attributes are in the picture. We can still infer two different groups of beings, plants and animals. Nonetheless, the original lattice is much more refined. For example, the original concept lattice is more distinct in the subsets of beings that need *chlorophyll* or those who *can move* around. We consider the pq-core to be a more coarse representation of the entailed domain knowledge.

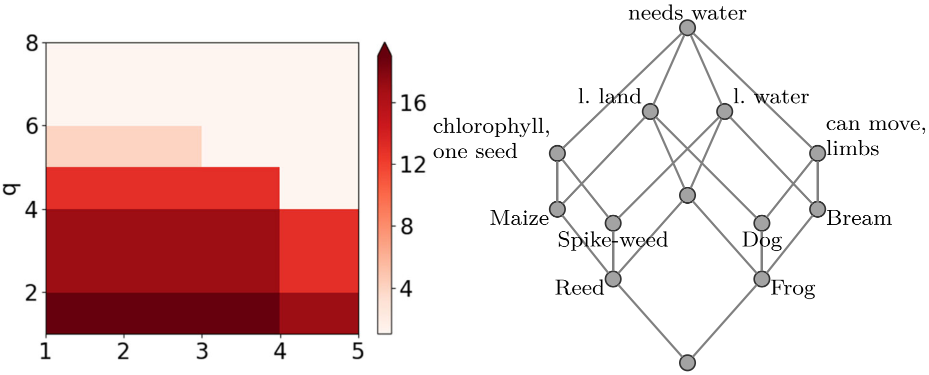


Fig. 6 Figure on the left shows the concept lattice sizes for all pq-cores of living beings and water data set, the abscissa indicates p and the ordinate q. On the right we present the 4,3-core

5.0.4 Experiment: Spices

In this experiment we analyze a spice recommendation data set. This context is derived from a spice planer published in [23]. It contains 56 meals and 37 Spices. Meals in the data set cover nine categories which are not part of the formal context. There are fifteen vegetables, nine meat, three poultry, five fish, five potato, four rice dishes, as well as three sauces, eight baked goods and four diverse dishes. The incidence relation is which meal requires which Spices. The resulting concept lattice of the original context contains 531 formal concepts. The results of applying pq-cores to this data set with different parameters are depicted in Fig. 7. There is a great number of candidate cores to be considered, i.e., cores with a steep decrease in the number of formal concepts while increasing parameters p or q. However, many of those are still very large with respect to the number of formal concepts, e.g., 5,7-core or the 9,4-core. Following our pragmatic criterion for human readability those are not interesting. In contrast is the 5,11-core (cf. light red color in figure) which covers a dense

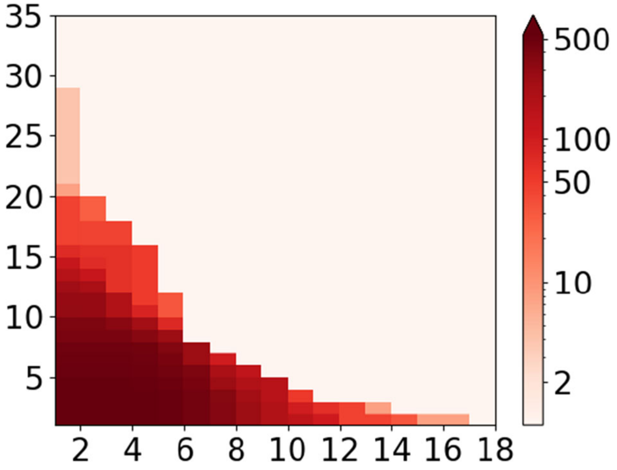


Fig. 7 Concept lattice sizes for all pq-cores of Spices data set, the abscissa indicates p and the ordinate q

object and attribute selection of readable size. There are twelve dishes using six Spices in the 5,11-core.

As another selection we present two different cores exhibiting a large attribute coverage and large object coverage respectively. A real-world motivation for this is: one wants to cook lots of different dishes with possibly fewer Spices; one is focused on a diverse usage of Spices with possibly fewer meals. We exemplify this with the 2,18-core and the 14,1-core, as depicted in Fig. 8. The 2,18-core includes 28 concepts with 33 out of the 56 dishes. The 14,1-core has 32 concepts with 29 out of the 37 Spices. While having less than 10% of the size of the original concept lattice, both concept lattices cover a vast amount of human recognizable knowledge. A thorough investigation with respect to implications is done later in this work.

6 The problem of large contexts

Large formal contexts constitute an infeasible problem for classical formal concept analysis. This is in particular true when computing implicational theories of them. Applying FCA notions only to pq-cores may be a possible resort. However, this results in a large number of pq-cores to be considered, which constitutes a problem in its own, see Table 1. Since our ultimate goal in this work is to present a novel method for coping with large formal contexts, we demonstrate and evaluate an approach for reducing the search space for p and q in this section. For $\mathbb{S} \leq_{p,q} \mathbb{K}$ we know from Proposition 2 that $|\mathfrak{B}(\mathbb{S})|$ decreases monotonously in p and q. Let $\hat{p} \in \mathbb{N}$ be the maximal number such that for all $\mathbb{S} \leq_{p,q} \mathbb{K}$ with $p \geq q$ and $|\mathfrak{B}(\mathbb{S})| \leq 30$ we have that $\mathbb{S} \leq_{p,q} \mathbb{T} \leq_{\hat{p},1} \mathbb{K}$. Furthermore, let $\hat{q} \in \mathbb{N}$ be the maximal number such that for all $\mathbb{S} \leq_{p,q} \mathbb{K}$ with $p < q$ and $|\mathfrak{B}(\mathbb{S})| \leq 30$ we have that $\mathbb{S} \leq_{p,q} \mathbb{T} \leq_{1,\hat{q}} \mathbb{K}$. This implies that cores with human readable sized concept lattices are sub-contexts of particular object- and attribute cores. Our computational approach now is based on finding those particular cores. Equipped with these contexts we only need to consider pq-cores $\mathbb{S} \leq_{p,q} \mathbb{K}$ that are sub-contexts of $\mathbb{T} \leq_{\hat{p},1} \mathbb{K}$ or $\mathbb{T} \leq_{1,\hat{q}} \mathbb{K}$. Since a direct computation of $\hat{p}$ and $\hat{q}$ is infeasible we suggest an estimation. A naïve solution for this would be to examine the derivation size distribution of all objects or attributes. For the data sets investigated in this work this approach was unsuccessful. More fruitful is a binary search among the parameters. We set for this the bound for the concept lattice size to 60 as threshold (which is twice as large as what we consider as readable). Therefore, even if the $\hat{p}$, 1-core is not human readable, we may encounter $\hat{p}, q$-core with $q > 1$ that is readable. A general observation for large formal contexts in the following experiments is that cores with readable concept lattices tend to having extreme values for parameters p, q, i.e., either $p \ll q$ or $q \ll p$.

6.0.5 Binary Search for cores in mushroom

Due to its size (in context as well as in concept lattice terms) the Mushroom data set is an ideal candidate for the just proposed binary search. Computing the sizes of all core concept lattices is infeasible. We search as an initial core for our search paradigm $\hat{q}$ with $p = 1$. We start with $\hat{q} = |G|$, which results almost surely in an empty context for real-world data sets. The binary search in $[1, |G|]$ gives a pq-core with $p = 1$ and $q = 4937$. With 38 formal concepts the concept of this sub-context has less than two times 30 concepts, which we considered human readable. Using this core we reduce the search space to 12832 different p, q, which are all bound by 38 in the number of formal concepts. We may note

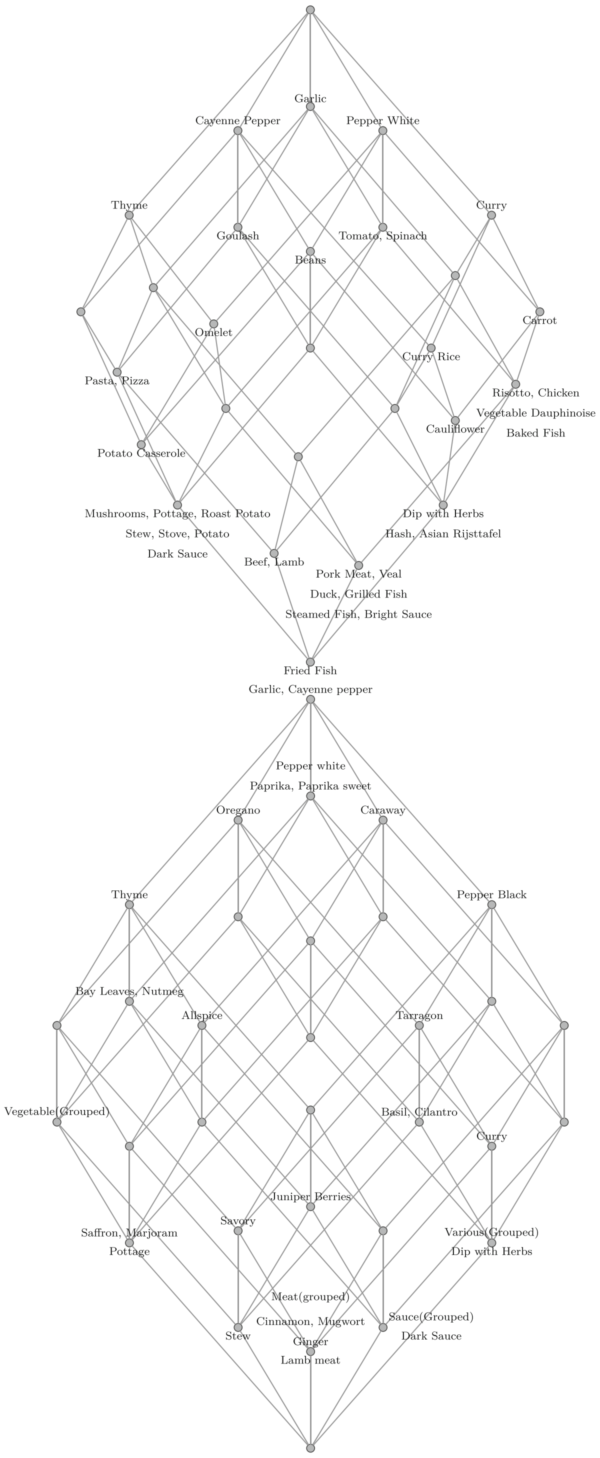


Fig. 8 The concept lattice diagrams of the 2,18-core (top) and the 14,1-core (bottom) of the spice data set

that searching for some $\hat{p}$ is impractical for this data set. This is due to the fact that it was created by scaling twenty-three non-binary attributes into 119 nominal-scaled attributes. Hence, there are only two sub-contexts of the Mushroom context which are in core relation for $q = 1$. More accurately, these are the Mushroom context and the empty context. We depicted a heat-map of the core concept lattices in Fig. 9 for $q \in [4937, 8123]$ and $p \in [1, 5]$. We are interested in cores with as much readable conceptual information as possible, which are cores with $4937 < q < 5176$ (indicated by the darker red color), that are also interesting. Out of those we find the 5,5176-core (at the corner of the dark red area) to be the most interesting. This core contains seven distinct attributes and 7930 Mushrooms. In the depiction of the corresponding concept in Fig. 9 we refrained from annotating all objects and indicated the number of Mushrooms instead (using shorthand notation from FCA). Hence, to get the total number of objects associated to some concept one has to add to the object count all numbers from concepts in the order ideal of that concept. When comparing the core lattice with the original lattice we notice that the object number for all concepts with at least five attributes is similar, which is expected from our theoretical considerations.

6.0.6 Binary search for cores in Wiki44k

To provide another example, we perform the same search in the Wiki44k data set. The corresponding concept lattice contains 21,923 formal concepts and we were able to compute that there are approximately 98,000 non-empty pq-core contexts. Hence, computing all interesting (Section 5) cores is costly. Therefore, we resort again to the binary search approach. As the largest attribute core with a readable concept lattice we identified 1,5202-core, having 54 formal concepts. We display a heat-map for the concept lattice size distribution of all sub-cores starting from this bound in Fig. 10. As for the object core we discovered that the 15,1-core has 139 concepts. However, the 16,1-core is empty, thus we are constrained to employ the 15,2-core. Starting from this we can report that the 15,3-core and the 15,4-core have twenty-five concepts and beyond that the cores are empty. Hence, those two are interesting candidates. Despite having more concepts than we considered readable we looked more thoroughly into the 15,2-core. Using background knowledge about the Wikidata properties we are able to present a well-drawn diagram of its lattice, as depicted in Fig. 11. We realized that in this core we only cover eighteen out of 101 attributes. This is, for example, in contrast to our observations for the Spice data set, where more than 50% were covered using a similar sized pq-core. Nonetheless, the 15,2-core provides a rough overview about the most important properties in the Wiki44k data set, in terms of usage for items, and how they are connected.

Coming back to the object core investigation, we start with the 1,5202-core. From there we find two candidates for interesting pq-core contexts, namely the 1,8290-core on 41735 objects, seven attributes with 34 formal concepts and the 4,7115-core on 20748 objects, eight attributes with 38 formal concepts. Although the latter covers more attributes we decided to look into the former. The reason for this is the increased readability (due to a lower number of concepts) and the higher object coverage. Cores with a higher object coverage entail implications with a higher confidence in the original concept lattice, see Proposition 7. For the visualizations of Fig. 11 we decided to indicate the objects using their Wikidata item numbers instead of their labels. This core describes a majority of the WikiData entities contained in the data set. The Wiki44k data set employs properties used for countries or people for the majority of statements. Using our proposed core analysis we are able to provide a human readable diagram representing how these properties are related. This, in turn, enables us to identify logical errors. For example, we found that there are

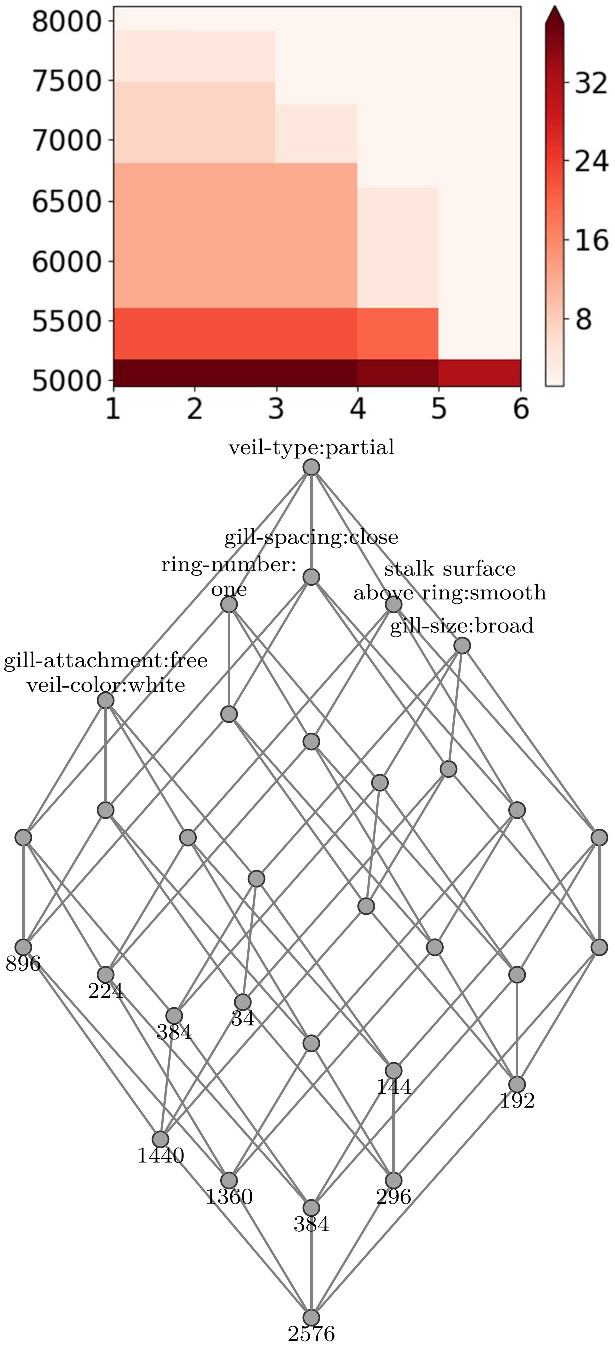


Fig. 9 Heat-map for the core concept lattice sizes (above) and the concept lattice of the 5,5176-core of the Mushroom context (below)

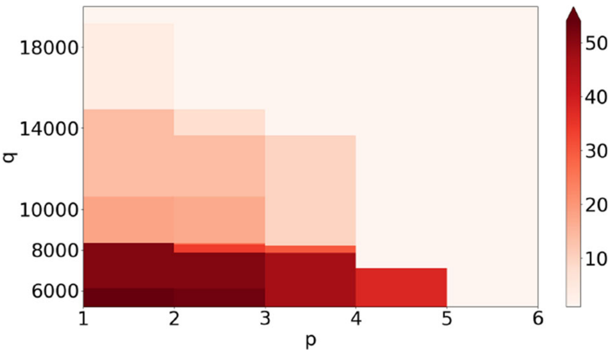


Fig. 10 The heatmapt of all core concept lattice sizes of the 1,5202-core of the Wiki44k data set.

entities which are countries with an occupation and a gender, see the concept in Fig. 11 indicated in red. The Wikidata description of these properties, however, states that the country property should not be used on human. By a closer look into the data set we found that one of these entities is ”Alfred A. Knopf”, which is both a person (Q61108) and the name of an American book publisher (Q1431868). Hence, someone added claim to Wikidata on a wrong item. Besides the study of property usage we can also employ our analysis method for the identification of missing information, i.e., missing statements in Wikidata. We see in Fig. 11 that all properties that are depicted on the right part of the diagram describe human features, e.g., occupation (P106), country of citizenship (P27), and gender (P21). Honoring the constraint that occupation is only to be used for instances of (P31) human (Q5), we find 66 items having P106 but missing the property P27. For example, one is ”James Blunt” (Q130799), an English singer-songwriter.

The approach described above can be conducted for arbitrary combinations of Wikidata properties. Hence, pq-cores enable the user to validate or contradict reasonable constraints in incomprehensibly sized data sets, at least to some confidence. Furthermore, the pq-core approach enables an automated procedure for checking implicational bases, cf. Proposition 7. In particular, one could employ methods from [18] to investigate implicational bases in Wikidata through pre-computing pq-core contexts of feasible size.

6.1 Comparison with the TITANIC approach

TITANIC [30] is an Apriori based approach that computes all formal concepts satisfying the minimum-support threshold in the data set. TITANIC computes these concepts in a bottom-up fashion, with respect to the concept lattice. The result is an ordered set of minimum supported concepts, which constitutes a join-semilattice. We show the TITANIC concept lattice of the Mushroom data set in Fig. 12 (right). In the following we compare this join-semilattice to the concept lattice of a pq-core. For this, we reuse the pre-identified interesting 5,5176-core $\mathbb{S}$ of Mushroom (see Fig. 12, left) and noted the support-values (in $\mathbb{S}$) for all object concepts, i.e., for all concepts that fulfill (g^{JJ}, g^{J}) for $g \in H$. These numbers are to be read as follows: the true support value for some concept c is the sum of all support values of concepts in the order ideal $\downarrow c$ from c. We refer the reader to Proposition 7 regarding the support estimations in $\mathbb{K}$. We observe that $\mathbb{S}$ comprises seven attributes compared to the TITANIC result, which has twelve. Both conceptual structures are built on thirty-two formal concepts.

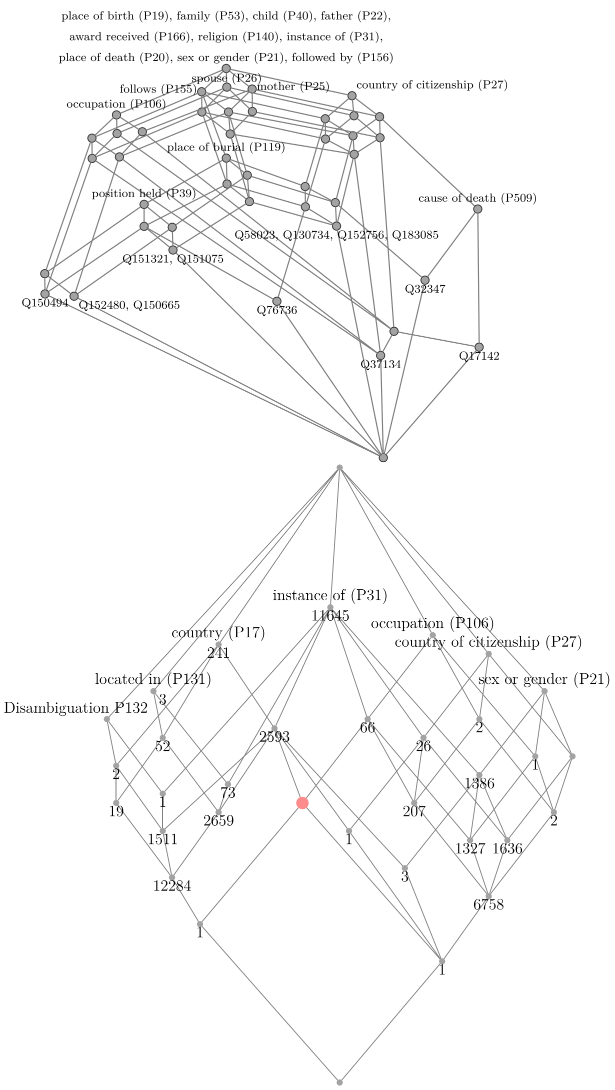


Fig. 11 The concept lattice of the 15,2-core (above) and the 1,8290-core (bottom) of the Wiki44k data set.

In the following, we want to study more thoroughly the difference in applicability between pq-cores and TITANIC concept lattices on real-world data sets. We applied both procedures to Mushroom and Wiki44k data sets. For TITANIC we used the (hyper-) parameters min-support $\in \{0.05, 0.1, 0.3\}$ for Wiki44k and $\in \{0.1, 0.3, 0.55\}$ for Mushroom. The latter values we drew from Stumme et al [30]. The pq-core (hyper-)parameters for p and q were selected using the method described in Section 5.

We show the results of our analysis in Fig. 2. First, we observe in column two that the sizes of the resulting concept lattices for the Wiki44k data set are comparably sized whereas the TITANIC concept lattices of the Mushroom data set are larger in general compared to pq-cores. To compare the applicability of the knowledge that results form TITANIC and

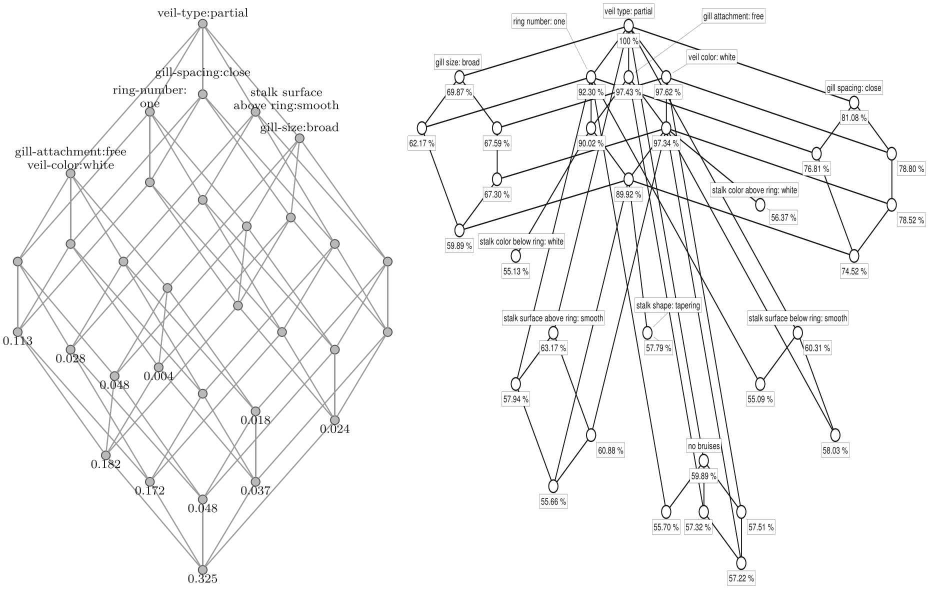


Fig. 12 The concept lattice of the 5,5176-core of the Mushroom (left) and the TITANIC output with minimum support value of 55% of the same context (right)

Table 2 Comparing the canonical base of the TITANIC iceberg lattice $\mathcal{C}_{\mathbb{T}}$ and pq-cores for the Wiki44k and Mushroom data set $\mathcal{C}_{\mathbb{C}}$

Data Set	$\lvert\mathfrak{B}\rvert$	$\lvert\mathcal{C}\rvert$	Valid[%]	Conf[%]	Sup[%]	Conf$_{\text{min}}$[%]
Mushroom	238710	2323				
6,4464-core	338	15	86.6	99.9± 0	32.5± 36.8	99.8
6,4852-core	63	3	33.3	99.9± 0	98.3±1.2	99.8
5,5176-core	32	3	33.3	99.9± 0	98.3±1.2	99.8
TITANIC m=0.1	4885	2237	24.7	24.7± 43.1	9.9±7.5	0
TITANIC m=0.3	427	414	14.3	14.3± 35.0	21.0±13.6	0
TITANIC m=0.55	30	134	6.7	6.7± 25.0	21.3±22.1	0
Wiki44k	21923	7040				
5,3432-core	100	14	50	87.3± 24.4	5.9±11.1	25
5,3579-core	71	10	30	78.4± 25.5	11.8±16.1	25
4,7115-core	38	6	0	65.6± 25.5	17.6±18.1	25
TITANIC m=0.05	120	140	29.3	29.3± 45.5	1.3±2.8	0
TITANIC m=0.1	58	117	16.2	16.2± 36.9	1.6±3.2	0
TITANIC m=0.3	12	98	0	0± 0	2.8±5.0	0

Besides the size of their bases (column 3), we show for $\mathcal{C}_{\mathbb{T}}$ and $\mathcal{C}_{\mathbb{C}}$ how many implications semantically follow from the original context $\mathcal{C}_{\mathbb{K}}$ (column 4), the average confidence/support in $\mathbb{K}$ including the standard deviation (column 5 and 6), as well as the lowest confidence of an implication in $\mathbb{K}$ (column 7)

pq-cores, we analyze the quality of the implicational structures that arise from their respective concept lattices. To do this efficiently, we employ the canonical base representation of said implicational structures. In the case of pq-cores we compute the canonical base of the sub-context, for TITANIC this canonical base arises from closure system corresponding to frequent intents.

We see in column three (Fig. 2) that TITANIC leads to larger canonical bases by at least one magnitude in our experiments. We depicted in column four the proportion of implications from the canonical base that can be inferred from the original data set, i.e., *correct* implications. Apparently, pq-cores lead to a lower number, yet more correct implications. The observations in column five, in which we depicted the average confidence per implication with respect to the original data set, go in the same direction. This can be seen in particular, when comparing the standard deviation values for the average confidence. Finally, in column six, we find that the average support of the implications computed through pq-cores is higher with respect to' TITANICs results.

Overall, while both techniques are used to reduce the size of the data set, they reflect different parts of the original data. The pq-core provide an accurate view on a dense part whereas TITANIC produces a coarsened overview over the complete data set. For the Mushroom data set the computation of the largest investigated core including its set of concepts, i.e. 6,4464-core, was nine times faster then the computation of the smallest TITANIC output, i.e. for minimum support of 55%. However, neither our TITANIC implementation nor pq-core implementation was optimized for speed. Hence, a thorough run-time comparison is deemed future work.

7 Algorithms

For a novel data reduction approach it is essential to have efficient algorithms available. In this section we present two computational problems concerned with pq-cores and their algorithmic solution. We start with the fundamental problem of computing the pq-core $\mathbb{S}$ for a given formal context $\mathbb{K}$. Our solution to this problem is an adaption of an algorithm by Matula and Beck 1983 [24] for computing k-cores of graphs. Given some graph $G = (V, E)$ with $E \subseteq \binom{V}{2}$ it uses bucket queues to repeatedly find and remove vertices of small degree. The bucket queue `Q` is generated with `Q[k]` $:= \{v \in V \mid deg_G(v) = k\}$. After that, the algorithm removes iteratively all vertices in buckets with index smaller than k and reassigns the remaining vertices to buckets of corresponding degree. Our adaption to pq-cores employs this algorithm. However, due to the bipartite nature of our data we provision two bucket queues, for objects and attributes, respectively. The computational cost for initializing these bucket queues for a context (G, M, I) is $O(|G| \cdot |M|)$. The cost for one removal iteration on both queues is bound by $O(|H|p + |N|q)$, where H, N are the objects and attributes contradicting the pq-core property, i.e., $H = \{g \in G | |\{g\}'| \leq p\}$ and $N = \{m \in M \mid |\{m\}'| \leq q\}$. In particular the algorithm has to update at most p incidences of each removed object and q incidences of an removed attribute. The following iteration is bound by the number of the remaining objects and attributes, i.e., $O(|G \setminus H|p + |M \setminus N|q)$. Thus, the total cost for removal process bound by $O(|G|p + |M|q)$. A extreme case is to set $p = |M|, q = |G|$ which would result in a single iteration step of complexity $O(|G| \cdot |M|)$.

The total cost of the algorithm, as presented in Algorithm 1 is bound by $O(|G| \cdot |M|)$. A worst-case context is one of interordinal scale as seen in Fig. 13.

	1	2	3	4	5	6
1	×	×	×			
2		×	×	×		
3			×	×	×	
4				×	×	×

	1	2	3	4	5
1	×	×	×		
2		×	×	×	
3			×	×	×
4				×	×

	1	2	3	4	5
1	×	×	×		
2		×	×	×	
3			×	×	×

…

Fig. 13 Example for a worst-case instance data set for Algorithm 1. Computing the 3,2-core (right) results in a cascading sequence of removing either one object or attribute (left, middle) in each step

7.0.1 Navigating between *pq*-core lattices

In Section 5 we characterized the interestingness of cores. This required knowledge about the corresponding concept lattice sizes of pq-cores. However, every computation of such a concept lattice is (possibly) costly and the number of these computations is large. For example, we have seen that the Wiki44k data set has 97,773 non-empty pq-cores. To overcome this issue (to some extent), we developed an algorithm based on the theory presented in Fig. 2 (right).

Problem 1 (Core Lattice) Given $\mathbb{K}$ and the set of all its concepts $\mathfrak{B}(\mathbb{K})$ compute for $\mathbb{S} \leq_{p,q} \mathbb{K}$ the set of concepts $\mathfrak{B}(\mathbb{S})$.

For solving this problem we present Algorithm 2, which is based on Propositions 3 and 6. This algorithm employs a so far not recollected notion in FCA, *duality*. We say that the dual of a formal context $\mathbb{S} = (H, N, J)$ is $\mathbb{S}^d := (N, H, J^{-1})$. Furthermore, by abuse of notation, we denote by $\mathfrak{B}(\mathbb{S})^d$ the set of concepts of the dual context. The algorithm solves Problem 1 in the following manner. First, all attributes not in $\mathbb{S}$ are removed by the method `remove_attributes` and inputs $\mathbb{T} = (U, V, L), \mathbb{O} = (U, N, L \cap (U \times N))$ (Line 7). This is realized by intersecting all intents B with N (Line 4, left) where $B \cap (V \setminus N) \neq \emptyset$ (Line 2). We construct the new extent as follows: we compute all extents associated to the same intent, i.e., intersection with N yields $\int c \cap N$ and form the union of them (Line 4, right). We justify this using our considerations after Proposition 3.

Secondly, we remove all objects that are not contained in $\mathbb{S}$ from the extents of $\mathcal{B}$ and apply the same `remove_attributes` method to the duals (see Line 8).

The overall time complexity of Algorithm 2 is $O(|\mathfrak{B}(\mathbb{T})| + |\mathfrak{B}(\mathbb{T}) \setminus \mathfrak{B}(\mathbb{S})| \cdot |U|^2|V|)$, where the first summand accounts for the dualization (Line 8) and the latter summand is

Algorithm 1 Compute p, q-core.

```
  Input  : A context $\mathbb{K} = (G, M, I)$ and $p, q \in \mathbb{N}$
  Output: $\mathbb{S}$, with $\mathbb{S} \subseteq_{pq} \mathbb{K}$
  // initialize core context
1 init output $(H, N, J)$ as $(G, M, I)$
  // initialize bucket lists
2 init $A$, with $A[i] = \{g \in H \mid |g^J| = i\}$
3 init $B$, with $B[i] = \{m \in N \mid |m^J| = i\}$
4 while $\exists g \in A[i < p]$  or  $\exists m \in B[i < q]$ do
5     $H = H \setminus \{g \in A[i < p]\}$
6     $N = N \setminus \{m \in B[i < k]\}$
7     $J = J \cap H \times N$
8     update $A$ and $B$
  return : $\mathbb{S} = (H, N, J)$
```

Algorithm 2 Transform Core Concepts.

```
   Input  : $\mathbb{T} = (U, V, L)$ and $\mathbb{S} = (H, N, J)$
            with $\mathbb{S} \leq_{p,q} \mathbb{T}$ and $\mathfrak{B}(\mathbb{T})$ (as hashmap: intent –> extent)
   Output: $\mathfrak{B}(\mathbb{S})$
1  def remove_attributes($\mathbb{O}_1 = (O, P_1, F_1), \mathbb{O}_2 = (O, P_2, F_2), \mathfrak{B}$):
       // input: $\mathfrak{B} = \mathfrak{B}(\mathbb{O}_1)$ and $P_1 \supseteq P_2$
       // init order $\leq$ on attributes $P_1$ such that
       // $\forall m \in P_1 \setminus P_2, \forall n \in P_2 : m \leq n$
2      $\hat{\mathfrak{B}}$ = next_closure on $\mathbb{O}_1$ in lectic($\leq$) starting with $P_2$
3      for $(A, B) \in \hat{\mathfrak{B}}$ do
4          $\mathfrak{B}[B \cap P_2] = \mathfrak{B}[B \cap P_2] \cup A$   // Not existing key on r.h.s. implies empty set
5          remove the concept $(A, B)$ from $\mathfrak{B}$
       return : $\mathfrak{B}$
6  $\mathbb{O} := (U, N, L \cap U \times N)$
7  $\hat{\mathfrak{B}}$ = remove_attributes($\mathbb{T}, \mathbb{O}, \mathfrak{B}(\mathbb{T})$)
8  $\mathfrak{B}(\mathbb{S})$ = remove_attributes($\mathbb{O}^d, \mathbb{S}^d, \hat{\mathfrak{B}}^d)^d$
```

the computational effort for `remove_attributes` (Line 1-5). If the size of $\mathfrak{B}(\mathbb{T}) \setminus \mathfrak{B}(\mathbb{S})$ (see Corollary 1) is small, i.e., $\mathbb{S}$ and $\mathbb{T}$ differ in a small number of concepts, the computation via Algorithm 2 is superior to `next_closure` on $\mathbb{S}$, i.e., $O(|\mathfrak{B}(\mathbb{S})| \cdot |H|^2|N|)$. This is an improvement compared to the output polynomial time complexity of the common computation of $\mathfrak{B}(\mathbb{S})$.

The computational effort for Algorithm 2 can be reduced even further, which we did not address in the pseudocode, but want to reflect here. In cases where we only require to compute the set of all concept intents of a pq-core, we can apply Proposition 6 in combination with the cover relation of the concept lattice. This relation of $(\mathfrak{B}(\mathbb{K}), \leq)$ is given by $\prec \subseteq \leq$ such that for all $c, d \in \mathfrak{B}(\mathbb{K})$ we have $c \prec d$ iff $c < d$ and there is no $e \in \mathfrak{B}(\mathbb{K})$ with $c < e < d$. After removing all attributes that are not in the pq-core, (cf. Algorithm 2, Line 7) we need to remove intents that are no longer closed after a removal of non-core objects. This is equivalent to computing the p, 0-core and can be done using the cover relation $\prec$ (see Proposition 6).

We have shown that this can be achieved by removing meet-irreducible intents with cardinality $< p$. These can be identified easily using the cover relation among intents, i.e., the elements with exactly one upper neighbor, or using the cover relation of $\underline{\mathfrak{B}}(\mathbb{K})$, i.e. elements with one lower neighbor due to duality. Note that after the removal of a meet-irreducible element, its neighbor elements can become meet-irreducible and thereby may be removed to. This saves the dualization step in Line 8 and a second enumeration using `next_closure` in Line 2 of Algorithm 2.

In the following we illustrate a generalization of Problem 1 to arbitrary sub-contexts.

Problem 2 (Navigating contexts) Let $\mathbb{S} = (H, N, J)$ be a formal context and $\mathfrak{B}(\mathbb{S})$ its concepts. Compute the set of concepts $\mathfrak{B}(\mathbb{T})$ of $\mathbb{T} = (U, V, L)$, with $L \cap H \times N = J \cap U \times V$.

With Algorithm 3 we present an approach for solving Problem 2, which is based on Propositions 3 and 4. The algorithm starts by adapting the intents of $\mathbb{S}$ to the attribute set of $\mathbb{T}$ in two steps. First, attributes not included in $\mathbb{T}$ are removed. For this we apply the `remove_attributes` method of Algorithm 2. Second, to insert missing intents the algorithm employs the `insert_attributes` method which enumerates the set of missing intents from $\mathfrak{B}(\mathbb{T}) \setminus \mathfrak{B}(\mathbb{S})$. For this, we use the contexts $(H, N \cap V, \_)$ and $(H, V, \_)$

Algorithm 3 Transform Concepts.

```
   Input  : $\mathbb{K} = (G, M, I)$
            $\mathbb{S} = (H, N, J)$, induced sub-context of $\mathbb{K}$
            $\mathbb{T} = (U, V, L)$, induced sub-context of $\mathbb{K}$
            $\mathfrak{B}(\mathbb{S})$ (as hashmap: intent –>extent)
   Output: $\mathfrak{B}(\mathbb{T})$
1  def insert_attributes($\mathbb{O}_1=(O, P_1, F_1), \mathbb{O}_2=(O, P_2, F_2), \mathfrak{B}$):
       // input: $\mathfrak{B} = \mathfrak{B}(\mathbb{O}_1)$ and $P_1 \subseteq P_2$
       // init order $\leq$ on attributes $P_2$ such that
       // $\forall m \in P_2 \backslash P_1, \forall n \in P_1 : m \leq n$
2      $\hat{\mathfrak{B}}$ = next_closure on $\mathbb{O}_2$ in lectic($\leq$) starting with $P_1$
3      for $(A, B) \in \hat{\mathfrak{B}}$ do
4          if $B \cap P_1$ not closed in $\mathbb{O}_2$ then
5              remove the concept $((B \cap P_1)^{\mathbb{O}_2}, B \cap P_1)$ from $\mathfrak{B}$
       return : $\mathfrak{B} \cup \hat{\mathfrak{B}}$
   // Adjust the set of attributes
6  $\mathbb{O}_{a1} = (H, N \cap V, \_)$, $\mathbb{O}_{a2} = (H, V, \_)$
7  $\mathfrak{B}_{a_1}$=remove_attributes($\mathbb{S}, \mathbb{O}_{a1}, \mathfrak{B}(\mathbb{S})$)
8  $\mathfrak{B}_{a_2}$=insert_attributes($\mathbb{O}_{a1}, \mathbb{O}_{a2}, \mathfrak{B}a_1$)
   // Adjust the set of objects
9  $\mathbb{O}_{b1} = (H \cap U, V, \_)$
10 $\mathfrak{B}^d_{b_1}$=remove_attributes($\mathbb{O}^d_{a2}, \mathbb{O}^d_{b1}, \mathfrak{B}^d_{a_2}$)
11 $\mathfrak{B}^d_{b_2}$=insert_attributes($\mathbb{O}^d_{b1}, \mathbb{T}^d, \mathfrak{B}^d_{b1}$)
12 $\mathfrak{B}(\mathbb{T}) = \mathfrak{B}_{b_2}$
```

as inputs for insert_attributes. Since any intent of this set contains at least one element of $V \setminus N$ the algorithm starts with computing next_closure of N (see Line 9) in a pre-chosen order $\leq$ on V such that $\forall m \in V \backslash N, \forall n \in N : m \leq n$. Finally in this step, concepts in $\mathfrak{B}(\mathbb{S}) \setminus \mathfrak{B}(\mathbb{T})$ need to be removed (cf. Proposition 4, ii). Thus, we can perform the removal (see Line 5) using a simple check (see Line 4). The result $\mathfrak{B}(H, V, \_)$ is then stored as indicated (see Line 8). The necessary adjustment of the set of objects is performed in a similar fashion due to duality.

The overall run-time complexity of Algorithm 3 can be estimated analogously to Algorithm 2 by $O(|\mathfrak{B}(\mathbb{S})| + |\mathfrak{B}(\mathbb{T})| + |\mathfrak{B}(\mathbb{T}) \setminus \mathfrak{B}(\mathbb{S})| \cdot (|U|^2 \cdot |V|))$ (again enabled by Corollary 1). The additional summand $\mathfrak{B}(\mathbb{S})$ arises from the additional dualization in Line 11. This is apparent since the first step is the same as in Algorithm 2 and the second step employs one scan of $\mathbb{T}$. This result enables a fast solution of Problem 2, in particular in the case of pq-cores.

The advantage of the navigation algorithm over a new computation of $\underline{\mathfrak{B}}(\mathbb{T})$ for similar contexts is especially clear when comparing its computational complexity to that of the next-closure algorithm, i.e., $O(|\mathfrak{B}(\mathbb{T})| \cdot |G_{\mathbb{T}}|^2 \cdot |M_{\mathbb{T}}|)$. In settings where $\mathbb{S}, \mathbb{T}$ are similar, their difference in concepts becomes smaller $\mathfrak{B}(\mathbb{T}) \setminus \mathfrak{B}(\mathbb{S}) << \underline{\mathfrak{B}}(\mathbb{T})$, resulting in a better performance of our algorithm.

7.1 Run-Time Performance

We determine the practical applicability of pq-cores by measuring the run-time on a real-world data set. In particular, we investigate the performance of our methods Algorithms 1, 2, and 3. For this, we implemented them in conexp-clj[16], a contemporary research tool for FCA. We start with the computation of pq-cores using Algorithm 1, by applying it on the Wiki44k data set.

According to the two parameters of pq-cores, we split the evaluation into two parts. First, we measure the impact on the performance when increasing p and secondly q. We start the computations with the 2,2-core. For the parameters p and q we used step widths of 250 due to the number of objects in Wiki44k. This does also impact the number of pq-cores. We depicted the run-time performance (in seconds) on the ordinate axis in the first column in Fig. 14. We observe that the time per pq-core computation does not decrease monotonously with increasing p or q. This may be attributed to the cascading nature of the pq-core computation, as seen in Fig. 13, i.e., worst-case on the number of removal iterations. In the second column of Fig. 14 we show the number of concepts with respect to the pq-core parameters. We find the overall decrease in the number of concepts to be different for p and q.

In the third column of Fig. 14 we compare the run-times of Algorithm 2 and `next_closure` for computing all concepts of pq-cores. The task is to navigate the set of pq-core lattices, i.e., starting from the 1,1-core lattice we compute for all p, 2-cores (top) as well as $2, q$-cores (bottom) the concept lattice, with $p \in \{2, \ldots, 16\}$ and $q \in \{250, \ldots, 19000\}$ in steps of 250. The output of each pq-core concepts computation is used as input for the next iteration. We observe that excluding very low and very high values for p and q the Algorithm 2 outperforms the `next_closure` algorithm by at least one magnitude. Although, we may note that Algorithm 2 depends for the initial computation on `next_closure`. Given this initialization, or any other initialization for a p, q-core we can apply Algorithm 2 to efficiently derive the concept lattices of $\hat{p}, \hat{q}$-core for $p \leq \hat{p}, q \leq \hat{q}$.

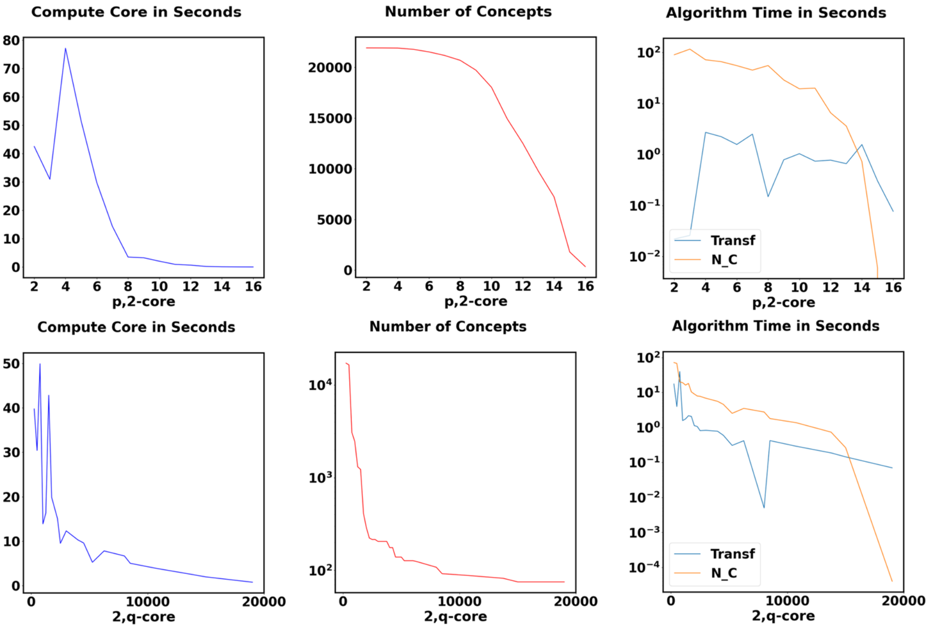


Fig. 14 Time performance of the core computation and concept intent navigation for the Wiki44k data set

7.2 Application for general context alterations

Next, we investigated the performance of the general navigation (Problem 2) among concept lattices that do not necessarily arise from pq-core contexts. For this, we employ Algorithm 3. The baseline for this algorithm is again the regular concept computation of conexp-clj. We evaluate the algorithm on randomly generated contexts [4] in three sizes, i.e., $|G| \times |M|$ equals 20×20, 50×50 and 100×100. In each of these size settings, we computed five initial random contexts with an incidence density between 10 and 90 percent in steps of 10 percent (cf. abscissa Fig. 15). We refer to these contexts as start contexts.

For each of the start contexts we then computed five random contexts by removing 5%, 10% and 20% of the start context's objects as well as attributes and adding new samples for them. We refer to the set as target contexts. We ensured that the number of incidence pairs in any start context equals the number of incidence pairs in the related target context.

The parameters are encoded into the plot's legends, e.g., s20p5N_C denotes a starting context size of 20 × 20, a randomization rate of five percent and the use of the next_closure algorithm. We observe that the performance impact is greater for larger contexts. In particular we find for the largest contexts in our experiment that the performance impact is about one magnitude in run-time. We suspect, since larger contexts tend to have larger concept lattices it becomes more efficient to update differences with our algorithm rather then recalculating the entire concept lattice. Yet, we may note that our report does include only lower densities for the larger contexts, due to the increasing computational intractability of applying next_closure. However, since real-world data sets are of particularly low density [18], we claim that our findings prevail in practical applications.

8 Related Work

As FCA is interested in representing knowledge through formal concepts and knowledge bases, it is computationally demanding. Hence, it is crucial to develop methods that can compute meaningful reductions of data sets or enable a computationally feasible navigation in them. A popular and simple technique to achieve this is random sampling from contexts. This approach, however, does not allow for a meaningful control of the result. Moreover, the computed concept lattices do mostly elude from interpretation or even explanation. Also, another disadvantage of randomly sampling objects (attributes) from $\mathbb{K}$, is that rare attribute (object) combinations of otherwise frequent attributes are unlikely to be drawn. Yet, these

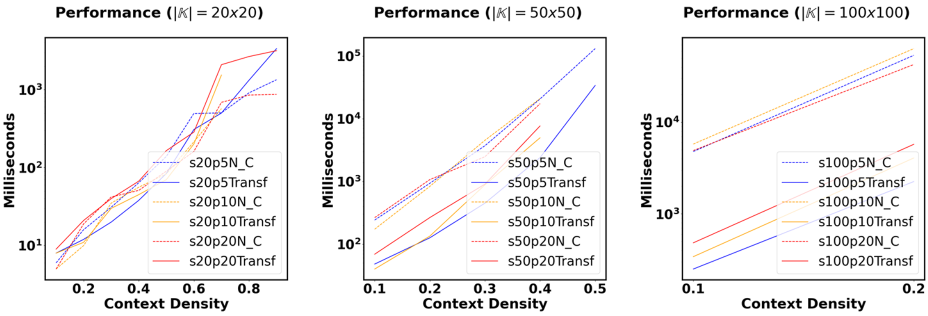


Fig. 15 Time performance of the conceptual navigation algorithm compared to regular concept computation

may represent essential counter-examples for learning a sound propositional Horn logic of the domain.

There are a multitude of methods to filter pre-computed concept lattices by importance measures, e.g., the stability index [22], and robustness of concepts [26, 31]. In empirical studies these reduction procedures performed well, however, in contrast to pq-cores, they require the computation of all formal concepts beforehand.

Other approaches compress formal contexts with popular machine learning procedures such as *latent semantic analysis* or unsupervised clustering algorithms on the object set/attribute set [3, 5]. However, we find the resulting concept lattices do lack on meaningfulness. Since all mentioned approaches introduce new attributes, e.g., as linear combinations of the original attributes, they often loose their human explainability. Contrary there are also procedures to automatically/manually select attributes and objects of relevance to the user [2, 17]. However, these approaches may require a fair amount of domain knowledge, which is not always available. Furthermore, such processes are very often time-consuming for large data sets, e.g., with hundreds of attributes, when done manually. A major shortfall of these techniques is that they do not provide proper estimations for their impact on the concept lattice of the original data set.

Another course of action to cope with large formal contexts are techniques such as TITANIC [29]. They address the computational and knowledge size issue by omitting rare attribute combinations, i.e., less supported ones. We consider this a problem as discussed in the first paragraph. Nonetheless, an advantage of TITANIC is that the resulting iceberg "lattice" is reasonably sized and does not introduce any error with respect to the original concept lattice. Nonetheless, the implicational knowledge derivable from the iceberg concept lattice is not well supported and lacks confidence, cf. Fig. 2.

A well-established method for data set reduction originates from the research field of network analysis, called *cores* [1, 8, 14, 19, 20, 25]. The original idea for this goes back to k-cores by Seidman [27]. In there, a network is reduced to a densely connected part. A variation of this notion for bipartite networks are pq-cores [1]. A further application for cores is in the realm of pattern structures as done in [28]. Our presented work on pq-cores is based on the research results mentioned in this paragraph and extends them to knowledge cores in formal contexts. Notions, like the impact of pq-cores on concept lattices and the canonical bases are so far not investigated, to the best of our knowledge.

9 Conclusion

In this work we presented an approach to define and investigate the knowledge core of a formal context. For this we employed a notion from two-mode networks, called pq-cores. We transferred the idea from graph theory to formal concept analysis and introduced the notion of pq-core formal contexts. Based on that, we show how one can identify the essential differences between pq-core lattices and their originating concept lattices. In particular we investigated conceptual differences for arbitrary sub-contexts and demonstrated their application to cores. Secondly, we derived several approaches to analyze data using pq-cores. Crucial for this was the formal characterization of *interestingness* among the set of core lattices.

To demonstrate the applicability of pq-cores to real data we analyzed seven different data sets from a qualitative and quantitative point of view. We were able to show that our method is capable to compute two meaningful core lattices for the *Spices* data set that are also human comprehensible in size. For the *Wiki44k* data set, we illustrated how to pinpoint wrongly used properties as well as missing information using pq-cores.

Furthermore, our theoretical findings allowed us to derive three algorithms for computing and transforming core structures from a formal context. As for implicational knowledge bases, we were able to show how to estimate the confidence and support of core implications within the original concept lattice. In particular, we presented core transformations that can be computed in time linear with respect to the size of the original concept lattice. An exceptionally interesting result is therewith achieved ability to navigate efficiently between arbitrary core lattices of a data set without recalculating shared concepts. The more two contexts have in common, with respect to their closure systems, the faster a transformation will perform. We have verified this theoretical result by means of a practical runtime analysis. All algorithms presented in this work are implemented and provided via the FCA software `conexp-clj`[16], a free and open-source research tool written in Clojure.

For future work we identify different meaningful lines of research. First of all a large experimental study on real-world data sets is required. In such a study domain experts from different fields should evaluate the meaningfulness of core knowledge to their respective research domain. Second, we envision a combination of pq-cores with other data reduction approaches. In our experiments we showed that core implications are, in general, more supported and have a higher confidence than implications derived from iceberg concept lattices. Yet, we are confident that both techniques might be coupled for applications on very large data sets. In such a setup one could compute an initial interesting core with our method and employ in a second step TITANIC to compute a highly supported fraction. In a third research thread we propose a more thorough investigation of the set of all pq-cores. Although we could show that this set does not constitute a lattice structure, one may draw meaningful knowledge from investigating the shown order relation with tools from directed graph analysis. Finally, we anticipate an application of pq-cores in temporal knowledge settings. Due to the shown efficient adaptability to small changes in objects or attributes pq-cores are an ideal candidate to maintain the dynamic conceptual knowledge of a domain.

Acknowledgements We thank Robert Jäschke for pinpointing us to the spice data set.

Funding Open Access funding enabled and organized by Projekt DEAL. This work was funded by the German Federal Ministry of Education and Research (BMBF) in its program “CIDA - Computational Intelligence & Data Analytics” under grant number 01IS17057.

Declarations

Conflicts of interests/Competing interests The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Ahmed, A., Batagelj, V., Fu, X., Hong, S.H., Merrick, D., Mrvar, A.: Visualisation and analysis of the internet movie database. In: S.H. Hong, K.L. Ma (eds.) APVIS, pp. 17–24. IEEE Computer Society (2007). http://dblp.uni-trier.de/db/conf/apvis/apvis2007.html#AhmedBFHMM07
2. Andrews, S., Orphanides, C.: Analysis of large data sets using formal concept lattices. In: M. Kryszkiewicz, S.A. Obiedkov (eds.) CLA, vol. 672, pp. 104–115. CEUR-WS.org. http://dblp.uni-trier.de/db/conf/cla/cla2010.html#AndrewsO10 (2010)
3. Aswanikumar, C., Srinivas, S.: Concept lattice reduction using fuzzy k-means clustering. Expert Syst. Appl. **37**(3), 2696–2704 (2010). http://dblp.uni-trier.de/db/journals/eswa/eswa37.html#AswanikumarS10
4. Borchmann, D., Hanika, T.: Some experimental results on randomly generating formal contexts. In: M. Huchard, S. Kuznetsov (eds.) CLA, CEUR Workshop Proceedings, vol. 1624, pp. 57–69. CEUR-WS.org (2016). http://dblp.uni-trier.de/db/conf/cla/cla2016.html#BorchmannH16
5. Codocedo, V., Taramasco, C., Astudillo, H.: Cheating to achieve formal concept analysis over a large formal context. In: A. Napoli, V. Vychodil (eds.) CLA, vol. 959, pp. 349–362. CEUR-WS.org (2011). http://dblp.uni-trier.de/db/conf/cla/cla2011.html#CodocedoTA11
6. Degens, P., Hermes, H., Opitz, O. (eds.): Implikationen Und Abhängigkeiten Zwischen Merkmalen. Studien Zur Klassifikation. Indeks, Frankfurt (1986)
7. Distel, F., Sertkaya, B.: On the complexity of enumerating pseudo-intents. Discrete Applied Mathematics **159**(6), 450–466 (2011). http://dblp.uni-trier.de/db/journals/dam/dam159.html#DistelS11
8. Doerfel, S., Jäschke, R.: An analysis of tag-recommender evaluation procedures. In: In: Q. Yang, I. King, Q. Li, P. Pu, G. Karypis (eds.) RecSys '13, pp. 343–346. ACM (2013). https://doi.org/10.1145/2507157.2507222
9. Dua, D., Graff, C.: UCI machine learning repository. http://archive.ics.uci.edu/ml (2017)
10. Fischer, J., Vreeken, J.: Sets of robust rules, and how to find them. In: ECML/PKDD (2019). https://ecmlpkdd2019.org/downloads/paper/650.pdf
11. Ganter, B.: Two basic algorithms in concept analysis. In: L. Kwuida, B. Sertkaya (eds.) Formal Concept Analysis, LNCS, vol. 5986, pp. 312–340. Springer Berlin Heidelberg (2010). https://doi.org/10.1007/978-3-642-11928-6_22
12. Ganter, B., Wille, R.: Implikationen Und Abhängigkeiten Zwischen Merkmalen. In: Degens, P. O., Hermes, H. J. Opitz, O.(eds.) Die Klassifikation Und Ihr Umfeld, pp. 171-185. Indeks, Frankfurt (1986)
13. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer-Verlag, Berlin (1999)
14. Ghani, A.C., Swinton, J., Garnett, G.P.: The role of sexual partnership networks in the epidemiology of gonorrhea. Sexually transmitted diseases **24**(1), 45–56 (1997)
15. 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). http://eudml.org/doc/94331
16. Hanika, T., Hirth, J.: Conexp-clj - a research tool for FCA. In: D. Cristea, F.L. Ber, R. Missaoui, L. Kwuida, B. Sertkaya (eds.) ICFCA (Supplements), vol. 2378, pp. 70–75. CEUR-WS.org (2019). http://dblp.uni-trier.de/db/conf/icfca/icfca2019suppl.html#HanikaH19
17. Hanika, T., Koyda, M., Stumme, G.: Relevant attributes in formal contexts. In: D. Endres, M. Alam, D. Sotropa (eds.) ICCS, LNCS, vol. 11530, pp. 102–116. Springer (2019). https://doi.org/10.1007/978-3-030-23182-8_8
18. Hanika, T., Marx, M., Stumme, G.: Discovering implicational knowledge in wikidata. In: D. Cristea, F.L. Ber, B. Sertkaya (eds.) Formal Concept Analysis - 15th International Conference, ICFCA 2019, Proceedings, LNCS, vol. 11511, pp. 315–323. Springer (2019). https://doi.org/10.1007/978-3-030-21462-3_21
19. Healy, J., Janssen, J.C.M., Milios, E.E., Aiello, W.: Characterization of graphs using degree cores. In: W. Aiello, A.Z. Broder, J.C.M. Janssen, E.E. Milios (eds.) WAW, LNCS, vol. 4936, pp. 137–148. Springer (2006). http://dblp.uni-trier.de/db/conf/waw/waw2006.html#HealyJMA06
20. Kitsak, M., Gallos, L.K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H.E., Makse, H.A.: Identification of influential spreaders in complex networks. Nature Physics **6**(11), 888–893 (2010). https://doi.org/10.1038/nphys1746
21. Kuznetsov, S.: On the intractability of computing the Duquenne-Guigues base. Journal of Universal Computer Science **10**(8), 927–933 (2004)
22. Kuznetsov, S.O., Obiedkov, S.A., Roth, C.: Reducing the representation complexity of lattice-based taxonomies. In: U. Priss, S. Polovina, R. Hill (eds.) Conceptual Structures: Knowledge Architectures for Smart Applications, 15th International Conference on Conceptual Structures, ICCS 2007, Sheffield, UK, July 22-27, 2007, Proceedings, Lecture Notes in Computer Science, vol. 4604, pp. 241–254. Springer (2007). https://doi.org/10.1007/978-3-540-73681-3_18

23. Mahn, M.: Gewürze : Das Standardwerk. Christian Verlag GmbH, München (2014)
24. Matula, D.W., Beck, L.L.: Smallest-last ordering and clustering and graph coloring algorithms. J. ACM **30**(3), 417–427 (1983). http://dblp.uni-trier.de/db/journals/jacm/jacm30.html#MatulaB83
25. Pastor-Satorras, R., Castellano, C., Van Mieghem, P., Vespignani, A.: Epidemic processes in complex networks. Reviews of Modern Physics **87**(3), 925–979 (2015). https://doi.org/10.1103/RevModPhys.87.925
26. Roth, C., Obiedkov, S.A., Kourie, D.G.: On succinct representation of knowledge community taxonomies with formal concept analysis. Int. J. Found. Comput. Sci. **19**(2), 383–404 (2008). http://dblp.uni-trier.de/db/journals/ijfcs/ijfcs19.html#RothOK08
27. Seidman, S.B.: Network structure and minimum degree. Soc. Networks **5**(3), 269–287 (1983)
28. Soldano, H., Santini, G., Bouthinon, D., Bary, S., Lazega, E.: Bi-pattern mining of two mode and directed networks. In: P. Champin, F.L. Gandon, M. Lalmas, P.G. Ipeirotis (eds.) WWW Companion, pp. 1287–1294. ACM (2018). https://doi.org/10.1145/3184558.3191568
29. Stumme, G.: Efficient Data Mining Based on Formal Concept Analysis. DEXA, LNCS, vol. 2453, pp. 534–546. Springer (2002)
30. Stumme, G., Taouil, R., Bastide, Y., Pasquier, N., Lakhal, L.: Computing iceberg concept lattices with titanic. Data & Knowledge Engineering **42**(2), 189–222 (2002). https://doi.org/10.1016/S0169-023X(02)00057-5. http://portal.acm.org/citation.cfm?id=606457
31. Tatti, N., Moerchen, F., Calders, T.: Finding robust itemsets under subsampling. ACM Trans. Database Syst. **39**(3), 20:1–20:27 (2014). https://doi.org/10.1145/2656261
32. Valtchev, P., Duquenne, V.: On the merge of factor canonical bases. In: R. Medina, S.A. Obiedkov (eds.) ICFCA, LNCS, vol. 4933, pp. 182–198. Springer (2008). https://doi.org/10.1007/978-3-540-78137-0_14
33. Wille, R.: Ordered Sets: Proc. of the NATO Adv. Study Institute Held at Banff, Canada, August 28 to September 12, 1981, Chap. Restructuring Lattice Theory 1 An Approach Based on Hierarchies of Concepts, pp. 445–470. Springer, Dordrecht (1982)
34. Zaki, M.J., Hsiao, C.: Efficient algorithms for mining closed itemsets and their lattice structure. IEEE Transactions on Knowledge and Data Engineering **17**(4), 462–478 (2005). https://doi.org/10.1109/TKDE.2005.60

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.