**Indonesian Journal of Electrical Engineering and Computer Science**
Vol. 30, No. 1, April 2023, pp. 366~387
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v30.i1.pp366-387 

# Survey on attribute and concept reduction methods in formal concept analysis

**Mohammed Alwersh, László Kovács**
Department of Information Technology, Faculty of Mechanical Engineering and Information Science, University of Miskolc, Miskolc, Hungary

**Article Info**

***Article history:***

Received Aug 8, 2022
Revised Nov 7, 2022
Accepted Nov 18, 2022

***Keywords:***

Concept lattice reduction
Concept lattices
Formal concept analysis
Formal context reduction
Knowledge reduction

**ABSTRACT**

Formal concept analysis (FCA) is now widely recognized as a useful approach for extracting, representing, and analyzing knowledge in various domains. The high computational cost of knowledge processing and the difficulty of visualizing the lattice are two key challenges in practical FCA implementations. Moreover, assessing the finalized built-up lattice may be problematic due to the enormous number of formal concepts and the complexity of their connections. The challenge of constructing concept lattices of adequate size and structure to convey high-importance context features remains a significant FCA aim. In the literature, various strategies for concept lattice reduction have been presented. In this work, we suggest a categorization of reduction methods for concept lattice based on three main categories: context pre-processing, non-essential distinctions elimination, and concept filtration, whereby using FCA-based analysis, the most important methods in the literature are analyzed and compared based on six pillars: the preliminary step of the reduction process, domain expert, changing the original data structure, final concept lattice, quality of reduction, and category of reduction method.



***Corresponding Author:***

Mohammed Alwersh
Department of Information Technology, Faculty of Mechanical Engineering and Information Science
University of Miskolc
3515 Miskolc, Miskolc-Egyetenvaros, Hungary
Email: alwersh.mohammed.ali.daash@student.uni-miskolc.hu

## 1. INTRODUCTION

Formal concept analysis (FCA), is a practical conceptual model used to interpret, extract, and analyze knowledge. Wille's work in 1982 was instrumental in formalizing it [1]. He presented in his work the mathematical aspects of the FCA model. A key characteristic of the FCA is the fundamental incorporation of three main knowledge extraction components, namely, the exploration of data concepts (formal concepts in FCA), the exploration and interpretation of relationships in data (implications in FCA), and visualization of the information system in the form of a conceptual hierarchy (concept lattice). Each lattice element can be considered a formal concept. The extension component includes all objects sharing the same set of characteristics, while the intention part includes all attributes corresponding to the same set of objects [2]. Together, these make up a formal concept, a basic unit of concept that plays a vital role in knowledge processing. FCA has a wide range of applications in a variety of fields. Such as data mining [3], information retrieval [4], neural networks [5], [6], ontology engineering [7]-[9], reliability engineering [10], and social networks [11]-[16]. Make use of FCA modelling techniques in order to capture knowledge. A comprehensive survey of FCA applications in knowledge discovery and information science can be found in [17].

The major factor in the success of FCA is the size of the final concept lattice that is build up when formal concepts are retrieved from their formal context in a sets of extents (set of objects) and set of intents (set of attributes). The usual characteristics of formal contexts are their size and complexity, as well as the presence of a great deal of information duplication. As a consequence of this, one of the primary problems that have been observed in practical implementations of FCA is the high computing cost of knowledge processing as well as the difficulty in representing the lattice. Moreover, assessing the finalized built-up lattice may be problematic due to the enormous number of formal concepts and the complexity of their connections. As a result, important traits will be lost in a tangle of useless information, especially the information that is meant to be tracked and disclosed. The problem of generating concept lattices of adequate size and structure to display highly important characteristics of the context remains a critical goal of FCA [18], [19]. Even small data might generate a substantial number of formal concepts [20], [21].

In this day and age of big data, FCA may also be utilized as a powerful tool for analyzing large dataset too. Therefore, while working with large datasets, it is essential to have fast and accurate FCA algorithms for knowledge discovery and knowledge representation. In recent years, several parallel and distributed approaches have been suggested to speed up the process of enumerating all possible concepts (relevant concepts). Parallel implementations utilizing the CloseByOne algorithm are proposed by the authors of the study [22]. The authors use the MapReduce programming model [23] to develop the first distributed algorithm [24], this was accomplished in 2009. Chunduri *et al.* [25] introduced a distributed FCA method that they termed the UNConceptGeneration to extract new knowledge from binary object-attribute relational data. This method was based on the fast concept analysis technique that was introduced by Lindig [26]. This paper presents both an analysis and an implementation of the FCA Upper Neighbor algorithm. This algorithm extracts the upper concepts from binary matrices by making use of the MapReduce architecture, which is a distributed method that is typically applied to large datasets. The proposed approach provides scalable, distributed algorithms to address the challenges of working with large datasets and generating formal concepts. As a result, they built the method utilizing the MapReduce framework, which provides a viable solution. In recent years, MapReduce, a novel model for distributed computing, has been increasingly popular due to the prevalence of its implementation utilizing Hadoop. Hadoop is built to operate over massive datasets, improving retrieval performance. The MapReduce architecture conceals the complexities of parallel processing, such as node availability and fault tolerance, which makes the implementation of distributed computing much more manageable. In a nutshell, the MapReduce process divides the computation of jobs into two distinct phases. In the first step of the process, map functions are used to delegate a computing task to various machines, each of which will process a distinct subset of the data. Due to the fact that it stores data using the hadoop distributed file system (HDFS), the throughput of MapReduce is particularly effective. In order to facilitate the processing of maps at the most local level possible, the data is replicated in blocks and distributed across multiple machines. The results of the Map function are referred to as intermediate outputs, and after being temporarily stored on the processing machine, they are combined by a function known as Reducer, which is the second phase of the MapReduce framework.

Parallel algorithms, in general, have the drawback of demanding machinery that is outfitted with multiple processors or processor cores. This is a disadvantage for general use. In spite of the fact that the trend in the development of technology is toward multicore microprocessors, hardware configurations that contain a significant number of processing cores remain relatively expensive and uncommon. On the other hand, distributed algorithms can be executed on connected commodity hardware if necessary. In general, parallel algorithms have significantly lower overheads for managing computations compared to distributed algorithms. However, distributed algorithms are more efficient in terms of cost because they may be executed on regular personal computers linked together over a network. The reader can get more information about these kinds of algorithms here [22]-[25], [27], where in this paper, we will compare and analyse the traditional methods for attribute and concept lattice reduction proposed in the literature.

Although big data methods speed up the calculation to generate concept sets, attribute reduction still has its importance in the FCA domain. Attribute reduction of a concept lattice facilitates the discovery and expression of hidden information in large datasets, providing a new approach for constructing a concept lattice and enhancing the theory of concept lattices, which is essential for theoretical study and practical application. The goal of attribute reduction is to identify the smallest number of attributes that can accurately capture the concept and structure of the initial formal context. Because the concept lattice of the reduced subset is isomorphic to that of the original, the reduced concept lattice may be used to compute the original lattice, reducing the complexity of the concept lattice of the original formal context. As a result, we get a smaller concept and knowledge graph, only the relevant concepts. This graph can be easier understood and interpreted by human users.

Several methods for attribute and concept lattice reduction are proposed in the literature to tackle the problem of generating concept lattices of adequate size and structure to display highly important characteristics of the context, each method with its own set of properties. Some methods use a context-level representation to start the process of reducing concept lattice complexity. On the other hand, for some methods using a lattice version to reduce complexity, we can perform such lattice level pruning like the iceberg reduction. Generally, some reduction methods try to determine the smallest number of objects\attributes that preserve the original lattice's structure in the reduction. This paper will put such methods under the pre-processing context category [28]-[31]. Other methods, which aim for a high level of simplicity that highlights the most significant features, will be categorized as non-essential distinctions elimination [32]-[34]. Alternatively, a method works by using a relevance criterion to choose formal concepts, objects, or attributes; these methods will be categorized as concept filtration [35], [36].

The three categories of concept lattice reduction methods stated above are discussed in further depth in this study, and the most important methods for each category are highlighted. The methods are analyzed using formal concept analysis, which is based on six major pillars: the preliminary step of the reduction process, domain expert, changing the original data structure, final concept lattice, quality of reduction, and category of reduction method. In addition to the FCA-based analysis, complexity of an algorithm, scalability, and reliability of the concept lattice are evaluated. We used a well-known FCA tool, ConExp [37], for drawing all the line diagrams in this study.

This paper is divided into five sections. A brief review of FCA terminology and notions is offered in the section 2, along with a small demonstration example. Section 3 divides the methods for reducing concept lattice into three categories: context pre-processing, non-essential distinction elimination, and concept filtration. The analysis and comparison of FCA reduction methods are presented in section 4 based on six major pillars: the preliminary step of the reduction process, domain expert, changing the original data structure, final concept lattice, quality of reduction, and category of reduction method, by using FCA-based analysis. Section 5, the last part of this work, discusses future work and concludes this work.

## 2. FCA: KEY TERMINOLOGY AND NOTIONS

FCA is a mathematical field that dates to the early 1980s. Its key feature is the representation of information using particular diagrams known as line diagrams (Hasse diagrams), which mathematicians refer to as diagrams of concept lattices. To interpret our research, we will provide a brief overview that explains key concepts and definitions of FCA and an example that will be used throughout the study. The terminology and notions of FCA shown in this paper are based on [2].

**Definition 1.** Formal context: a formal context is denoted by the notation $(G, M, I)$ is a cross table, where $G$ is is a group of objects, $M$ denotes a group of attributes, and $I \subseteq G \times M$ denotes an occurrences connection that exists between $G$ and $M$. For any object g $\in G$ and attribute $m \in M$ there is a binary relation $gIm$ ($(g, m) \in I$) which denotes that $g$ (object) has $m$ (attribute). It is usual practice to use a "cross table" to depict a formal context. In a formal context, the presence or absence of a link between objects and their attributes is represented by crosses and blank spaces or (1s and 0s). Such a context is known as a binary context, as shown in Table 1.

In Table 1, the group of objects $G$ are $\{o_1, o_2, o_3, o_4, o_5\}$, and the group of attributes $M$ are $\{a_1, a_2, a_3, a_4, a_5, a_6\}$, if there is a 1's in the intersection of the row for object $g$ and the column for attribute $m$, then object g possesses attribute $m$, and if there is 0's, then object $g$ and attribute $m$ are unrelated. For instance, in the Table 1, the attributes $\{a_1, a_2\}$ belong to the object $o_1$.

Table 1. Formal context

| Objects/attributes | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ |
|---|---|---|---|---|---|---|
| $o_1$ | 1 | 1 | 0 | 0 | 0 | 0 |
| $o_2$ | 1 | 0 | 0 | 0 | 0 | 0 |
| $o_3$ | 1 | 1 | 1 | 1 | 0 | 1 |
| $o_4$ | 0 | 1 | 0 | 1 | 1 | 1 |
| $o_5$ | 1 | 0 | 0 | 0 | 0 | 0 |

**Definition 2.** Derivation operators: for each subset of objects $A \subseteq G$ and a subset of attributes $B \subseteq M$ in a formal context $(G, M, I)$, the derivation operators $\uparrow = 2^G \rightarrow 2^M$, $\downarrow = 2^M \rightarrow 2^G$ are explicitly defined as $A^{\uparrow} = \{m \in M | \forall g \in A: (g, m) \in I\}$, $B^{\downarrow} = \{g \in G | \forall m \in B: (g, m) \in I\}$. This definition holds true for all formal contexts. The up operator $A^{\uparrow}$ is just the collection of all attributes that all objects from $A$ have in common, and the down operator $B^{\downarrow}$ is the collection of all objects that share all of $B$ 's attributes. The derivation $(\uparrow, \downarrow)$ operators $A^{\uparrow}$ and $B^{\downarrow}$ are also known as $A'$ and $B'$. For instance, given the formal context depicted in Table 1. One can simply notice that:

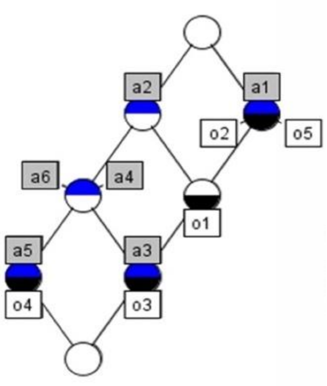
$\{o_1\}^{\uparrow} = \{a_1, a_2\}$
$\{o_1, o_1\}^{\uparrow} = \{a_1\}$
$\{a_1\}^{\downarrow} = \{o_1, o_2, o_3, o_5\}$
$\{a_1, a_2\}^{\downarrow} = \{o_1, o_3\}$

**Definition 3.** Formal concepts: is a pair $(A, B)$, derived from a formal context $(G, M, I)$, such that, $A \subseteq G$ which indicate the extent part for $(A, B)$, and $B \subseteq M$ is indicate the intention part of the formal concept $(A, B)$, such that $A' = B, B' = A$. For instance, the pair ({$o_3$, $o_4$}, {$a_2$, $a_4$, $a_6$}) is a formal concept according to the formal context in Table 1, with {$o_3$, $o_4$} being the extent part and {$a_2$, $a_4$, $a_6$} being the intent part.

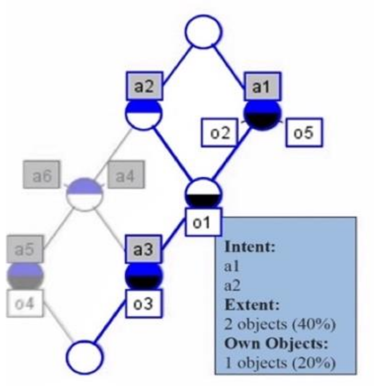
**Definition 4.** Concept lattices**:** given two formal concepts like $c_1 = (A_1, B_1), c_2 = (A_2, B_2)$ of a context can be sorted using the subconcept-super concept sorting relation $\leq$, which is described as follows: $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2$ (or equivalently $B_1 \subseteq B_2$ ), where $c_1$ is a subconcept (more specific) and $c_2$is a super concept (more general). In a formal context $K$=$(G, M, I)$, the set of all formal concepts along with the partial order $\leq$ usually defines a complete lattice known as a concept lattice [1], denoted by $\mathcal{B}(G, M, I)$.

According to the first section of the "concept lattice fundamental theorem" given in [1], a concept lattice $\mathcal{B}(G, M, I)$ is a “complete lattice” in which the infimum and supremum exist for any random set $C \subseteq \mathcal{B}(G, M, I)$, and given by $(A_1, B_1) \wedge (A_2, B_2) = (A_1 \cap A_2, (B_1 \cup B_2)'')$ and $(A_1, B_1) \vee (A_2, B_2) = ( (A_1 \cup A_2)'', B_1 \cap B_2)$.

The line diagram for the concept lattice is shown in Figure 1 and it is created using the formal context of Table 1, where Figure 1(a), indicates the set of formal concepts created from the formal context and the relationship between them, known as the subconcept-superconcept, could make up the concept lattice [2]. The line diagram's nodes represent the formal concepts. The terms of object concept $(\{g\}'', \{g\}')$ and attribute concept $(\{m\}', \{m\}'')$ may also be used to explain the labeling of the formal concepts in the line diagram, where the object concept is denoted by $\gamma(g)$ and the attribute concept is denoted by $\mu(m)$. The tagging (labeling) of $\mathcal{B}(G, M, I)$ is therefore accomplished in the following manner: the formal concept $\gamma(g)$ is assigned the tag $g$ for every object $g$, and the formal concept $\mu(m)$ is assigned the tag $m$ for every attribute $m$. Some concept nodes have objects drawn underneath them, while others have attributes drawn above them (usually, concept object labeling is drawn below a node in a line diagram while the concept attribute labeling is drawn above the node). Not every concept in a context is an object or an attribute concept. Any concept can be an object concept, an attribute concept, a combination of the two, or none [38]. From the concept lattice, the extent part (concept objects) for a formal concept can be reached by a downward path from the node to capture all concept objects. In the highlighted node in Figure 1(b), we can easily notice that $\{o_1, o_3\}$ is the extent part of the node, whereas the intent part (concept attributes) of the highlighted node is $\{a_1, a_2\}$ is obtainable by following the upwards path for the highlighted node to capture the concept attributes.




(a)




(b)

Figure 1. Concept lattice structure (a) concept lattice of Table 1 and (b) clarify the path to reach the concept objects and concept attributes

**Definition 5.** Attribute Implications: an attribute implication is constructed across the set $M$ of attributes. The following is the interpretation of the implication of the form $(A \rightarrow B)$ with $(A, B \subseteq M)$: if an object has all the attributes of the set $A$, it also has all of the attributes of the set $B$, and in the provided context $K = (G, M, I)$ is the same holds if ($A^{\downarrow} \subseteq B^{\downarrow}$). Then it is respected by all concept intents. The following are the definitions of the support and confidence measures for any implication ($A \to B$): $\sup(A \to B) = |(A \cup B)'|/|G|$, $\text{conf}(A \to B) = |(A \cup B)'|/|A'|$. The set $I$ of implications shown in Table 2 are constructed fron the concept lattice of Figure 1(a). For instance, the implication $a_1, a_4 \to a_3$ denotes those objects with the attributes $a_1$ and $a_4$ (really, just the object$o_3$) also have the attribute $a_3$.

The duquenne-guigues (DG) and Luxenberger bases of implications are often used to define the association rules in any concept lattice. We make use of the DG implications here. The percentage of confidence levels is what distinguishes these two bases. The DG basis has a confidence level of 100%, but the Luxenberger basis has a confidence level of less than 100% [39]. The terms "finer", "consistent set", "reduct", and "core" are defined in the following definitions. These concepts are required for reducing the subset of attributes in FCA effectively. Attribute reduction aims to reduce attributes while maintaining the number of concepts and their relationships, resulting in a new concept lattice that is isomorphic to the original [40], [41].

Table 2. Attribute implications derived from a formal context of Table 1

| S. No | Implications |
|---|---|
| 1 | $a_3 \to a_1$ |
| 2 | $a_4 \to a_2$ |
| 3 | $a_5 \to a_2$ |
| 4 | $a_3 \to a_2$ |
| 5 | $a_6 \to a_2$ |
| 6 | $a_5 \to a_4$ |
| 7 | $a_3 \to a_4$ |
| 8 | $a_6 \to a_4$ |
| 9 | $a_4 \to a_6$ |
| 10 | $a_5 \to a_6$ |
| 11 | $a_3 \to a_6$ |
| 12 | $a_1, a_4 \to a_3$ |
| 13 | $a_1, a_6 \to a_3$ |

**Definition 6.** For a given two concept lattices $L_1(B, A_1, I_1)$ and $L_2(B, A_2, I_2)$. If for any $(X, Y) \in L_2(B, A_2, I_2)$ there exists $(X', Y') \in L_1(B, A_1, I_1)$ such that $X' = X$, then we said that $L_1(B, A_1, I_1)$ is finer than $L_2(B, A_2, I_2)$, denoted by:

$$L_1(B, A_1, I_1) \leq L_2(B, A_2, I_2).$$

If $L_1(B, A_1, I_1) \leq L_2(B, A_2, I_2)$ and $L_2(B, A_2, I_2) \leq L_1(B, A_1, I_1)$, If this is the case, we can say that the two concept lattices are isomorphic to one another, as indicated by the following:

$$L_1(B, A_1, I_1) \cong L_2(B, A_2, I_2).$$

Let $(G, M, I)$ be a formal context. For any $B \subsetneq M$ and $I_B = I \cap (G \times B)$, $(G, B, I_B)$ is also a formal context, which can be interpreted as a subcontext of the original one. However, we can apply the mapping $\uparrow$ and $\downarrow$ , which will be rewritten in this subcontext as $\uparrow^B$ and $\downarrow^B$ . For any $A \subsetneq G$, we obtain that $A^{\uparrow B} = A^{\uparrow} \cap B$, which implies that $A^{\uparrow\downarrow\uparrow B} = A^{\uparrow\downarrow\uparrow} \cap B = A^{\uparrow} \cap D = A^{\uparrow B}$. Note that when we consider a subset $X \subseteq B$, then $X^{\downarrow B} = X^{\downarrow}$. As a result, $A^{\uparrow B\downarrow\uparrow B} = A^{\uparrow B\downarrow B\uparrow B} = A^{\uparrow B}$.

Let $(G, M, I)$ be a formal context, for any $B \subsetneq M$ such that $B \neq \emptyset$, $L_1(G, M, I) \leq L_2(G, B, I_B)$ holds.

**Definition 7.** For a given formal context $(G, M, I)$, for a set of attributes $B \subseteq M$ such that $L_2(G, B, I_B) \cong L_1(G, M, I)$, then $B$ is called a consistent set of $(G, M, I)$. Furthermore, if $L_2(G, B - \{b\}, I_{B-\{b\}}) \ncong L_1(G, M, I)$ for all $b \in B$, then $B$ is called a reduct of $(G, M, I)$. The intersection of all reducts of $(G, M, I)$ is called the core of $(G, M, I)$.

The subsequent result is straightforward to validate. Let $(G, M, I)$ be a formal context, for any $B \subsetneq M$ such that $B \neq \emptyset$, then $B$ is a consistent $\Leftrightarrow L_2(G, B, I_B) \leq L_1(G, M, I)$. Antoni *et al.* [31], established three distinct kinds of attributes in a formal context, which are analogous to the rough set theory described in [42].

**Definition 8.** For a formal context $(G, M, I)$, the set $\{B_i |\, B_i \textit{ is a reduct}, i \in T\}$, where $T$ is an index set that indicates all reducts of $(G, M, I)$. Then $M$ is divides into three parta as follows:

a. Absolute necessary attribute (core attribute) $b$: $b \in \bigcap_{i \in T} B_i$.
b. Relative necessary attribute $c$: $c \in \bigcup_{i \in T} B_i - \bigcap_{i \in T} B_i$.

c. Absolute unnecessary attribute $d$: $d \in M - \bigcup_{i \in T} B_i$.

We use the term "unnecessary attribute" to describe a feature that is not necessary. $e$: $e \in M - \bigcap_{i \in T} B_i$, and can be classified as either "relatively necessary" or "absolutely unnecessary". Assuming that b, c, and d are absolute necessary, relative necessary, and absolute unnecessary, respectively, it is obvious that $b' \neq c', c' \neq d', b' \neq d'$. Let's say that $(G, M, I)$ represents a formal context. Evidently, the following outcomes have taken place [40].

Corollary 1. The core is a reduct ⟺ there is only one reduction.

Corollary 2. $m \in M$is an unnecessary attribute ⟺ $M - \{m\}$ is a consistent set.

Corollary 3. $m \in M$is a core attribute ⟺ $M - \{m\}$ is not a consistent set.

Considering the formal context depicted in Table 1, it contains two reduces: $B_1 = \{a_1, a_2, a_4, a_5\}$, $B_2 = \{a_1, a_2, a_5, a_6\}$. As a result, attributes $a_1, a_2$ and $a_5$ are absolutely necessary, $a_4$ and $a_6$ are relative necessary attributes, and $a_3$ is the unnecessary attribute.

## 3. CATEGORIZATION OF FCA REDUCTION METHODS

For FCA real-world applications, one essential aspect is the visualization of formal concepts in hierarchical order within the concept lattice structure. The magnitude of the concept lattice, which is created from a very vast formal context, is one of the most significant challenges associated with this methodology. A large formal context results in a concept lattice that is difficult to work with in practice. As a result, problems with FCA applications, such as dealing with a large formal context and reducing the size of the concept lattice, are highlighted as critical issues in FCA applications [38].

Attribute reduction of a concept lattice facilitates the discovery and expression of hidden information in large datasets, providing a new approach for constructing a concept lattice and enhancing the theory of concept lattices, which is important for theoretical study and practical application. The goal of attribute reduction is to identify the smallest number of attributes that can accurately capture the concept and structure of the initial formal context. Because the concept lattice of the reduced subset is isomorphic to that of the original, the reduced concept lattice may be used to compute the original lattice, reducing the complexity of the concept lattice of the original formal context. The literature describes various techniques of attribute reduction to control the complexity and size of formal contexts, formal concepts, concept lattices, and implications. Popular reduction techniques for improving FCA scalability include iceberg concept lattices [35], matrix decompositions [43], [44], conceptual scaling for many-valued contexts [45], the reduction of the concept lattices based on rough set theory [46]-[48], and others.

Priss and Old [49] proposed a kind of categorization of the concept lattice reduction methods called “data weeding techniques”, where they categorized the existing reduction methods into four categories. Visual reduction methods are the first category. These methods aim at how data is shown without altering the mathematical structure of the fundamental concept lattices. Faceting and plain scaling methods are the second category; these methods divide the original concept lattices into smaller ones without losing information. The divide should make sense in terms of the lattice's content. Pruning and restricting methods are the third category, which includes removing objects, attributes, or concepts from concept lattices using statistical methods. The fourth category, decomposition and general scaling methods, decomposes a lattice while also reducing complexity, for instance, by aggregating objects or attributes. Their study looked at several "data weeding" methods that may be used to shrink a concept lattice and generate "good" graphical representations for concept lattices. Techniques for data weeding tend to be very dependent on the type of application. Unlike their work, Dias and Vieira [50] classify concept lattice reduction strategies into three groups. In the first group of reduction strategies, the context is stripped of superfluous information. The second group of reduction approaches simplifies formal context and concept lattices. The third group of reduction strategies is the selection of formal concepts, objects, and attributes. Based on their comparison methodology, we will categorize the most significant works and recent contributions in the direction of attribute reduction of a concept lattice. We will identify three categories of reduction methods for concept lattices. In the first category, “context pre-processing reduction,” the methods under this category try to determine the smallest number of objects\ attributes that preserve the original concept lattice's structure. In the second category, “non-essential distinctions elimination,” these reduction methods aim for a high level of simplicity that highlights the most significant features. In the third category, “concept filtration,” such a method works by using a relevance criterion to choose formal concepts, objects, or attributes.

The three categories of reduction methods for concept lattices stated above are discussed in further depth in this comprehensive overview, mostly with important primary methods for each category highlighted. The methods are analyzed using formal concept analysis, which is based on six major pillars: the preliminary step of the reduction process, domain expert, changing the original data structure, final concept lattice, quality of reduction, and category of reduction method. Each pillar has its own set of features. In addition to the FCA-based analysis, complexity of an algorithm, scalability, and reliability of the concept lattice are evaluated. It is essential to say that the focus of this study will be on classical FCA. The three categories of concept lattice reduction methods are defined and discussed in the following subsections, along with each category's most important reduction methods.

### 3.1. Context pre-processing

These methods focus on finding a formal context with the fewest possible objects or attributes while preserving the concept lattice's structure intact by working on eliminating or transforming redundant information in the formal context. It is possible to assert that $g \in G$ (set of objects), $m \in M$(set of attributes) are redundant if eliminated or altered one of them or all in such a manner leads to leads to generating a concept lattice preserves the initial concept lattice's structure intact.

A first feasible reduction is to replace a group of objects with precisely identical attributes with a single object or a group of attributes that appear in identical objects with a single attribute. A clarified formal context has been acquired after such redundancies have been removed. As an illustration, Table 1 of the formal context shows that the objects demonstrate that the objects $o_1, o_5 \in G$ such that $o_1' = o_5'$, then $o_1, o_5$can be reduced to a single representative object. Similarly, the attributes $a_4, a_6 \in M$ such that $a_4' = a_6'$, then $a_4, a_6$ can be reduced to a single representative attribute [2].

Removing attributes that a set of other attributes may represent is another type of reduction that does not alter the concept lattice's structure, termed a reducible attribute [2]. More formally, if an attribute $m \in M$, and a set of attributes $B \subseteq M$, where $m \notin B$, such that $m' = B'$, then $\mu m$ (attribute concept) can be considered as the infimum of $\mu b$ (attribute concepts) where $b \in B$. As a consequence of this, if the attribute $m$ is removed, the lattice that is derived from the concluding formal context is equivalent to the initial lattice in terms of both its form and its relations. For instance, in the formal context depicted in Table 1, we can notice that $a_3' = \{a_1, a_4\}'$ such that $\mu a_3$ is the infimum of $a_1$ and $a_4$. As a result, removing the attribute $a_3$ from that formal context yields a final lattice (new one) that is similar in form and relations to the initial lattice; the equivalent concept lattice structure with reduced tagging is that of Figure 1(a) without concept attribute tagged $a_3$. Likewise, eliminating the reducible objects from a formal context can also result in a reduced formal context that can be used to derive a concept lattice that is similar in form and relations "isomorphic" to the initial lattice. More formally, an object $g \in G$ such that $\gamma(g)$ (object concept) is the infimum of $\gamma(a)$ (object concepts) such that $a \in A$, where $A \subseteq G$ and $g \notin A$, by removing such objects result in a concept lattice that is equivalent to the initial lattice in terms of both its form and its relations (isomorphic) [2].

In order to minimize the size of the formal contexts while preserving the integrity of the concept lattice structure, numerous approaches have been developed. Zhang *et al.* [51], the authors suggested the use of a "discernibility matrix" in order to construct a minimal set of characteristics. This was accomplished by considering $(G, M, I)$ as a formal context and two formal concepts $(A_1, B_1), (A_2, B_2) \in \mathcal{B}(G, M, I)$. The symmetric difference between the intention parts $B_1, B_2$ determines the discernibility between the concepts $(A_1, B_1), (A_2, B_2)$ as follows: $Dis_{(A_1,B_1),(A_2,B_2)} = (B_1 \cup B_2)\backslash(B_1 \cap B_2)$. After establishing discernibility matrix (Dis) from a given formal context $(G, M, I)$, a minimal set of attributes $B \subseteq M$ may be identified that resulting in a lattice $\mathcal{B}(G, M, I')$ isomorphic to the original $\mathcal{B}(G, M, I)$, where $I' = I \cap (G \times A)$ and $A$ is the minimal set of attributes with the smallest cardinality. Qi *et al.* [52] the authors set guidelines for reducing the number of discernibility computations while maintaining the potential for getting a minimal set of attributes. Zhang *et al.* [51] determine if a formal context's attributes are "absolutely necessary", "relatively necessary", or "absolutely unnecessary". To be "absolutely necessary," an attribute must be present in every minimal set. It is "relatively necessary" if it is included in at least one minimum set but not all of them. Lastly, an attribute is " absolutely unnecessary " if it is not present in any minimal set.

Medina [29] looked at attribute reduction in three frameworks: "object-oriented concept lattices", "property-oriented concept lattices", and "concept lattices". Regardless of the framework, it has been discovered that the attributes may be divided into three degrees of requirement, with the attribute reducts being similar at each level. Wang and Zhang [53], the authors describe the process of reducing formal contexts by eliminating attributes. Wang *et al.* [54] offers a heuristic method for identifying the smallest possible collection of attributes. To deal with approximation sets, the authors in their work [54] expanded the work of [51].

Belohlavek [55], the author presented a method for factorizing concept lattices according to concept similarity. It has also been shown how to efficiently compute similarity relations. They developed and investigated the relationships between similarity at three levels: the similarity of concept lattices, the similarity of the set of objects\attributes, and finally, the similarity of formal concepts. The granular structure of concept lattices is investigated in [56] and how it might be utilised to decrease knowledge in FCA. Both attribute reduction and size reduction in concept lattices were discussed in the of [57], which was an important contribution to FCA research. The authors offer a method for concurrently shrinking both attribute sizes and concept lattice sizes by employing an irreducible-cut concept lattice.

### 3.2. Non-essential distinctions elimination

These methods aim for a high level of simplicity that highlights the most significant features by extracting non-essential distinctions (based on some criterion) between formal concepts, sets of objects, or attributes from a formal context or a concept lattice. When applied to the information retrieval process [58], object clustering's primary objective is to lessen the number of dimensions that concept lattices contain. In this work, object equivalence groups are constructed by the application of the singular value decomposition (SVD) method. “SVD” is one of numerous linear algebra matrix decomposition techniques for reducing a large matrix to a smaller one. The authors construct an equivalency relation using reduced matrices created by the SVD approach, as described in the following: let $h_1, h_2$ be two objects denoting documents that are equivalent if and only if the cosine of the angle between them reaches a specified threshold. To decrease the formal context, non-negative matrix factorization was applied [59]. Recently, Sumangali and Kumar [60] have developed a unique strategy that decomposes the original context in terms of dimensionally reduced low-rank matrices comprising real columns and rows using the CUR matrix decomposition technique. As a result, using CUR decomposition in FCA reduction techniques might help us extract the most significant data from the datasets.

The authors used fuzzy k means (FKM) clustering to make the concept lattices more manageable. Using equivalence relations developed from FKM clustering, the context matrix is shrunk, and quotient lattices are generated. Each variable represents a range of membership levels; a given record may have many cluster memberships [21]. In their work, Cheung and Vogel [58] introduce a new technique to reduce the formal context significantly. They accomplish this by identifying a new object $g$ derived by the intersection of all objects within a parity group whose attributes match the union of the attributes of the elements $|g|$. For instance, if $I$ is the incidence relation in the initial formal context, then $g'$={$oIm\ for\ each\ o \in |g|$} in the reduced formal context. Considering the formal context in Table 1, Assume that all equivalence classes have a size of one, with the exception of a class containing objects $o_1$ and $o_3$. In this instance, objects $o_1$ and $o_3$ are replaced by a new object $g$, which shares the attributes of both $o_1$ and $o_3$: $g'$=$o_1' \cup o_3'$={$a_1, a_2, a_3, a_4, a_6$}, while the remaining objects are unaffected.

To facilitate the reduction process, domain experts (prior knowledge) about the issue area can be integrated into the reduction method. This is done in several studies like [18], [20], [61]. Junction based on object similarity (JBOS) uses expert knowledge of a domain (prior knowledge) to replace similar objects with representative elements that are similar to a certain degree [32]. To assess the similarity of objects, they used weight assignment ($0 \leq w^m \leq 1$) to every attribute $m \in M$, where $w^m$ is used to measure the relevancy of an attribute in the range of 0 (no relevance) to 1 (high relevance) and should be determined by an expert of the domain. The degree of similarity between objects like $g_1, g_2$ is stated by a range from 0 (absolutely dissimilar) to 1 (absolutely similar), defined as follows:

$$sim(g_1, g_2) = \sum_{m\in M} \mu(g_1, g_2, m)/\sum_{m\in M} w^m,$$

$$\mu(g_1, g_2, m)=\begin{cases} w^m\ se(g_1, m) \in I \leftrightarrow (g_2, m) \in I \\ 0\ otherwise. \end{cases} \quad (1)$$

where $g_1, g_2$ are similar if their similarity is greater than the threshold.

JBOS employs the intersection of the attributes of the objects that are regarded as similar, in addition to using prior knowledge. This feature prohibits objects from being created outside of the formal context. The research presented in [18] showed that it is feasible to drastically diminish a formal context by studying the implications given by the JBOS technique while also achieving sufficient performance on a particular work. By eliminating incidents from a formal context, the authors managed the complexity of a concept lattice [62].

### 3.3. Concept filtration

The methods under this category work by using a relevance criterion to choose formal concepts, objects, or attributes. In many circumstances, more knowledge about the sets of objects and the sets of attributes is available. This knowledge is used by certain filtration methods to direct the reduction process. Some of these methods make use of attribute weighting [20], [62]. Wu *et al.* [56] suggested using all of the user's prior knowledge to set limits on attributes. Only formal concepts that fulfill the limitations are maintained when constructing the concept lattice and formal concepts. [20] present an approach that uses a weight assignment to every attribute to convey its importance and then chooses formal concepts that are judged important by using the same method of [32] the authors in their work attempting to capture the significance of concepts through information conveyed by weights. Zhang *el al.* [61], the authors gave weights to attributes to associate significance with formal concepts. The importance of formal concepts is quantified similarly to the authors’ work here [20]. However, the previous authors solve concept lattice completeness difficulties by establishing virtual formal concepts.

Object, attribute, and concept selection procedures that employ a relevance criterion are types of concept filtration methods. Pasquier *et al*. [63] contributed significantly by connecting frequent items and formal concepts. The terms "support" and "frequent sets" are described as: Let $B \subseteq M$, where M is a set of attributes and $Sup(B, G)$, is the count of objcts in $G$ that contain all the attributes of $B$. We can say that a set of attributes $B \subseteq M$ is frequent iff $Sup(B, G) \geq minSup$(minimal support previously set). Iceberg concept lattices are concept lattices constructed by selecting only frequent item sets. In this particular instance, the generated lattice is partial; just the formal concepts that occur most frequently were employed. i.e., given two formal concepts $(A_1, B_1), (A_2, B_2)$, where $(A_1, B_1) \leq (A_2, B_2)$, sup $(B_1, G) \leq$ sup $(B_2, G)$, where $G$ is the sets of objects. Stumme *et al*. [35], the authors present the "Titanic algorithm" for the creation of "iceberg concept lattices" and illustrate the functionality of these lattices in a variety of applications, like mining association rules, the visualization of implications as well as the analysis of "large-scale" databases

Soldano *et al*. [64], the authors presented a new model of concept lattice that can be considered of as being similar to iceberg concept lattice. The concept lattice that was produced as a result was given the name "alpha concept lattice" by the authors. An unlimited lattice that only contains frequent formal concepts can form the basis of what is known as an iceberg concept lattice. The frequency of specific formal concept elements is used by several strategies to make their selections. For instance, a formal concept can be chosen if it meets a limitation $\alpha$(user parameter). These limitations, like $|A| > \alpha, |B| > \alpha \ or \ (|A| \times |B|) > \ \alpha$ and so on, in the area of data mining, such limitations are considered in [65]. In concept lattice building and implication extraction, Belohlavek and Vychodil [66] developed a set of restrictions that could be directly exposed to derivation operators and evaluated their use. The authors put out a similarity metric for fuzzy formal concepts. A subset of formal concepts related to one another is chosen using the similarity measure; this subset may be much less than the initial set of formal concepts. Below is the definition of a metric used to compare formal concept extensions.

**Definition 9.** Given two formal concepts $(A_1, B_1), (A_2, B_2)$, the similarity between the extensions $A_1$ and $A_2$ given by $Sim(A_1, A_2)$=1- $|A_1' \cap A_2'|/\ |A_1' \cup A_2'|$.

This method is also discussed in [67], [68]. When minimizing the number of formal concepts, the quality of the formal context is also important to consider. Many researchers have shown how difficult it is to construct reliable concept lattices when noisy underlying data is. Several concept filtration methods are considered for this [69]-[72].

The term "stability" is associated with formal concepts, so judgment based on stability is a regularly employed strategy. Stability tries to construct an indicator for concepts that illustrates how the intention of the concept varies depending on the collection of objects. All formal concepts with a stability index less than a certain threshold are eliminated [69]. Due to the necessity to construct all subsets of each formal concept, the computation of this indicator is #P-complete. Kuznetsov *et al.* [70] proposed two heuristics to get around this difficulty. Markov chains were offered to calculate the stability index [73]. Klimushkin *el al.* [72] addressed how probabilistic factors are utilized in the probability-based method to choose specific formal concepts. They referred to this as "the selection of specific concepts". The factor is defined according to the likelihood that a specific object possesses a particular property.

## 4. ANALYSIS AND COMPARISON OF REDUCTION METHODS

In this section, we will analyze and compare the reduction methods that previously described each one with its category. The methods will be analyzed using formal concept analysis, which is based on six major pillars: the preliminary step of the reduction process, domain expert, changing the original data structure, the final concept lattice, quality of reduction, and category of reduction method. In addition to the FCA-based analysis, the complexity of an algorithm, scalability, and reliability of the concept lattice are evaluated. We used a well-known FCA tool, ConExp [37], for drawing all the line diagrams in this study.

a) The preliminary step of the reduction process

In this pillar, the focus is on the specific stage of the concept formation process at which the reduction methods being analyzed can be applied. Some methods [74] may be appropriate for use on the formal context. Other methods [75] may be more suitable for use on the set of formal concepts itself, which is a complete and irreducible set of concepts that can be derived from the formal context. Still other methods [76] may be more appropriate for use on the concept lattice, which is a graphical representation of the relationships between the concepts in the set of formal concepts.

b) Domain expert

In this pillar, the focus is on whether or not the reduction methods being analyzed incorporate prior knowledge from domain experts in the reduction process. Some methods [32] incorporate this knowledge, while others [77] do not. This distinction is important because the use of prior knowledge from domain experts can potentially improve the quality of the reduction process and the resulting concept lattice. On the other hand, not using prior knowledge may result in a more objective and unbiased reduction process but may also potentially compromise the quality of the results.

c) Changing the original data structure

In this pillar, the focus is on reduction methods that involve modifying the set of objects, attributes, or occurrences (incidence relation) in the formal context or concept lattice. Some methods [32] may modify the set of objects, which are the entities being described in the formal context. Other methods [78] may modify the set of attributes, which are the properties or characteristics being used to describe the objects. Still other methods [61] may modify the occurrences or incidence relation, which is the relationship between the objects and attributes in the formal context. These modifications can have a significant impact on the resulting concept lattice and may be used to achieve specific goals in the reduction process. It is important to carefully consider the potential consequences of making such modifications before using these methods.

d) Final concept lattice

In this pillar, the focus is on the characteristics of the final concept lattice resulting from the reduction process, as compared to the original concept lattice. Some methods [29] result in a final concept lattice that is isomorphic to the original, meaning that it is structurally identical and preserves the relationships between the concepts. Other methods [61] result in a final concept lattice that is partial to the original, meaning that it is a subset of the original concept lattice and contains only a portion of the concepts and relationships. Still other methods [32] result in a final concept lattice that is different from the original, meaning that it is structurally distinct and does not preserve the relationships between the concepts in the same way as the original.

e) Quality of reduction

During the process of reduction, it is possible that some of the information that is being represented by a concept lattice will be lost. It is important that such a loss be quantified in an objective manner using indicators that reflect the ability of the concept lattice and that enable comparisons to be made between the various techniques. A concept lattice's complexity is typically evaluated using a variety of different metrics, including the cardinality of the covering relation, the number of formal concepts, objects, or occurrences, and the attribute cardinality. Other reduction methods utilize particular metrics for selecting formal concepts in the original lattice that represent the quality of the final lattice from the standpoint of the specific methodology. Indexes depending on stability [69], frequency [35], or distance [79] are only a few examples. Such metrics may not be applicable throughout all situations. Furthermore, such metrics exclude losses linked to structural characteristics of the lattice, which can be significant.

Dias and Vieira [32] suggested indexes related to the usage of implications as a different perspective on the knowledge provided by the lattice. The objective is to compare the efficiency of such lattices using the sets of implications that correspond to the original and reduced formal contexts. The adjustments made to shrink the concept lattice are intended to be mirrored in the set of implications, decreasing the capacity of the implications to characterize a collection of reference objects. Two measures are provided to quantify the degree of comparability between both the original formal context and the reduced formal context: fidelity and descriptive loss. Let $G_o$ represent the set of objects from the original formal context, $G_r$ represent the set of objects from the reduced formal context, and $r_1$, $r_2$,..., $r_k$ indicate the set of rules derived from the reduced concept lattice.

The fidelity ($F$) metric is used to find out the percentage of successful rule applications to original objects. If there is a rule $r_i = P \to Q$ for which $P \subset g'$ and $Q \not\subset g'$ for an object $g \in G_o$, the rule $r_i$ is said to have failed. The following equation calculates fidelity:

$$F = (\sum_{i=1}^{k}(1 - \frac{Nf_i}{|G_o|})/k) \times 100, \tag{2}$$

where, $Nf_i$ denotes the number of failures of the rule $r_i$(i.e., the number of objects in $G_o$ for which the rule fails:). As a result, $F$ is the percentage of non-failures when considering the $|G_o| \times k$ applications of $k$ rules applied to $|G_o|$objects. To avoid numerous contributions to $F$ from objects with precisely the same attributes, such objects must be reduced to just one object first.

The descriptive loss (DL) is a metric that quantifies the reduction in the capacity to characterize a set of $G_o$ objects as a result of the removal of attributes during the reduction process. In order to define it, an "onto function" denoted by $v$: $G_o \to G_r$r will be utilized. This function will map each object $g$ in $G_o$ onto an object $v(g)$ in $G_r$. This mapping occurs as a direct result of the reduction process, which may have resulted in $g$ being simplified. It is presumed that the object $(g)$ will have its attributes comprised of a selection from the set of $g$ attributes, i.e., According to the definition provided in [32], the descriptive loss measure can be written as:

$$DL = \left(1 - \frac{\sum_{g \in G_O}\left(\frac{|\nu(g)'|}{|g'|}\right)}{|G_O|}\right) \times 100 \quad (3)$$

where $|\nu(g)'|$ is the reduced object's attribute count $\nu(g)$ and $|g'|$ is the original object's attribute count $g$. When attributes that are regarded "redundant" in the real application are removed, DL might show descriptive losses. As a result, such attributes must be eliminated before the reduction technique may be used.

It is important to note that the level of fidelity is decided by a collection of implications that are obtained from the concept lattice. This collection of implications indicates the degree to which the information that is available in the generated formal context is consistent with the original. The descriptive loss is evaluated directly on the acquired formal context, and it evaluates the loss of capacity to represent the features that were provided in the framework that was given in the initial formal context [32]. Fidelity is a valuable metric for assessing non-essential distinctions elimination methods and concept filtration methods. Descriptive loss is beneficial for all types of reduction methods, but it requires formal context accessibility. This pillar considers several quality indexes of the final lattice, such as fidelity and descriptive loss, as stated above.

f) Category of the reduction method

Eventually, the method's reduction category is outlined in the last pillar. Three categories comprise all methods: context pre-processing methods, non-essential distinctions elimination methods, and concept filtration methods, which have been covered in section 3.

Based on the six pillars described above, we will analyze and compare the most important reduction methods in the literature. The formal context (binary table) shown in Table 3 highlights the reduction methods, where objects denote the method's reference and attributes indicate a method's features. Figure 2, depicts the conceptual lattice (Hasse diagram) derived from the formal context shown in Table 3. At first glance, we can notice that all reduction methods modify the occurrence (actually, attribute p3c).

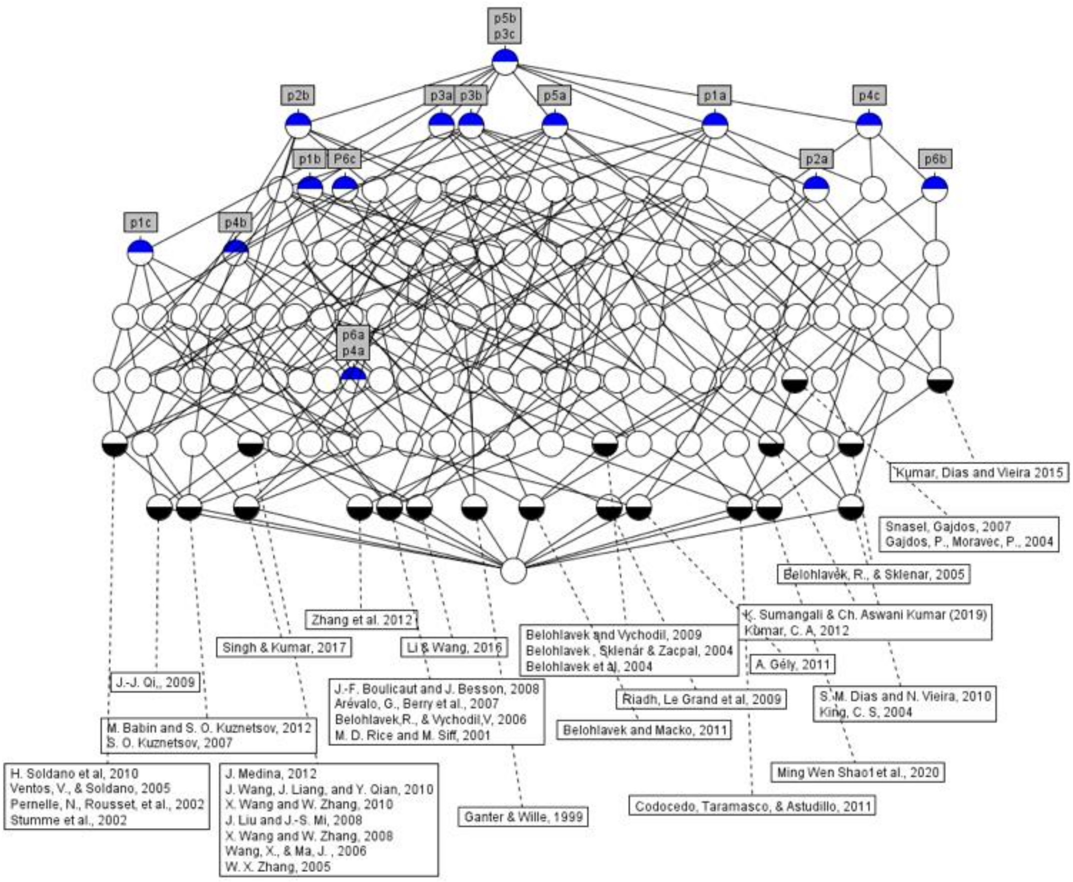



Figure 2. Concept lattice of reduction methods extracted from the formal context in Table 3

Table 3. Formal context (binary table) represents the reduction methods

| Objects/Attributes | p1 a | p1 b | p1 c | p2 a | p2 b | p3 a | p3 b | p3 c | p4 a | p4 b | p4 c | p5 a | p5 b | p6 a | p6 b | P6 c |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ganter and Wille [2] | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Belohlavek and Macko [20] | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Medina [29] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Dias and Vieira 2010 [32] | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Codocedo *et al.* [33] | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Singh and Kumar [36] | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Stumme [39] | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Zhang *et al*. [40] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Snasel *et al.* [43] | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Liu *et al*. [46] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Qi [52] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Wang and Zhang [53] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Wang *et al.* [54] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Kumar *et al.* [59] | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| Sumangali and Kumar [60] | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Zhang *et al.* [61] | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Soldano *et al.* [64] | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Boulicaut and Besson [65] | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Kuznetsov [69] | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Grand *et al*. [71] | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| Babin and Kuznetsov [73] | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Gajdos *et al.* [75] | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Gély [76] | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Liu and Mi [77] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Wang and Zhang [78] | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Rice and Siff [79] | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Belohlavek and Vychodil [80] | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Li and Wang [81] | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| Belohlavek *et al.* [82] | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Cheung [83] | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Belohlavek and Sklenar [84] | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Kumar [85] | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Shao *et al.* [86] | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| Belohlavek and Vychodil [87] | 1 | 0 | 0 | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Belohlavek *et al.* [88] | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Arévalo *et al.* [89], [90] | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Pernelle *et al.* [91] | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| Ventos and Soldano [92] | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |

The preliminary step of the reduction process (Pilar1): p1a: formal context, p1b: formal concepts, p1c: concept lattice.
Domain expert (Pilar2): p2a: used), p2b: unused.
Changing the original data structure (Pilar3): p3a: set og objects, p3b: set of attributes, p3c: occurrences(incidences).
Final concept lattice (Pilar4): p4a: isomorphic to the original lattice, p4b: partial from the original lattice, p4c: different from the original lattice.
Quality of reduction (Pilar5): p5a: fidelity losses, p5b: descriptive losses.
Category of reduction method (Pilar6): p6a: Context pre-processing, p6b: Non-essential distinctions elimination, p6c: Concept filtration.

Deleting characteristics (attributes)\objects modifies the set of occurrences (set of incidences) in the category of context pre-processing methods. Likewise, by grouping "similar" objects, "non-essential distinctions elimination reduction methods" diminish the occurrence relation (set of incidences). While choosing a set of formal concepts, a set of objects, or a set of attributes, concept filtration reduction methods modify the set of occurrences (set of incidences) by excluding any of those items. The line diagram also clearly shows that all reduction methods exhibit some descriptive losses (attribute p5b).

A. Pillar 1 (The preliminary step of the reduction process)

Considering the preprocessing steps for attribute or concept set reduction, one of the key elements is the selection of the object set representation. Based on the literature review, the relationships among the dominating approaches can be presented in the form of a concept lattice as shown in Figure 3. The performed analysis on the reduction methods in the literature highlights the following key approaches for preparing the appropriate object set.

– One important aspect is the format of the data sets, namely it can be a) a context, b) a concept set, or c) a concept lattice. It can be seen that the different object set alternatives provide different ways and different options for the later reduction steps. For example, using a context-level representation, the attribute-level operations are more flexible and efficient than a lattice-based approach. On the other hand, using a lattice version, we can perform such lattice-level pruning like the iceberg reduction. The most dominating approach is the application of concept set representation; we can mention here, among others, the works [29], [51]-[54], [70], [77]-[80]. Regarding the exceptions, we can mention the method in [2], which has formal context as a point to start the reduction process and the method in [81], which has both formal concepts and concept lattice as a point to begin the reduction process. Under the non-essential distinctions elimination category, all of those [32], [33], [43], [59], [60], [82]-[85] reduction methods have formal context as a preliminary step for the reduction process, except for the method in [86], which has both formal context and formal concepts as a point to start the reduction process and the method in [76], which has both formal concepts and concept lattice as a preliminary step for the reduction process. In the concept filtration category, eight methods [65], [71], [79], [82], [87]–[90] use formal context as a preliminary step for the reduction process, four methods [35], [64], [91], [92] use concept lattice as a preliminary step for the reduction process, three reduction methods [61], [69], [73] use both formal concepts and concept lattice as a preliminary step for the reduction process, and two reduction methods [20], [36] use both formal context and formal concepts as a preliminary step for the reduction process.

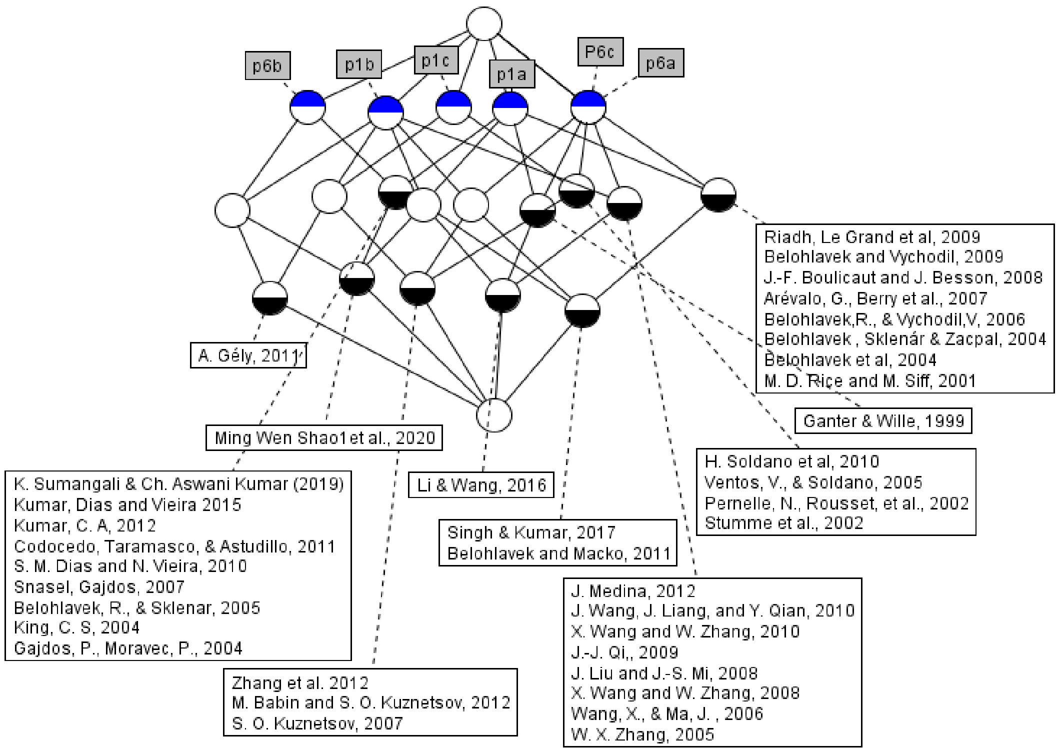



Figure 3. Concept lattice of Pillar 1.

– Besides the generation of the object set representations, another key element is the application of standard data conversation methods. These methods transform the attribute domains of real values into categories using a discretization technique.
– Application of statistical attribute reduction methods. In this case, the dominating approach calculates the correlation between the attributes to determine the strong relationships. Regarding the popular methods, we can mention the PCA and SVD algorithms. The preprocessing step's main benefit is that it yields a smaller dataset requiring fewer costs during the computations.

B. Pillar 2 (Domain expert)

The approach can be enhanced by using domain experts (prior knowledge) about the domain problem to lead the reduction process. In this context many works integrated the prior knowledge for better reduction such as [17], [70]. The JBOS approach [15] is used in the clustering domain based on prior knowledge. It aims to substitute a set of similar objects with relevant objects based on a qualitative evaluation of their features. Other methods, like [52], and [72], suggest restricting features using prior knowledge. These restrictions are known as "attribute-dependency rules." Only formal concepts that adhere to the restrictions are kept when constructing formal concepts and the concept lattice. In this study, the analysis and comparison of the reduction methods for each category under the concept lattice of this pillar as shown in Figure 4 revealed the following features:

- For the most popular reduction methods in the literature that we selected in this study under the category of context pre-processing, no domain expert (prior knowledge) has been used to assist the reduction.
- Only five strategies [32], [59], [83], [84], [86] in the non-essential distinction's elimination category involve a domain expert to lead the reduction process.
- Only five strategies [20], [61], [82], [87], [88] in the concept filtration category involve a domain expert to lead the reduction process, and twelve do not.

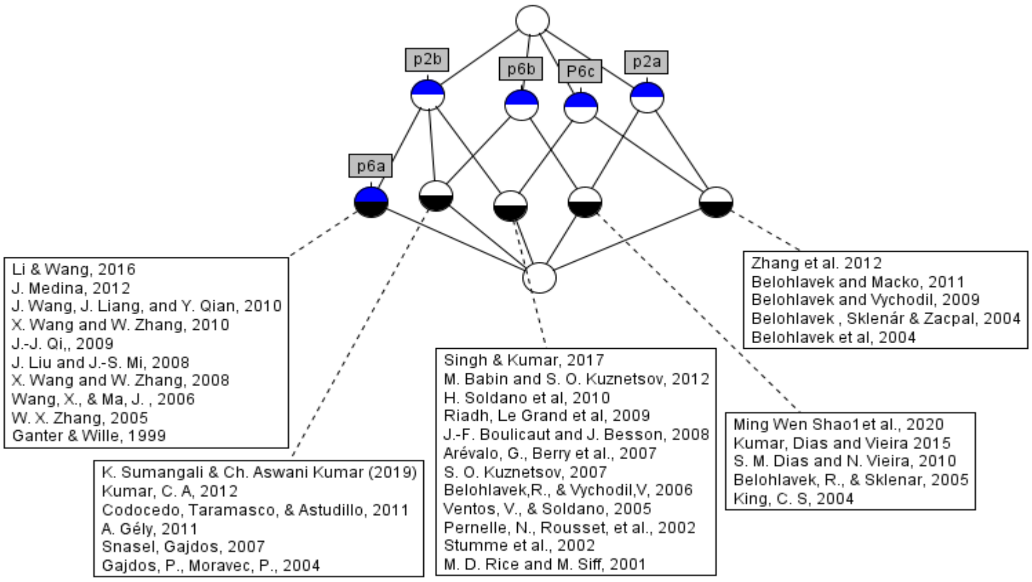



Figure 4. Concept lattice of Pillar 2

C. Pillar 3 (Changing the original data structure)

A first feasible reduction is to replace a group of objects with precisely identical attributes with a single object, or a group of attributes that appear in identical objects with a single attribute. A clarified formal context has been acquired after such redundancies have been removed. More precisely, the formal context depicted in Table 1 demonstrates that the objects $o_1, o_5 \in G$ such that $o_1' = o_5'$, then $o_1, o_5$can be reduced to a single representative object. Similarly, the attributes $a_4, a_6 \in M$ such that $a_4' = a_6'$, then $a_4, a_6$ can be reduced to a single representative attribute [2]. A variety of methods have been founded with the aim of reducing formal contexts. JJ Qi [52] the authors set guidelines for reducing the number of discernibility computations while maintaining the potential for getting a minimal set of attributes. Zhang *et al.* [51] determine if a formal context's attributes are "absolutely necessary", "relatively necessary", or "absolutely unnecessary". If an attribute appears in all minimal sets, it is considered "absolutely necessary". If it appears in at least one, but not all, minimal sets, it is considered "relatively necessary". Lastly, if an attribute is not included in any minimal set, it is "absolutely unnecessary". Based on the literature review, the dominant approaches' relationships can be presented as a concept lattice as shown in Figure 5. The performed analysis of the reduction methods in the literature highlights the following key features:

- In the pre-processing context category, all reduction methods alter the set of attributes and the set of occurrences (incidence relation), except one method [2] that also changes the set of objects.
- In the non-essential distinctions elimination category, all methods modify occurrences (incidences); as well as another two methods [32], [83] modify the set of objects, five methods [59], [60], [84]–[86] alter the set of attributes too, and one [33] alters both the set of objects and the set of attributes along with incidences.
- In the concept filtration category, all reduction methods alter the set of attributes, set of objects as well as the set of occurrences (incidence relation), except two methods, one of them [71] doesn't alter the set of attributes and the other one [36] alters the set of incidences only.

D. Pillar 4 (Final concept lattice)

Regarding the structure of the generated reduced concept sets, we can distinguish the following main approaches: a) only concept set without lattice; b) concept hierarchy, c) not-isomorphic concept lattice, d) isomorphic sub concept lattice or e) not-isomorphic concept lattice. In Figure 6, these variants are depicted pillar 4. The analysis and comparison of the reduction methods for each category under the concept lattice of this pillar revealed the following features:

- In this case, we get the list of the selected concepts without the ordering relationship among them. The main benefit of this approach is the simplicity; the performed calculations require fewer time costs. On the other hand, the result can not be used to show the specialization relationship.
- The hierarchy structure means a simplification of the original lattice structure. This structure is very popular in many fields of knowledge engineering and software engineering; it can be used for additional processing steps. We can mention the single parent class approach in UML or the taxonomy of some ontology.
- Under the non-essential distinction’s elimination category, there are no methods for constructing an isomorphic final lattice or a subset of the original concept lattice. All of them have a different final lattice than the original. In this case, only some selected concepts remain in the goal lattice, and this method results in a different ordering relationship. Another approach is when the output reduced lattice consists of blocks of the input lattice, and the edges correspond to the ordering relationship among these blocks.
- In the concept filtration category, all methods generate a final lattice that is a part of the original lattice, with no resultant lattice being isomorphic to the original but only one [71] being different.
- In this case, the input and output concepts are isomorphic, usually having a different attribute set. The main goal is to keep only those attributes representing the existing relationship in the input space.

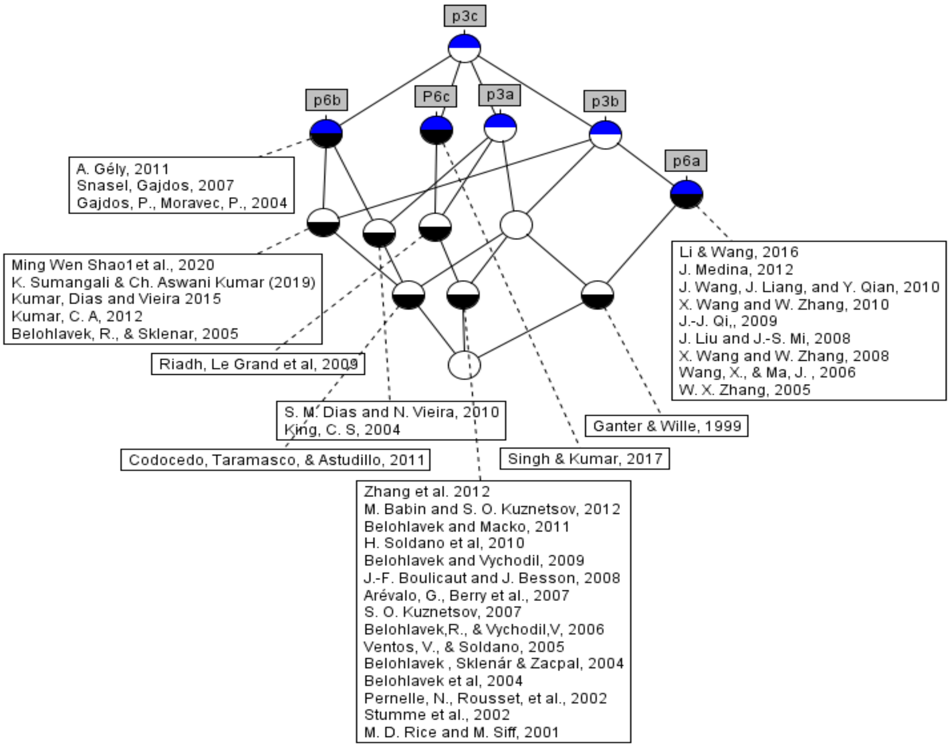



Figure 5. Concept lattice of Pillar 3

E. Pillar 5 (Quality of reduction)

During the process of reduction, it is possible that some of the information that is being represented by a concept lattice will be lost. It is important that such a loss be quantified in an objective manner using indicators that reflect the ability of the concept lattice and that enable comparisons to be made between the various techniques. A concept lattice's complexity is typically evaluated using a variety of different metrics, including the cardinality of the covering relation, the number of formal concepts, objects, or occurrences, and the attribute cardinality. Other reduction methods utilize particular metrics for selecting formal concepts in the original lattice that represent the quality of the final lattice from the standpoint of the specific methodology. Indexes depending on stability [69], frequency [35], or distance [79] are only a few examples. Such metrics may not be applicable throughout all situations. Furthermore, such metrics exclude losses linked to structural characteristics of the lattice, which can be significant. Figure 7, depicted the concept lattice for pillar 5 derived from Table 3. The analysis and comparison of the reduction methods for each category under the concept lattice of this pillar revealed the following features:

- For the analysis and comparison of the reduction methods in this work we will consider two measures to quantify the degree of comparability between both the original formal context and the reduced formal context: a) fidelity loss and b) descriptive loss; fidelity loss is determined from a collection of implications derived from the concept lattice, which measures the level of consistency of the information available in the generated formal context concerning the original. The descriptive loss is calculated directly on the acquired formal context and assesses the loss of capacity to express the features provided in the model given in the original formal context [23]. Fidelity is a valuable metric for assessing non-essential elimination methods and concept filtration methods. Descriptive loss is beneficial for all reduction methods but requires formal context accessibility. In the pre-processing context category, we can mention that no method has fidelity loss, but they all have a descriptive loss.

– Under the non-essential distinction's elimination category, all reduction methods have fidelity loss and descriptive loss, except one method [59] which has only descriptive loss.
– All reduction methods in the concept filtration category have both fidelity loss and descriptive loss.

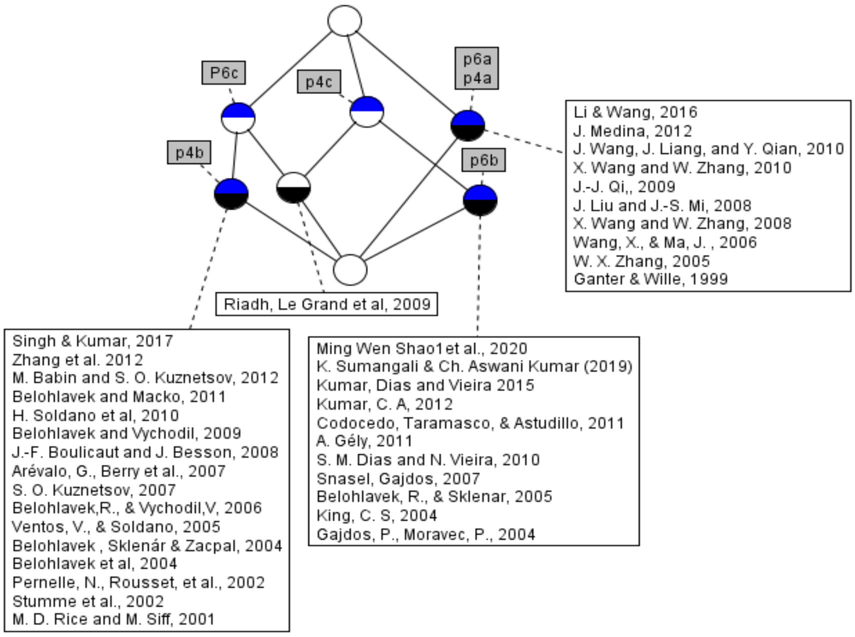



Figure 6. Concept lattice of Pillar 4

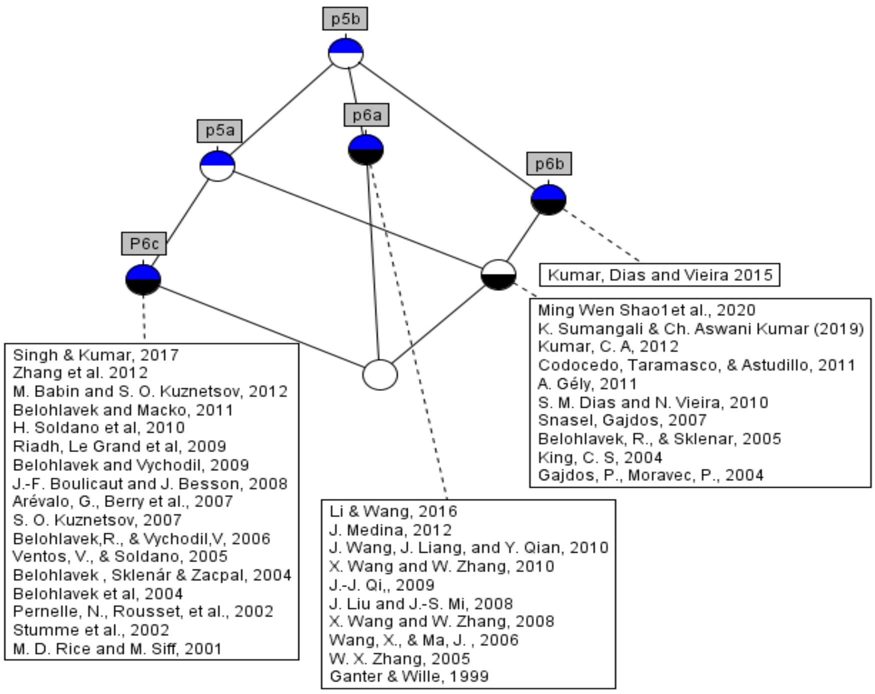
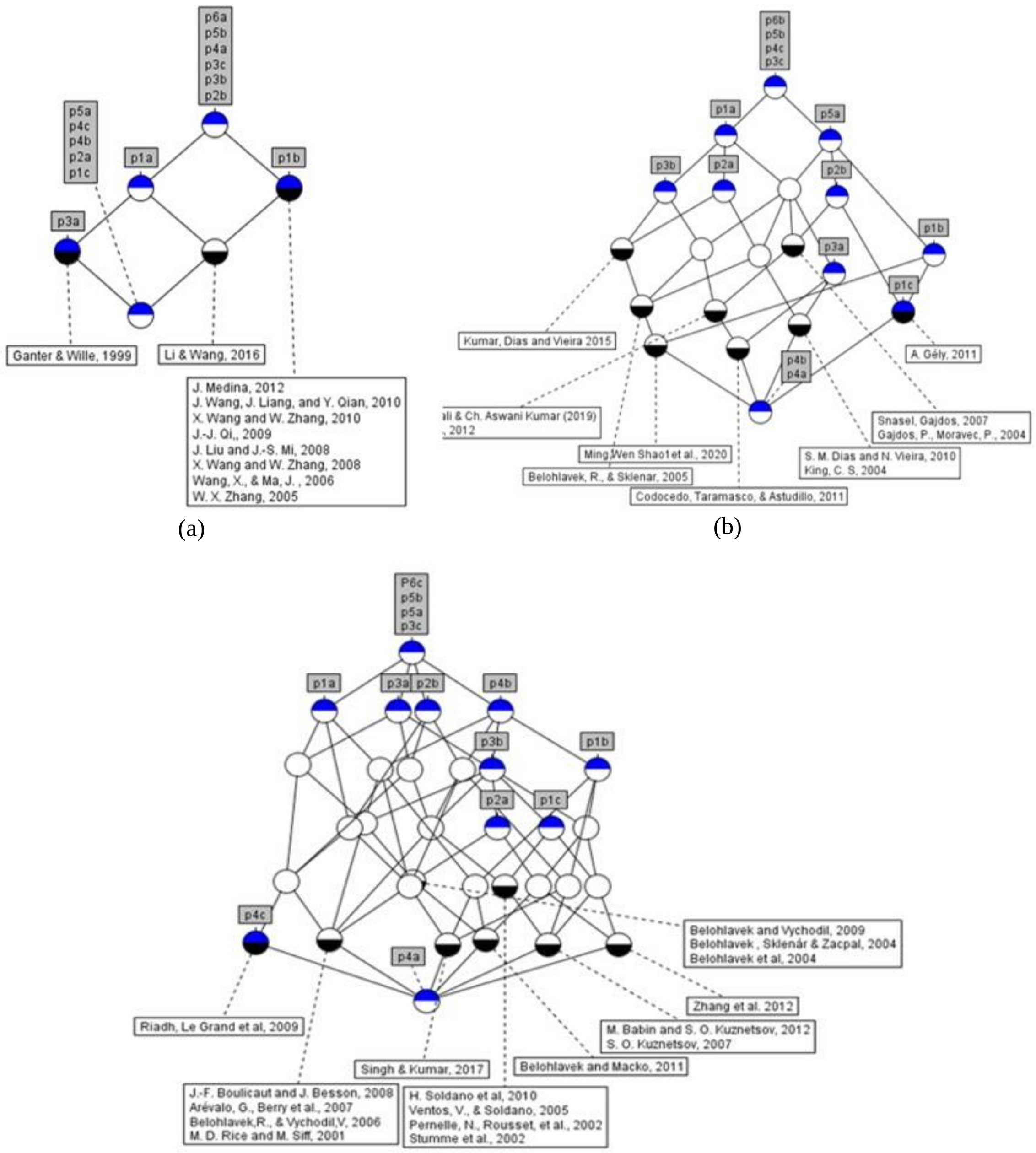



Figure 7. Concept lattice of Pillar 5

F. Pillar 6 (Category of reduction method)

Figure 8, shows the concept lattice for each category of reduction methods that were derived from Table 3: Figure 8(a), indicates the concept lattice for context pre-processing methods, Figure 8(b) indicates the concept lattice for non-essential distinctions elimination methods, and Figure 8(c) indicates the concept lattice for concept filtration methods. The ten methods in the pre-processing context category are distinguished because most methods begin the reduction process with a set of formal concepts and do not employ previous information. All reduction methods provide an isomorphic final lattice to the original. Most of the seventeen methods in the category of concept filtration modify the set of objects, attributes, and occurrences and yield a final lattice that is a part of the original lattice. In a nutshell, the majority of techniques for non-essential distinctions elimination methods begin the process of reduction with the formal context, which ultimately leads to a concept lattice that is not isomorphic and is not a subset of the initial concept lattice. Some are using prior information to aid in the reduction and elimination process.




(c)

Figure 8. Concept lattice for each category of the reduction methods (a) context pre-processing methods, (b) non-essential distinctions elimination methods, and (c) concept filtration methods

Context pre-processing methods yield a final lattice with fewer attributes, ideal for applications that need direct interaction with the user, either via a basic analysis and visualization of the lattice or exploitation of the new formal context's implications. Non-essential distinctions elimination methods are often used in the formal context and have a level of complexity that allows for controlling extremely vast formal contexts. The final concept lattice may diverge significantly from the original and have an inadequate quality. Concept filtration methods operate by reducing the space of concepts based on some relevance criterion, such as using an objective function on the concept lattice to eliminate pathways that appear to be irrelevant. The building of iceberg concept lattices, introduced by [35], is a highly respected approach in this category of methods. The downside of this method seems to be that significant formal concepts may be ignored over the process. Some techniques list all formal concepts first, then use a criterion to choose the ones that are relevant [20], [61], [69]. A strategy that focuses on this method can be costly because the whole search space is investigated.

## 5. CONCLUSION AND FUTURE WORK

FCA is an appropriate tool in many application areas for constructing concept (class) hierarchies or lattices to provide an efficient knowledge representation. One promising area is the field of software engineering, where concept lattices can be used to build standard UML diagrams. In this case, the goal of the reduction module is to select the relevant key classes for model generation needed for UML modeling. The proposed engine will extract the context from existing documents, and the generated FCA lattice will be converted into a UML model. Another important application area is automated educational systems, where the knowledge base covers content ontology and the student's competency maps. One way to generate the knowledge units in the content ontology is the application of FCA tools with concept reduction to determine only the important elements in the ontology model. FCA can also be used as a general method for clustering, too. In this case, every concept in the lattice corresponds to a cluster of the covered atomic items. The reduction here is a powerful step to determine only the important clusters in the hierarchy. A special case of clustering is the field where items are represented with sequences like genes in bioinformatic or event sequences in business process mining. Another important FCA application area is association rule mining; the goal of association rule mining, a subfield of data mining, is to find intriguing common patterns, correlations, or associations in a dataset. Various issues, including the extraction of redundant rules, the enormous volume of extracted rules, and the potential loss of significant rules, make it difficult to extract traditional association rules from large amounts of data. Integrating formal concept analysis into the mining process becomes one of the most promising applications to solve these issues. Only the frequently closed item sets may be directly derived when FCA is used in the association rule mining (ARM) process. Utilizing FCA to extract association rules (ARs) is regarded as one of the lattice-based techniques (LBT) that works better than the related traditional mining techniques compared to the extraction time.

Additionally, without reprocessing the raw data, ARs may be derived with various pairings of minimal support and confidence using the resulting formal concept lattice. As a result, database searches might be considerably reduced. Therefore, it has been noted that a formal concept lattice is a crucial tool for obtaining the association rules basis, which has been hypothetically demonstrated to be a minimal, efficient, and nonredundant association rule set. Additionally, association rule mining with formal concept analysis may be utilized to categorize data efficiently. Concept lattice reduction techniques are useful for reducing the complexity of the concept lattice and improving its readability and usability in various application areas. The choice of the appropriate reduction technique depends on the specific goals and requirements of the application and the trade-off between the quality of the resulting concept lattice and the computational complexity of the technique. In this work, we suggest categorizing concept lattice reduction methods into three main categories: First, context pre-processing methods that try to identify the smallest number of objects\attributes that will preserve the original lattice's structure. Non-essential distinctions elimination methods are the second category. They aim to keep things as simple as possible while highlighting the most important features. Finally, concept filtration methods aim to choose formal concepts, objects, or attributes based on a relevance criterion. Each of these categories has its own strengths and limitations, and the appropriate technique will depend on the specific needs and goals of the application. Overall, it was determined that context pre-processing methods, which remove redundant information and create a concept lattice with fewer attributes, may be effective in situations where user interaction is important. Non-essential distinctions elimination methods, which simplify the concept lattice by reducing the space of concepts, may be more suitable for cases where the size of the concept lattice is more important than its quality. Concept filtration methods, which select relevant concepts based on a criterion, can be computationally expensive if they require enumerating all formal concepts before applying the selection process.

Thirty-eight methods were chosen from among the most important methods in the literature. The methods have been compared, analyzed, and categorized using six major pillars: a preliminary step of the reduction process, domain experts, changing the original data structure, final concept lattice, quality of reduction, and category of reduction method. The analysis was carried out using FCA, where the selected methods are summarized in a formal context. It was demonstrated that all reduction methods modify the occurrence (incidences) and result in some level of descriptive loss. Considering each category of methods, most reduction methods under the context of pre-processing start the reduction process from the set of formal concepts and modify the set of attributes. They don't use prior information (domain expert), resulting in a final isomorphic lattice to the original. While most methods for eliminating non-essential distinctions use formal context as a preliminary step in the reduction process, and some involve a domain expert to lead the reduction process, where there are no methods for constructing an isomorphic final lattice or a subset of the original concept lattice, all of them have a different final lattice than the original. Considering the methods of concept filtration, most have the formal context as a preliminary step for the reduction process. Most don't use domain experts, resulting in a final lattice that is partial to the original lattice. Furthermore, all reduction methods in this category have both fidelity loss and descriptive loss.

The most significant reductions are those that can significantly reduce the amount of space to be examined. Such reductions are acquired through "non-essential distinctions elimination" and "concept filtration" methods. Methods for "non-essential distinctions elimination" may be regarded as "risky" because they have the potential to alter the set of formal concepts dramatically. It's important to ensure that the process that leads to these changes keeps the concept lattice's core components. Although concept filtration methods are intriguing because they reduce the concept space, traversing such a place should be done in such a way that the important concepts are reached. Our future work will move in two directions: first, we will categorize the reduction techniques for FCA extensions such as fuzzy formal context, fuzzy concept lattice, rough set theory, and decision formal context, based on our categorization for the reduction methods of classical FCA in this work. Second, based on the findings of the reduction methods analyzed in this study, we will choose the methods that appear to be the best for making some modifications and improvements.

## REFERENCES


[1] R. Wille, "Restructuring lattices theory: an approach on hierarchies of concepts," *International conference on formal concept analysis,* Dordrecht, Holland: Springer, 1982, doi: 10.1007/978-94-009-7798-3_15.

[2] R. Ganter and R. Wille, *Formal concept analysis: Mathematical foundations*, Germany, Berlin: Springer-Verlag, 1999, doi: 10.1007/978-3-642-59830-2.

[3] K. Sumangali and C. A. Kumar, "Critical analysis on open source LMSs using FCA," *International Journal of Distance Education Technologies (IJDET),* vol. 11, no. 4, pp. 97–111, 2013, doi: 10.4018/ijdet.2013100107.

[4] S. Hao, C. Shi, Z. Niu, and L. Cao, "Concept coupling learning for improving concept lattice-based document retrieval," *Engineering Applications of Artificial Intelligence,* vol. 69, pp. 65–75, 2018, doi: 10.1016/j.engappai.2017.12.007.

[5] L. E. Zárate and S. M. Dias, "Qualitative behavior rules for the cold rolling process extracted from trained ANN via the FCANN method," *Engineering Applications of Artificial Intelligence,* vol. 22, no. 4–5, pp. 718–731, 2009, doi: 10.1016/j.engappai.2008.11.009.

[6] M. Priya and A. K. Ch, "A novel method for merging academic social network ontologies using formal concept analysis and hybrid semantic similarity measure," *Library Hi Tech,* 2019, doi: 10.1108/LHT-02-2019-0035.

[7] T. T. Quan, S. C. Hui, A. C. M. Fong, and T. H. Cao, "Automatic generation of ontology for scholarly semantic web," in *International Semantic Web Conference,* 2004, pp. 726–740, doi: 10.1007/978-3-540-30475-3_50.

[8] C. de Maio, G. Fenza, V. Loia, and S. Senatore, "Hierarchical web resources retrieval by exploiting fuzzy formal concept analysis," *Information Processing & Management,* vol. 48, no. 3, pp. 399–418, 2012, doi: 10.1016/j.ipm.2011.04.003.

[9] L. González and A. Hogan, "Modelling dynamics in semantic web knowledge graphs with formal concept analysis," in *Proceedings of the 2018 World Wide Web Conference,* 2018, pp. 1175–1184, doi: 10.1145/3178876.3186016.

[10] C. M. Rocco, E. Hernandez-Perdomo, and J. Mun, "Introduction to formal concept analysis and its applications in reliability engineering," *Reliability Engineering & System Safety,* vol. 202, p. 107002, 2020, doi: 10.1016/j.ress.2020.107002.

[11] N. Jay, F. Kohler, and A. Napoli, "Analysis of social communities with iceberg and stability-based concept lattices," in *International Conference On Formal Concept Analysis,* 2008, pp. 258–272, doi: 10.1007/978-3-540-78137-0_19.

[12] F. Hao, Y. Yang, G. Min, and V. Loia, "Incremental construction of three-way concept lattice for knowledge discovery in social networks," *Information Sciences,* vol. 578, pp. 257–280, 2021, doi: 10.1016/j.ins.2021.07.031.

[13] F. Hao, Y. Yang, B. Pang, N. Y. Yen, and D.-S. Park, "A fast algorithm on generating concept lattice for symmetry formal context constructed from social networks," *Journal of Ambient Intelligence and Humanized Computing,* pp. 1–8, 2019, doi: 10.1007/s12652-019-01274-6.

[14] Y. Yang, F. Hao, B. Pang, G. Min, and Y. Wu, "Dynamic maximal cliques detection and evolution management in social internet of things: A formal concept analysis approach," *IEEE Transactions on Network Science and Engineering,* vol. 9, no. 3, pp. 1020–1032, 2021, doi: 10.1109/TNSE.2021.3067939.

[15] F. Hao, G. Min, Z. Pei, D.-S. Park, and L. T. Yang, "$ K $-clique community detection in social networks based on formal concept analysis," *IEEE Systems Journal,* vol. 11, no. 1, pp. 250–259, 2015, doi: doi:10.1109/JSYST.2015.2433294.

[16] F. Hao, D.-S. Park, G. Min, Y.-S. Jeong, and J.-H. Park, "k-Cliques mining in dynamic social networks based on triadic formal concept analysis," *Neurocomputing,* vol. 209, pp. 57–66, 2016, doi: 10.1016/j.neucom.2015.10.141.

[17] J. Poelmans, D. I. Ignatov, S. O. Kuznetsov, and G. Dedene, "Formal concept analysis in knowledge processing: A survey on applications," *Expert Systems with Applications,* vol. 40, no. 16, pp. 6538–6560, 2013, doi: 10.1016/j.eswa.2013.05.009.

[18] S. M. Dias and N. J. Vieira, "Applying the JBOS reduction method for relevant knowledge extraction," *Expert Systems with Applications,* vol. 40, no. 5, pp. 1880–1887, 2013, doi: 10.1016/j.eswa.2012.10.010.

[19] A. K. Ch, "Concept lattice simplification in formal concept analysis using attribute clustering," *Journal of Ambient Intelligence and Humanized Computing,* vol. 10, pp. 2327–2343, 2019, doi: 10.1007/s12652-018-0831-2.

[20] R. Belohlavek and J. Macko, "Selecting important concepts using weights," in *International Conference On Formal Concept Analysis*, 2011, pp. 65–80, doi: 10.1007/978-3-642-20514-9_7.
[21] C. A. Kumar and S. Srinivas, "Concept lattice reduction using fuzzy K-means clustering," *Expert Systems with Applications,* vol. 37, no. 3, pp. 2696–2704, 2010, doi: 10.1016/j.eswa.2009.09.026.
[22] P. Krajca, J. Outrata, and V. Vychodil, "Parallel recursive algorithm for FCA," in *CLA*, 2008, pp. 71–82.
[23] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," *OSDI'04: Sixth Symposium on Operating System Design and Implementation,* San Francisco, 2004, pp. 137-150.
[24] P. Krajca and V. Vychodil, "Distributed algorithm for computing formal concepts using map-reduce framework," in *International Symposium on Intelligent Data Analysis,* pp. 333–344, 2009, doi: 10.1007/978-3-642-03915-7_29.
[25] R. K. Chunduri, A. K. Cherukuri, and M. Tamir, "Concept generation in formal concept analysis using MapReduce framework," in 2017 *International Conference on Big Data Analytics and Computational Intelligence (ICBDAC),* 2017, pp. 191–204, doi: 10.1109/ICBDACI.2017.8070834.
[26] C. Lindig, "Fast concept analysis," *Working with Conceptual Structures-Contributions to ICCS*, vol. 2000, pp. 152–161, 2000.
[27] R. K. Chunduri and A. K. Cherukuri, "Scalable formal concept analysis algorithms for large datasets using spark," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 11, pp. 4283–4303, 2019, doi: 10.1007/s12652-018-1105-8.
[28] J. Li, C. Mei, and Y. Lv, "A heuristic knowledge-reduction method for decision formal contexts," *Computers & Mathematics with Applications,* vol. 61, no. 4, pp. 1096–1106, 2011, doi: 10.1016/j.camwa.2010.12.060.
[29] J. Medina, "Relating attribute reduction in formal, object-oriented and property-oriented concept lattices," *Computers & Mathematics with Applications,* vol. 64, no. 6, pp. 1992–2002, 2012, doi: 10.1016/j.camwa.2012.03.087.
[30] H. Wang and W.-X. Zhang, "Approaches to knowledge reduction in generalized consistent decision formal context," *Math Comput Model,* vol. 48, no. 11–12, pp. 1677–1684, 2008, doi: 10.1016/j.mcm.2008.06.007.
[31] L. Antoni, M. E. Cornejo, J. Medina, and E. Ramírez-Poussa, "Attribute classification and reduct computation in multi-adjoint concept lattices," *IEEE Transactions on Fuzzy Systems,* vol. 29, no. 5, pp. 1121–1132, 2020, doi: 10.1109/TFUZZ.2020.2969114.
[32] S. M. Dias and N. Vieira, "Reducing the size of concept lattices: The JBOS approach," in Cla, 2010, vol. 672, pp. 80–91.
[33] V. Codocedo, C. Taramasco, and H. Astudillo, "Cheating to achieve formal concept analysis over a large formal context," in *The Eighth International Conference on Concept Lattices and their Applications-CLA* 2011, 2011, pp. 349–362.
[34] C. A. Kumar and S. Srinivas, "Mining associations in health care data using formal concept analysis and singular value decomposition," *J Biol Syst,* vol. 18, no. 04, pp. 787–807, 2010, doi: 10.1142/S0218339010003512.
[35] G. Stumme, R. Taouil, Y. Bastide, N. Pasquier, and L. Lakhal, "Computing iceberg concept lattices with titanic," *Data & Knowledge Engineering,* vol. 42, no. 2, pp. 189–222, 2002, doi: 10.1016/S0169-023X(02)00057-5.
[36] P. K. Singh and C. A. Kumar, "Concept lattice reduction using different subset of attributes as information granules," *Granular Computing,* vol. 2, no. 3, pp. 159–173, 2017, doi: 10.1007/s41066-016-0036-z.
[37] S. A. Yevtushenko, "System of data analysis concept explorer," *in Proc. 7th National Conference on Artificial Intelligence (KII'00),* 2000, pp. 127–134.
[38] K. Sumangali and C. A. Kumar, "A comprehensive overview on the foundations of formal concept analysis," *Knowledge Management & E-Learning: An International Journal*, vol. 9, no. 4, pp. 512–538, 2017, doi: 10.34105/j.kmel.2017.09.032.
[39] G. Stumme, "Efficient data mining based on formal concept analysis," in *International conference on database and expert systems applications,* 2002, pp. 534–546, doi: 10.1007/3-540-46146-9_53.
[40] W.-X. Zhang, L. Wei, and J.-J. Qi, "Attribute reduction in concept lattice based on discernibility matrix," in *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing,* 2005, pp. 157–165, doi: 10.1007/11548706_17.
[41] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Śle¸ zak, "A computational procedure for variable selection preserving different initial conditions," *International Journal of Computer Mathematics,* vol. 97, no. 1–2, pp. 387–404, 2020, doi: 10.1080/00207160.2019.1613530.
[42] Z. Gong, B. Sun, and D. Chen, "Rough set theory for the interval-valued fuzzy information systems," *Information Sciences,* vol. 178, no. 8, pp. 1968–1985, 2008, doi: 10.1016/j.ins.2007.12.005.
[43] V. Snasel, P. Gajdos, H. M. D. Abdulla, and M. Polovincak, "Using matrix decompositions in formal concept analysis," *ISIM,* vol. 252, 2007.
[44] M. Trnecka and M. Trneckova, "Data reduction for Boolean matrix factorization algorithms based on formal concept analysis," *Knowledge-Based Systems,* vol. 158, pp. 75–80, 2018, doi: 10.1016/j.knosys.2018.05.035.
[45] N. Messai, M.-D. Devignes, A. Napoli, and M. Smail-Tabbone, "Many-valued concept lattices for conceptual clustering and information retrieval," *in ECAI*, 2008, vol. 178, pp. 127–131.
[46] M. Liu, M. Shao, W. Zhang, and C. Wu, "Reduction method for concept lattices based on rough set theory and its application," *Computers & Mathematics with Applications,* vol. 53, no. 9, pp. 1390–1410, 2007, doi: 10.1016/j.camwa.2006.03.040.
[47] M. J. Benítez-Caballero, J. Medina, and E. Ramírez-Poussa, "Unifying reducts in formal concept analysis and rough set theory," *Trends in Mathematics and Computational Intelligence*, pp. 89–95, 2019, doi: 10.1007/978-3-030-00485-9_10.
[48] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Ślęzak, "Rough-set-driven approach for attribute reduction in fuzzy formal concept analysis," *Fuzzy Sets and Systems,* vol. 391, pp. 117–138, 2020, doi: 10.1016/j.fss.2019.11.009.
[49] U. Priss and L. J. Old, "Data weeding techniques applied to roget's thesaurus," in *Knowledge Processing and Data Analysis, Springer*, 2007, pp. 150–163. doi: 10.1007/978-3-642-22140-8_10.
[50] S. M. Dias and N. J. Vieira, "Concept lattices reduction: definition, analysis and classification," *Expert Systems with Applications,* vol. 42, no. 20, pp. 7084–7097, 2015, doi: 10.1016/j.eswa.2015.04.044.
[51] W. Zhang, L. Wei, and J. J. Qi, "Attribute reduction theory and approach to concept lattice," *Science in China Series F: Information Sciences,* vol. 35, pp. 628–639, 2005, doi: 10.1360/122004-104.
[52] J.-J. Qi, "Attribute reduction in formal contexts based on a new discernibility matrix," *Journal of Applied Mathematics and Computing volume,* vol. 30, no. 1, pp. 305–314, 2009, doi: 10.1007/s12190-008-0174-9.
[53] X. Wang and W. Zhang, "Attribute reduction in concept lattices based on deletion transformations," in 2010 *Sixth International Conference on Natural Computation,* 2010, vol. 4, pp. 2065–2069, doi: 10.1109/ICNC.2010.5582340.
[54] J. Wang, J. Liang, and Y. Qian, "A heuristic method to attribute reduction for concept lattice," in 2010 *International Conference on Machine Learning And Cybernetics,* vol. 1, 2010, pp. 483–487, doi: 10.1109/ICMLC.2010.5581015.
[55] R. Belohlavek, "Similarity relations in concept lattices," *Journal of Logic and Computation,* vol. 10, no. 6, pp. 823–845, 2000, doi: 10.1093/logcom/10.6.823.
[56] W.-Z. Wu, Y. Leung, and J.-S. Mi, "Granular computing and knowledge reduction in formal contexts," *IEEE Transactions on Knowledge and Data Engineering,* vol. 21, no. 10, pp. 1461–1474, 2008, doi: 10.1109/TKDE.2008.223.

[57] M. E. Cornejo, J. Medina, and E. Ramírez-Poussa, "Attribute and size reduction mechanisms in multi-adjoint concept lattices," *Journal of Computational and Applied Mathematics*, vol. 318, pp. 388–402, 2017, doi: 10.1016/j.cam.2016.07.012.
[58] K. S. K. Cheung and D. Vogel, "Complexity reduction in lattice-based information retrieval," *Information Retrieval*, vol. 8, no. 2, pp. 285–299, 2005, doi: 10.1007/s10791-005-5663-y.
[59] D. S. M. CH AK and N. J. VIEIRA, "Knowledge reduction in formal contexts using non-negative matrix factorization," *Mathematics and Computers in Simulation*, vol. 109, pp. 46–63, 2015, doi: 10.1016/j.matcom.2014.08.004.
[60] K. Sumangali and C. A. Kumar, "Knowledge reduction in formal contexts through CUR matrix decomposition," *Cybernetics and Systems*, vol. 50, no. 5, pp. 465–496, 2019, doi: 10.1080/01969722.2019.1602300.
[61] S. Zhang, P. Guo, J. Zhang, X. Wang, and W. Pedrycz, "A completeness analysis of frequent weighted concept lattices and their algebraic properties," *Data & Knowledge Engineering*, vol. 81, pp. 104–117, 2012, doi: 0.1016/j.datak.2012.08.002.
[62] M. Kauer and M. Krupka, "Removing an incidence from a formal context," in *CLA*, 2014, pp. 195–206.
[63] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal, "Discovering frequent closed itemsets for association rules," in *International Conference on Database Theory*, 1999, pp. 398–416, doi: 10.1007/3-540-49257-7_25.
[64] H. Soldano, V. Ventos, M. Champesme, and D. Forge, "Incremental construction of alpha lattices and association rules," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2010, pp. 351–360, doi: 10.1007/978-3-642-15390-7_36.
[65] J.-F. Boulicaut and J. Besson, "Actionability and formal concepts: A data mining perspective," in *International Conference on Formal Concept Analysis*, 2008, pp. 14–31, doi: 10.1007/978-3-540-78137-0_2.
[66] R. Belohlavek and V. Vychodil, "Closure-based constraints in formal concept analysis," *Discrete Applied Mathematics*, vol. 161, no. 13–14, pp. 1894–1911, 2013, doi: 10.1016/j.dam.2013.02.029.
[67] J. Muangprathub, V. Boonjing, and P. Pattaraintakorn, "A new case-based classification using incremental concept lattice knowledge," *Data & Knowledge Engineering*, vol. 83, pp. 39–53, 2013, doi: 10.1016/j.datak.2012.10.001.
[68] F. Alqadah and R. Bhatnagar, "Similarity measures in formal concept analysis," *Annals of Mathematics and Artificial Intelligence*, vol. 61, no. 3, pp. 245–256, 2011, doi: 10.1007/s10472-011-9257-7.
[69] S. O. Kuznetsov, "On stability of a formal concept," *Annals of Mathematics and Artificial Intelligence*, vol. 49, no. 1, pp. 101–115, 2007, doi: 10.1007/s10472-007-9053-6.
[70] S. Kuznetsov, S. Obiedkov, and C. Roth, "Reducing the representation complexity of lattice-based taxonomies," in *International Conference on Conceptual Structures*, 2007, pp. 241–254, doi: 10.1007/978-3-540-73681-3_18.
[71] B. le Grand, M. Soto, and M.-A. Aufaure, "Conceptual and spatial footprints for complex systems analysis: Application to the semantic web," in *International Conference on Database and Expert Systems Applications*, 2009, pp. 114–127, doi: 10.1007/978-3-642-03573-9_9.
[72] M. Klimushkin, S. Obiedkov, and C. Roth, "Approaches to the selection of relevant concepts in the case of noisy data," in *International Conference on Formal Concept Analysis*, 2010, pp. 255–266, doi: 10.1007/978-3-642-11928-6_18.
[73] M. A. Babin and S. O. Kuznetsov, "Approximating concept stability," in *International Conference on Formal Concept Analysis*, 2012, pp. 7–15, doi: 10.1007/978-3-642-29892-9_7.
[74] R. Belohlavek, B. de Baets, and J. Konecny, "Zoom-in/zoom-out algorithms for FCA with attribute granularity," in *Computer And Information Sciences II, Springer*, pp. 549–555, 2011, doi: 10.1007/978-1-4471-2155-8_70.
[75] P. Gajdoš, P. Moravec, and V. Snášel, "Concept lattice generation by singular value decomposition," in *CLA*, vol. 2004, 2004, pp. 13–22.
[76] A. Gély, "Links between modular decomposition of concept lattice and bimodular decomposition of a context," in *CLA*, 2011, pp. 393–403.
[77] J. Liu and J.-S. Mi, "A novel approach to attribute reduction in formal concept lattices," in *International Conference on Rough Sets and Knowledge Technology*, 2008, pp. 426–433, doi: 10.1007/978-3-540-79721-0_59.
[78] X. Wang and W. Zhang, "Relations of attribute reduction between object and property oriented concept lattices," *Knowledge-Based Systems*, vol. 21, no. 5, pp. 398–403, 2008, doi: 10.1016/j.knosys.2008.02.005.
[79] M. D. Rice and M. Siff, "Clusters, concepts, and pseudometrics," *Electron Notes Theor Comput Sci*, vol. 40, pp. 323–346, 2001, doi: 10.1016/S1571-0661(05)80060-X.
[80] R. Bělohlávek and V. Vychodil, "Formal concept analysis with constraints by closure operators," in *International Conference on Conceptual Structures*, 2006, pp. 131–143, doi: 10.1007/11787181_10.
[81] M. Li and G. Wang, "Approximate concept construction with three-way decisions and attribute reduction in incomplete contexts," *Knowledge-Based Systems*, vol. 91, pp. 165–178, 2016, doi: 10.1016/j.knosys.2015.10.010.
[82] R. Belohlávek, V. Sklenar, and J. Zacpal, "Concept lattices constrained by attribute dependencies," in *DATESO*, 2004, pp. 63–73.
[83] S. K. Cheung, "Complexity reduction in lattice-based information retrieval: theory, prototype development, and evaluation," Doctoral Thesis, City University of Hong Kong, Hong Kong, 2004.
[84] R. Belohlávek and V. Sklenar, "Formal concept analysis over attributes with levels of granularity," in *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*, vol. 1, 2005, pp. 619–624, doi: 10.1109/CIMCA.2005.1631332.
[85] C. A. Kumar, "Fuzzy clustering-based formal concept analysis for association rules mining," *Applied Artificial Intelligence*, vol. 26, no. 3, pp. 274–301, 2012, doi: 10.1080/08839514.2012.648457.
[86] M.-W. Shao, W.-Z. Wu, X.-Z. Wang, and C.-Z. Wang, "Knowledge reduction methods of covering approximate spaces based on concept lattice," *Knowledge-Based Systems*, vol. 191, p. 105269, 2020, doi: 10.1016/j.knosys.2019.105269.
[87] R. Belohlavek and V. Vychodil, "Formal concept analysis with background knowledge: attribute priorities," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 39, no. 4, pp. 399–409, 2009, doi: 10.1109/TSMCC.2008.2012168.
[88] R. Bělohlávek, V. Sklenář, and J. Zacpal, "Formal concept analysis with hierarchically ordered attributes," *International Journal of General Systems*, vol. 33, no. 4, pp. 383–394, 2004, doi: 10.1080/03081070410001679715.
[89] G. Arévalo, A. Berry, M. Huchard, G. Perrot, and A. Sigayret, "Performances of galois sub-hierarchy-building algorithms," in *International Conference on Formal Concept Analysis*, 2007, pp. 166–180, doi: 10.1007/978-3-540-70901-5_11.
[90] R. Bělohlávek, J. Outrata, and V. Vychodil, "Thresholds and shifted attributes in formal concept analysis of data with fuzzy attributes," *in International Conference on Conceptual Structures*, 2006, pp. 117–130, doi: 10.1007/11787181_9.
[91] N. Pernelle, M.-C. Rousset, H. Soldano, and V. Ventos, "Zoom: a nested Galois lattices-based system for conceptual clustering," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 14, no. 2–3, pp. 157–187, 2002, doi: 10.1080/09528130210164189.
[92] V. Ventos and H. Soldano, "Alpha galois lattices: An overview," in *International Conference on Formal Concept Analysis*, 2005, pp. 299–314, doi: 10.1007/978-3-540-32262-7_21.

## BIOGRAPHIES OF AUTHORS

**Mohammed Alwersh** received the B.Sc. degree in Computer Science in 2008 from the University of Al-Qaddissiyah, Iraq, and in 2014 the M.Sc. degree in Information Technology from Belarusian state university of informatics and radioelectronics (BSUIR), Minsk-Belarus. He is a PhD student at the University of Miskolc, Hungary, since 2020. The Topic title is "Attribute set reduction of categorical attributes for machine learning and FCA methods". He can be contacted at email: alwersh.mohammed.ali.daash@student.uni-miskolc.hu.

**Prof. Dr. László Kovács** head of the Department of the Institute of Informatics of the University of Miskolc. His main research interests are optimization of neural network-based prediction methods, theory of concept lattices, NLP and automatic question generation procedures, applications of ontology in knowledge engineering. He is an author of more than 90 journal publications in the related research fields. He is an active member of the editorial board at 7 international journals. He can be contacted at email: kovacs@iit.uni-miskolc.hu.