La Rochelle Université The REDUCEDCONTEXTCOMPLETION Algorithm

The ReducedContextCompletion Algorithm ating the reduced context of a lattice in linear time and linear men

The Galactic Organization <contact@thegalactic.org>



¹C 2018-2022 the Galactic Organization. This document is licensed under CC-by-nc-nd (https://cnsatiwecommons.org/licenses/by-nc-nd/4.0/deed.en)

The ReducedContextCompletion Algorithm

Updating the reduced context of a lattice in linear time and linear memory

The Galactic Organization <contact@thegalactic.org>



 $^1 \odot$ 2018-2022 the Galactic Organization. This document is licensed under CC-by-nc-nd (https://creativecommons.org/licenses/by-nc-nd/4.0/deed.en)

03-11

2022-

Motivation Interractive navigation Example (Application on integer lattice) Exemple of the reduced context completion

The REDUCEDCONTEXTCOMPLETION Algorithm Motivation Motivation Motivation Motivation



Motivation

La Rochelle Université

Exporting the FCA outside of its inner self

The FCA is a powerfull tool to deal with complex data, but the output can be hard to hard for non-lt users. The following work will be a first step to:

- Allow non-It users to uses FCA algorithms.
- Put the data-scientist in the center of the process.



Figure 1: Sub-sequence match of touristics trajectories of La Rochelle

Motivation Interractive navigation Example (Application on integer lattice) Exemple of the reduced context completion

The REDUCEDCONTEXTCOMPLETION Algorithm



Motivation

La Rochelle Université

Exporting the FCA outside of its inner self

- Build in an iterative and intuitive way a lattice of concepts.
- Allow the user to change the strategy during the building of the lattice.



Figure 2: Sub-sequence match of touristics trajectories of La Rochelle



Change the strategy

-11

2022-03-





-Motivation

-Motivation

Change the strategy



The REDUCEDCONTEXTCOMPLETION Algorithm
Motivation
Motivation
Change the strategy



Motivation

Introduction The reduced context completion Theorem The algorithm of ReducedContextCompletion Conclusion

La Rochelle Université Motivation Interractive navigation Example (Application on integer lattice) Exemple of the reduced context completion

A first step to the conceptual navigation into lattice-shaped data-structure : The conceptual navigation

- Achieve a proper interactive navigation into complex data structures such as lattices.
- Put the data-scientist at the center of the analysis, because only him know the semantic of its data
- Giving the possibility to change strategies "on the road" (Allowed by the NextPriorityConcept algorithm)

- The REDUCEDCONTEXTCOMPLETION Algorithm
- -Motivation

11

0ġ

,

202

- └─Interractive navigation
 - A first step to the conceptual navigation into lattice-shaped data-structure :

It tigs to the conceptual exception into Littles-shaped data-structure = the conceptual exception. P Athene a proper interactive marigation into complex data structures such as tartice. P Put the data-scientific at the conter of the analysis, because only him know the semantic of its data. B Giving the possibility to change strategies "on the road" (Alowed by the NextPriori/Concert Alerrithm).

Motivation

Introduction The reduced context completion Theorem The algorithm of ReducedContextCompletion Conclusion

Motivation Interractive navigation Example (Application on integer lattice) Exemple of the reduced context completion

A first step to the conceptual navigation into lattice-shaped data-structure : The conceptual navigation

- Achieve a proper interactive navigation into complex data structures such as lattices.
- Put the data-scientist at the center of the analysis, because only him know the semantic of its data
- Giving the possibility to change strategies "on the road" (Allowed by the NextPriorityConcept algorithm)

Optimization of the solution

La Rochelle

Université

Rather than rebuilding the lattice (that can be huge) :

- Maintain a condensed form (the reduced context) of a lattice.
- Update this reduced context according to the changes made by the analysis.

The REDUCEDCONTEXTCOMPLETION Algorithm

└─ Motivation

11

Ö

,

202

- └─Interractive navigation
 - A first step to the conceptual navigation into lattice-shaped data-structure :

The conceptual navigation > Achieve a proper interactive navigation into complex data structures such as lattices. > Put the data-scientist at the center of the analysis, baccase only him know the semantic of its data and the structure structure of the structure of the source of the - Gring the possibility to change strategies "on the road" (Allowed by the Nuclfrieinfycomest algorithm)

ptimization of the soluti

ther than rebuilding the lattice (that can be huge) : Maintain a condensed form (the reduced context) of a lattice. Uddate this reduced context according to the chanses made by the analysis. Motivation

Introduction M The reduced context completion In Theorem E The algorithm of ReducedContextCompletion E Conclusion

Motivation Interractive navigation Example (Application on integer lattice) Exemple of the reduced context completion

prime factor lattice

La Rochelle

Université

Definition of a prime factor lattice

In the prime factor lattice, elements are tuple of prime number divisor. The meet operation between two points seeks the greatest common multiple of those two numbers, while the join operation return the greatest common divisor such as shown in figure 5.



Figure 5: Example of a prime factor

-11

-03

d'

202

The REDUCEDCONTEXTCOMPLETION Algorithm

Example (Application on integer lattice)





(2)

Figure 6: Initial lattice

 $\frac{J^{\perp} - M^{\perp}}{\emptyset - \emptyset}$



Figure 6: Initial lattice





 $\frac{J^{\scriptscriptstyle A}}{(1,3)} = \frac{M^{\scriptscriptstyle A}}{(2,6)}$ (1,2) = (3,6)



 $egin{array}{cccc} \hline J^{\perp} & M^{\perp} \\ \hline (1,3) & (2,2) \\ (1,2) & (3,3) \\ (1,7) & (2,42) \\ & (14,42) \\ & (21,42) \\ \hline \end{array}$

Figure 8: Add the element 7

La Rochelle
UniversitéMotivation
Introduction
The reduced context completion
The algorithm of ReducedContextCompletion
ConclusionMotivation
Interractive navigation
Example (Application on integer lattice)
Exemple of the reduced context completion
 $x_4 = 9$



Figure 9: Add the element **9**

21

126

-Exemple of the reduced context completion

2022-03-11

12/61

 $L_{x_4} = 9$



J^L M^L



Relation order

The REDUCEDCONTEXTCOMPLETION Algorithm

Relation order

Contractoring contracts is a binary relation on the set S which satisfy three properties such as : $\blacktriangleright \leq is$ reflexive : for all $x \in S, x \leq x$ $\vdash \leq is$ antisymmetric : for all $x, y \in S$ if $x \leq y$ and $y \leq x$ then x = y $\vdash \leq is transitive : with <math>x, y, z \in S$, if we have $x \leq y$ and $y \leq x$ then $x \leq z$.

Ordinal definition

- \leq is a binary relation on the set S which satisfy three properties such as :
- ► \leq is **reflexive** : for all $x \in S, x \leq x$
- ▶ ≤ is **antisymmetric** : for all $x, y \in S$ if $x \leq y$ and $y \leq x$ then x = y
- ▶ ≤ is transitive : with $x, y, z \in S$, if we have $x \le y$ and $y \le z$ then $x \le z$.

11

0ġ

Ń.



The meet and join operator

Ordinal definition

- The greatest lower bound of two elements (also named a join) x, y is noted x ∨ y refers to the greatest element of the predecessors of both x and y such that z ≤ x and z ≤ y, z ∈ S.
- The least upper bound of two elements (also named a meet) x, y is noted x ∧ y refers to the smallest element of the successor of both x and y such that x ≥ z and y ≥ z, z ∈ S.
- \blacktriangleright \lor and \land can be used to defines the partial order relation \leq on S in the context of a lattice structure.

The REDUCEDCONTEXTCOMPLETION Algorithm
Introduction
Basic definition
The meet and join operator

The meet and join operator

The greatest lower bound of two elements (also named a join) x, y is noted $x \lor y$
refers to the greatest element of the predecessors of both x and y such that $z \le x$
and $z \le y, z \in S$.
The least upper bound of two elements (also named a meet) x, y is noted $x \land y$
refers to the smallest element of the successor of both x and y such that $x \ge x$ and
$y \ge x, x \in S$.
\lor and \land can be used to defines the partial order relation \le on S in the context of a
lattice structure.

11

0ġ

,



Definition of a lattice

Lattice definition

- A lattice $L = \langle S, \leq \rangle$ is a poset such that $x \lor y$ and $x \land y$ exist for any x and $y \in S$.
- By considering ∨ and ∧ as meet and join operators, we can also defines a lattice as L = ⟨S, ∨, ∧⟩^{S≠∅}.



The REDUCEDCONTEXTCOMPLETION Algorithm



-11

0ġ

Ċ.



The irreducible

Reduced context definition

- An element j ∈ S is called join-irreducible if for any subset X of S, j ≠ ∧X. We noted j a join-irreducible on the lattice L, j_L. The set of the join-irreducible of L is noted J_L. In a lattice, a join-irreducible element only has one immediate predecessor noted j⁻.
- An element $m \in S$ is called meet-irreducible if for any subset X of S, $m \neq \forall X$. We noted m a join-irreducible on the lattice L, m_L . The set of the meet-irreducible of L is noted M_L . In a lattice, a meet-irreducible element only has one immediate successor noted m^+ .

The REDUCEDCONTEXTCOMPLETION Algorithm

11

0ġ

2022-

The irreducibl

context definition

▶ An element j ∈ S is called join-invelocible if for any subset X of S, j + AX. We noted j is join-invelocible of L is and the transmission of the significant structure of the significant structure is a significant structure of the significant st

La Rochelle Université ______ The algorithm of Reduced Context Completion ______ Conclusion

The irreducible

Lattice definition

Irreducible elements of a lattice can't be obtain by \lor operator (for join irreducible) or \land (for meet irreducible), they represent the very structure of a lattice.

- We can't have more join irreducible elements than the number of individuals in our data.
- We can't have more meet irreducible elements than the number of different attributes in all our individuals.



Figure 11: Lattice L, with colored join irreducible and meet irreducible

11

0ġ

,

202

The REDUCEDCONTEXTCOMPLETION Algorithm

The irreducible





The reduced context definition

Reduced context definition

- ▶ The reduced context (Also called the table) of a lattice is defined as both all elements in J_L and M_L and the relational order \leq on S such that $R_L = \langle J_L, M_L, \leq \rangle$.
- Based on the fundamental theorem, any lattice L is isomorphic to the concept lattice of its reduced context, we can then rebuilt any lattice L based on its irreducible.

- The REDUCEDCONTEXTCOMPLETION Algorithm -Introduction

11

0ġ

d'

202

- The reduced context
 - The reduced context definition

The reduced context (Also called the table) of a lattice is defined as both all lements in h and M_i and the relational order $< \alpha n S$ such that $R_i = (h M_i)$ Based on the fundamental theorem, any lattice L is isomorphic to the concept lattice of its reduced context, we can then rebuilt any lattice I, based on its



The reduced context definition

Maintaining the reduced context

- As we can rebuild a lattice L from its reduced context (And as the reduced context only contains irreducible elements and binaries operators of L), we only need to stock the structure R.
- By maintaining and working only on R_L, we drastically reduce the number of elements to take into account during the process and the storage.

- The REDUCEDCONTEXTCOMPLETION Algorithm

11

0ġ

,

202

- └─The reduced context
 - └─The reduced context definition

The reduced context definition

ng the reduced context

 As we can rebuild a lattice L from its reduced context (And as the reduced contex only contains irreducible elements and binniss operators of L), we only need to solock the structure R.
 By maintaining and working only on R₁, we drastically reduce the number of elements to take into account during the process and the storage.

The function ompletion ompletion Conclusion The function Conclusion

Definition : joins and immediate predecessors

La Rochelle Université

Covering relation between joins and immediate predecessors

For the purpose of the algorithm, we introduce a covering relation between joins irreducible and theirs immediate predecessor such as :

 $L \prec_{|J} = \{(j^-, j) | j \in J\}$ Where we note $J_L^{\wedge} = L \prec_{|J}$ such that J_L^{\wedge} is a list of couple (j^-, j) . The REDUCEDCONTEXTCOMPLETION Algorithm

The reduced context completion

-11

-03

2022-

- -Problematic Covering relation meet/successor and join/predecessor
- Definition : joins and immediate predecessors

Definition : joins and immediate predecesso

ering relation between joins and immediate predecessors the purpose of the algorithm, we introduce a covering relation between joins ucible and theirs immediate predecessor such as :

 ${}^L \prec_{|J} = \{(j^-,j)|j \in J\}$ Where we note $J_L^{\perp} = {}^L \prec_{|J}$ such that J_L^{\perp} is a list of couple $(j^-,j).$

La Rochelle Université
 Motivation

 Introduction
 Problematic Covering relation meet/successor and join/predecessor

 completion
 The function

 Theorem
 Example

 Conclusion
 Example

Definition : meets and immediate successors

Covering relation between meets and immediate successors

For the purpose of the algorithm, we introduce a covering relation between meets irreducible and theirs immediate successors such as :

 ${}^{L}\prec_{|M}=\{(m,m^{+})|m\in M\}$

Where we note $M_L^{\wedge} = {}^L \prec_{|M}$ such that M_L^{\wedge} is a list of couple (m, m^+) .

The REDUCEDCONTEXTCOMPLETION Algorithm

└─The reduced context completion

- -Problematic Covering relation meet/successor and join/predecessor
- Definition : meets and immediate successors

Definition : meets and immediate success

ring relation between meets and immediate successors he purpose of the algorithm, we introduce a covering relation between meets ucble and theirs immediate successors such as :

 ${}^L\prec_{|M}=\{\{m,m^+)|m\in M\}$ Where we note $M_L^{\perp}={}^L\prec_{|M}$ such that M_L^{\perp} is a list of couple $(m,m^+).$

-03-11

2022-

La Rochelle Université _____ The algorithm of ReducedContextCompletion _____ Conclusion

Example

Example

In the following example figure 19 we have :





- The REDUCEDCONTEXTCOMPLETION Algorithm
 - Problematic Covering relation meet/successor and join/predecessor
 - -Example

-03-11

2022-



La Rochelle Université The algorithm of ReducedContextCompletion Conclusion

Problematic

Two sets of couple

In this work, we will uses a special reduced context of L where $R_L^{\wedge} = \langle J_L^{\wedge}, M_L^{\wedge}, \wedge, \vee \rangle$.

Two sets of couple

Based on R_L^{λ} , the reduced context of L, and X, $X \subseteq S$, find the reduced context $R_{L_X}^{\lambda}$ of the sublattice L_X , $L_X \subseteq L$, the smallest lattice containing X.

The REDUCEDCONTEXTCOMPLETION Algorithm

 $\begin{array}{ccc} \overrightarrow{H} & & & \\ \overrightarrow{P} & & & \\ \end{array}$ The reduced context completion \overrightarrow{P} roblematic Covering relation \overrightarrow{r}

Problematic Covering relation meet/successor and join/predecessor

Problematic

2022-

Two sets of couple In this work, we will uses a special reduced context of *L* where a Two sets of couple

Based on R_L^+ , the reduced context of L, and X, $X \subseteq S$, find the reduced context $R_{L_X}^+$ of the sublattice L_X , $L_X \subseteq L$, the smallest lattice containing X. La Rochelle Université The algorithm of Reduced Context Completion Conclusion

The reduce context completion function

Foreword

- ▶ Let L be a finite lattice, with a bottom element noted \perp_L and a top element noted \top_L
- With CL an application that build a the concept lattice, where L = CL(⟨M^A_X, J^A_X, ⊤_X, ⊥_X, ∧, ∨⟩)
 Let L_X = CL(⟨M^A_Y, J^A_Y, ⊤_X, ⊥_X, ∧, ∨⟩), the smallest lattice containing X ⊂ S.

The λ case

We introduce λ , a particular case where $\lambda^x = C\mathcal{L}(\langle \emptyset, \emptyset, x, x, \wedge, \vee, \rangle)$ is a lattice that only contains an element $\{x\}$ and the knowledge of join and meet operator of L.

The REDUCEDCONTEXTCOMPLETION Algorithm 다 나 The reduced context completion 다 나 The function

d'

202

La The reduce context completion function

The reduce context completion funct

Let L be a finite lattice, with a bottom element noted \perp_{\perp} and a top element noted \top_L With CC an application that build a the concept lattice, where $L = CE((M_X^c, J_X^c, \top_X, \bot_X, \wedge, \vee))$, the smallest lattice containing $X \subseteq S$.

ht λ case is introduce λ , a particular case where $\lambda^{\pi} = CL((\emptyset, \emptyset, x, x, \wedge, \vee,))$ is a lattice that by contains an element $\{x\}$ and the knowledge of join and mest operator of L.

La Rochelle Université

Motivation Introduction The reduced context completion Theorem

The algorithm of ReducedContextCompletion

The reduce context completion function

The reduce context completion function

 κ is our reduced context completion function, returning the smallest sub-lattice of L with the element $\{x\}$. Where the function κ^x find J_x^{λ} , M_x^{λ} from the reduced context of L, the minimal number of irreducible from L required to build L_x , the smallest lattice containing the element $\{x\}$; based on \wedge and \vee operator of L. With $X = \{x_1, x_2...x_n\}$ we note :

The function

 $L_{\mathbf{X}} = \mathcal{CL}(\kappa^{x_1}(\kappa^{x_2}(\dots\lambda^{x_n}))) = \mathcal{CL}(\langle M_{\mathbf{X}}^{\lambda}, J_{\mathbf{X}}^{\lambda}, \top_{\mathbf{X}}, \bot_{\mathbf{X}}, \wedge, \vee \rangle)$

The REDUCEDCONTEXTCOMPLETION Algorithm -11 -The reduced context completion -03 \square The function 2022-

The reduce context completion function

 $L_X = C\mathcal{L}(\kappa^{x_1}(\kappa^{x_2}(...\lambda^{x_n}))) = C\mathcal{L}(\langle M_Y^{\perp}, J_Y^{\perp}, \top_X, \bot_X, \wedge, \vee \rangle)$

La Rochelle Université The algorithm of ReducedContextCompletion Conclusion The REDUCEDCONTEXTCOMPLETION Algorithm
The reduced context completion
Example
Example

Let L be a lattice $L = (S, \leq)$. Let compute the reduced context completion with a subset of $S, X = \{b, e, d, c\}$ and the goal is to find L_X such that :

 $L_X = C\mathcal{L}(\langle M_X^{\perp}, J_X^{\perp}, \top_X, \bot_X, \wedge, \vee \rangle) = C\mathcal{L}(\kappa^c(\kappa^d(\kappa^d(\lambda^b))))$

Exemple of a reduced context completion

Example

Let L be a lattice $L = \langle S, \leq \rangle$. Let compute the reduced context completion with a subset of S, $X = \{b, e, d, c\}$ and the goal is to find L_X such that :

 $L_X = \mathcal{CL}(\langle M_X^{\scriptscriptstyle \bot}, J_X^{\scriptscriptstyle \bot}, \top_X, \bot_X, \wedge, \vee \rangle) = \mathcal{CL}(\kappa^c(\kappa^d(\kappa^e(\lambda^b))))$

-03-11

2022-

Motivation The REDUCEDCONTEXTCOMPLETION Algorithm Introduction -03-11 -The reduced context completion The reduced context completion La Rochelle Université Theorem Example The algorithm of ReducedContextCompletion Conclusion 2022-

> The original lattice $L = \langle S, \leq \rangle$, $S = \{a, b, c, d, e, f, g, h, i\}$

Initialisation



Figure 13: Lattice L $< S, \lor, \land >$

-Example

-Initialisation

The original lattice $L = \langle S, \leq \rangle$, $S = \{a, b, c, d, e, f, g, h, i\}$

Figure 13: Lattice L $< S, \lor, \land >$

La Rochelle Université The algorithm of Reduced Context Completion Conclusion

The REDUCEDCONTEXTCOMPLETION Algorithm
The reduced context completion
Example
n=1

We create the first sub-lattice 3^b where

 $\lambda^{b} = C\mathcal{L}((\emptyset, \emptyset, b, b, \wedge, \vee,))$

_**b**____

Figure 14: λ^{b}



We create the first sub-lattice λ^{b} where $\lambda^{b} = C\mathcal{L}(\langle \emptyset, \emptyset, b, b, \wedge, \vee, \rangle)$

Figure 14: λ^b

Problematic Covering relation meet/successor and join/predecessor The function Example

Apply the function of reduced context completion $\kappa^{e}(\lambda^{b})$ on L, resulting in the sub-lattice noted

 $L_1 = \mathcal{CL}(\langle \{(e, b)\}, \{(e, b)\}, b, e, \wedge, \vee \rangle)$



Apply the function of reduced context completion $\kappa^{e}(\lambda^{b})$ on L. resulting in the sub-lattice noted

 $L_1 = CL((\{(e, b\}), \{(e, b)\}, b, e, \land, \lor))$

(**)**

Figure 15: The sub-lattice L₂



La Rochelle Université



Figure 15: The sub-lattice L_2

Problematic Covering relation meet/successor and join/predecessor The function $% \left({{{\left[{{{\rm{T}}_{\rm{T}}} \right]}}} \right)$

Apply the function of reduced context completion $\kappa^d(L_1)$ on L, resulting in the sub-lattice noted $L_2 = C\mathcal{L}(\langle \{(g,d), (g,e)\}, \{(d,b), (e,b)\}, b, g, \land, \lor \rangle)$



Apply the function of reduced context completion $\kappa^{d}(L_{1})$ on L, resulting in the sub-lattice noted L_{2} -

 $CE((\{(x, d), (x, e)\}, \{(d, b), (e, b)\}, b, x, \land, \lor))$

(a) - 1

Figure 16: The sub-lattice L₃



La Rochelle Université



Figure 16: The sub-lattice L_3

Problematic Covering relation meet/successor and join/predecessor The function Example

Apply the function of reduced context completion $\kappa^{c}(L_{2})$ on L, resulting in the sub-lattice $\kappa^{n}(\lambda^{b})$

 $\mathcal{CL}(\langle \{(g,d),(g,e)\},\{(d,b),(c,a),(b,a)\},a,g,\wedge,\vee\rangle)$



Apply the function of nuclead context completion $c^{+}(L_{2})$ on L-neuling in the sublattice $s^{+}(\Lambda^{+})$ model $L_{2} = C\mathcal{L}(((g, d), (g, a)), ((d, b), (c, a), (b, a)), a, g, \wedge, \vee))$

Figure 17: The sub-lattice L₃



La Rochelle Université

n=4

Figure 17: The sub-lattice L_3

noted $L_3 =$





Given a lattice $\mathcal{L} = (S, \wedge, \vee)$, then the algorithm reduced completion computes the minimal sublattice of \mathcal{L} containing a subset $X \subseteq S$.

Theorem

Theorem

Given a lattice $\mathcal{L} = (S, \wedge, \vee)$, then the algorithm reduced completion computes the minimal sublattice of \mathcal{L} containing a subset $X \subseteq S$.



The lattice of sublattices

Let us consider each sublattice of \mathcal{L} given by its set of elements, and the relation between these sublattices by inclusion on these sets of elements. The Moore family of all the sublattices of \mathcal{L} equipped with this inclusion relation, and an emptyset \emptyset forms a lattice. **The lattice of sublattice**.



Figure 18: The lattice of the Moore families of sublattices of \mathcal{L} (Partial)



The leaster of subleasters	(acoustry)
The factore of subfactores	
Let us consider each sublattice of	(intigi) (alben) (about) (desp)
L given by its set of elements, and	
the relation between these	(abox) [bilag] (agh() [sub]
sublattices by inclusion on these	
sets of elements. The Moore family	(m)
of all the sublattices of L equipped	
with this inclusion relation, and an	
emptyset Ø forms a lattice.	5 10 Th 1 10 10 10
The lattice of sublattice.	rigure 10: The lattice of the Moore



The REDUCEDCONTEXTCOMPLETION Algorithm

A lattice L

A lattice L



Theorem

With φ as closure operator. The smallest sublattice of \mathcal{L} containing X is then uniquely defined as $\mathcal{CL}(\varphi(X))$ in this lattice of sublattices.

The REDUCEDCONTEXTCOMPLETION Algorithm
Theorem
Proof
Lattice of sublattices

Lattice of sublattice

Theorem With φ as closure operator. The smallest sublattice of \mathcal{L} containing X is then uniquely defined as $C\mathcal{L}(\varphi(X))$ in this lattice of sublattices.

2022-03-11



Theorem

With φ as closure operator. The smallest sublattice of \mathcal{L} containing X is then uniquely defined as $\mathcal{CL}(\varphi(X))$ in this lattice of sublattices.

Closure operator

A closure operator is a function on a set *S* that satisfy :

- \blacktriangleright $X \subset \varphi(X)$: Extensive
- $\blacktriangleright X \subset Y \to \varphi(X) \subset \varphi(Y) : \text{ Isotone}$
- $\varphi(\varphi(X)) = \varphi(X)$: Idempotent

Closure operator are determined by their **closed set** (Or Moore Families ...). The closure $\varphi(X)$ of a set X is the smallest set containing X.



Lattice of sublattic

Theorem With φ as closure operator. The smallest sublattice of \mathcal{L} containing X is then uniquely defined as $C\mathcal{L}(\varphi(X))$ in this lattice of sublattices.

Closure operator A closure operator is a function on a set S that satisfy : $X \subset [V]$: Extensive $X \subset [V]$: Isotone $Y \subset [V]$: Isotone $Y \subseteq [V]$: Isotone $Y \subseteq [V]$: Isotone $Y \subseteq [V]$: Isotone Closure operator are distermined by their closed set (Or Moree Families...). The closure V[X] of a set X is the smallest set containing X.



Problems definition

Problems definition

The algorithm successively computed a reduced context for x_1 , then x_2, \ldots , then x_n . Let X_i be the set of element of the lattice of the reduced context computed at each iteration, i.e. for $\{x_1, ..., x_i\}$. Then X_n is the set of elements of the reduced context of the lattice computed by the algorithm. Therefore we have to proove that the algorithm computes the minimal set of elements that generate $\mathcal{CL}(\varphi(X))$, i.e. that $X_n = \mathcal{CL}(\varphi(X))$.









Let us assume that we have the reduced context of the smallest context containing $\{x_1, ..., x_{-1}\}$, then $X_{-1} = C\mathcal{L}(x_1^{-}\{x_1, ..., x_{-1}\})\}$. as compares the reduced context of the smallest sublattice containing $X_{i-1} \cup \{x_i\} \subseteq X_i$. Then X_i is the smallest lattice containing $C\mathcal{L}(x_i^{-}\{x_1, ..., x_{-1}\} \cup \{x_i\})$. Therefore $X_i = C\mathcal{L}(x_i^{-}, ..., x_{-1})$ and the remember $X_i = C\mathcal{L}(x_i^{-})$.

Proof

Proof

Let us assume that we have the reduced context of the smallest context containing $\{x_1, ..., x_{i-1}\}$, then $X_{i-1} = C\mathcal{L}(\varphi(\{x_1, ..., x_{i-1}\}))$. κ computes the reduced context of the smallest sublattice containing $X_{i-1} \cup \{x_i\} \subseteq X_i$. Then X_i is the smallest lattice containing $C\mathcal{L}(\varphi(\{x_1, ..., x_{i-1}\} \cup \{x_i\}))$. Therefore $X_i = C\mathcal{L}(\varphi(\{x_1, ..., x_{i-1}\}))$, and by recursion, $X_n = C\mathcal{L}(\varphi(X))$.



Proof Let un assume that we have the reduced context of the smallest context containing $\{x_1, ..., x_{-1}\}$, then $X_{-1} = CC(\varphi(1x_1, ..., x_{-1})))$. a comparise the reduced context of the smallest sublattice containing $X_{-1} \cup \{x_i\} \subseteq$. Then X is the smallest lattice containing $CC(\varphi(1x_1, ..., x_{-1}) \cup \{x_i\})$. Therefore $K_{-1} = CC(\varphi(1x_1, ..., x_{-1}))$ and the surveys $K_{-1} = CC(\varphi(1x_1, ..., x_{-1}))$.

K; is composed of these updated irreducible elements, together with join and me susual from these irreducible elements.

Proof

Proof

Let us assume that we have the reduced context of the smallest context containing $\{x_1, ..., x_{i-1}\}$, then $X_{i-1} = C\mathcal{L}(\varphi(\{x_1, ..., x_{i-1}\}))$. κ computes the reduced context of the smallest sublattice containing $X_{i-1} \cup \{x_i\} \subseteq X_i$. Then X_i is the smallest lattice containing $C\mathcal{L}(\varphi(\{x_1, ..., x_{i-1}\} \cup \{x_i\}))$. Therefore $X_i = C\mathcal{L}(\varphi(\{x_1, ..., x_{i-1}\}))$, and by recursion, $X_n = C\mathcal{L}(\varphi(X))$.

Proof

 X_i is composed of these updated irreducible elements, together with join and meet issued from these irreducible elements.



s a closure operator This computation is extensive since $X \subseteq X_n = CL(\psi(X))$

Moreover, if κ is applied twice, then no element is added $CL(\varphi(\varphi(X))) = CL(\varphi(X))$: idempotent. And this computation add new ineducible elements to generate new elements, then it is isotone: $X \subseteq Y$ then $CL(\varphi(X)) \subseteq CL(\varphi(Y))$.

κ is a closure operator

- ▶ This computation is extensive since $X \subseteq X_n = C\mathcal{L}(\varphi(X))$.
- Moreover, if κ is applied twice, then no element is added CL(φ(φ(X))) = CL(φ(X)) : idempotent.
- And this computation add new irreducible elements to generate new elements, then it is isotone: if X ⊆ Y then CL(φ(X)) ⊆ CL(φ(Y)).



Input of the algorithme

The REDUCEDCONTEXTCOMPLETION Algorithm
The algorithm of ReducedContextCompletion
Input of the algorithme
Input of the algorithme

Input The reduced context of a lattice LAn element $x \in S$

be the reduced context of the sublattice L_{Y} , $L_{Y} \subseteq I$

Input

The reduced context of a lattice L
 An element x ∈ S

Output

▶ the reduced context of the sublattice L_X , $L_X \subseteq L$

-11

-03

2022-

Input of the algorithme Startup check Add new maximum and new minimun Extend maximum or\and minimum The main execution

The algorithm of ReducedContextCompletion Input

- ▶ M^{\land} : Set of couples (m, m^+)
- ► J^{\downarrow} : Set of couples (j^-, j)
- \blacktriangleright \top : The top element of the current lattice
- \blacktriangleright \perp : The bottom element of the current lattice
- \blacktriangleright \land : The join operator of L
- \blacktriangleright \lor : the meet operator of L
- \blacktriangleright X : Subset of S

Three step

La Rochelle

Université

- Startup Check
- Maximum and minimum updating
- Irreducibles updating

The REDUCEDCONTEXTCOMPLETION Algorithm
The algorithm of ReducedContextCompletion
Input of the algorithme

11

0ġ

,

202

40 / 61

The algorithm of ReducedContextCompletion

F	M^{\perp} : Set of couples (m, m^{+})
	J ^L : Set of couples (i ⁻ , i)
÷	T : The top element of the current lattice
÷	⊥ : The bottom element of the current lattice
÷	∧ : The join operator of L
	V : the meet operator of L
•	X : Subset of S
'n	te step
	Startup Check
÷	Maximum and minimum updating

Input of the algorithme **Startup check** Add new maximum and new minimum Extend maximum or\and minimum The main execution

Startup check of ReducedContextCompletion

Startup check

La Rochelle Université

- if (., x) in J^{\wedge} : continue
- ▶ if (x, .) in M^{\land} : continue
- if x is \top : continue
- ▶ if x is \bot : continue

- The REDUCEDCONTEXTCOMPLETION Algorithm
 The algorithm of ReducedContextCompletion
 Startup check
 - Startup check of ReducedContextCompletion

Startup check of ReducedContextCompletion

rtup check	
if $(., x)$ in J^{\perp} : continue	
if $(x, .)$ in M^{\perp} : continue	
if x is T : continue	
if x is ⊥ : continue	

-11

-03

2022-

Input of the algorithme **Startup check** Add new maximum and new minimum Extend maximum or\and minimum The main execution

Startup check of ReducedContextCompletion

Startup check

La Rochelle

Université

- if (., x) in J^{\wedge} : continue
- ▶ if (x, .) in M^{\land} : continue
- ▶ if x is \top : continue
- if x is \perp : continue

Maximum and minimum redefinition

- ▶ if $x > \top$: ADDMAXIMUM(x)
- ▶ if $x < \bot$: ADDMINIMUM(x)
- ▶ if $x \not\leq \top$ nor $\geq \top$: EXTENDMAXIMUM(x)
- ▶ if $x \not\leq \bot$ nor $\not\geq \bot$: EXTENDMINIMUM(x)

- The REDUCEDCONTEXTCOMPLETION Algorithm
 The algorithm of ReducedContextCompletion
 Startup check
 - Startup check of ReducedContextCompletion

	rtup check
٠	if (., x) in J ^A : continue
٠	if (x,.) in M ^A : continue
٠	if x is T : continue
٠	if x is ⊥ : continue

For the set of the second sec

11

0ġ

,



Add new maximum

The REDUCEDCONTEXTCOMPLETION Algorithm The algorithm of ReducedContextCompletion Add new maximum and new minimum Add new maximum

Add new maximum

AddMaximum(x) ► add(T, x) in M^A // add x as successor of T ► add(x, T) in J^A // add T as predecessor of x ► T ← x // x become the new top element continue

AddMaximum(x)

- ▶ $add(\top, x)$ in M^{\star} // add x as successor of \top
- ▶ $add(x, \top)$ in J^{\land} // add \top as predecessor of x
- ▶ $\top \leftarrow x / / x$ become the new top element

continue

-11

2022-03-



Add new minimum

The REDUCEDCONTEXTCOMPLETION Algorithm The algorithm of ReducedContextCompletion Add new maximum and new minimum Add new minimum

addMinimum(x)

add(⊥, x) in J^k // add x as predecessor of ⊥
 add(x, ⊥) in M^k // add ⊥ as successor of x
 ⊥ ← x // x become the bottom element

addMinimum(x)

- ▶ $add(\bot, x)$ in J^{\downarrow} // add x as predecessor of \bot
- ▶ $add(x, \bot)$ in M^{\land} // add \bot as successor of x
- ▶ $\bot \leftarrow x / / x$ become the bottom element

continue

-11

2022-03-

Input of the algorithme Startup check Add new maximum and new minimum **Extend maximum or\and minimum** The main execution

Extend a new maximum or new minimum

ExtendMinimum(x)

La Rochelle Université

- add((⊥ ∧ x), ⊥) in J[⊥] // Add the bottom element of κ^x into the set of join irreducibles
- ► $\bot \leftarrow (\bot \land x) / /$ Update bottom element

ExtendMaximum(x)

- add(⊤, (⊤∨x)) in M[↓] // Add the top element of x into the set of meet irreducibles
- ► $\top \leftarrow (\top \lor x) / / \text{Update top element}$

- The REDUCEDCONTEXTCOMPLETION Algorithm
 The algorithm of ReducedContextCompletion
 Extend maximum or\and minimum
 - Extend a new maximum or new minimum

Extend a new maximum or new minimum

 add((⊥ ∧ x), ⊥) in J[⊥] // Add the bottom element of κ[×] into the set of join irreducibles ⊥ ← (⊥ ∧ x) // Update bottom element
TextendMaximum(v)

add(T,(T ∨ x)) is M^k // Add the top element of x into the set of meet irreducibles
 T ← (T ∨ x) // Update top element

-11

-03

d'

Input of the algorithme Startup check Add new maximum and new minimum Extend maximum or\and minimum The main execution

Classical case : the main execution

Setting immediate successor and immediate predecessor of x

At this step, x is between \top and \bot . We initialize \bot as potential immediate predecessor of $x \in J^{\lambda}$, and top as potential immediate successor of $x \in M^{\lambda}$, and then we maintain the irreducible.

▶ add(\bot, x) in J^{\land}

La Rochelle

Université

▶ $add(x, \top)$ in M^{\land}

The REDUCEDCONTEXTCOMPLETION Algorithm
The algorithm of ReducedContextCompletion
The main execution

Classical case : the main execution

Classical case : the main execution

Setting immediate successor and immediate predecessor of xat this step, x is between T and L. We initiate L are potential immediate predecessor of $x \in M^+$, and top as potential immediate successor of $x \in M^+$, and then we maintain the irreducible. • add(L, x) in M^+

-11

0ġ

,

Input of the algorithme Startup check Add new maximum and new minimum Extend maximum or\and minimum The main execution

Classical case : the main execution

Setting immediate successor and immediate predecessor of x

At this step, x is between \top and \bot . We initialize \bot as potential immediate predecessor of $x \in J^{\lambda}$, and top as potential immediate successor of $x \in M^{\lambda}$, and then we maintain the irreducible.

▶ $add(\bot, x)$ in J^{\land}

La Rochelle

Université

▶ add (x, \top) in M^{\land}

Entering into the main execution

- ▶ INSERTJOINIRREDUCIBLE(*x*)
- ► INSERTMEETIRREDUCIBLE(*x*)

The REDUCEDCONTEXTCOMPLETION Algorithm
The algorithm of ReducedContextCompletion
The main execution
Classical case : the main execution

Classical case : the main execution

Setting immediate successor and immediate predecessor of x		
At this step, x is between T and \bot . We initialize \bot as potential immediate predecessor of $x \in J^{A}$, and then we maintain the irreducible. \mathbb{P} add (L, x) in J^{A} \mathbb{P} add (L, T) in M^{A}		
Entering into the main execution		
INSERTJOINTREEDUCILE(x) INSERT/INTEREDUCILE(x)		

-11

0ġ

,

Input of the algorithme Startup check Add new maximum and new minimum Extend maximum or\and minimum The main execution

Maintening the join and meet operator

La Rochelle

Université

Maintening the join and meet operator

Currently, we are working on finite sub-lattice and therefore we maintain the join and meet operator. In case of a join-semi-lattice, we can only maintain the join operator (And only focus on join irreducible) and the same is also true for a meet-semi-lattice, where we only seeks for the meet irreducible. Future works will address this question directly, but for now, we are looking for both the join and meet irreducible.



Figure 20: A lattice L

The REDUCEDCONTEXTCOMPLETION Algorithm
The algorithm of ReducedContextCompletion
The main execution

Maintening the join and meet operator

Maintening the join and meet operat

Maintening the join and meet operator Currently, was envolving on finite sub-lattice and therefore we maintain the join and meet operator. In case of a join-smultatice, we can only maintain the join operator (And only focus on join irreductible) and the same is also true for a meet enreductible. Where we only seeks for the meet irreductible. Future works will address this question directly, but for now, we are looking for both the join and meet irreductible.

Firms 20: A lattice I

-11

0ġ

,

Input of the algorithme Startup check Add new maximum and new minimum Extend maximum or\and minimum The main execution

Maintening the join and meet operator

La Rochelle

Université

Maintening the join and meet operator

Currently, we are working on finite sub-lattice and therefore we maintain the join and meet operator. In case of a join-semi-la-ttice, we can only maintain the join operator (And only focus on join irreducible) and the same is also true for a meet-semi-lattice, where we only seeks for the meet irreducible. Future works will address this question directly, but for now, we are looking for both the join and meet irreducible.



Figure 21: A join-semi-lattice from L

- The REDUCEDCONTEXTCOMPLETION Algorithm
 - └─Maintening the join and meet operator





Input of the algorithme Startup check Add new maximum and new minimum Extend maximum or\and minimum The main execution

Maintening the join and meet operator

La Rochelle

Université

Maintening the join and meet operator

Currently, we are working on finite sub-lattice and therefore we maintain the join and meet operator. In case of a join-semi-lattice, we can only maintain the join operator (And only focus on join irreducible) and the same is also true for a meet-semi-lattice, where we only seeks for the meet irreducible. Future works will address this question directly, but for now, we are looking for both the join and meet irreducible.



Figure 22: A meet-semi-lattice from L

The REDUCEDCONTEXTCOMPLETION Algorithm

└─Maintening the join and meet operator

Maintening the join and meet operat





insertJoinIrreducible(x)

end

The REDUCEDCONTEXTCOMPLETION Algorithm The algorithm of ReducedContextCompletion The main execution insertJoinIrreducible(x)

insertJoinIrreducible(>

insertJoinIrreducible(x)	insert.loinIrreducible(x)
for (j^-, j) in J^+ do $r = j \wedge x$ $r^- = (j^- \wedge x) \vee (j \wedge x^-)$ add (r, r) in J^+ $x \neg V = r$ $j \neg V = r$ end	$ \begin{array}{l} \mbox{for } (j^-, j) \mbox{ if } j^- = -j \mbox{ then } \\ & \mbox{if } j^- = -j \mbox{ then } \\ & \mbox{ remove}(j^-, j) \mbox{ from } J^{\perp} \\ & \mbox{ end } \end{array} $

insert JoinIrreducible(x) for (j^-, j) in J^{\wedge} do $r = j \wedge x$ $r^- = (j^- \wedge x) \lor (j \wedge x^-)$ add(r-, r) in J^{\wedge} $x^- \lor = r$ $j^- \lor = r$

insertJoinIrreducible(x) for (j^-, j) in J^{\wedge} do | if $j^- == j$ then | remove (j^-, j) from J^{\wedge} end end



 $r = j \land x$ $x^- \land = r$

i = r



iner/Joinfrackibi(+)



The REDUCEDCONTEXTCOMPLETION Algorithm The algorithm of ReducedContextCompletion The main execution insertJoinIrreducible(x)





insertJoinIrreducible(x)



$(j^- \wedge x) \lor (j \wedge x^-)$

The REDUCEDCONTEXTCOMPLETION Algorithm
The algorithm of ReducedContextCompletion
The main execution

 \Box insertJoinIrreducible(x)



2022-03-11



insertMeetIrreducible(x)

insertMeetIrreducible(x)insertMeetIrreducible(x)for (m, m^+) in M^{\wedge} do $r = m \lor x$ $r^+ = (m^+ \lor x) \land (m \lor x^+)$ add(r, r+) in M^{\downarrow} $M^{\scriptscriptstyle \wedge}$ $x^+ \lor = r$ end $m^+ \lor = r$ end end

for (m, m^+) in M^{\wedge} do if $m^+ == m$ then *remove*(m, m+) from

The REDUCEDCONTEXTCOMPLETION Algorithm -The algorithm of ReducedContextCompletion └─The main execution \Box insertMeetIrreducible(x)

insertMeetIrreducible(x)	insertMeetIrreducible(x)
for (m, m^+) in M^+ do $r = m \lor x$ $r^+ = (m^+ \lor x) \land (m \lor x^+)$ add (r, r_+) in M^+ $x^+ \lor = r$ end	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$

-11

-03

2022-



 $r = m \lor x$ $x^+ \lor = r$

 $\blacktriangleright m^+ \lor = r$

Figure 22



artHornbrokeke(-)

 $m^+ \vee x$



 $m \vee x^+$

Figure 20



 $r^+ = (m^+ \lor x) \land (m \lor x^+)$

Figure 30



Theorem

The REDUCEDCONTEXTCOMPLETION Algorithm The algorithm of ReducedContextCompletion The main execution Theorem

 $\label{eq:corresponse} \frac{\mathsf{Theorem}}{\mathsf{At} \text{ the end of the algorithm } (J^{\lambda}, M^{\lambda}, \top, \bot, \lor, \land) \text{ is the reduced context of } L_X.$

Theorem

At the end of the algorithm $\langle J^{\lambda}, M^{\lambda}, \top, \bot, \vee, \wedge \rangle$ is the reduced context of L_X .

-11

-03

2022-









Figure 31: Deep learning



Figure 32: FCA

The Galactic Organization <contact@thegalactic.org> The ReducedContextCompletion Algorithm



-Conclusion



Future works

The reduced context completion, is a first step in data navigation, but not only. With this algorithm, we are able to only work on reduced contexts of lattices, drastically reducing the number of elements to take into account.

