TECHNISCHE UNIVERSITÄT DRESDEN

Faculty of Computer Science • Institute of Theoretical Computer Science • Chair of Automata Theory

How KR benefits from Formal Concept Analysis

Session 2/2

Francesco Kriegel*, Barış Sertkaya (Frankfurt UAS)

*Funded by DFG in Project 430150274 (Repairing Description Logic Ontologies) and partially in Project 389792660 (TRR 248: Foundations of Perspicuous Software Systems).

ReDLO Repairing Description Logic Ontologies

CPEC CENTER FOR PERSPICUOUS COMPUTING

20th International Conference on Principles of Knowledge Representation and Reasoning (KR 2023), 2–8 September 2023

Tutorial Outline

Session 1:

1. Conceptual clustering with FCA (Francesco)
2. Extracting dependencies with FCA (Barış)

Session 2:

3. Acquiring complete knowledge about an application domain, enriching OWL ontologies (Barış)
4. Mining axioms from interpretations and knowledge graphs (Francesco)
5. Computing Stable Extensions of Argumentation Frameworks using FCA (Barış)

Axiomatization of Concept Inclusions

Axiomatization of Concept Inclusions

DL can be seen as an extension of FCA with role names, where attributes and concept names coincide. In particular, concept inclusions (CIs) $C \sqsubseteq D$ and implications $P \to Q$ are similar.

- Given a formal context $\mathbb{K} := (G, M, I)$, we consider the interpretation $\mathcal{I}$ with domain G, signature M (only concept names), and where $m^{\mathcal{I}} := \{ g \mid (g, m) \in I \}$ for each $m \in M$.
- Then $\mathbb{K} \models P \to Q$ iff $\mathcal{I} \models \bigsqcap P \sqsubseteq \bigsqcap Q$, where $\bigsqcap\{m_1, \ldots, m_\ell\} := m_1 \sqcap \cdots \sqcap m_\ell$ and $\bigsqcap\emptyset := \top$.

Recall from Session 1 that we can compute implication bases for formal contexts.

Can we also compute concept inclusion bases for interpretations?

Definition. A concept inclusion base (CI base) for an interpretation $\mathcal{I}$ is a TBox $\mathcal{B}$ that is

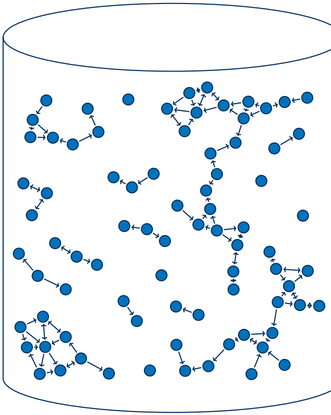
- sound: all CIs in $\mathcal{B}$ hold in $\mathcal{I}$,
- complete: $\mathcal{B}$ entails all CIs that hold in $\mathcal{I}$.

Contrary to FCA, existence of CI bases is not obvious since infinitely many concept descriptions C, D can be constructed from the signature.

A Closer Look on the Input

Resembling the FCA definition, we have defined CI bases for interpretations.

- Datasets in form of labelled graphs (with node labels and edge labels) can be used as interpretations, e.g., sets of RDF triples.
- ABoxes could also be treated as interpretations, but thereby switching from open-world assumption to closed-world assumption.
- Without any closure assumption on the input data no CIs except tautologies could be axiomatized (as there could exist a still unknown counterexample to any CI).



No Overfitting

Recall that a CI base axiomatizes the given interpretation in a complete manner.

If the axiomatization approach should be employed for "ontology learning" such that the computed CI base is suitable for real-world applications, then

1. **overfitting must be avoided**: otherwise the input dataset could be simply be rewritten into CIs.
2. **abstraction is necessary**: in order to understand a concept, it is often better to find the commonalities of all objects in this concept instead of just memorizing them and their descriptions.

We therefore impose the following restrictions on the considered DL.

- No nominals.
- No disjunction.
- No negation.

Computation of CI Bases

The problem of computing CI bases can be reduced to computing implication bases.

Definition. Given an interpretation $\mathcal{I}$ and a set M of concept descriptions, the induced context is defined as $\mathbb{K}_{\mathcal{I}} := (G, M, I)$ where G is the domain of $\mathcal{I}$ and $(x, C) \in I$ iff $x \in C^{\mathcal{I}}$.

Lemma. For all subsets $P, Q \subseteq M$, the CI $\bigsqcap P \sqsubseteq \bigsqcap Q$ holds in $\mathcal{I}$ iff the implication $P \to Q$ holds in $\mathbb{K}_{\mathcal{I}}$.

Thus, if $\mathcal{B}$ is an implication base for $\mathbb{K}_{\mathcal{I}}$, then $\{ \bigsqcap P \sqsubseteq \bigsqcap Q \mid P \to Q \in \mathcal{B} \}$ is sound for $\mathcal{I}$, but completeness depends on the choice of the attribute set M.

Sebastian Rudolph: **Exploring Relational Structures via $\mathcal{FLE}$.** ICCS 2004.

Computation of CI Bases

1. For $\mathcal{EL}$ and its extension with greatest fixed-point semantics, such an attribute set M exists that guarantees completeness. Moreover, the CI base obtained from the canonical implication base for $\mathbb{K}_\mathcal{I}$ is minimal.
2. An attribute set M that yields a complete CI base also exists for $\mathcal{EL}$ extended with value restrictions, number restrictions, negations of concept names, and existential self-restrictions.
3. Also for the Horn fragment of the latter DL a complete set M exists.

In all above cases, the complexity is not higher than for computing implication bases in FCA: a CI base can be computed in exponential time, and there exist interpretations that have no CI base of polynomial size.

Franz Baader, Felix Distel: **A Finite Basis for the Set of $\mathcal{EL}$-Implications Holding in a Finite Model.** ICFCA 2008.
Daniel Borchmann, Felix Distel, Francesco Kriegel: **Axiomatisation of general concept inclusions from finite interpretations.** J. Appl. Non Class. Logics 26.1, 2016.
Francesco Kriegel: **Joining Implications in Formal Contexts and Inductive Learning in a Horn Description Logic.** ICFCA 2019.

Computation of CI Bases

FCA cannot look into the concept descriptions in M, but can only conjoin them by conjunction.

- For $\mathcal{EL}$, it thus suffices that M consists of concept names and existential restrictions.
- To make it even unnecessary to look into the concept descriptions in M, we require that C is a "closure" of $\mathcal{I}$ for each $\exists r.C \in M$: if the CI $C \sqsubseteq D$ holds in $\mathcal{I}$, then C is subsumed by D.

  Otherwise, FCA could generate implications $\{A\} \to \{B\}$ and $\{\exists r.A\} \to \{\exists r.B\}$ for concept names A, B, but does not recognize that the second follows from the first with DL semantics.

All "closures" are model-based most specific concept descriptions.

Definition. Given a subset X of the domain of $\mathcal{I}$, its model-based most specific concept description (MMSC) $X^{\mathcal{I}}$ is defined by:

1. $X \subseteq (X^{\mathcal{I}})^{\mathcal{I}}$
2. For each concept description C, if $X \subseteq C^{\mathcal{I}}$, then $X^{\mathcal{I}}$ is subsumed by C.

Computation of CI Bases

In $\mathcal{EL}$, the MMSC $X^{\mathcal{I}}$ describes the commonalities of all objects in X.

With these MMSCs, we can define the attribute set M consisting of

- the bottom concept $\bot$,
- all concept names,
- and all existential restrictions of the form $\exists r.X^{\mathcal{I}}$.

To avoid the computation of tautologies, we further define the background implication set $\mathcal{L}$ consisting of all $\{C\} \to \{D\}$ where $C, D \in M$ and C is subsumed by D.

Theorem. If $\mathcal{B}$ is the canonical implication base of the induced context $\mathbb{K}_{\mathcal{I}}$ (with the above attribute set M) w.r.t. the background implications in $\mathcal{L}$, then the TBox

$$\{\bigsqcap P \sqsubseteq \bigsqcap Q \mid P \to Q \in \mathcal{B}\}$$

is a minimal CI base for $\mathcal{I}$ and it can be computed in exponential time.

Extensions and Variations of the Axiomatization Approach

Axiomatization of Confident Concept Inclusions

In addition to the CIs that hold in the interpretation $\mathcal{I}$, we might also be interested in the CIs that are confident in the following sense.

Definition. The confidence of $C \sqsubseteq D$ in $\mathcal{I}$ is defined as $\mathrm{conf}(C \sqsubseteq D) := |(C \sqcap D)^{\mathcal{I}}| / |C^{\mathcal{I}}|$.

Given a bound p with $0 < p < 1$, we say that $C \sqsubseteq D$ is confident of its confidence is at least p.

To obtain a CI base that is complete also for all confident CIs, it suffices to add to the canonical CI base all confident CIs of the form $X^{\mathcal{I}} \sqsubseteq Y^{\mathcal{I}}$ where X, Y are subsets of the domain of $\mathcal{I}$.

Theorem. If $\mathcal{B}$ is a CI base for $\mathcal{I}$, then $\mathcal{B} \cup \{\, X^{\mathcal{I}} \sqsubseteq Y^{\mathcal{I}} \mid \mathrm{conf}(X^{\mathcal{I}} \sqsubseteq Y^{\mathcal{I}}) \geq p \,\}$ is a confident CI base for $\mathcal{I}$ and p.

Daniel Borchmann: **Towards an Error-Tolerant Construction of $\mathcal{EL}^{\perp}$-Ontologies from Data Using Formal Concept Analysis.** ICFCA 2013.

Axiomatization with an Existing TBox

Recall that each interpretation $\mathcal{I}$ induces the closure operator $\varphi_{\mathcal{I}} \colon C \mapsto C^{\mathcal{I}\mathcal{I}}$.

Definition. Given a concept description C and a TBox $\mathcal{T}$, the most specific consequence $C^{\mathcal{T}}$ is defined by:

1. $\mathcal{T}$ entails $C \sqsubseteq C^{\mathcal{T}}$.
2. For each concept description D, if $\mathcal{T}$ entails $C \sqsubseteq D$, then $\mathcal{T}$ entails $C^{\mathcal{T}} \sqsubseteq D$.

With this, also each TBox $\mathcal{T}$ induces a closure operator $\varphi_{\mathcal{T}} \colon C \mapsto C^{\mathcal{T}}$.

Francesco Kriegel: **Constructing and Extending Description Logic Ontologies using Methods of Formal Concept Analysis.** Doctoral Thesis, 2019.

Axiomatization with an Existing TBox

Example: Axiomatization

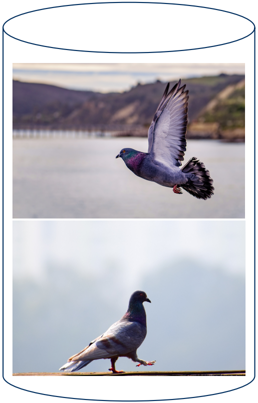
$$\mathsf{Eagle} \sqsubseteq \mathsf{Bird}$$

$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$$

$$\vdots$$

Axiomatization with an Existing TBox

Example: Completion

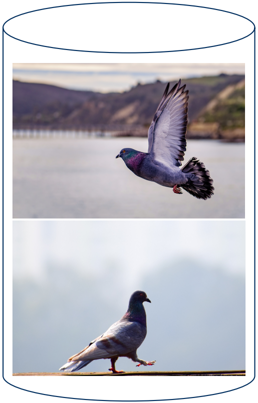


$$\mathsf{Eagle} \sqsubseteq \mathsf{Bird}$$

$$\begin{aligned}\mathsf{Bird} \sqsubseteq\ & \exists \mathsf{hasBodyPart}.\mathsf{Wings} \\ & \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}\end{aligned}$$

Axiomatization with an Existing TBox

Example: Completion



$$\mathsf{Eagle} \sqsubseteq \mathsf{Bird}$$

$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$$

$$\mathsf{Pigeon} \sqsubseteq \mathsf{Bird}$$

$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasAbility}.\mathsf{Walking}$$

Axiomatization with an Existing TBox

Closure operators can be ordered: $\varphi \trianglelefteq \psi$ iff $\mathsf{Closures}(\varphi) \supseteq \mathsf{Closures}(\psi)$. With that, we have:

- $\varphi_{\mathcal{I}} \trianglerighteq \varphi_{\mathcal{T}}$ iff $\mathcal{I} \models \mathcal{T}$
- $\varphi_{\mathcal{I}} \trianglelefteq \varphi_{\mathcal{T}}$ iff $\mathcal{T}$ is complete for $\mathcal{I}$
- $\varphi_{\mathcal{S}} \trianglerighteq \varphi_{\mathcal{T}}$ iff $\mathcal{S} \models \mathcal{T}$

Moreover, as operations on closure operators we have the infimum $\vartriangle$ and the supremum $\triangledown$:

- Infimum: $C^{\varphi \vartriangle \psi} = C^{\varphi} \vee C^{\psi}$
- Supremum: $C^{\varphi \triangledown \psi}$ is the fixed point of the sequence $C,\ C^{\varphi},\ (C^{\varphi})^{\psi},\ ((C^{\varphi})^{\psi})^{\varphi},\ \ldots$

Both are certain types of intersections:

- $\mathsf{Theory}(\varphi \vartriangle \psi) = \mathsf{Theory}(\varphi) \cap \mathsf{Theory}(\psi)$
- $\mathsf{Closures}(\varphi \triangledown \psi) = \mathsf{Closures}(\varphi) \cap \mathsf{Closures}(\psi)$

Francesco Kriegel: **Constructing and Extending Description Logic Ontologies using Methods of Formal Concept Analysis.** Doctoral Thesis, 2019.

Axiomatization with an Existing TBox

Example: Adjusting then Completing

$\mathsf{Eagle} \sqsubseteq \mathsf{Bird}$ $\mathsf{Pigeon} \sqsubseteq \mathsf{Bird}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasAbility}.\mathsf{Walking}$

Axiomatization with an Existing TBox

Example: Adjusting then Completing

$$\mathsf{Eagle} \sqsubseteq \mathsf{Bird} \qquad \mathsf{Pigeon} \sqsubseteq \mathsf{Bird}$$
$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$$
$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasAbility}.\mathsf{Walking}$$

$$\mathsf{Eagle} \sqsubseteq \mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$$
$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings}$$
$$\mathsf{Pigeon} \sqsubseteq \mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$$
$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasAbility}.\mathsf{Walking}$$

Axiomatization with an Existing TBox

Example: Adjusting then Completing

$\mathsf{Eagle} \sqsubseteq \mathsf{Bird} \qquad \mathsf{Pigeon} \sqsubseteq \mathsf{Bird}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasAbility}.\mathsf{Walking}$

$\downarrow$

$\mathsf{Eagle} \sqsubseteq \mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Wings}$

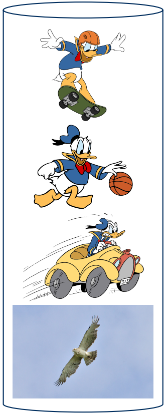
$\mathsf{Pigeon} \sqsubseteq \mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{Flying}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasAbility}.\mathsf{Walking}$

$\mathsf{Penguin} \sqsubseteq \mathsf{Bird}$

Axiomatization with an Existing TBox

Example: Filtering then Completing

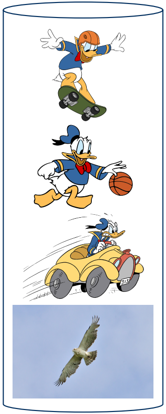


$\mathsf{Duck} \sqsubseteq \mathsf{Bird}$

$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasBodyPart}.\mathsf{Wings}$

Axiomatization with an Existing TBox

Example: Filtering then Completing

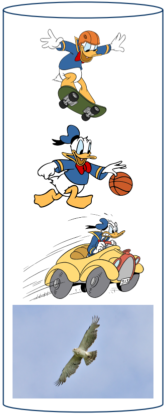


$$\mathsf{Duck} \sqsubseteq \mathsf{Bird}$$

$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasBodyPart}.\mathsf{Wings}$$

Axiomatization with an Existing TBox

Example: Filtering then Completing



$$\mathsf{Duck} \sqsubseteq \mathsf{Bird}$$

$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasBodyPart}.\mathsf{Feet} \sqcap \exists \mathsf{hasBodyPart}.\mathsf{Wings}$$

$$\mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{Skateboarding} \sqsubseteq \bot$$

$$\mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{PlayingBasketball} \sqsubseteq \bot$$

$$\mathsf{Bird} \sqcap \exists \mathsf{hasAbility}.\mathsf{DrivingCars} \sqsubseteq \bot$$

$$\mathsf{Eagle} \sqsubseteq \mathsf{Bird}$$

$$\mathsf{Bird} \sqsubseteq \exists \mathsf{hasAbility}.\mathsf{Flying}$$

Do you have questions or comments?