



A Logical Generalization of Formal Concept Analysis

Sébastien Ferré and Olivier Ridoux – Irisa/{CNRS,IFSIC}

Darmstadt, August 18th – ICCS 2000

Introduction

Concept Analysis has a wide and increasing range of applications

- data analysis
ex: program understanding [Snelting et al.]
- knowledge extraction (learning, datamining)
ex: learning from positive and negative examples [Kuznetsov]
- information systems, information retrieval
ex: software component retrieval [Lindig]

⇒ we are especially interested in information systems

Information Systems (IS) with FCA

[Godin, Missaoui, April] **boolean** and **hierarchical** IS are unsatisfactory for organizing and retrieving information

- no free combination of **querying** and **navigation**
- flat or tedious **organization**
- no consistency with **updates**

⇒ **FCA** is an interesting alternative

- *concept*: unifies **query** (*intent*) and **directory** (*extent*)
- *lattice*: automatic and consistent **organization**

Need for extending FCA ...

... because of the wide range of applications

- data analysis

ex: attributes \rightarrow cubes [Chaudron and Maille]

- learning

ex: attributes \rightarrow conceptual graphs [Kuznetsov]

- information system

We wish to describe objects and ask queries in an arbitrary logical language (in place of mere attributes)

\Rightarrow Logical Information System based on Logical Concept Analysis

Towards Logical Concept Analysis (LCA)

FCA

- object description/query = set of attributes
- subsumption: \supseteq
- operations: \cap, \cup

LCA

- object description/query = **formula** of a language \mathcal{L} (infinite)
- subsumption: $\dot{\models}$ (**logical deduction**)
- operations: $\dot{\vee}, \dot{\wedge}$ (**disjunction** and **conjunction**)
- $\langle \mathcal{L}; \dot{\models}, \dot{\vee}, \dot{\wedge} \rangle$ *has to be* a **lattice**

→ LCA can be seen as an abstraction of FCA
because $\langle 2^{\mathcal{A}}; \supseteq, \cap, \cup \rangle$ forms a lattice

LCA: logical context

- logical context

\mathcal{O} : set of objects

$\langle \mathcal{L}; \models, \dot{\vee}, \dot{\wedge} \rangle$: logical language

$i \in \mathcal{O} \rightarrow \mathcal{L}$: maps a description to each object

- Galois connection

- the most accurate formula satisfied by a set of objects

$$\sigma(O) \doteq \dot{\bigvee}_{o \in O} i(o), \forall O \subseteq \mathcal{O}$$

- all objects satisfying a formula (its extent)

$$\tau(f) = \{o \in \mathcal{O} \mid i(o) \dot{\models} f\}, \forall f \in \mathcal{F}$$

LCA: a context example

object : **function**

formula : **types** with unknowns

subsumption : type **instanciation**

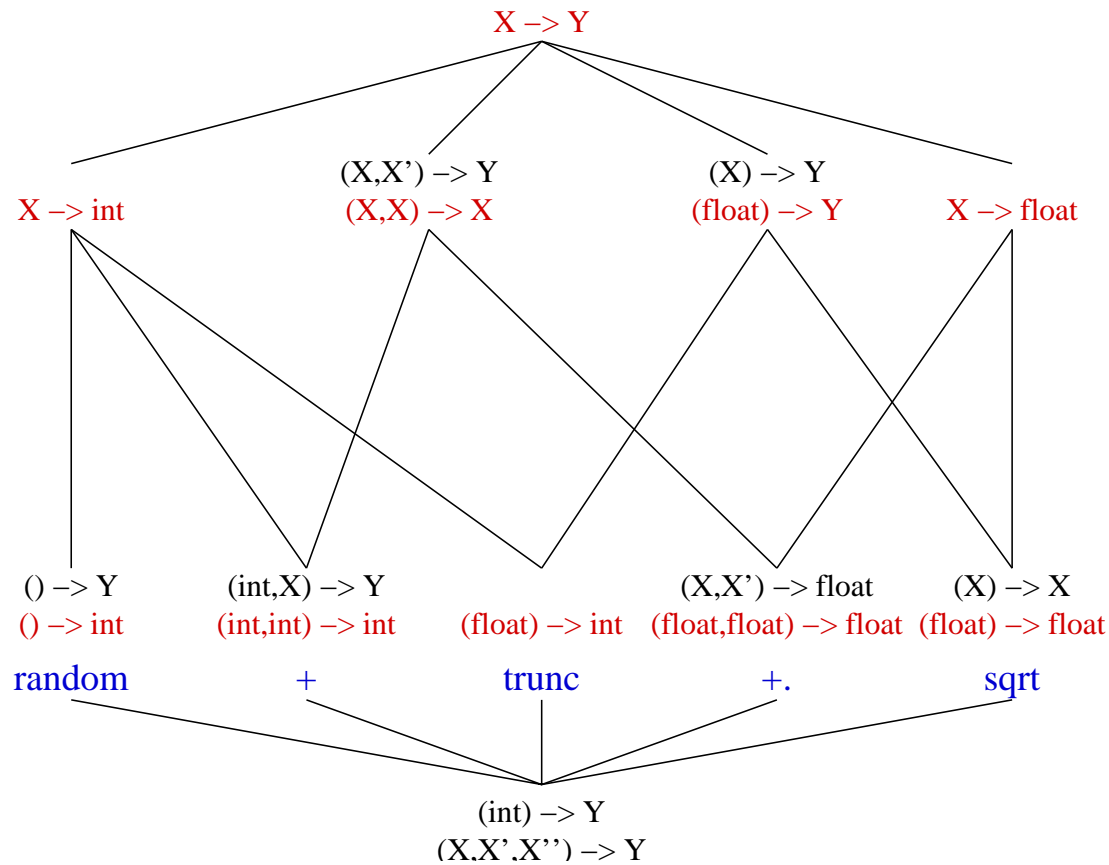
disjunction, conjunction : type **anti-unification** and **unification**

function	description
+	(int,int) -> int
+ .	(float,float) -> float
random	() -> int
trunc	(float) -> int
sqrt	(float) -> float

LCA: concept lattice

- **concept** : pair (**extent,intent**) = (O, f)
with $\sigma(O) = f$ and $\tau(f) = O$
- specialization/generalization ordering
 $(O_1, f_1) \leq (O_2, f_2) : \iff O_1 \subseteq O_2 \iff f_1 \dot{\models} f_2$
 $\longrightarrow \langle \mathcal{C}; \leq \rangle$ is a **complete lattice**
- labelling
 - by objects: $\gamma(o) = (-, \sigma(\{o\}))$
 - by formulas: $\mu(f) = (\tau(f), -)$

LCA: concept lattice example



- $() \rightarrow Y \implies X \rightarrow int$
- $(X) \rightarrow Y \iff (float) \rightarrow Y$
- $(X,X') \rightarrow Y \iff (X,X) \rightarrow X$
- $(int,X) \rightarrow Y \iff (int,int) \rightarrow int$
- $(X,X') \rightarrow float \iff (float,float) \rightarrow float$
- $(X) \rightarrow X \iff (float) \rightarrow float$

concept intents
objects

LCA: contextualized logic (1/2)

- $f \dot{\vDash} g$: **logical** deduction (true in every context)
- $f \dot{\vDash}^K g$: **contextualized** deduction (true in the context K)

definition: $f \dot{\vDash}^K g : \iff \tau(f) \subseteq \tau(g) \iff \mu(f) \leq \mu(g)$

ex:
 $(\) \rightarrow Y \dot{\vDash}^K X \rightarrow \text{int}$
 $(X, X') \rightarrow Y \dot{\vDash}^K (X, X) \rightarrow X$

- $f \dot{\vDash} g \implies f \dot{\vDash}^K g$: the contextualized deduction *extends* the logical one

Important theorem: $\langle \mathcal{L} / \dot{=}^K ; \dot{\vDash}^K \rangle$ is isomorphic to $\langle \mathcal{C} ; \leq \rangle$ by

the labelling mapping μ

\implies it gathers all facts about logic \mathcal{L} and a context K

in a full logical way

LCA: contextualized logic (2/2)

Applications

- **data analysis:** $f \stackrel{\cdot}{\models}^K g \iff \mu(f) \leq \mu(g)$
 → choice between $\langle \text{extents}; \subseteq \rangle$, $\langle \text{intents}; \stackrel{\cdot}{\models} \rangle$ and $\langle \text{formulas}; \stackrel{\cdot}{\models}^K \rangle$ for representing concepts
 - **datamining/learning:** the rule $f \rightarrow g$ stands iff $f \stackrel{\cdot}{\models}^K g$
 → generalizes *implications* between attributes
 - **information systems:** x is a navigation link from f to g iff $f \wedge x \stackrel{\cdot}{\models}^K g$ (see paper for more details)
 → objects are accessed through *several concise paths* (completion)
- ⇒ the **contextualized logic** enables to take into account properties that are true in a context while false in general

Conclusion

Prototype of a Logical File System based on LCA

- bibtex references described by authors, title, year, keywords, ...
% cd author:Wille/keywords:"concept analysis" &
- "conceptual graph"/year:1990..1999 (querying)
% ls -v /author:* (navigating)
% mv -r . location:Darmstadt (updating)

Future work for Logical Information Systems

- [datamining/learning](#) by navigating rather than an exhaustive production of rules
- [terminology, taxonomy](#) (what is true in every context of a domain)
- [composite](#) objects and [relationships](#) between objects