Rule-based resoning over conceptual knowledge

Pascal Hitzler

Artificial Intelligence Institute, Dresden University, Germany

Contents

- A motivation: Semantic Web
- A prototype of a generic reasoning system

... work in progress ...

Motivation

Semantic Web Challenge: making the internet machine-usable.

Conceptual knowledge extracted from/provided with web pages.
$\leadsto$ Ontology engineering/text-/data mining etc.

A current effort:
How to do **rule-based reasoning over conceptual knowledge**.

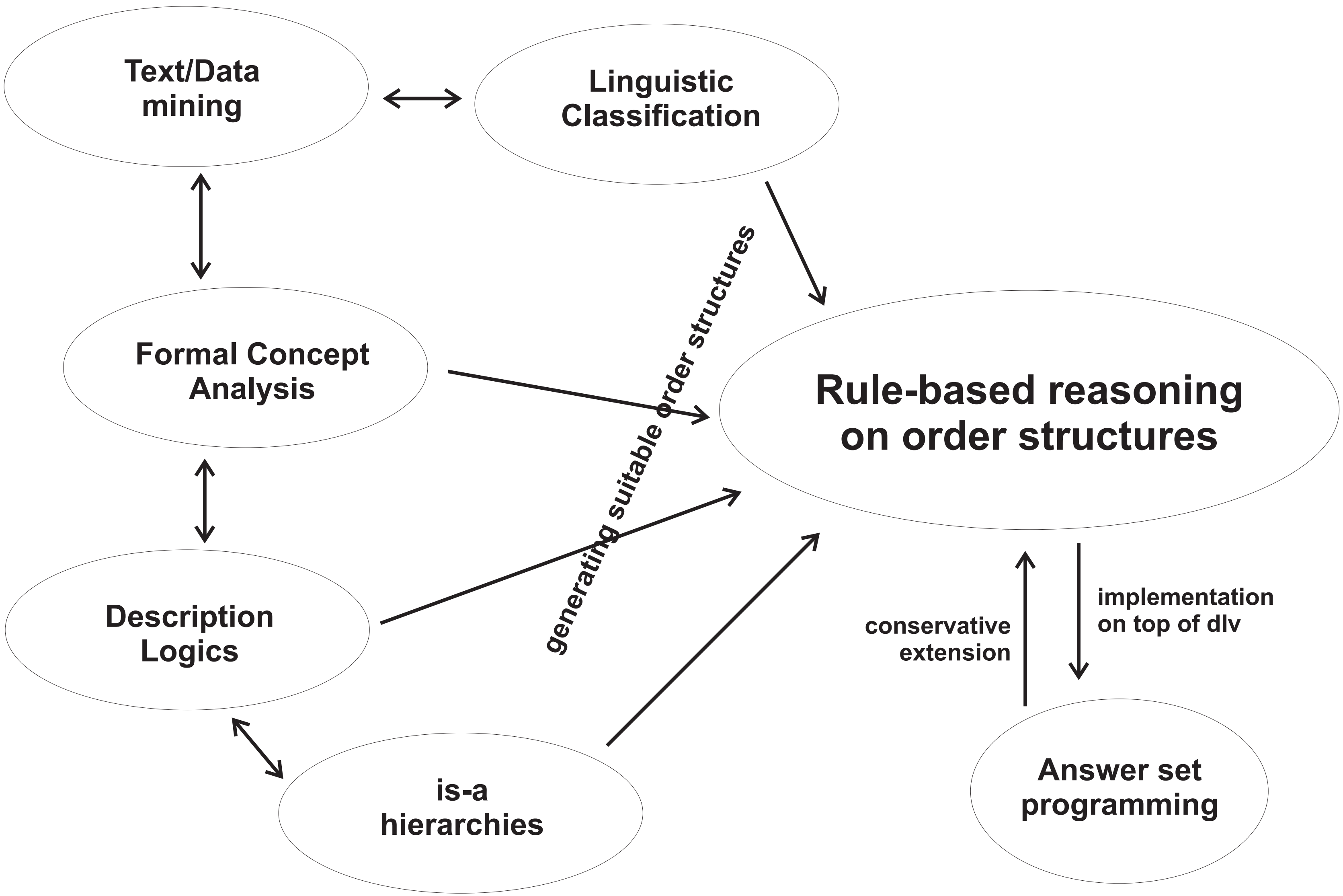
Towards a reasoning system

Seek:

- Generic
- rule-based
- non-monotonic reasoning system
- covering different kinds of conceptual structures,
- in the tradition of answer set programming (ASP).

Conceptual knowledge is **hierarchical**

$\rightsquigarrow$ Reasoning on **order structures**.


Text/Data mining
Linguistic Classification
Formal Concept Analysis
Rule-based reasoning on order structures
generating suitable order structures
Description Logics
conservative extension
implementation on top of dlv
is-a hierarchies
Answer set programming

Logic RZ on order structures

(Rounds & Zhang IC 2001)

$(D, \sqsubseteq)$: coherent algebraic cpo (e.g. finite)

Clause: $X \subseteq \mathsf{K}(D)$ finite (disjunction)

$w \in D$, then $\quad w \models X \quad$ iff $\quad \exists x \in X. x \sqsubseteq w$

Theory T: set of clauses (conjunktion)

$w \models T \quad$ iff $\quad \forall X \in T. w \models X$

$T \models X \quad$ iff $\quad \forall w \in D. w \models T \implies w \models X$

- compact; comes with proof theory
- *domain logic*: for characterizing Smyth powerdomains

Formal Concept Analysis (FCA)

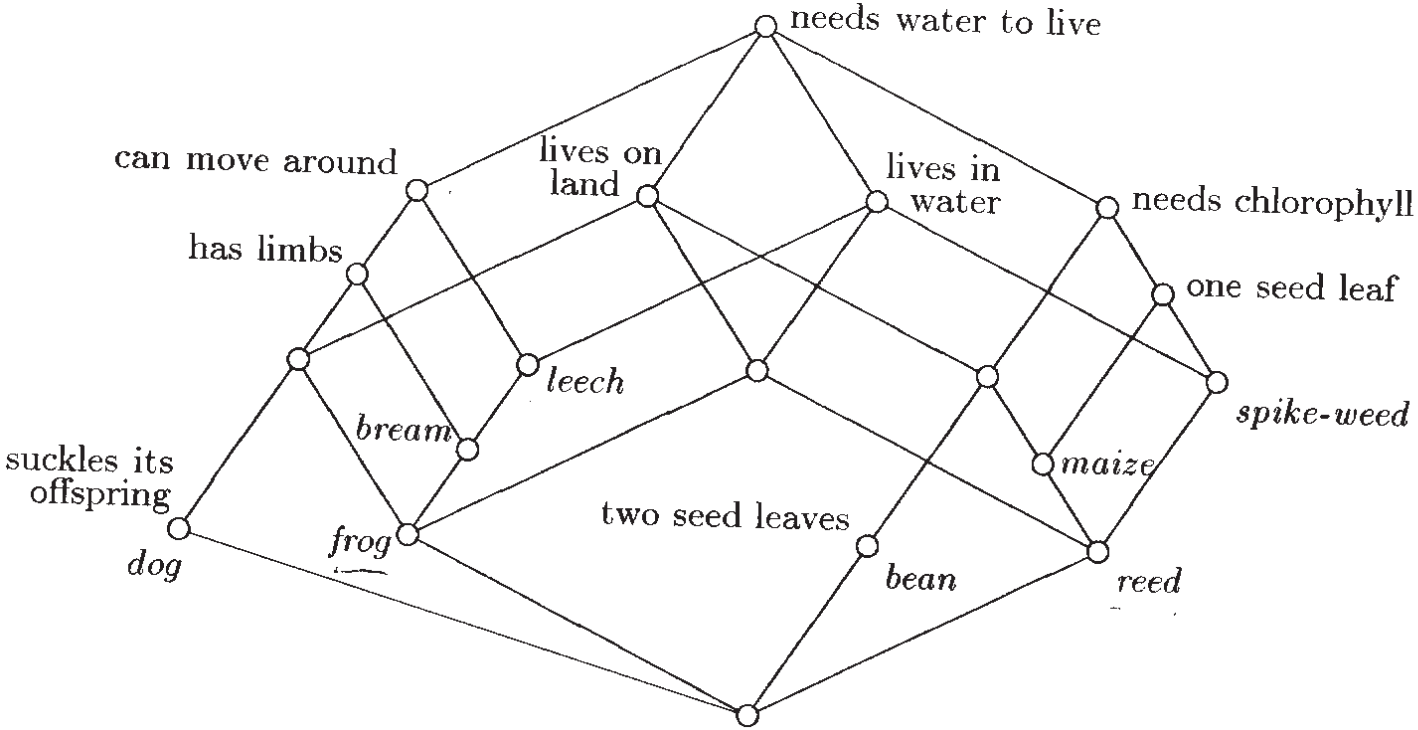


Figure 1.4 Concept lattice for the educational film "Living beings and water".

(picture: Ganter & Wille, Formal Concept Analysis, Springer, 1999. (also next slide))

FCA and Logic RZ

(H & Wendt ICCS 2003; H 2004)

Theorem

$(D, \sqsubseteq)$ coherent algebraic cpo

$(L, \leq)$ AOC of a formal context (G, I, M)

$\iota : L \to D$ order-reversing injection
with $\mathsf{K}(D) \subseteq \iota(L)$

$B = \{m_1, \ldots, m_n\} \subseteq M$ mit $\iota(m_i) \in \mathsf{K}(D)$ for all i

Then

$$B'' = \{m \in M \mid \{\{\iota(m_1)\}, \ldots, \{\iota(m_n)\}\} \models \{\iota(m)\}\}$$

RZ logic programming

(Rounds & Zhang IC 2001)

Add material implication: $X \leftarrow Y$ for X, Y clauses.

$w \models P$: if $w \models Y$ for $X \leftarrow Y \in P$, then $w \models X$.

Propagation rule $\mathrm{CP}(P)$:

$$\frac{X_1 \quad \dots \quad X_n; \quad a_i \in X_i; \quad Y \leftarrow Z \in P; \quad \mathrm{mub}\{a_1, \dots, a_n\} \models Z}{Y \cup \bigcup_{i=1}^{n}(X_i \setminus \{a_i\})}$$

Semantic operator on theories:

$$\mathcal{T}_P(T) = \mathrm{cons}\left(\{Y \mid Y \text{ is a } \mathrm{CP}(P)\text{-consequence of } T\}\right).$$

- ▶ $\mathcal{T}_P$ is Scott continuous.
- ▶ $\mathrm{fix}\left(\mathcal{T}_P\right) = \mathrm{cons}(P)$.

Addition of default negation

(H 2004)

P program, $w \in D$. Define P/w:

Replace $Y, \sim Z$ by Y if $w \not\models Z$.

Remove rule if $w \models Z$.

w *min-answer model* for P if w is minimal with $w \models \mathrm{fix}(\mathcal{T}_{P/w})$.

Theorem

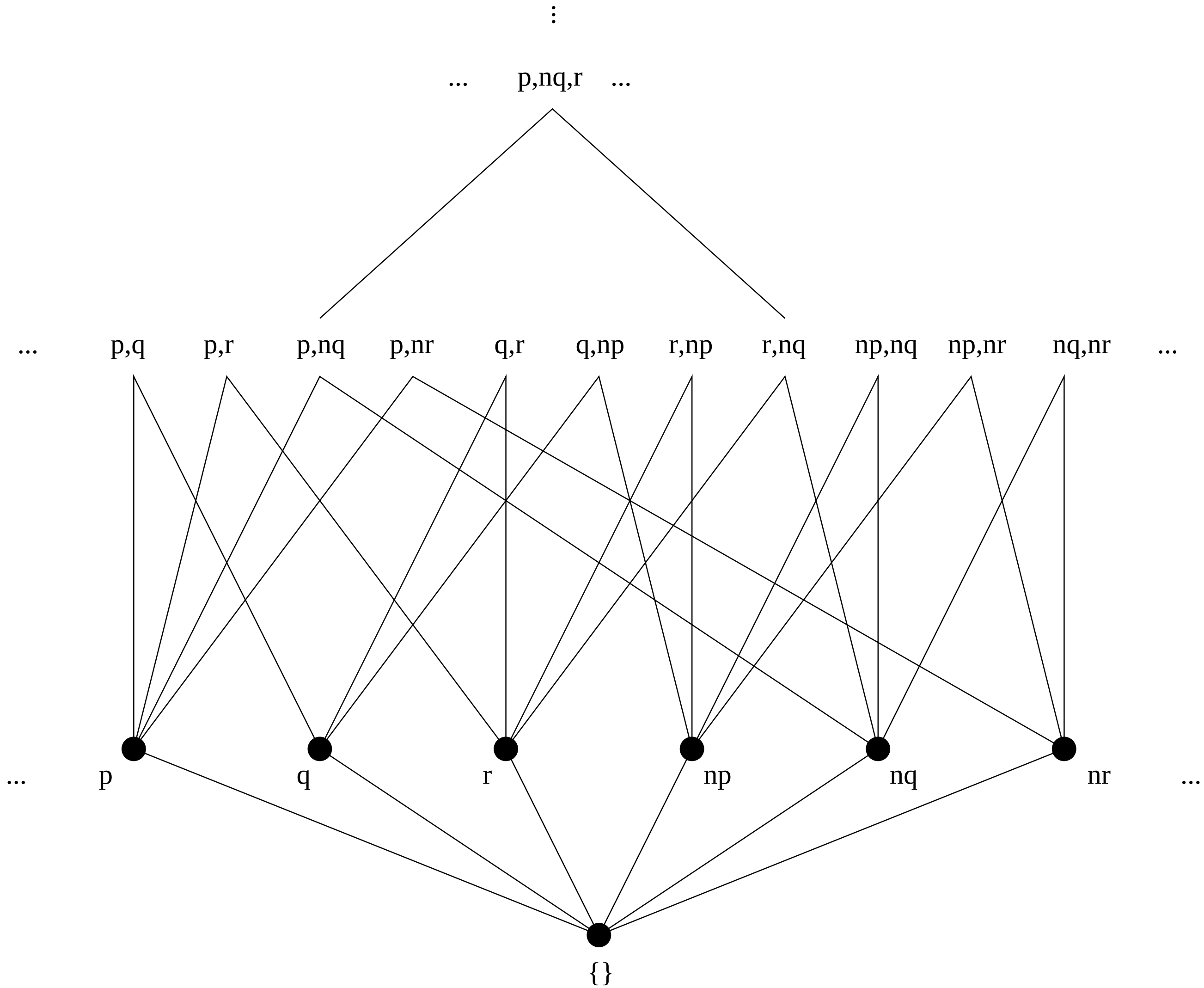
Answer set programming

with extended disjunctive programs (e.g. dlv-system)

is exactly

RZ logic programming with $D = \mathbb{T}^{\omega}$.

Plotkin's $\mathbb{T}^\omega$



Outlook

Idea:

Extraction of hierarchical conceptual knowledge.

Rule-based reasoning over this knowledge.

e.g.

Attribute exploration over T-boxes.

FCA-textmining.

Other methods (linguistics?).

In the sense of mainstream ASP.

Implementation e.g. on top of dlv.

Formal Concept Analysis (FCA)

(FCA: an approach to data mining and analysis; Ganter & Wille 1999)

G set of objects; M set of attributes. $C \subseteq G \times M$ *formal context*.

$A \subseteq G$ then $A' = \{m \in M \mid (\forall g \in A)(g, m) \in C\}$.
$B \subseteq M$ then $B' = \{g \in G \mid (\forall m \in B)(g, m) \in C\}$.

Formal concept: Pair (A, B) with $A' = B$, $A = B'$.
Equivalently: All (B', B'') for $B \subseteq M$.

Formal concept lattice:
complete lattice of all concepts ordered by $\supseteq$ in second argument.

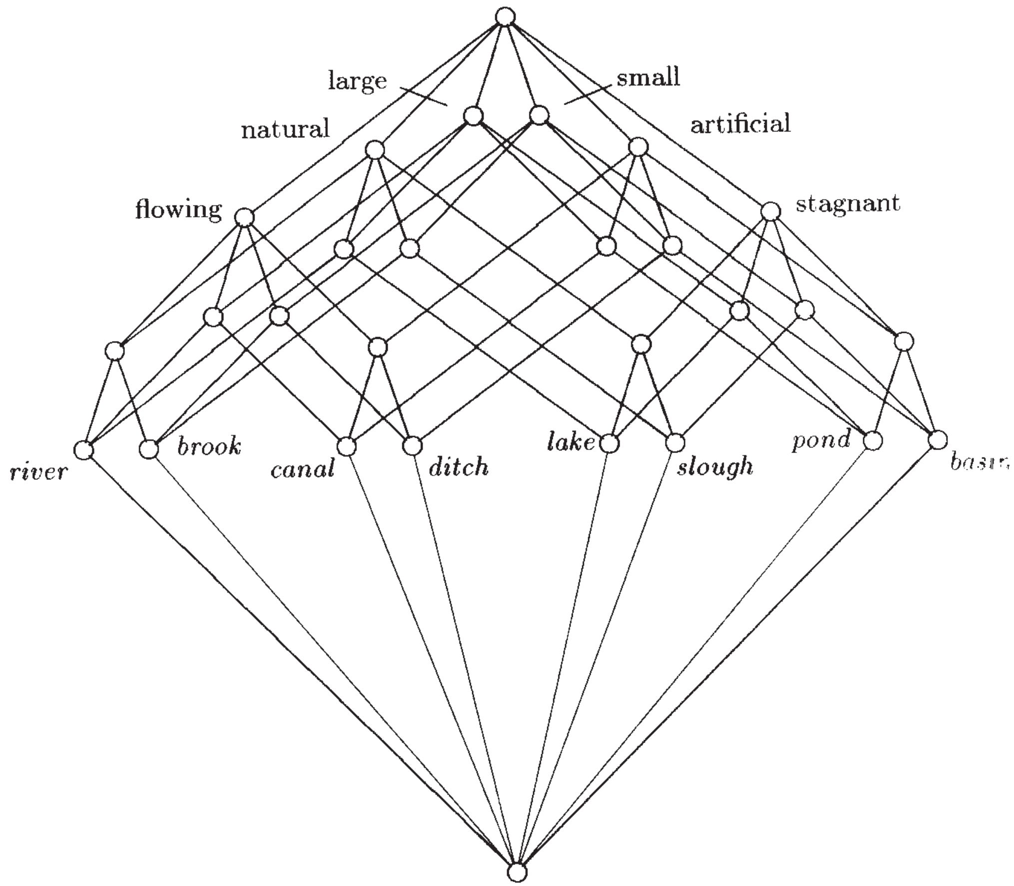


Figure 2.9 An additive line diagram of the concept lattice of a *lexical field "waters"*. The set representation is based on the irreducible attributes, i.e. the positioning of the attribute concepts determines that of all remaining concepts. If we interpret the line segments between the unit element and the attribute concepts as vectors, we obtain the position of an arbitrary concept by the sum of the vectors belonging to attributes of its concept intent starting from the unit element. Other diagrams for the same lattice can be found in Figure 2.10.

Domain theory: coherent algebraic cpos

cpo: directed complete partial order with bottom $(D, \sqsubseteq)$

$c \in \mathsf{K}(D)$ (compact) iff $(\forall A$ directed$)(d \sqsubseteq \bigsqcup A \implies (\exists a \in A) d \sqsubseteq a)$

cpo *algebraic*: $(\forall x)\,(x = \bigsqcup(x\!\downarrow \cap \mathsf{K}(D)))$

Scott topology: base $\{\uparrow c \mid c \in \mathsf{K}(D)\}$

coherent: finite intersections of compact-opens are compact-open

Examples: Finite posets with bottom. Powersets. $\mathbb{T}^\omega$.

Logic RZ

Proof theory: (H JEEEC 2004)

$$\frac{}{\{\perp\}}$$

$$\frac{X; \qquad a \in X; \qquad y \sqsubseteq a}{\{y\} \cup (X \setminus \{a\})}$$

$$\frac{X; \qquad y \in \mathsf{K}(D)}{\{y\} \cup X}$$

$$\frac{X_1 \quad X_2; \qquad a_1 \in X_1 \quad a_2 \in X_2}{\mathrm{mub}\{a_1, a_2\} \cup (X_1 \setminus \{a_1\}) \cup (X_2 \setminus \{a_2\})}$$

Logic RZ is compact.

Logic RZ and Formal Concept Analysis (FCA)

(H & Wendt ICCS 2003)

Consider subposet D of all $(\{b\}', \{b\}'')$, $b \in M$,
and all $(\{a\}'', \{a\}')$, $a \in G$, ordered reversely (add $\bot$).

If D is finite, then

for $D \supseteq \{b_i \mid i \in I\} = B \subseteq M$ we have

$$B'' = \{b \in M \mid \{\{b_i\} \mid i \in I\} \models \{b\}\}.$$

Extension of answer set programming

Consider $\mathbb{T}^\omega$.

Consider programs P with rules $X \leftarrow Y, \sim Z$ such that:

X contains only atoms in $\mathbb{T}^\omega$.

Y is a singleton clause.

Z contains only atoms in $\mathbb{T}^\omega$ or $\perp$.

These programs are exactly extended disjunctive programs.

Min-answer models w correspond to *answer sets* $\{L \text{ atom} \mid w \models \{L\}\}$ and vice-versa.