Scalable authoritative reasoning: a logical reconstruction

P. A. Bonatti[1] A. Hogan G. Lukacsy A. Polleres[2]

[1] Università di Napoli Federico II

[2] DERI Galway

COST WG1 annual meeting
Cyprus, Dec 15, 2009

Outline I

Motivations

- Processing billions of triples, distributed over the web
- Even polynomial inference procedures are slow
- An interesting strategy: incomplete reasoning
  - LarKC
  - [Hogan, Hart & Polleres, Int. J. Semantic Web Inf. Syst., 2009]

Motivations

- Processing billions of triples, distributed over the web
- Even polynomial inference procedures are slow
- An interesting strategy: incomplete reasoning
    - LarKC
    - [Hogan, Hart & Polleres, Int. J. Semantic Web Inf. Syst., 2009]

Principled incompleteness

- Normal form for equality (sameAs)
  - use a representative member c for each equivalence class C
  - produce only triples d sameAs c for $d \in C$
- Ontology hijacking (authoritativeness)
  - non-conservative extensions of ontology by other organizations
  - non-authoritative, hence untrusted
  - ignore them
- Blacklisted knowledge
  - ill-formed, e.g. empty key attributes or incomplete URIs
  - many equalities erroneously inferred
  - ignore blacklisted knowledge
- Inconsistencies are not exploded

Principled incompleteness II

In summary, three main ideas

1. Partial materialization; completing inference from it is cheap
2. Ignore unreliable or ill-formed knowledge
3. Paraconsistent semantics

Here we focus on points 2 and 3

Contributions

- A logical model of reasoning with untrusted knowledge
    - extensible to further notions of trusted/reliable knowledge
    - e.g. based on provenance, page ranks, etc.
- Alternative notion of completeness
    - guarantees on the results of reasoning (coverage)
    - guidelines for implementations
    - a basis for interoperability
        - through shared meaning of less-than-perfect knowledge bases
- Without reinventing the wheel

Rule syntax

- As a first step, standard syntax
  - We want to process the rules adopted *today*
- This presentation does not bother with XML details
- Standard rules:

$$A \leftarrow B_1, \ldots, B_n \quad (n \geq 0)$$

- and constraints (or *denials*)

$$\leftarrow B_1, \ldots, B_n \quad (n > 0)$$

- where $A, B_1, \ldots, B_n$ are logical atoms.
- However, we shall adopt a nonclassical semantics

Confidence values

- Each fact (and each conclusion) is associated to a *confidence value*
- Confidence values are structured to capture difference facets
  - authoritativeness
  - blacklisted conclusions
  - reputation
  - ...
- Example: $\langle \mathit{blacklisted}, \mathit{authoritative}, 0.8 \rangle$

Confidence ordering

- Confidence values are ordered by their strength
- Example:

$$\langle \mathit{blacklisted}, \mathit{authoritative}, 0.8\rangle \preceq \langle \mathit{correct}, \mathit{non\_authoritative}, 0.9\rangle$$

$$\langle \mathit{correct}, \mathit{non\_authoritative}, 0.9\rangle \preceq \langle \mathit{correct}, \mathit{authoritative}, 0.7\rangle$$

Formalizing confidence values

Definition (Domain of confidence values, CV)

CV is the Cartesian product $D_1 \times \ldots \times D_z$ of z total orders $(D_i, \leq_i)$ with finite l.u.b. $\max_i$ and finite g.l.b. $\min_i$ $(i = 1, \ldots, z)$.

CV is totally ordered by $\preceq$ that is the lexicographic ordering of CV based on $\leq_i$ $(1 \leq i \leq z)$.

Intuitions

Basic idea

Deriving A with confidence level v means:
there is a proof of A *whose components all have a confidence value* $\geq v$.

Example

A can be labelled $\langle \mathit{non\_blacklisted}, \mathit{authoritative}, 0.3\rangle$ only if some of its proofs adopts only

- non-blacklisted and authoritative rules and facts
- with confidence value ≥ 0.3.

- **Q**: shall we invent a new semantics for rules?
- **A**: not completely...

Annotated programs [Kifer & Subrahmanian, JLP, 1992]

Definition (Syntax)

Rules are expressions

$$A{:}f(v_1,\ldots,v_n) \leftarrow B_1{:}v_1 \,\&\ldots\&\, B_n{:}v_n$$

where

- $A, B_1, \ldots, B_n$ are atoms
- $v_1, \ldots, v_n$ are constants or variables ranging over confidence values
- $f : \mathrm{CV}^n \longrightarrow \mathrm{CV}$

Annotated programs [Kifer & Subrahmanian, JLP, 1992]

Definition (Semantics, specialized to total orderings)

An interpretation $\mathcal{I}$ is a mapping $\mathit{Atoms} \rightarrow \mathrm{CV}$

$\mathcal{I} \models A{:}v$ iff $v \preceq \mathcal{I}(A)$

$\mathcal{I} \models \neg A{:}v$ iff $\neg v \preceq \mathcal{I}(A)$ (arbitrary operator $\neg : \mathrm{CV} \rightarrow \mathrm{CV}$)

$\mathcal{I} \models F \,\&\, G$ iff $\mathcal{I} \models F$ and $\mathcal{I} \models G$

$\mathcal{I} \models F \leftarrow G$ iff $\mathcal{I} \models F$ or $\mathcal{I} \not\models G$

Remark

With this definition, when two sources with different reliability both say "*A*" there is no inconsistency:

$$A{:}v \,\&\, A{:}w \text{ entails } A{:}\max(v, w)$$

(the strongest evidence "wins")

Annotated programs [Kifer & Subrahmanian, JLP, 1992]

- Under suitable conditions one can define an immediate consequence operator R_P which is monotonic and continuous
    - the study of such preconditions in our particular case is still in progress
- Its least fixpoints is the least model of the annotated program
- There exist also top-down calculi based on resolution

Application of annotated programs to our framework

The key steps are:

1. Implicit annotation of (standard) rules and facts
    - based on the approach of [Hogan et al. 2009]
    - of course, explicit annotations can be used of so desired
2. Definition of a suitable negation
3. Suitable semantics of denials
    - for paraconsistent contradiction handling

Paraconsistent annotated semantics

Intuitions:

- Each atom A is assigned two confidence values
  - associated to A's truth and falsity, respectively
  - this allows for independent arguments in favor and against A
  - the closest the two values, the highest the uncertainty about A's (classical) truth value
- The negation operator ($\neg$) swaps those two values
  - the confidence in the truth of A is the confidence in the falsity of $\neg A$ and viceversa
- Given a constraint $\leftarrow A, B$, if A has confidence value v then it follows that also $\neg B$ should *at least* have confidence value v
- therefore, if $A{:}v$ and $B{:}w$ hold, also $\neg B{:}v$ and $\neg A{:}w$ hold
  - $\leftarrow A_1, \ldots, A_n$ abbreviates $\neg A_i \leftarrow A_1, \ldots, A_{i-1}, A_{i+1}, \ldots, A_n$ $(1 \leq i \leq n)$

Paraconsistent annotated language

Definition (Syntax, simplified)

Annotated rules are expressions $v{:}R$ where R is

$$L_0 \leftarrow L_1 \,\&\, \ldots \,\&\, L_n$$

where $L_0, L_1, \ldots, L_n$ are literals (A or $\neg A$)

Annotated constraints are annotated rules with empty head

Paraconsistent annotated semantics

Definition (Models and valuation)

An interpretation $\mathcal{I}$ is a mapping *Atoms* $\longrightarrow \mathrm{CV}^2$

$$\mathcal{I}(A) = \langle \mathcal{I}^+(A), \mathcal{I}^-(A) \rangle$$

$$\mathcal{I}(\neg A) = \langle \mathcal{I}^-(A), \mathcal{I}^+(A) \rangle$$

If R is $v : L_0 \leftarrow L_1, \ldots, L_n$ then

$$I \models R \text{ iff } \min_{\preceq}\{v, I^+(L_1), \ldots, I^+(L_n)\} \preceq I^+(L_0)$$

If R is $w :\leftarrow L_1, \ldots, L_n$ then

$$I \models C \text{ iff } \min_{\preceq}\{w, I^+(L_1), \ldots, I^+(L_{i-1}), I^+(L_{i+1}), \ldots, I^+(L_n)\} \preceq I^-(L_i)$$

for $i = 1, \ldots, n$

Paraconsistent annotated semantics

Properties relevant to implementation

- Each literal has one confidence value
    - no matter how many proofs it has
    - no storage explosion
- Anytime algorithms are possible
    - store the current value (best proof so far)
    - replace it when a “better” proof is found
- Predefined thresholds can be set
    - literals below threshold can be ignored
    - they can’t lead to any consequence above threshold

Conclusions

- A natural generalization of annotated programs nicely addresses
    - inconsistency confinement
    - authoritative reasoning
    - ill-formed knowledge

  in a uniform way
- Its properties are compatible with scalability needs
    - no storage explosion
    - anytime algortithms
    - thresholds

Further work

- Complete theoretical analysis
- Scalable, parameterizable implementations
- Suitable metalanguage for specifying confidence value domains
- Possibly drop restriction to total orders

QUESTION TIME