max planck institut
informatik

YAGO++ Query Answering

Christoph Weidenbach, Patrick Wischnewski

Translation

max planck institut informatik

AlbertEinstein BornIn Ulm

$$\Longrightarrow \text{BornIn}(\text{AlbertEinstein}, \text{Ulm})$$

LocatedIn Type YagoTransitiveRelation

$$\Longrightarrow \forall x, y, z\ \text{LocatedIn}(x, y) \wedge \text{LocatedIn}(y, z) \rightarrow \text{LocatedIn}(x, z)$$

BornIn Type YagoFunction

$$\Longrightarrow \forall x, y, z\ \text{BornIn}(x, y) \wedge \text{BornIn}(x, z) \rightarrow y \approx z$$

AngelaMerkel Type Human

$$\Longrightarrow \text{Human}(\text{AngelaMerkel})$$

Human SubClassOf Mammal

$$\Longrightarrow \forall x\ \text{Human}(x) \rightarrow \text{Mammal}(x)$$

Bernays–Schönfinkel Horn fragment with Equality (BSHE) + unique name

YAGO++

max planck institut informatik

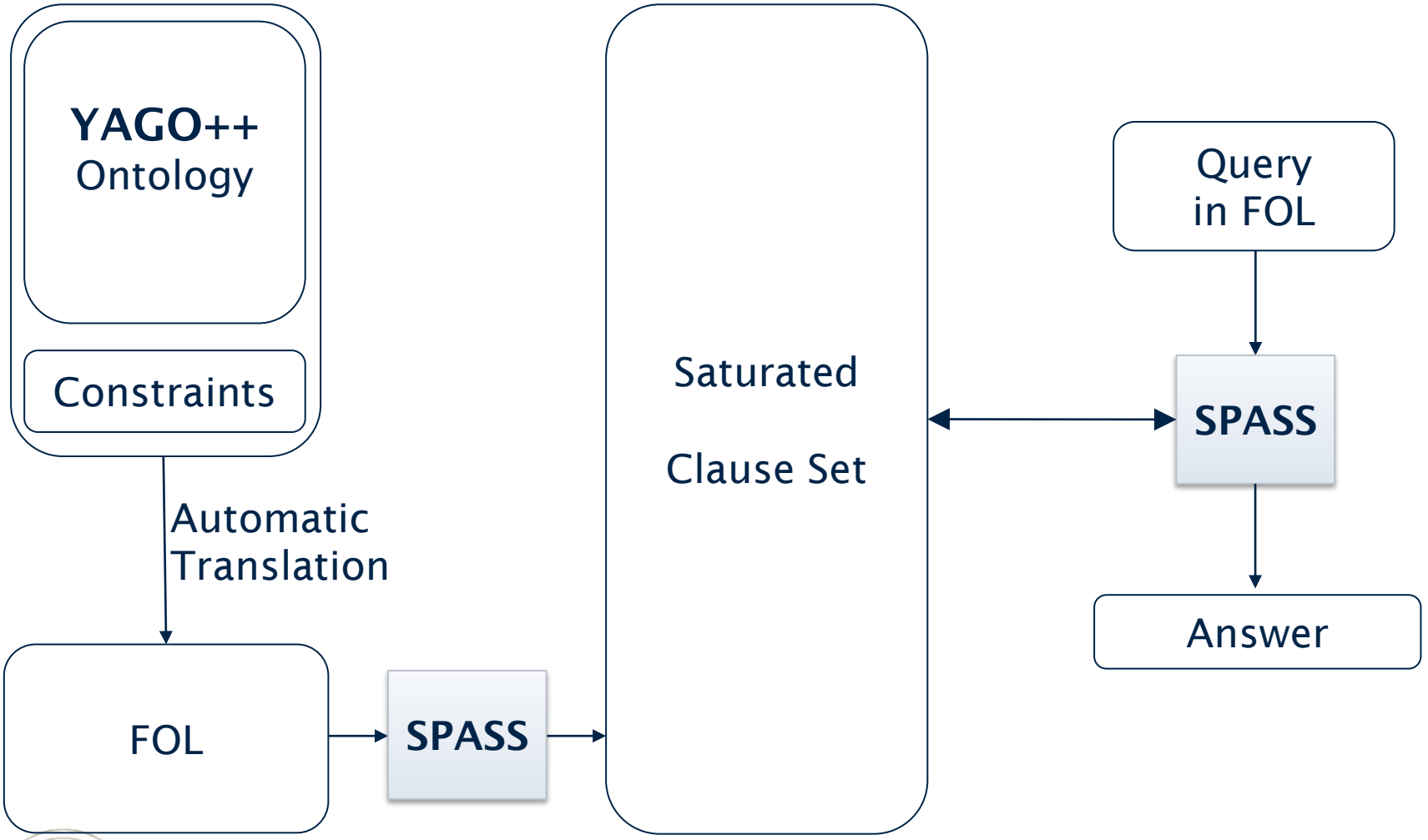
Negative Facts

$$\text{bornIn}(\text{AlbertEinstein}, \text{Munich}) \rightarrow$$

Definitions

$$\text{bornIn}(x, y) \wedge \text{locatedIn}(\text{y}, \text{z}) \wedge \text{country}(z) \rightarrow \text{bornInCountry}(x, z)$$

SPASS-YAGO


YAGO++
Ontology
Constraints
Automatic
Translation
FOL
SPASS
Saturated
Clause Set
Query
in FOL
SPASS
Answer

Calculus for BSHE

max planck institut informatik

Hyperresolution

$$\frac{A_1 \quad \ldots \quad A_n \quad \neg B_1 \vee \ldots \vee \neg B_n \vee P}{P\sigma}$$

- $A_1, \ldots, A_n$ are positive unit clauses,
- P is a positive literal or false
- σ is the simultaneous mgu of A_i and B_i for all $i \in \{1 \ldots n\}$

OECut

$$\frac{a \approx b}{\square}$$

- a and b are two different constants.

Positive Chaining

$$\frac{Q(l,s) \quad Q(t,r)}{Q(l,r)\sigma}$$

- σ is the mgu of s and t
- $l\sigma \not\succeq s\sigma$ and $r\sigma \not\succeq t\sigma$

Negative Chaining

$$\frac{Q(l,s) \quad D \vee \neg Q(t,r)}{D\sigma \vee \neg Q(t,l)\sigma}$$

- σ is the mgu of s and r
- $l\sigma \not\succeq s\sigma$ and $t\sigma \not\succ r\sigma$

Neg. Chaining and Definitions

$$\text{bornIn}(x, y) \wedge \text{locatedIn}(y, z) \wedge \text{country}(z) \rightarrow \text{bornInCountry}(x, z)$$

$$\frac{Q(l,s) \qquad D \vee \neg Q(t,r)}{D\sigma \vee \neg Q(t,l)\sigma}$$

- σ is the mgu of s and r
- $l\sigma \not\succeq s\sigma$ and $t\sigma \not\succ r\sigma$

Neg. Chaining and Definitions

$$\text{bornIn}(x, y) \wedge \text{locatedIn}(y, z) \wedge \text{country}(z) \rightarrow \text{bornInCountry}(x, z)$$

$$\frac{\Gamma_i \rightarrow A_i \qquad T_1, \ldots, T_m, B_1, \ldots, B_n \rightarrow \Delta}{(T_1, \ldots, T_m, \Gamma_1, \ldots, \Gamma_n \rightarrow \Delta)\sigma}$$

1. σ simultaneous unifier of all A_i and B_i
2. T_i are transitive atoms
3. B_i are non-transitive atoms

Queries

$$\Phi \quad := \quad \Gamma \quad | \quad \forall x.\Gamma \rightarrow \Phi \quad | \quad \exists x.\Gamma \wedge \Phi$$

Example:

$$\exists x, y.\mathrm{Phy}(x) \wedge \mathrm{bornIn}(x, y) \wedge \forall z.\mathrm{hasChild}(x, z) \rightarrow \mathrm{bornIn}(z, y)$$

Calculus

N saturated clause set

$$\frac{\forall \overline{x}.\, T_1, \ldots, T_m, P_1, \ldots, P_n \;\rightarrow\; \Phi \qquad P'_1, \ldots, P'_{n_2}}{\Phi\sigma}$$

1. P'_i positive ground unit clauses with $P'_i \in N$ for all $i \in \{1, \ldots, n\}$
2. σ simultanous unifier of $P_1, \ldots, P_n$ and $P'_1, \ldots, P'_n$
3. $N \models T_i\sigma$ for all $i \in \{1, \ldots, m\}$

Query Processing

max planck institut informatik

$\text{ProcessQuery}(\text{Query } \Phi)$

$\vdots$

$\textbf{if}(\Phi = \forall x.\Gamma \rightarrow \Phi') \{$

$\quad \textbf{ForEach}(\Phi'\sigma \in \text{forall}(\Phi, N)) \{$

$\quad\quad \textbf{if}(\neg\ \text{ProcessQuery}(\Phi'\sigma, N)\ \textbf{return}(\text{false})$

$\quad \}$

$\quad \textbf{return}(\text{true})$

$\}$

$\vdots$

Results

- We saturate YAGO++ in about 1 hour
  2.4 million clauses generated,
  134,000 kept

- SPASS-YAGO answer queries with quantifier alternations

$$\exists x, y.\, \mathrm{Phy}(x) \;\wedge\; \mathrm{bornIn}(x, y) \;\wedge\; \forall z.\, \mathrm{hasChild}(x, z) \;\rightarrow\; \mathrm{bornIn}(z, y)$$

Summary

- Extended YAGO with neg. facts and definitions
- New saturation calculus
- Defined the query language as a fragment of first-order logic
- New query answering calculus
- Implemented in SPASS-YAGO++
- Answers queries with quantifier alternations in < 1second
- Webfrontend and SPASS-YAGO++ as server backend

Future Work

- Faster query answering: Query optimization,…
- Robustness of implementation
- more stable Web-Interface and server backend
- Natural language interface
- Confidence values
- Time

Webfrontend: http://spassyago.spass-prover.org

Thank you for your attention