Quantitative Variants of Language Equations and their Applications to Description Logics

Pavlos Marantidis

Supervisor: Prof. Dr. Franz Baader

Motivation: Unification in Description Logics

While creating knowledge bases redundancies may occur.

**Woman ⊑ ∀ has-child, Woman**

**Human ⊑ Female ⊑ ∀ has-child, (Human \( \cap \) Female)**

Unification in DL can discover and remedy redundancies by introducing unifiers e.g., **Woman \( \rightarrow \) Human \( \cap \) Female**

Problem

Exact unifiers need not always exist

**Woman ⊑ ∀ has-child, Woman**

**Human ⊑ Female ⊑ ∀ has-child, Female**

Finding approximate solutions

1. Determine meaningful ways to assess solutions.
2. Provide algorithms for computing optimal solutions.

Approaches: Reduction to Automata Problems

Classical construction (introduced by Baader & Narendran)

Unification in \( FL_0 \) reduces to Language Equations solved using Automata

\[
C = D
\]

\[
X \cap \forall X
\]

\[
A \cap B \cap \forall (A \cap B)
\]

New construction (JELIA '16)

Approximate Unification in \( FL_0 \) reduces to Approximate Language Equations solved using Modified Versions of Automata

Measure of Closeness of Concepts

Distance of Languages

Modified Emptiness Test

Linear Programming

Current & Future Work

Matching

Special case of unification: \( K_0 = L_0 \cup L_1 X_1 \cup \cdots \cup L_n X_n \)

- The classical case is in P
- Approximate matching w.r.t. \( d_1 \) stays in P in NP for a wide class of distances

Extensions

- Adding terminological knowledge

  \[
  \text{Human} \cap \forall \text{has-child, Human}
  \]

- Approximate unification and matching in other DLs
- Connection to weighted automata (LATA '17)

Results

Warm-up: Approximate set equations (UNIF '16)

P- or NP-complete, for the measures considered

Language distances induce concept measures

Distances already considered

\[
d_1(K, L) = 2^{-n}
\]

with

\[
n = \min \{ |w| : w \in K \Delta L \}
\]

- Existence of optimal solution guaranteed for both cases.
- Approximate unification is ExpTime-complete w.r.t. \( d_1, d_2 \).