Finding $\mathcal{EL}^+$ Justifications using the Earley Parsing Algorithm

Riku Nortje **Katarina Britz** **Thomas Meyer**

Meraka Institute, CSIR
Pretoria, South Africa

Outline

1. Description Logics
   - A Brief Introduction
   - Reasoning Tasks
   - The $\mathcal{EL}$ Family of DLs
   - Reasoning Complexity
2. Module Extraction
   - Reachability Modules
   - Backward Reachability Modules
   - Bi-Directional Reachability Modules
3. The Earley Parsing Algorithm for Extracting Modules
   - Context Free Grammars
   - Earley Parsing Algorithm
4. Conclusion and Future Work

Outline

1. Description Logics
   - A Brief Introduction
   - Reasoning Tasks
   - The $\mathcal{EL}$ Family of DLs
   - Reasoning Complexity
2. Module Extraction
   - Reachability Modules
   - Backward Reachability Modules
   - Bi-Directional Reachability Modules
3. The Earley Parsing Algorithm for Extracting Modules
   - Context Free Grammars
   - Earley Parsing Algorithm
4. Conclusion and Future Work

Outline

1. Description Logics
   - A Brief Introduction
   - Reasoning Tasks
   - The $\mathcal{EL}$ Family of DLs
   - Reasoning Complexity
2. Module Extraction
   - Reachability Modules
   - Backward Reachability Modules
   - Bi-Directional Reachability Modules
3. The Earley Parsing Algorithm for Extracting Modules
   - Context Free Grammars
   - Earley Parsing Algorithm
4. Conclusion and Future Work

Outline

Outline

Description Logics

An overview

- A class of knowledge representation languages
- Frequently used to represent ontologies formally
- Decidable fragments of first-order logic
- Makes it possible to **reason** over ontologies

Description Logics

An overview

- Concepts: classes of individuals
- Roles: (binary) relationships between individuals
- Define a terminology: TBox
- Provide assertions: ABox
- We are only considered with Tboxes

Outline

Reasoning Tasks

Classical Reasoning Tasks

- Determining the subsumption hierarchy between concepts (Tbox)
- Instance checking (Abox)
- Consistency checking (Tbox)
- Concept equivalence (Tbox)

Non-Classical Reasoning Tasks

- Ontology repair (Tbox)
- Computing justifications (Tbox)
- Module extraction (Tbox)

Reasoning Tasks

Classical Reasoning Tasks

- Determining the subsumption hierarchy between concepts (Tbox)
- Instance checking (Abox)
- Consistency checking (Tbox)
- Concept equivalence (Tbox)

Non-Classical Reasoning Tasks

- Ontology repair (Tbox)
- Computing justifications (Tbox)
- Module extraction (Tbox)

Outline

1. Description Logics
   - A Brief Introduction
   - Reasoning Tasks
   - The $\mathcal{EL}$ Family of DLs
   - Reasoning Complexity
2. Module Extraction
   - Reachability Modules
   - Backward Reachability Modules
   - Bi-Directional Reachability Modules
3. The Earley Parsing Algorithm for Extracting Modules
   - Context Free Grammars
   - Earley Parsing Algorithm
4. Conclusion and Future Work

The description logic $\mathcal{EL}^+$

Concepts and Roles

- Concept names: Person, Mother, Female
- Special concepts: $\top$, $\bot$
- Roles: hasChild
- Concept construction: Person $\sqcap$ Female, $\exists$ hasChild.Female

Tbox (and Rbox) statements

- Subsumption statements: Mother $\sqsubseteq$ Female $\sqcap$ $\exists$ hasChild.$\top$
- Role inclusion: hasFather $\sqsubseteq$ hasAncestor

Abox statements

- Mother(Susan)
- hasChild(Susan,Samantha)

The description logic $\mathcal{EL}^+$

Concepts and Roles

- Concept names: Person, Mother, Female
- Special concepts: $\top$, $\bot$
- Roles: hasChild
- Concept construction: Person $\sqcap$ Female, $\exists$ hasChild.Female

Tbox (and Rbox) statements

- Subsumption statements: Mother $\sqsubseteq$ Female $\sqcap$ $\exists$ hasChild.$\top$
- Role inclusion: hasFather $\sqsubseteq$ hasAncestor

Abox statements

- Mother(Susan)
- hasChild(Susan,Samantha)

The description logic $\mathcal{EL}^+$

Concepts and Roles

- Concept names: Person, Mother, Female
- Special concepts: $\top$, $\bot$
- Roles: hasChild
- Concept construction: Person $\sqcap$ Female, $\exists$ hasChild.Female

Tbox (and Rbox) statements

- Subsumption statements: Mother $\sqsubseteq$ Female $\sqcap$ $\exists$ hasChild.$\top$
- Role inclusion: hasFather $\sqsubseteq$ hasAncestor

Abox statements

- Mother(Susan)
- hasChild(Susan,Samantha)

Classical Reasoning Tasks

Subsumption checking

- Is Female subsumed by Person?

Instance checking

- Is Susan a Female?

Consistency checking

- Is Female subsumed by $\bot$?

Equivalence checking

- Is Female subsumed by Person **and** Person subsumed by Female?

Non-Classical Reasoning Tasks

- Let $\mathcal{O}$ be a set of Tbox (and Rbox) statements

Justification (MinA)

- Let $A \sqsubseteq B$ be a statement that follows **logically** from $\mathcal{O}$
- Justification: A minimal subset of $\mathcal{O}$ from which $A \sqsubseteq B$ follows

Module extraction

- Given a statement $A \sqsubseteq B$
- Module: a subset $\mathcal{O}'$ of $\mathcal{O}$ such that
  $A \sqsubseteq B$ follows from $\mathcal{O}'$ iff
  $A \sqsubseteq B$ follows from $\mathcal{O}$

Outline

Complexity for Description Logics

Classical Reasoning

- Classical reasoning for DLs in general is hard (intractable in general)
- Reasoning for the $\mathcal{EL}$ family is much better
- Computing the subsumption hierarchy can be done in polynomial time
- The $\mathcal{EL}$ family is surprisingly useful

Non-classical Reasoning

- Non-classical reasoning is based on classical reasoning
- Finding justifications involve consistency checking
- So is module extraction
- In general non-classical reasoning tasks are hard (intractable)
  - Even for DLs for which classical reasoning is tractable
- A solution is to use heuristics

Outline

Reachability Modules

Subsumption Module for A: $\mathcal{O}'$

- $\mathcal{O}'$ is a subset of $\mathcal{O}$
- $A \sqsubseteq B$ follows from $\mathcal{O}'$ iff $A \sqsubseteq B$ follows from $\mathcal{O}$ for every B in $\mathcal{O}$

Strong Subsumption Module for A: $\mathcal{O}'$

- Subsumption module for A
- If $A \sqsubseteq B$ follows from $\mathcal{O}$ then all MinAs for $A \sqsubseteq B$ are contained in $\mathcal{O}'$

Definition of A-reachability

- A is A-reachable
- For all axioms $\alpha_L \sqsubseteq \alpha_R$, if x is A-reachable for every x in α_L then y is A-reachable for every y in α_R

A-reachability

- $\alpha_L \sqsubseteq \alpha_R$ is A-reachable if all xs in α_L are A-reachable
- $\mathcal{O}^{A\rightarrow}$: set of A-reachable axioms
- $\mathcal{O}^{A\rightarrow}$ **is a strong subsumption module for** A

Example

1. $A \sqsubseteq \exists r.D$ 2. $\exists r.D \sqsubseteq B$ 3. $E \sqsubseteq B$ 4. $A \sqsubseteq F$

$\mathcal{O}^{A\rightarrow}$: 1, 2, 4

Observe: Axiom 4 is irrelevant

Outline

Backward Reachability modules

Backward Subsumption module for B: $\mathcal{O}'$

- $\mathcal{O}'$ is a subset of $\mathcal{O}$
- $A \sqsubseteq B$ follows from $\mathcal{O}'$ iff $A \sqsubseteq B$ follows from $\mathcal{O}$ for every A in $\mathcal{O}$

Strong Backward Subsumption Module for B: $\mathcal{O}'$

- Backward subsumption module for B
- If $A \sqsubseteq B$ follows from $\mathcal{O}$ then all MinAs for $A \sqsubseteq B$ are contained in $\mathcal{O}'$

Definition of backward B-reachability

- B is backward B-reachable
- For all axioms $\alpha_L \sqsubseteq \alpha_R$, if y is backward B-reachable for some y in α_R, or if $\alpha_R = \bot$, then x is backward B-reachable for every x in α_L

Backward B-reachability

- $\alpha_L \sqsubseteq \alpha_R$ is backward B-reachable if some y in α_R is backward B-reachable, or if $\alpha_R = \bot$,
- $\mathcal{O}^{\to B}$: set of backward B-reachable axioms
- $\mathcal{O}^{\to B}$ **is a strong backward subsumption module for** B

Example

1. $A \sqsubseteq \exists r.D$ 2. $\exists r.D \sqsubseteq B$ 3. $E \sqsubseteq B$ 4. $A \sqsubseteq F$

$\mathcal{O}^{\to B}$: 1, 2, 3

Observe: Axiom 3 is irrelevant

Outline

Definition of a bi-directional reachability-based module: $\mathcal{O}^{A \to B}$

All axioms $\alpha_L \sqsubseteq \alpha_R \in \mathcal{O}$ such that
for every x in α_L, x is A-reachable and,
for some y in α_R, y is backward B-reachable

Example

1. $A \sqsubseteq \exists r.D$ 2. $\exists r.D \sqsubseteq B$ 3. $E \sqsubseteq B$ 4. $A \sqsubseteq F$

$\mathcal{O}^{A \to B}$: 1, 2

Observe: Axioms 3 and 4 don't feature anymore

Result

If $A \sqsubseteq B$ follows from $\mathcal{O}$ then $\mathcal{O}^{A \to B}$ contains all MinAs for $A \sqsubseteq B$

Outline

1. Description Logics
   - A Brief Introduction
   - Reasoning Tasks
   - The $\mathcal{EL}$ Family of DLs
   - Reasoning Complexity

2. Module Extraction
   - Reachability Modules
   - Backward Reachability Modules
   - Bi-Directional Reachability Modules

3. The Earley Parsing Algorithm for Extracting Modules
   - Context Free Grammars
   - Earley Parsing Algorithm

4. Conclusion and Future Work

Context Free Grammars

- Well-known method for modeling the structure of languages
- Set of productions
  - Expresses the ways the symbols in a language can be grouped together
  - Lexicon of symbols

Example

S → VP
S → NP VP
VP → VP NP
NP → Det Noun
VP → Verb
Det → that
Verb → book
Noun → flight

Outline

Earley Parsing Algorithm

- Dynamic programming approach
- A single left-to-right, top-down, depth-first parallel search strategy to compute all possible parses
- Does this in polynomial worst case time (n^3)

Converting $\mathcal{EL}^+$ to Earley

- An $\mathcal{EL}^+$ ontology can be converted to a Context Free Grammer
- The Earley Algorithm is used to compute all reachability paths
- Used to compute bi-directional subsumption modules

Conclusion

- Extended the notion of reachability
  - Backward reachability
  - Bi-directional reachability
- Conversion of $\mathcal{EL}^+$ statements to a Context Free Grammar
- Use the Earley parsing algorithm to compute bi-drectional subsumption modules

Future Work

- Seems promising in principle: will it be worthwhile in practice?
- Efficient implementation
- Protégé plugin
- Investigate the link with automata-based pinpointing approach (Penãloza)