Knowledge representation and ontology engineering

3. knowledge engineering with PL and FOL

Nicolas Troquard
nicolas.troquard@gssi.it

Outline

Language

The language of propositional logic is inductively defined from:

- Propositional variables: atomic statements that can be true or false
- Symbol $\top$: truth
- Propositional connectives:
  - $\neg$: not
  - $\vee$: or
- Parentheses (and)

Formally:

$$A ::= \top \mid p \mid \neg A \mid A \vee A$$

where p is a propositional variable.

Defined connectives:

- $A \wedge B := \neg(\neg A \vee \neg B)$
- $A \rightarrow B := \neg A \vee B$
- $A \leftrightarrow B := (A \rightarrow B) \wedge (B \rightarrow A)$
- $\bot := \neg\top$

Examples

A simple knowledge base of the domain of tumours:

- $\text{Benign} \rightarrow \neg\text{Metastasis}$
- $\text{Stage4} \leftrightarrow \neg\text{Benign}$
- $\text{Treatment} \rightarrow \text{Surgery} \vee \text{Chemo} \vee \text{Radio}$

Meaning through interpretations

An interpretation for PL is a tuple $\mathcal{I} = (P, .^{\mathcal{I}})$, where:

- P is a set of propositional variables
- $.^{\mathcal{I}} : P \longrightarrow \{true, false\}$ assigns truth values to propositional variables

The assignment $.^{\mathcal{I}}$ can be inductively extended to all PL formulas:

- $(\neg A)^{\mathcal{I}} = true$ iff $A^{\mathcal{I}} = false$
- $(A \vee B)^{\mathcal{I}} = true$ iff $A^{\mathcal{I}} = true$ or $B^{\mathcal{I}} = true$

We write $\mathcal{I} \models A$ when $A^{\mathcal{I}} = true$, and say that A is satisfied in $\mathcal{I}$, or that $\mathcal{I}$ is a model of A.

Reasoning, computational complexity of PL

A formula A is satisfiable if there is an interpretation that is a model of A.
A formula A is valid if A is satisfied in every model.
A set of formulas Γ entails a formula B if every interpretation that is model of all formulas in Γ is also a model of B.

Deciding satisfiability in PL is NP-complete.
Deciding unsatisfiability in PL is coNP-complete.
Deciding validity in PL is coNP-complete. (A valid iff $\neg A$ is not satisfiable)
Deciding entailment in PL is coNP-complete (Γ entails B iff $(\bigwedge_{A \in \Gamma} A) \to B$ is valid)

Reminder:
... $AC^0 \subseteq$ LOGSPACE $\subseteq$ NLOGSPACE $\subseteq$ P $\subseteq$ NP, coNP $\subseteq$ ... $\subseteq$ PH $\subseteq$ PSPACE $\subseteq$ EXPTIME $\subseteq$ NEXPTIME $\subseteq$ EXPSPACE $\subseteq$ 2EXPTIME $\subseteq$ N2EXPTIME $\subseteq$ 2EXPSPACE $\subseteq$ ... $\subseteq$ E $\subseteq$ TOWER $\subseteq$ RE $\subseteq$ ...
... and much more, before, after, and in-between.

Limitations of PL (1)

Consider the following statements from a medical domain:

- A juvenile disease affects only children or teenagers
- Children and teenagers are not adults
- Juvenile arthritis is a kind of arthritis and a juvenile disease
- Arthritis affects some adults

Expected consequence (this could be a competency question): Juvenile arthritis does not affect adults.

Attempt at formalisation in PL:

- $\mathsf{JuvDisease} \rightarrow \mathsf{AffectsChild} \lor \mathsf{AffectsTeenager}$
- $\mathsf{Child} \lor \mathsf{Teenager} \rightarrow \neg\mathsf{Adult}$
- $\mathsf{JuvArthritis} \rightarrow \mathsf{JuvDisease} \land \mathsf{Arthritis}$
- $\mathsf{Arthritis} \rightarrow \mathsf{AffectsAdult}$

Does it entail: $\mathsf{JuvArthritis} \rightarrow \neg\mathsf{AffectsAdult}$?

Limitations of PL (1)

Consider the following statements from a medical domain:

- A juvenile disease affects only children or teenagers
- Children and teenagers are not adults
- Juvenile arthritis is a kind of arthritis and a juvenile disease
- Arthritis affects some adults

Expected consequence (this could be a competency question): Juvenile arthritis does not affect adults.

Attempt at formalisation in PL:

- $\text{JuvDisease} \rightarrow \text{AffectsChild} \vee \text{AffectsTeenager}$
- $\text{Child} \vee \text{Teenager} \rightarrow \neg\text{Adult}$
- $\text{JuvArthritis} \rightarrow \text{JuvDisease} \wedge \text{Arthritis}$
- $\text{Arthritis} \rightarrow \text{AffectsAdult}$

Does it entail: $\text{JuvArthritis} \rightarrow \neg\text{AffectsAdult}$?

No. Worse, we obtain an unsatisfiable set of formulas when we add:

- $\text{JuvArthritis} \rightarrow \neg\text{AffectsAdult}$?
- JuvArthritis

Limitations of PL (2)

PL cannot make a distinction between objects, relationships between objects, and quantifier restrictions.

- A juvenile disease affects only children or teenagers
- Children and teenagers are not adults
- Juvenile arthritis is a kind of arthritis and a juvenile disease
- Arthritis affects some adults

We need a more expressive language for knowledge representation.

Outline

Language

FO languages are inductively defined from:

- Predicate Symbols, each with an arity
- Function symbols, each with an arity
- Constants
- Variables
- Symbol $\top$: truth
- Propositional connectives: $\neg$, $\vee$
- The existential and universal quantifiers: $\exists$, $\forall$
- Parentheses (and)

Formally:

$$t ::= x \mid c \mid f(t, \ldots, t)$$

$$\beta ::= t = t \mid R(t, \ldots, t)$$

$$\alpha ::= \top \mid \beta \mid \neg\alpha \mid \alpha \vee \alpha \mid \exists x.\alpha$$

where t are terms, f are functions mapping tuples of terms to terms, and R are relations over terms.
In the formula $\mathsf{MotherOf}(\mathsf{ann}, \mathsf{john}) \wedge \exists x.\mathsf{BrotherOf}(\mathsf{bob}, x)$, x is a bound variable.
In the formula $\mathsf{FatherOf}(\mathsf{john}, x)$, x is a free variable.
A FO sentence is a formula without free variables.

Meaning through interpretations

An interpretation for FOL is a tuple $\mathcal{I} = (D, .^{\mathcal{I}})$, where:

- D is non-empty set; the domain of interpretation
- $.^{\mathcal{I}}$ is the interpretation function that associates:
  - every constant c an object $c^{\mathcal{I}} \in D$.
  - every n-ary function symbol f, a function $f^{\mathcal{I}} : D^n \longrightarrow D$
  - every n-ary predicate symbol R, a relation $R^{\mathcal{I}} \subseteq D^n$.

Meaning through interpretations and assignments

Interpreting terms:

- To interpret free variables, given an interpretation $\mathcal{I}$, an assignment is a function g that assigns an element of D to every variable of the language.
- We can extend the assignment g: to constants $g(c) = c$, and to functions $g(f(t_1, \ldots, t_n)) = f(g(t_1), \ldots, g(t_n))$.

Given an interpretation $\mathcal{I}$ and an assignment g, every FOL formula is either true or false:

- $R(t_1, \ldots, t_n)^{\mathcal{I}}[g] = true$ iff $(g(t_1), \ldots, g(t_n)) \in R^{\mathcal{I}}$
- $(t_1 = t_2)^{\mathcal{I}}[g] = true$ iff $g(t_1) = g(t_2)$
- $(\neg\alpha)^{\mathcal{I}}[g] = true$ iff $\alpha^{\mathcal{I}}[g] = false$
- $(\alpha_1 \vee \alpha_2)^{\mathcal{I}}[g] = true$ iff $\alpha_1^{\mathcal{I}}[g] = true$ or $\alpha_2^{\mathcal{I}}[g] = true$

$$(\exists x.\alpha)^{\mathcal{I}}[g] = true \text{ iff there is } a \in D \text{ such that } \alpha^{\mathcal{I}}[g/x \mapsto a] = true$$

That is, there is an a in the domain of interpretation that we can (re)assign to x, that makes α true in $\mathcal{I}$ under the (modified) assignment.

Satisfiability of sentences

For interpreting a sentence, assignments are irrelevant (no free variables).

Given a sentence α, we write $\mathcal{I} \models \alpha$ when $\alpha^{\mathcal{I}} = true$, and say that α is satisfied in $\mathcal{I}$, or that $\mathcal{I}$ is a model of α.

Validity and entailment are defined from satisfiability.

Example in FOL (1)

- Child, Arthritis, ... Unary predicates
- Affects Binary predicate
- ssnOf Unary function
- johnSmith, maryJones, jra Constants[1]
- x, y, z variables

E.g.:

- $\text{Child}(\text{johnSmith})$
- $\text{Affects}(\text{jra}, \text{johnSmith})$
- $\forall x.(\text{Affects}(\text{jra}, x) \rightarrow \text{Child}(x) \vee \text{Teenager}(x))$
- $\neg(\exists x.\exists y.(\text{JuvArthritis}(x) \wedge \text{Affects}(x, y) \wedge \text{Adult}(y)))$

[1] jra: juvenile rheumatoid arthritis

Example in FOL (2)

- A juvenile disease affects only children or teenagers
- Children and teenagers are not adults
- Juvenile arthritis is a kind of arthritis and a juvenile disease
- Arthritis affects some adults

Formalisation in FOL:

- $\forall x.(\forall y.(\text{JuvDisease}(x) \land \text{Affects}(x, y) \rightarrow \text{Child}(y) \lor \text{Teenager}(y)))$
- $\forall x.(\text{Child}(x) \lor \text{Teenager}(x) \rightarrow \neg\text{Adult}(x))$
- $\forall x.(\text{JuvArthritis}(x) \rightarrow \text{Arthritis}(x) \land \text{JuvDisease}(x))$
- $\exists x.(\exists y.(\text{Arthritis}(x) \land \text{Affects}(x, y) \land \text{Adult}(y)))$

PL vs FOL

A juvenile disease affects only children or teenagers

- JuvDisease $\rightarrow$ AffectsChild $\vee$ AffectsTeenager
  - 8 possible interpretations (over the three propositional variables)
  - 7 models
- $\forall x.(\forall y.(\text{JuvDisease}(x) \wedge \text{Affects}(x, y) \rightarrow \text{Child}(y) \vee \text{Teenager}(y)))$
  - infinity of interpretations (over arbitrary domains)
  - infinity of models

The role of reasoning

Why are we interested in reasoning?

- Discover new knowledge
- Detect undesired consequences
  - Γ entails $\exists x.(\text{Teenager}(x) \wedge \text{JuvDisease}(x))$
  - broken knowledge: Γ entail $\perp$

Juvenile arthritis does not affect adults?

Knowledge base Γ:

1. $\forall x.(\forall y.(\text{JuvDisease}(x) \wedge \text{Affects}(x, y) \rightarrow \text{Child}(y) \vee \text{Teenager}(y)))$
2. $\forall x.(\text{Child}(x) \vee \text{Teenager}(x) \rightarrow \neg\text{Adult}(x))$
3. $\forall x.(\text{JuvArthritis}(x) \rightarrow \text{Arthritis}(x) \wedge \text{JuvDisease}(x))$
4. $\exists x.(\exists y.(\text{Arthritis}(x) \wedge \text{Affects}(x, y) \wedge \text{Adult}(y)))$

Question:

- Does Γ entail $\forall x.(\forall y.(\text{JuvArthritis}(x) \wedge \text{Affects}(x, y) \rightarrow \neg\text{Adult}(y))$?

Exercise

Answer the question.

Juvenile arthritis does not affect adults? (solution)

Knowledge base Γ:

1. $\forall x.(\forall y.(\mathsf{JuvDisease}(x) \wedge \mathsf{Affects}(x,y) \rightarrow \mathsf{Child}(y) \vee \mathsf{Teenager}(y)))$
2. $\forall x.(\mathsf{Child}(x) \vee \mathsf{Teenager}(x) \rightarrow \neg\mathsf{Adult}(x))$
3. $\forall x.(\mathsf{JuvArthritis}(x) \rightarrow \mathsf{Arthritis}(x) \wedge \mathsf{JuvDisease}(x))$
4. $\exists x.(\exists y.(\mathsf{Arthritis}(x) \wedge \mathsf{Affects}(x,y) \wedge \mathsf{Adult}(y)))$

Question:

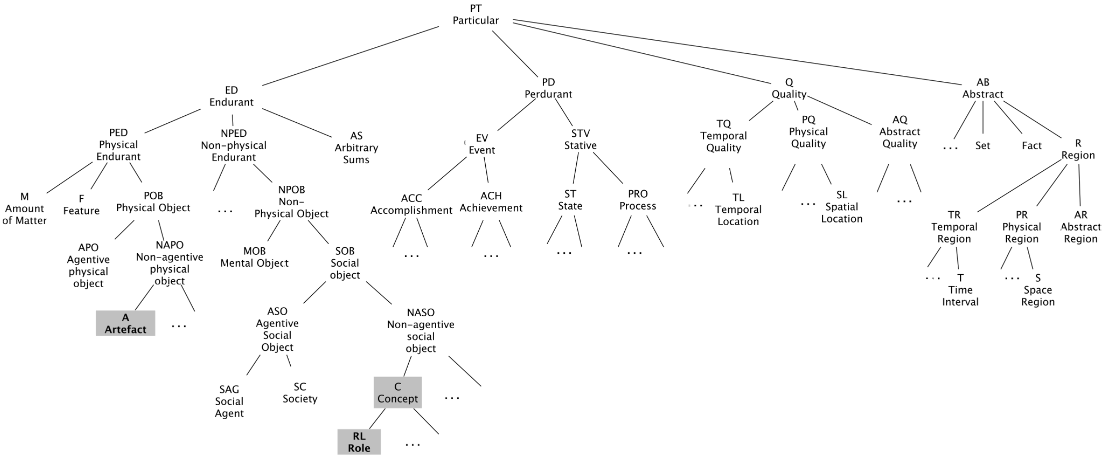
- Does Γ entail $\forall x.(\forall y.(\mathsf{JuvArthritis}(x) \wedge \mathsf{Affects}(x,y) \rightarrow \neg\mathsf{Adult}(y))$?

Answer:

- $\mathsf{JuvArthritis}(x)$ implies $\mathsf{Arthritis}(x)$ and $\mathsf{JuvDisease}(x)$ (use axiom 3)
- so we have $\mathsf{JuvDisease}(x)$ and $\mathsf{Affects}(x,y)$
- $\mathsf{JuvDisease}(x)$ and $\mathsf{Affects}(x,y)$ imply $\mathsf{Child}(y) \vee \mathsf{Teenager}(y)$ (use axiom 1)
- $\mathsf{Child}(y) \vee \mathsf{Teenager}(y)$ implies $\neg\mathsf{Adult}(x)$ (use axiom 2)
- so $\mathsf{JuvArthritis}(x) \wedge \mathsf{Affects}(x,y)$ imply $\neg\mathsf{Adult}(x)$
- so juvenile arthritis does not affect adults.

FOL as a language for foundational ontologies (1)

DOLCE [Masolo et al. 2003, Borgo et al. 2022][2], a foundational ontology. The taxonomy:



[2] Stefano Borgo et al. "DOLCE: A descriptive ontology for linguistic and cognitive engineering". In: *Applied Ontology* 17.1 (2022), pp. 45–69.

FOL as a language for foundational ontologies (2)

(ASO: agentive social object, SOB: social object, SC: society, P: (temporal) parthood, ED: endurant, PD: perdurant, T: time, PRE: presence, PC(C): (constant) participation)

Example of taxonomy (Agent):

- $\forall x.(\text{ASO}(x) \rightarrow \text{SOB}(x))$
- $\forall x.(\text{SC}(x) \rightarrow \text{ASO}(x))$
- ...

Example of typing (Mereology):

- $\text{P}(x, y, t) \rightarrow \text{ED}(x) \wedge \text{ED}(y) \wedge \text{T}(t)$
- ...

Example of definition ((Constant) Participation):

- $\text{PC}(x, y, t) \rightarrow \text{ED}(x) \wedge \text{PD}(x) \wedge \text{T}(t)$
- ...
- $\text{PCC}(x, y) := \exists t.(\text{PRE}(y, t)) \wedge \forall t.(\text{PRE}(y, t) \rightarrow \text{PC}(x, y, t))$

Computational complexity of FOL

The set of valid formulas in FOL can be characterized with a finite, sound and complete axiomatization. Validities in FOL are recursively enumerable [Gödel 1929].

Satisfiability in FOL is undecidable [Church 1936, Turing 1937].

We need a language computationally easier for knowledge representation and reasoning.

This is what we look at next.

Credits

Many slides and examples based on Ian Horrocks's KRR lectures
`https://www.cs.ox.ac.uk/people/ian.horrocks/`.
`https://www.cs.ox.ac.uk/teaching/courses/2020-2021/KRR/`