Logics for Data and Knowledge Representation

2b. Reasoning in First Order Logic

Chiara Ghidini

FBK-irst, Trento, Italy

September 27, 2012

Outline

- Reasoning in FOL:
    - General Concepts;
    - Tableaux calculus with examples.

;

Reasoning tasks in FOL

Model checking

Check if $\mathcal{I} \models \phi$

Satisfiability

Find a model for a formula ϕ

Validity

Check if a formula ϕ is valid, i.e., if it is true in all the interpretations.

Reasoning tasks in FOL

Logical consequence

Check if a formula ϕ is a logical consequence of a set of formulas Γ, i.e., $\Gamma \models \phi$

Instance checking

Given a set of formulas Γ (a Knowledge Base) check if a constant c is an instance of a formula $\phi(x)$, i.e., if $\Gamma \models \phi(c)$.

Query answering

Given a set of formulas Γ (a Knowledge Base) list all the constants c such that $\Gamma \models \phi(c)$.

Deciding logical consequence

Problem

Is there an algorithm to determine whether a formula ϕ is the logical consequence of a set of formulas Γ?

Naïve solution

- Apply directly the definition of logical consequence. That is:
  - build all the possible interpretations $\mathcal{I}$;
  - determine for which interpretations $\mathcal{I} \models \Gamma$;
  - for those interpretations check if $\mathcal{I} \models A$
- This solution can be used when Γ is finite, and there is a finite number of relevant interpretations.

Deciding logical consequence, is not always possible

Propositional Logics

The truth table method enumerates all the possible interpretations of a formula and, for each formula, it computes the relation $\models$.

Other logics

For first order logic there no general algorithm to compute the logical consequence. This because there may be an infinite number of relevant interpretations.

The Naïve solution in Propositional logic

Exercise

Determine, via truth table, if the following statements about logical consequence holds

- $p \models q$
- $p \supset q \models q \supset p$
- $p, \neg q \supset \neg p \models q$
- $\neg q \supset \neg p \models p \supset q$

Reasoning in FOL

Instead of building all possible interpretations of Γ and check whether $\Gamma \models \phi$ build a proof of ϕ from Γ (in symbols, $\Gamma \vdash \phi$)

1. Direct method: Try to build a proof of ϕ from Γ using axioms and syntactic manipulation rules (reasoning rules). If I'm able to find a proof then $\Gamma \models \phi$ holds.
2. Refutation method: Try to derive a contradiction from $\Gamma, \neg\phi$ using axioms and syntactic manipulation rules (reasoning rules). If I'm able to derive the contradiction then $\Gamma \models \phi$ holds.
3. Semantic method: Try to build an interpretation for $\Gamma, \neg\phi$ using axioms and syntactic manipulation rules (reasoning rules). If I'm **not** able to build this interpretation then $\Gamma \models \phi$ hold.

Soundness & Completeness

How can we be sure that we derive exactly what we can logically infer?

Theorem (Soundness)

The syntactic manipulation rules are such that we do not derive "wrong" logical consequences.
If $\Gamma \vdash A$ *then* $\Gamma \models A$.

Theorem (Completeness)

The syntactic manipulation rules are such that we can derive all logical consequences.
If $\Gamma \models A$ *then* $\Gamma \vdash A$.

Different proof methods, but all need to be sound and complete!

Decidability of FOL

Definition

A logical system is decidable if there is an effective method for determining whether arbitrary formulas are logically valid.

- Propositional logic is decidable, because the truth-table method can be used to determine whether an arbitrary propositional formula is logically valid.
- First-order logic is not decidable in general; in particular, the set of logical validities in any signature that includes equality and at least one other predicate with two or more arguments is not decidable.

Tableaux Calculus

- The Tableaux Calculus is an algorithm solving the problem of satisfiability.
- If a formula is satisfiable, then there exists an open branch in the tableaux of this formula.
- the procedure attempts to construct the tableaux for a formula. Sometimes it's not possible since the model of the formula is infinite.
- The basic idea is to incrementally build the model by looking at the formula, by decomposing it in a top/down fashion. The procedure exhaustively looks at all the possibilities, so that it can possibly prove that no model could be found for unsatisfiable formulas.

Tableaux method: Simple example

A model $\mathcal{I}$ for ...

$$\exists x P(x) \wedge \forall x (P(x) \supset Q(a, f(x)))$$

must be such that ...

$$\mathcal{I} \models \exists x P(x) \quad \text{and} \quad \mathcal{I} \models \forall x (P(x) \supset Q(a, f(x)))$$

... which implies that there is a c s.t. ...

$$\mathcal{I} \models P(c)$$

... furthermore, whe have that ...

$$\mathcal{I} \models P(c) \supset Q(a, f(c))$$

... which implies that either ...

$$\mathcal{I} \not\models P(c)$$

... or ...

$$\mathcal{I} \models Q(a, f(c))$$

Tableaux production rules

. . . for propositional connectives

α rules

$$\frac{\phi \wedge \psi}{\begin{array}{c}\phi\\ \psi\end{array}} \qquad \frac{\neg(\phi \vee \psi)}{\begin{array}{c}\neg\phi\\ \neg\psi\end{array}} \qquad \frac{\neg\neg\phi}{\phi} \qquad \frac{\neg(\phi \supset \psi)}{\begin{array}{c}\phi\\ \neg\psi\end{array}}$$

β rules

$$\frac{\phi \vee \psi}{\phi \mid \psi} \qquad \frac{\neg(\phi \wedge \psi)}{\neg\phi \mid \neg\psi} \qquad \frac{\phi \equiv \psi}{\begin{array}{c|c}\phi & \neg\phi\\ \psi & \neg\psi\end{array}} \qquad \frac{\neg(\phi \equiv \psi)}{\begin{array}{c|c}\phi & \neg\phi\\ \neg\psi & \psi\end{array}}$$

$$\frac{\phi \supset \psi}{\neg\phi \mid \psi}$$

Tableaux production rules

...for quantifiers

γ rules	$\dfrac{\forall x.\phi(x)}{\phi[x/t]}$	$\dfrac{\neg\exists x.\phi(x)}{\neg\phi[x/t]}$	Where t is a term free for x in ϕ
δ rules	$\dfrac{\neg\forall x.\phi(x)}{\neg\phi[x/c]}$	$\dfrac{\exists x.\phi(x)}{\phi[x/c]}$	where c is a new constant not previously appearing in the tableaux

Substitution $\phi[x/t]$

Substitution

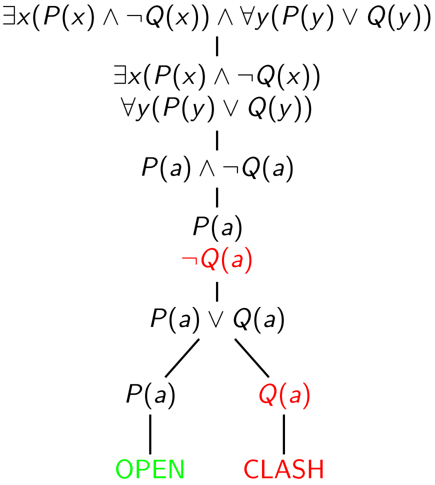
$\phi[x/t]$ denotes the formula we get by replacing each free occurrence of the variable x in the formula ϕ by the term t

Substitution $\phi[x/t]$

Example (of substitution)

$$P(x, y, f(x))[x/a] = P(a, y, f(a))$$
$$\forall x P(x, y)[x/b] = \forall x P(x, y)$$
$$\exists x P(x, x) \wedge Q(x)[x/c] = \exists x P(x, x) \wedge Q(c)$$
$$P(x, g(y))[y/f(x)] = P(x, g(f(x)))$$
$$\forall x.P(x, y)[y/f(x)] = \text{Not allowed since } f(x) \text{ is not free for } y \text{ in } \forall x.P(x, y)$$

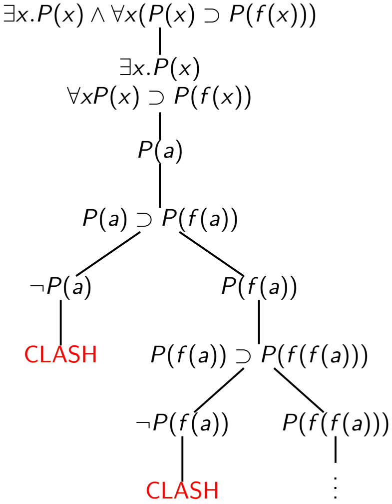
Example of tableaux



Termination

For certain formulas the tableaux can go on for ever generating an infinite tree.

Example:
$\exists x.P(x) \wedge (\forall x P(x) \supset P(f(x)))$



Some definition for tableaux

Definition (Closed branch)

A closed branch is a branch which contains a formula and its negation.

Definition (Open branch)

An open branch is a branch which is not closed

Definition (Closed tableaux)

A tableaux is closed if all its branches are closed.

Definition

Let ϕ be a first-order formula and Γ a finite set of such formulas. We write $\Gamma \vdash \phi$ to say that there exists a closed tableau for $\Gamma \cup \{\neg\phi\}$

Tableaux

Exercise

Give a tableau proof for the following logical consequence:

$$\forall x.P(x) \vee \forall x.Q(x) \models \neg\exists x(\neg P(x) \wedge \neg Q(x))$$

Hint for solution

Try to build a closed tableau for

$$\{\forall x.P(x) \vee \forall x.Q(x)\} \cup \{\neg\neg\exists x(\neg P(x) \wedge \neg Q(x))\}$$

Tableaux

Solution

$$\forall x.P(x) \lor \forall x.Q(x) \land \exists x(\neg P(x) \land \neg Q(x))$$

$$\forall x.P(x) \lor \forall x.Q(x)$$
$$\exists x(\neg P(x) \land \neg Q(x))$$

$$\neg P(a) \land \neg Q(a)$$

$$\neg P(a)$$
$$\neg Q(a))$$

$$\forall x P(x) \qquad \forall x Q(x)$$

$$P(a) \qquad Q(a)$$

CLASH CLASH

Homework

Exercise

Give tableau proofs for the following logical consequence:

$$\models \exists x.(P(x) \vee Q(x)) \equiv \exists x.P(x) \vee \exists x.Q(x)$$

Soundness and completeness

Theorem (Soundness)

$$\Gamma \vdash \phi \quad \Longrightarrow \quad \Gamma \models \phi$$

Theorem (Completeness)

$$\Gamma \models \phi \quad \Longrightarrow \quad \Gamma \vdash \phi$$

Remark

The mere existence of a closed tableau does not mean that we have an effective method to build it! Concretely: we don't know how often and in which way we have to apply the γ-rules ($\forall x \phi(x) \Rightarrow \phi[x/t]$), and what term to use in the substitution.

Apply the γ-rules

Example

Show that $\forall x, y(P(x) \supset Q(y)) \supset (\exists x P(x) \supset \forall y Q(y))$ is valid:

Solution

$$\neg(\forall x, y(P(x) \supset Q(y)) \supset (\exists x P(x) \supset \forall y Q(y)))$$

$$\forall x, y(P(x) \supset Q(y))$$
$$\neg(\exists x P(x) \supset \forall y Q(y))$$

$$\exists x P(x)$$
$$\neg \forall y Q(y)$$

$$P(a)$$

$$\neg Q(b)$$

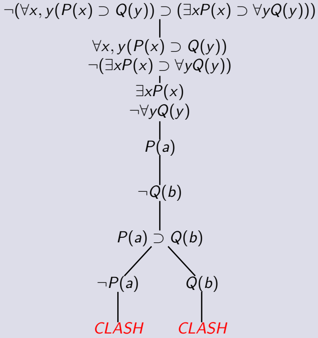
$$P(a) \supset Q(a) \qquad P(b) \supset Q(a) \qquad P(b) \supset Q(b) \qquad P(a) \supset Q(b) \qquad \ldots$$

Apply the γ-rules

Example

Show that $\forall x, y(P(x) \supset Q(y)) \supset (\exists x P(x) \supset \forall y Q(y))$ is valid:

Solution



Construction Strategies and Termination

- From the example in the previous slide we can understand that strategies to expand the tableau are important;
- Nevertheless, because of γ-rules we cannot guarantee the existence of an effective procedure to construct a closed tableau. Thus we may not be able to build a counter-model for a formula in a finite number of steps.
- There are fragments of FOL for which you can guarantee termination. These are for instance Description Logics and you will see them later during the course.