CS4004/CS4504: FORMAL VERIFICATION

Lecture 12: Semantics of First Order Logic

Vasileios Koutavas

17 Nov 2015

School of Computer Science and Statistics
Trinity College Dublin

Semantics of First Order Logic

- Let Γ be a list of formulas $\phi_1, \ldots, \phi_n$
- $\Gamma \vdash \psi$ means that there is a proof of ψ from premises Γ using the natural deduction rules.
- In propositional logic we also defined $\Gamma \models \psi$ to mean that any model (valuation) which makes Γ true, also makes ψ true.
  - we used finite truth-tables as the semantics of formulas

- Let Γ be a list of formulas $\phi_1, \ldots, \phi_n$
- $\Gamma \vdash \psi$ means that there is a proof of ψ from premises Γ using the natural deduction rules.
- In propositional logic we also defined $\Gamma \models \psi$ to mean that any model (valuation) which makes Γ true, also makes ψ true.
  - we used finite truth-tables as the semantics of formulas
- In propositional logic we showed soundness and completeness:
  - $\Gamma \vdash \psi$ iff $\Gamma \models \psi$

- Let Γ be a list of formulas $\phi_1, \ldots, \phi_n$
- $\Gamma \vdash \psi$ means that there is a proof of ψ from premises Γ using the natural deduction rules.
- In propositional logic we also defined $\Gamma \models \psi$ to mean that any model (valuation) which makes Γ true, also makes ψ true.
  - we used finite truth-tables as the semantics of formulas
- In propositional logic we showed soundness and completeness:
  - $\Gamma \vdash \psi$ iff $\Gamma \models \psi$
- How can we define a semantic entailment $\Gamma \models \psi$ in FOL?
  - What is the semantics of formulas?
  - What sort of models can we consider for quantifiers $\forall x.\phi$ and $\exists x.\phi$?

Syntactic entailment is useful to show existence of proofs.

- How can we show that $\Gamma \vdash \psi$?
  - we need to find one syntactic proof

USEFULNESS OF SEMANTICS

Syntactic entailment is useful to show existence of proofs.

- How can we show that $\Gamma \vdash \psi$?
  - we need to find one syntactic proof

Semantic entailment is useful to show absence of proofs.

- How can we show that $\Gamma \nvdash \psi$?
  - we need to consider **all possible (infinite) proofs**
- How can we show that $\Gamma \not\models \psi$?
  - we need to find **one model that makes Γ true and ψ false**.
  - OTOH, proving $\Gamma \models \psi$ is more difficult than proving $\Gamma \vdash \psi$ because we have to consider **all models making Γ true**

Syntactic entailment is useful to show existence of proofs.

- How can we show that $\Gamma \vdash \psi$?
  - we need to find one syntactic proof

Semantic entailment is useful to show absence of proofs.

- How can we show that $\Gamma \nvdash \psi$?
  - we need to consider **all possible (infinite) proofs**
- How can we show that $\Gamma \not\models \psi$?
  - we need to find **one model that makes Γ true and ψ false**.
  - OTOH, proving $\Gamma \models \psi$ is more difficult than proving $\Gamma \vdash \psi$ because we have to consider **all models making Γ true**

Semantics gives us a sanity check of our syntactic logic

- Consider a model of something familiar (e.g. natural numbers)
- Are the provable entailments reasonable theorems for this model?
- A lot of effort has gone into defining models for familiar mathematics
  - natural numbers
  - real numbers
  - set theory
  - ...

In predicate logic $(p \vee q)$ had a finite semantics: a truth-table with four rows, because there were only four models for (p, q):

- $(p \mapsto \texttt{True}, q \mapsto \texttt{True})$, $(p \mapsto \texttt{True}, q \mapsto \texttt{False})$, $(p \mapsto \texttt{False}, q \mapsto \texttt{True})$, $(p \mapsto \texttt{False}, q \mapsto \texttt{False})$.

In predicate logic $(p \vee q)$ had a finite semantics: a truth-table with four rows, because there were only four models for (p, q):

- $(p \mapsto \texttt{True}, q \mapsto \texttt{True}), (p \mapsto \texttt{True}, q \mapsto \texttt{False}),$
  $(p \mapsto \texttt{False}, q \mapsto \texttt{True}), (p \mapsto \texttt{False}, q \mapsto \texttt{False}).$

What should be the semantics of $\forall x. P(x, y)$?

In predicate logic $(p \vee q)$ had a finite semantics: a truth-table with four rows, because there were only four models for (p, q):

- $(p \mapsto \texttt{True}, q \mapsto \texttt{True}), (p \mapsto \texttt{True}, q \mapsto \texttt{False}),$ $(p \mapsto \texttt{False}, q \mapsto \texttt{True}), (p \mapsto \texttt{False}, q \mapsto \texttt{False}).$

What should be the semantics of $\forall x.P(x, y)$?

- It depends on the semantics of the parameters of FOL: the set of terms and predicates

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

$$\phi ::= P(t_1, \ldots, t_n) \mid \ldots$$

  where c, f are from the parameter set $\mathcal{F}$
  - e.g. natural numbers: $zero, succ$

  where P is from the parameter set $\mathcal{P}$
  - e.g. predicates on natural numbers: $(\cdot < \cdot), (\cdot \leq \cdot), (\cdot = \cdot), (\cdot \neq \cdot), \ldots$

Definition

:et $\mathcal{F}$ be a set of functions and $\mathcal{P}$ a set of predicate symbols (with known, fixed arity). A model $\mathcal{M}$ of $(\mathcal{F}, P)$ consists of the following:

1. A non-empty set A: the universe of concrete values.
   - → These are the objects we range over by quantified variables $\forall x / \exists x$
2. for each nullary function $c \in \mathcal{F}$, a concrete element $c^{\mathcal{M}} \in A$
   - → These are the values that correspond to constant terms
3. for each $f \in \mathcal{F}$ with arity $n > 0$, a concrete mathematical function $f^{\mathcal{M}} : A^n \to A$, taking n-tuple of A-values to A-values
   - → These are the functions that correspond to functional terms
4. for each $P \in \mathcal{P}$ with arity $n > o$, a subset $P^{\mathcal{M}} \subset A^n$ of $n - tuples$ over A.
   - → These are the tuples of values that make P true

Natural numbers:

$$\mathcal{F} = \{zero^0, succ^1\} \qquad \mathcal{P} = \{(\cdot < \cdot)^2\}$$

A model $\mathcal{M}$ may be:

1. $A = \{0, 1, 2, \ldots\}$
2. $zero^{\mathcal{M}} \stackrel{\text{def}}{=} 0$
3. $succ^{\mathcal{M}} \stackrel{\text{def}}{=} fun(x) \Rightarrow (x + 1)$
4. $<^{\mathcal{M}} \stackrel{\text{def}}{=} fun(x, y) \Rightarrow (\textit{if x less than y then true else false})$

Natural numbers:

$$\mathcal{F} = \{zero^0, succ^1\} \qquad\qquad \mathcal{P} = \{(\cdot < \cdot)^2\}$$

A model $\mathcal{M}$ may be:

1. $A \stackrel{\text{def}}{=} \{0, 1, 10, 11, 100, \ldots\}$
2. $zero^{\mathcal{M}} \stackrel{\text{def}}{=} 0$
3. $succ^{\mathcal{M}} \stackrel{\text{def}}{=} fun(x) \Rightarrow (x + 1)$
4. $<^{\mathcal{M}} \stackrel{\text{def}}{=} fun(x, y) \Rightarrow (\text{...}binary\ comparison\text{...})$

Natural numbers:

$$\mathcal{F} = \{zero^0, succ^1\} \qquad \mathcal{P} = \{(\cdot < \cdot)^2\}$$

A model $\mathcal{M}$ may be:

1. $A \stackrel{\text{def}}{=} \{$ "0", "0 + 1", "0 + 1 + 1", "0 + 1 + 1 + 1", $\ldots\}$
2. $zero^{\mathcal{M}} \stackrel{\text{def}}{=} 0$
3. $succ^{\mathcal{M}} \stackrel{\text{def}}{=} fun(x) \Rightarrow (x \ concatenate$ " + 1" $)$
4. $<^{\mathcal{M}} \stackrel{\text{def}}{=} fun(x, y) \Rightarrow (x \ isprefixof \ y)$

Models are extremely liberal (e.g., lookup the Church encoding of numerals in the lambda-calculus)

The only mild requirement imposed on all models is that the concrete functions and relations on A-values have the same number of arguments as their syntactic counterparts.

Models should abstract away aspects of the world.

We will give semantics to closed formulas (no free variables) using the semantics of open formulas.

We will give semantics to closed formulas (no free variables) using the semantics of open formulas.

The semantics of $\forall x.\phi$ means that for all values $a \in A$, $\phi[\alpha/x]$ is true.

However it's not a valid syntax to have formulas containing semantic values from a. We need to use environments.

Definition

l is an environment if it is a function that maps syntactic variables to semantic values. (lookup tables)

Definition

Given a model $\mathcal{M}$ for a pair $(\mathcal{F}, \mathcal{P})$ and given an environment l, we define the satisfaction relation $M \models_l \phi$ for each logical formula ϕ over the pair $(\mathcal{F}, \mathcal{P})$ and l as follows.

$\mathcal{M} \models_l P(t_1, \ldots, t_n)$: find the values $a_1, \ldots, a_n$ that correspond to $t_1, \ldots, t_n$, replacing any variable x with $l(x)$. This computes to `True` if $(a_1, \ldots, a_n) \in P^{\mathcal{M}}$

$\mathcal{M} \models_l \forall x.\psi$ computes to `True` if $\mathcal{M} \models_{l,(x \mapsto a)} \psi$ does, for all $a \in A$.

$\mathcal{M} \models_l \exists x.\psi$ computes to `True` if $\mathcal{M} \models_{l,(x \mapsto a)} \psi$ does, for some $a \in A$.

$\mathcal{M} \models_l \phi \vee \psi$ computes to `True` if $\mathcal{M} \models_l \phi$ or $\mathcal{M} \models_l \psi$ does

$\mathcal{M} \models_l \phi \vee \psi$ computes to `True` if $\mathcal{M} \models_l \phi$ and $\mathcal{M} \models_l \psi$ does

$\mathcal{M} \models_l \neg\psi$ computes to `True` if $\mathcal{M} \models_l \psi$ does not

$\mathcal{M} \models_l \phi \to \psi$ computes to `True` if $\mathcal{M} \models_l \psi$ does whenever $\mathcal{M} \models_l \phi$ does

Definition

$\phi_1, \ldots, \phi_n \models \psi$ if for all models $\mathcal{M}$ and environments l for which

$$\mathcal{M} \models_l \phi_1 \qquad \ldots \qquad \mathcal{M} \models_l \phi_n$$

we have $\mathcal{M} \models_l \psi$.

Definition

$\phi_1, \ldots, \phi_n \models \psi$ if for all models $\mathcal{M}$ and environments l for which

$$\mathcal{M} \models_l \phi_1 \qquad \ldots \qquad \mathcal{M} \models_l \phi_n$$

we have $\mathcal{M} \models_l \psi$.

* The symbol $\models$ is overloaded.

Definition

$\phi_1, \ldots, \phi_n \models \psi$ if for all models $\mathcal{M}$ and environments l for which

$$\mathcal{M} \models_l \phi_1 \qquad \ldots \qquad \mathcal{M} \models_l \phi_n$$

we have $\mathcal{M} \models_l \psi$.

* The symbol $\models$ is overloaded.

* The above semantic entailment is able to express properties that are true in all models, no matter how (un-)reasonable. For example:

$$1 < 2 \not\models 2 > 1$$

because there are models with the above symbols which don't have the "right" properties of $(<)$ and $(>)$. (Remember there are very few requirements for a model $\mathcal{M}$).

Definition

$\phi_1, \ldots, \phi_n \models \psi$ if for all models $\mathcal{M}$ and environments l for which

$$\mathcal{M} \models_l \phi_1 \qquad \ldots \qquad \mathcal{M} \models_l \phi_n$$

we have $\mathcal{M} \models_l \psi$.

* The symbol $\models$ is overloaded.

* The above semantic entailment is able to express properties that are true in all models, no matter how (un-)reasonable. For example:

$$1 < 2 \not\models 2 > 1$$

because there are models with the above symbols which don't have the "right" properties of $(<)$ and $(>)$. (Remember there are very few requirements for a model $\mathcal{M}$).

How can we compare provability ($\vdash$) with semantic entailment ($\models$)?

Results about FOL

Theorem (Soundness)

For a given $(\mathcal{F}, \mathcal{P})$*, if* $\vdash \phi$ *then* $\models \phi$
which means for any model $\mathcal{M}$ *of* $(\mathcal{F}, \mathcal{P})$ *and any environment l,* $\mathcal{M} \models_l \phi$.

Theorem (Strong soundness)

For a given $(\mathcal{F}, \mathcal{P})$*, if* $\Gamma \vdash \psi$ *then* $\Gamma \models \phi$
which means for any model $\mathcal{M}$ *of* $(\mathcal{F}, \mathcal{P})$ *and any l, if* $\mathcal{M} \models_l \Gamma$ *then* $\mathcal{M} \models_l \phi$.

Theorem (Soundness)

For a given $(\mathcal{F}, \mathcal{P})$*, if* $\vdash \phi$ *then* $\models \phi$
which means for any model $\mathcal{M}$ *of* $(\mathcal{F}, \mathcal{P})$ *and any environment* l*,* $\mathcal{M} \models_l \phi$*.*

Theorem (Strong soundness)

For a given $(\mathcal{F}, \mathcal{P})$*, if* $\Gamma \vdash \psi$ *then* $\Gamma \models \phi$
which means for any model $\mathcal{M}$ *of* $(\mathcal{F}, \mathcal{P})$ *and any* l*, if* $\mathcal{M} \models_l \Gamma$ *then* $\mathcal{M} \models_l \phi$*.*

This involves properties that are true in all models. How can we talk about properties of certain models (e.g., numbers with some standard predicates over them)?

Theorem (Soundness)

For a given $(\mathcal{F}, \mathcal{P})$*, if* $\vdash \phi$ *then* $\models \phi$
which means for any model $\mathcal{M}$ *of* $(\mathcal{F}, \mathcal{P})$ *and any environment l,* $\mathcal{M} \models_l \phi$*.*

Theorem (Strong soundness)

For a given $(\mathcal{F}, \mathcal{P})$*, if* $\Gamma \vdash \psi$ *then* $\Gamma \models \phi$
which means for any model $\mathcal{M}$ *of* $(\mathcal{F}, \mathcal{P})$ *and any l, if* $\mathcal{M} \models_l \Gamma$ *then* $\mathcal{M} \models_l \phi$*.*

This involves properties that are true in all models. How can we talk about properties of certain models (e.g., numbers with some standard predicates over them)?

A: Encode the necessary properties of these models in Γ. Γ can contain the axioms we want to hold in these models.

- → famous axiomatisation of natural numbers: Peano axioms

Terms: $\mathcal{F} = \{O^0, S^1\}$

Axioms:

- The reflexive, symmetric and transitive properties of equality
- $\forall x. \neg(S(x) = O)$
- $\forall x. \forall y. (S(x) = S(y) \rightarrow x = y)$
- $\forall x. (x + O = O)$
- $\forall x. (x \cdot O = O)$
- A countably infinite set of axioms to do induction over numbers:

$$\forall \vec{y}. (\ \ \phi(O, \vec{y}) \wedge (\forall x. (\phi(x, \vec{y}) \rightarrow \phi(S(x), \vec{y}))) \quad \rightarrow \quad \forall x. \phi(x, \vec{y})\ \ )$$

  one such axiom for every $\phi \in \mathcal{P}$ with the right number of arguments. Here $\vec{y}$ means $y_1, \ldots, y_n$ for some value of n (this value is determined by the arity of ϕ).

Terms: $\mathcal{F} = \{O^0, S^1\}$

Axioms:

- The reflexive, symmetric and transitive properties of equality
- $\forall x. \neg(S(x) = O)$
- $\forall x. \forall y. (S(x) = S(y) \rightarrow x = y)$
- $\forall x. (x + O = O)$
- $\forall x. (x \cdot O = O)$
- A countably infinite set of axioms to do induction over numbers:

$$\forall \vec{y}. (\ \ \phi(O, \vec{y}) \wedge (\forall x. (\phi(x, \vec{y}) \rightarrow \phi(S(x), \vec{y}))) \quad \rightarrow \quad \forall x. \phi(x, \vec{y}) \ \ )$$

one such axiom for every $\phi \in \mathcal{P}$ with the right number of arguments. Here $\vec{y}$ means $y_1, \ldots, y_n$ for some value of n (this value is determined by the arity of ϕ).

*Russell and others agreed that Peano axioms encode what we mean by "natural numbers".

Theorem (Incompleteness)

Any set of axioms Γ *which is consistent (no contradictions such as* $0 = 1$ *are derivable) and contains "enough arithmetic" cannot be complete.*

That is, there are true facts ϕ *about arithmetic for which* $\Gamma \nvdash \phi$*.*

Theorem (Incompleteness)

Any set of axioms Γ *which is consistent (no contradictions such as* $0 = 1$ *are derivable) and contains "enough arithmetic" cannot be complete.*

That is, there are true facts ϕ *about arithmetic for which* $\Gamma \nvdash \phi$*.*

In other words it is not possible to formalise mathematics in logic.

This broke Russell's (and other's) lifelong dream of making mathematics entirely unambiguous.

Theorem (Incompleteness)

Any set of axioms Γ *which is consistent (no contradictions such as* $0 = 1$ *are derivable) and contains "enough arithmetic" cannot be complete.*

That is, there are true facts ϕ *about arithmetic for which* $\Gamma \nvdash \phi$*.*

In other words it is not possible to formalise mathematics in logic.

This broke Russell's (and other's) lifelong dream of making mathematics entirely unambiguous.

Proof.

Göedel gave a way to encode first-order logic itself in any axiomatisation Γ containing Peano (or any other encoding of) natural numbers.

Hence for any such system he was able to write an encoding of the formula

$\phi \stackrel{\text{def}}{=}$ "ϕ is not provable in the logic."

If $\Gamma \vdash \phi$ then obviously the logic is inconsistent ($\Gamma \vdash \phi \wedge \neg\phi$).

If $\Gamma \nvdash \phi$ then obviously the logic is incomplete (ϕ is true but not provable). $\square$

Theorem (Completeness)

$\models \phi$ then $\vdash \phi$.

* This means that anything that is true for all models, can be syntactically proven. Examples:

Theorem (Completeness)

$\models \phi$ then $\vdash \phi$.

* This means that anything that is true for all models, can be syntactically proven. Examples:

→ Yes: $\neg\forall x.\phi(x) \rightarrow \exists x.\neg\phi(x)$

Theorem (Completeness)

$\models \phi$ *then* $\vdash \phi$.

* This means that anything that is true for all models, can be syntactically proven. Examples:

- Yes: $\neg\forall x.\phi(x) \rightarrow \exists x.\neg\phi(x)$
- No: $1 + 1 = 2$

Theorem (Completeness)

$\models \phi$ then $\vdash \phi$.

* This means that anything that is true for all models, can be syntactically proven. Examples:

- Yes: $\neg\forall x.\phi(x) \rightarrow \exists x.\neg\phi(x)$
- No: $1 + 1 = 2$
- No: The Goldbach conjecture: "Every even integer greater than 2 can be expressed as the sum of two primes"