Logic: first-order logic

Zahoor Tanoli (PhD)

Limitations of propositional logic

Alice and Bob both know arithmetic.

$\text{AliceKnowsArithmetic} \wedge \text{BobKnowsArithmetic}$

All students know arithmetic.

$\text{AliceIsStudent} \rightarrow \text{AliceKnowsArithmetic}$

$\text{BobIsStudent} \rightarrow \text{BobKnowsArithmetic}$

...

Every even integer greater than 2 is the sum of two primes.

???

- If the goal of logic is to be able to express facts in the world in a compact way, let us ask ourselves if propositional logic is enough.
- Some facts can be expressed in propositional logic, but it is very clunky, having to instantiate many different formulas. Others simply can’t be expressed at all, because we would need to use an infinite number of formulas.

Limitations of propositional logic

All students know arithmetic.

AliceIsStudent $\rightarrow$ AliceKnowsArithmetic

BobIsStudent $\rightarrow$ BobKnowsArithmetic

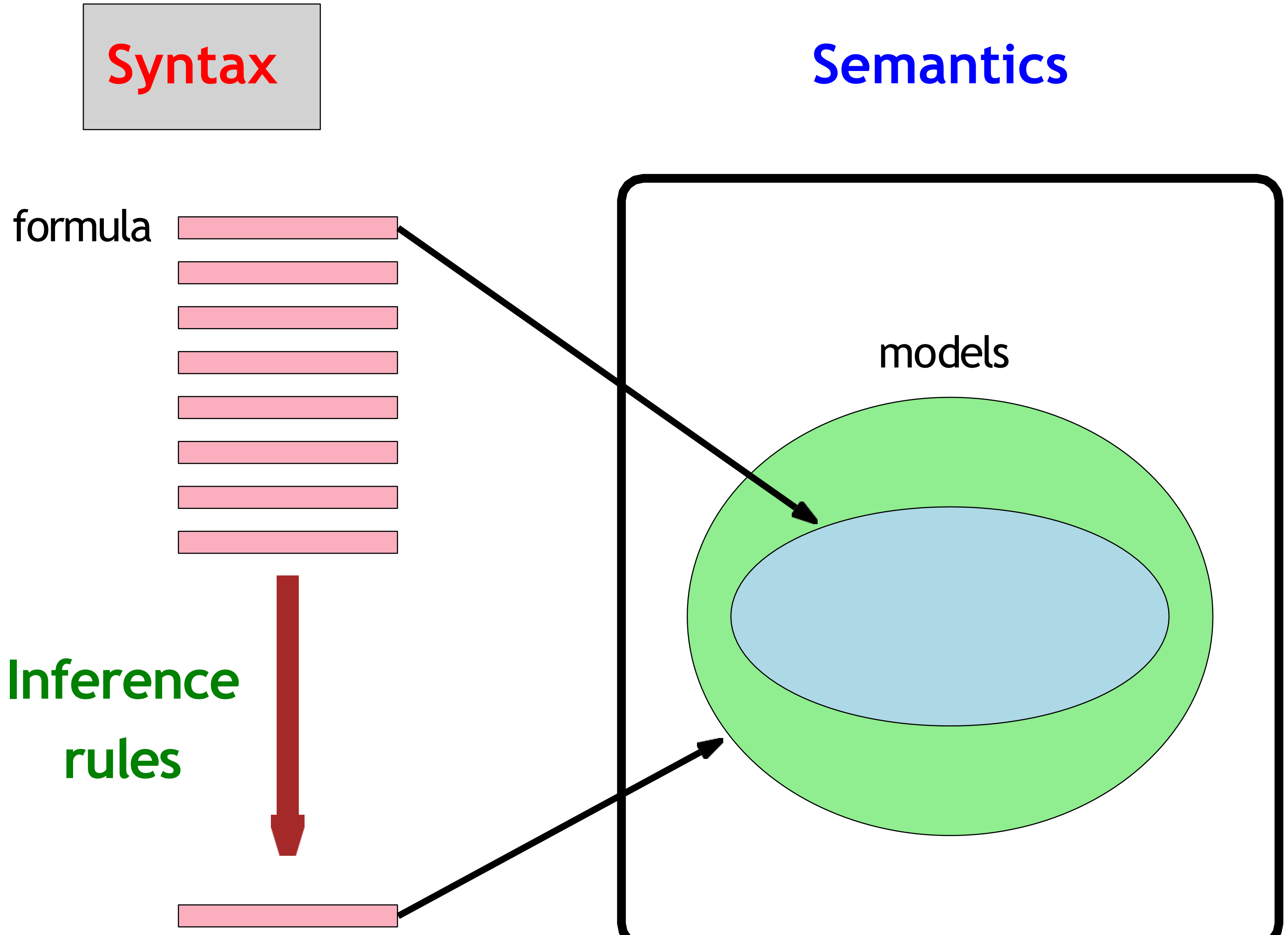
...

Propositional logic is very clunky. What's missing?

- Objects and predicates: propositions (e.g., AliceKnowsArithmetic) have more internal structure (alice, Knows, arithmetic)
- Quantifiers and variables: *all* is a quantifier which applies to each person, don't want to enumerate them all...

- What’s missing? The key conceptual observation is that the world is not just a bunch of atomic facts, but that each fact is actually made out of **objects** and **predicates** on those objects.
- Once facts are decomposed in this way, we can use **quantifiers** and **variables** to implicitly define a huge (and possibly infinite) number of facts with one compact formula. Again, where logic excels is the ability to represent complex things via simple means.

First-order logic



- We will now introduce **first-order logic**, which will address the representational limitations of propositional logic.
- Remember to define a logic, we need to talk about its syntax, its semantics (interpretation function), and finally inference rules that we can use to operate on the syntax.

First-order logic: examples

Alice and Bob both know arithmetic.

$$\text{Knows}(\text{alice}, \text{arithmetic}) \wedge \text{Knows}(\text{bob}, \text{arithmetic})$$

All students know arithmetic.

$$\forall x \; \text{Student}(x) \rightarrow \text{Knows}(x, \text{arithmetic})$$

- Before formally defining things, let's look at two examples. First-order logic is basically propositional logic with a few more symbols.

Syntax of first-order logic

Terms (refer to objects):

- Constant symbol (e.g., arithmetic)
- Variable (e.g., x)
- Function of terms (e.g., $\text{Sum}(3, x)$)

Formulas (refer to truth values):

- Atomic formulas (atoms): predicate applied to terms (e.g., $\text{Knows}(x, \text{arithmetic})$)
- Connectives applied to formulas (e.g., $\text{Student}(x) \to \text{Knows}(x, \text{arithmetic})$)
- Quantifiers applied to formulas (e.g., $\forall x\, \text{Student}(x) \to \text{Knows}(x, \text{arithmetic})$)

- In propositional logic, everything was a formula (or a connective). In first-order logic, there are two types of beasts: terms and formulas.
- There are three types of terms:
  - constant symbols (which refer to specific objects),
  - variables (which refer to some unspecified object to be determined by quantifiers),
  - functions (which is a function applied to a set of arguments which are themselves terms).
- Given the terms, we can form atomic formulas, which are the analogue of propositional symbols, but with internal structure (e.g., terms).
- From this point, we can apply the same connectives on these atomic formulas, as we applied to propositional symbols in propositional logic.
- At this level, first-order logic looks very much like propositional logic.
  - Finally, to make use of the fact that atomic formulas have internal structure, we have **quantifiers**, which are really the whole point of first-order logic!

Quantifiers

Universal quantification ($\forall$):

Think conjunction: $\forall x\ P(x)$ is like $P(A) \wedge P(B) \wedge \cdots$

Existential quantification ($\exists$):

Think disjunction: $\exists x\ P(x)$ is like $P(A) \vee P(B) \vee \cdots$

Some properties:

- $\neg\forall x\ P(x)$ equivalent to $\exists x\ \neg P(x)$

Natural language quantifiers

Universal quantification (∀):

Every student knows arithmetic.

$\forall x\ \text{Student}(x) \rightarrow \text{Knows}(x, \text{arithmetic})$

Existential quantification (∃):

Some student knows arithmetic.

$\exists x\ \text{Student}(x) \wedge \text{Knows}(x, \text{arithmetic})$

Note the different connectives!

- Universal and existential quantifiers naturally correspond to the words *every* and *some*, respectively. But when converting English to formal logic, one must exercise caution.
- *Every* can be thought of as taking two arguments P and Q (e.g., *student* and *knows arithmetic*). The connective between P and Q is an implication (not conjunction, which is a common mistake). This makes sense because when we talk about every P , we are only restricting our attention to objects x for which $P(x)$ is true. Implication does exactly that.
- On the other hand, the connective for existential quantification is conjunction, because we're asking for an object x such that $P(x)$ and $Q(x)$
  both hold.

Some examples of first-order logic

There is some course that every student has taken.

$$\exists\, y\, \mathrm{Course}(y) \wedge [\forall x\, \mathrm{Student}(x) \rightarrow \mathrm{Takes}(x, y)]$$

Every even integer greater than 2 is the sum of two primes.

$$\forall\, x\, \mathrm{EvenInt}(x) \wedge \mathrm{Greater}(x, 2) \rightarrow \exists\, y\, \exists\, z\, \mathrm{Equals}(x, \mathrm{Sum}(y, z)) \wedge \mathrm{Prime}(y) \wedge \mathrm{Prime}(z)$$

If a student takes a course and the course covers a concept, then the student knows that concept.

$$\forall\, x\, \forall\, y\, \forall\, z\, (\mathrm{Student}(x) \wedge \mathrm{Takes}(x, y) \wedge \mathrm{Course}(y) \wedge \mathrm{Covers}(y, z)) \rightarrow \mathrm{Knows}(x, z)$$

Some examples of first-order logic

All man drink coffee.

Some boys are intelligent.

All birds fly.

Every man respects his parent.

Some boys play cricket.

Solution

∀x man(x) → drink (x, coffee).

∃x: boys(x) ∧ intelligent(x)

∀x bird(x) →fly(x).

∀x man(x) → respects (x, parent).

∃x boys(x) → play(x, cricket)