First-order logic

• First-order logic (FOL) models the world in terms of
  – Objects, which are things with individual identities
  – Properties of objects that distinguish them from other objects
  – Relations that hold among sets of objects
  – Functions, which are a subset of relations where there is only one “value” for any given “input”

• Examples:
  – Objects: Students, lectures, companies, cars ...
  – Relations: Brother-of, bigger-than, outside, part-of, has-color, occurs-after, owns, visits, precedes, ...
  – Properties: blue, oval, even, large, ...
  – Functions: father-of, best-friend, second-half, one-more-than ...

Syntax of FOL

\[
S ::= \langle\text{Sentence}\rangle; \\
\langle\text{Sentence}\rangle ::= \langle\text{AtomicSentence}\rangle | \langle\text{Sentence}\rangle \langle\text{Connective}\rangle \langle\text{Sentence}\rangle | \langle\text{Quantifier}\rangle \langle\text{Variable}\rangle,... \langle\text{Sentence}\rangle | \text{"NOT" } \langle\text{Sentence}\rangle | \langle\text{ Predicate}\rangle \langle\text{Term}\rangle | \langle\text{Term}\rangle = \langle\text{Term}\rangle; \\
\langle\text{AtomicSentence}\rangle ::= \langle\text{Predicate}\rangle \langle\text{Term}\rangle | \langle\text{Term}\rangle = \langle\text{Term}\rangle; \\
\langle\text{Term}\rangle ::= \langle\text{Function}\rangle \langle\text{Term}\rangle | \langle\text{Constant}\rangle | \langle\text{Variable}\rangle; \\
\langle\text{Connective}\rangle ::= \text{"AND"} | \text{"OR"} | \text{"IMPLIES"} | \text{"EQUIVALENT"}; \\
\langle\text{Quantifier}\rangle ::= \text{"EXISTS"} | \text{"FORALL"}; \\
\langle\text{Constant}\rangle ::= \text{"A"} | \text{"X1"} | \text{"John"} | ...; \\
\langle\text{Variable}\rangle ::= \text{"a"} | \text{"x"} | \text{"s"} | ...; \\
\langle\text{Predicate}\rangle ::= \text{"Before"} | \text{"HasColor"} | \text{"Raining"} | ...; \\
\langle\text{Function}\rangle ::= \text{"Mother"} | \text{"LeftLegOf"} | ...;
\]

Constants, Functions, Predicates

• **Constant symbols**, which represent individuals in the world
  – Mary
  – 3
  – Green

• **Function symbols**, which map individuals to individuals
  – father-of(Mary) = John
  – color-of(Sky) = Blue

• **Predicate symbols**, which map individuals to truth values
  – greater(5,3)
  – green(Grass)
  – color(Grass, Green)

Variables, Connectives, Quantifiers

• **Variable symbols**
  – E.g., x, y, foo

• **Connectives**
  – Same as in PL: not (\(\sim\)), and (\(\land\)), or (\(\lor\)), implies (\(\rightarrow\)), if and only if (biconditional \(\leftrightarrow\))

• **Quantifiers**
  – Universal \(\forall x\) or (\(Ax\))
  – Existential \(\exists x\) or (\(Ex\))
Quantifiers

- **Universal quantification**
  - $(\forall x) P(x)$ means that $P$ holds for all values of $x$ in the domain associated with that variable
  - E.g., $(\forall x) \text{dolphin}(x) \rightarrow \text{mammal}(x)$

- **Existential quantification**
  - $(\exists x) P(x)$ means that $P$ holds for some value of $x$ in the domain associated with that variable
  - E.g., $(\exists x) \text{mammal}(x) \land \text{lays-eggs}(x)$
  - Permits one to make a statement about some object without naming it

Translating English to FOL

**Every gardener likes the sun.**

$(\forall x) \text{gardener}(x) \rightarrow \text{likes}(x, \text{Sun})$

**You can fool some of the people all of the time.**

$(\exists x)(\forall t) (\text{person}(x) \land \text{time}(t)) \rightarrow \text{can-fool}(x,t)$

**You can fool all of the people some of the time.**

$(\forall x)(\exists t) (\text{person}(x) \land \text{time}(t)) \rightarrow \text{can-fool}(x,t)$

**All purple mushrooms are poisonous.**

$(\forall x) (\text{mushroom}(x) \land \text{purple}(x)) \rightarrow \text{poisonous}(x)$

**No purple mushroom is poisonous.**

$\neg (\exists x) (\text{purple}(x) \land \text{mushroom}(x)) \lor \text{poisonous}(x)$

**There are exactly two purple mushrooms.**

$(\exists x)(\exists y) (\text{mushroom}(x) \land \text{purple}(x) \land \text{purple}(y) \land \neg (x = y) \land \neg (y = z))$

**Clinton is not tall.**

$\neg \text{tall}(\text{Clinton})$

**X is above Y if X is on directly on top of Y or there is a pile of one or more other objects directly on top of one another starting with X and ending with Y.**

$(\forall x)(\forall y) (\text{above}(x,y) \lor (\exists z) (\text{on}(x,z) \land \text{above}(z,y)))$

Sentences are built from terms and atoms

- A **term** (denoting a real-world individual) is a constant symbol, a variable symbol, or an n-place function of n terms.
  - x and $f(x_1, \ldots, x_n)$ are terms, where each $x_i$ is a term.
  - A term with no variables is a **ground term**

- An **atom** (which has value true or false) is either
  - an n-place predicate of n terms, or,
  - $\neg P$, $P \lor Q$, $P \land Q$, $P \rightarrow Q$, $P \leftrightarrow Q$ where $P$ and $Q$ are atoms

- A **sentence** is an atom, or, if $P$ is a sentence and $x$ is a variable, then $(\forall x)P$ and $(\exists x)P$ are sentences

- A **well-formed formula (wff)** is a sentence containing no “free” variables. That is, all variables are “bound” by universal or existential quantifiers.
  - $(\forall x)P(x,y)$ has x bound as a universally quantified variable, but y is free.

Quantifiers

- Universal quantifiers are often used with “implies” to form “rules”:
  - $(\forall x) \text{student}(x) \rightarrow \text{smart}(x)$ means “All students are smart”

- Universal quantification is rarely used to make blanket statements about every individual in the world:
  - $(\forall x) \text{student}(x) \land \text{smart}(x)$ means “Everyone in the world is a student and is smart”

- Existential quantifiers are usually used with “and” to specify a list of properties about an individual:
  - $(\exists x) \text{student}(x) \land \text{smart}(x)$ means “There is a student who is smart”

- A common mistake is to represent this English sentence as the FOL sentence:
  - $(\exists x) \text{student}(x) \rightarrow \text{smart}(x)$
    - But what happens when there is a person who is *not* a student?
Quantifier Scope

• Switching the order of universal quantifiers does not change the meaning:
  \[(\forall x)(\forall y)P(x,y) \Rightarrow (\forall y)(\forall x)P(x,y)\]
• Similarly, you can switch the order of existential quantifiers:
  \[(\exists x)(\exists y)P(x,y) \Rightarrow (\exists y)(\exists x)P(x,y)\]
• Switching the order of universals and existentials does change meaning:
  – Everyone likes someone: \[(\forall x)(\exists y)\text{likes}(x,y)\]
  – Someone is liked by everyone: \[(\exists y)(\forall x)\text{likes}(x,y)\]

Connections between All and Exists

We can relate sentences involving \(\forall\) and \(\exists\) using De Morgan’s laws:
\[
(\forall x) \neg P(x) \Rightarrow (\neg(\exists x)P(x))
\]
\[
\neg(\forall x)P \Rightarrow (\exists x)\neg P(x)
\]
\[
(\forall x)P(x) \Rightarrow (\exists x)\neg P(x)
\]
\[
(\exists x)P(x) \Rightarrow (\forall x)\neg P(x)
\]

Quantified inference rules

• Universal instantiation
  – \(\forall x P(x) \Rightarrow P(A)\)
• Universal generalization
  – \(P(A) \land P(B) \ldots \Rightarrow \forall x P(x)\)
• Existential instantiation
  – \(\exists x P(x) \Rightarrow P(F) \quad \text{skolem constant } F\)
• Existential generalization
  – \(P(A) \Rightarrow \exists x P(x)\)

An example from Monty Python

• FIRST VILLAGER: We have found a witch. May we burn her?
• ALL: A witch! Burn her!
• BEDEVERE: Why do you think she is a witch?
• SECOND VILLAGER: She turned me into a newt.
• B: A newt?
• V2 (after looking at himself for some time): I got better.
• ALL: Burn her anyway.
• B: Quiet! Quiet! There are ways of telling whether she is a witch.
Monty Python cont.

• B: Tell me… what do you do with witches?
• ALL: Burn them!
• B: And what do you burn, apart from witches?
• V4: …wood?
• B: So why do witches burn?
• V2 (pianissimo): because they’re made of wood?
• B: Good.
• ALL: I see. Yes, of course.

Monty Python cont.

• B: So how can we tell if she is made of wood?
• V1: Make a bridge out of her.
• B: Ah… but can you not also make bridges out of stone?
• ALL: Yes, of course… um… er…
• B: Does wood sink in water?
• ALL: No, no, it floats. Throw her in the pond.
• B: Wait. Wait… tell me, what also floats on water?
• ALL: Bread? No, no no. Apples… gravy… very small rocks…
• B: No, no, no,

Monty Python cont.

• KING ARTHUR: A duck!
• (They all turn and look at Arthur. Bedevere looks up, very impressed.)
• B: Exactly. So… logically…
• V1 (beginning to pick up the thread): If she… weighs the same as a duck… she’s made of wood.
• B: And therefore?
• ALL: A witch!

Monty Python cont.

• ∀x witch(x) → burns(x)
• ∀x wood(x) → burns(x)
• ------------------------
• ∀z witch(x) → wood(x)

• p → q
• r → q
• --------
• p → r          Fallacy: Affirming the conclusion
Monty Python Near-Fallacy #2

- \( \text{wood}(x) \rightarrow \text{bridge}(x) \)
- -----------------------------
- \( \therefore \text{bridge}(x) \rightarrow \text{wood}(x) \)

- B: Ah… but can you not also make bridges out of stone?

Monty Python Fallacy #3

- \( \forall x \text{ wood}(x) \rightarrow \text{floats}(x) \)
- \( \forall x \text{ duck-weight (x)} \rightarrow \text{floats}(x) \)
- -----------------------------
- \( \therefore \forall x \text{ duck-weight}(x) \rightarrow \text{wood}(x) \)

- \( p \rightarrow q \)
- \( r \rightarrow q \)
- ---------
- \( \therefore r \rightarrow p \)

Monty Python Fallacy #4

- \( \forall z \text{ light}(z) \rightarrow \text{wood}(z) \)
- \( \text{light}(W) \)
- -----------------------------
- \( \therefore \text{wood}(W) \)

- witch(W) \( \rightarrow \text{wood}(W) \)
  - applying universal instan.
    - to fallacious conclusion #1

- \( \text{wood}(W) \)
- -----------------------------
- \( \therefore \text{witch}(z) \)

Axioms for Set Theory in FOL

1. The only sets are the empty set and those made by adjoining something to a set:
   \( \forall s \text{ set}(s) \leftrightarrow (s=\text{EmptySet}) \lor (\exists r \text{ Set}(r) \land s=\text{Adjoin}(s,r)) \)
2. The empty set has no elements adjoined to it:
   \( \neg \exists x, s \text{ Adjoin}(x,s)=\text{EmptySet} \)
3. Adjoining an element already in the set has no effect:
   \( \forall x, s \text{ Member}(x,s) \leftrightarrow s=\text{Adjoin}(x,s) \)
4. The only members of a set are the elements that were adjoined into it:
   \( \forall x, s \text{ Member}(x,s) \leftrightarrow \exists y, r (s=\text{Adjoin}(y,r) \land (x=y \lor \text{Member}(x,r))) \)
5. A set is a subset of another iff all of the 1st set’s members are members of the 2nd:
   \( \forall s, r \text{ Subset}(s,r) \leftrightarrow (\forall x \text{ Member}(x,s) \rightarrow \text{Member}(x,r)) \)
6. Two sets are equal iff each is a subset of the other:
   \( \forall s, r (s=r) \leftrightarrow (\text{Subset}(s,r) \land \text{Subset}(r,s)) \)
7. Intersection
   \( \forall x, s_1, s_2 \text{ member}(X, \text{intersection}(S_1, S_2)) \leftrightarrow \text{member}(X, s_1) \lor \text{member}(X, s_2) \)
8. Union
   \( \exists x, s_1, s_2 \text{ member}(X, \text{union}(s_1, s_2)) \leftrightarrow \text{member}(X, s_1) \lor \text{member}(X, s_2) \)
Axioms, definitions and theorems

- **Axioms** are facts and rules that attempt to capture all of the (important) facts and concepts about a domain; axioms can be used to prove **theorems**
  - Mathematicians don’t want any unnecessary (dependent) axioms
  - Dependent axioms can make reasoning faster, however
  - Choosing a good set of axioms for a domain is a kind of design problem

- A **definition** of a predicate is of the form “p(X) _ …” and can be decomposed into two parts
  - Necessary description: “p(x) → …”
  - Sufficient description “p(x) ← …”
  - Some concepts don’t have complete definitions (e.g., person(x))

Extensions to FOL

- **Higher-order logic**
  - Quantify over relations
- **Representing functions with the lambda operator (λ)**
- **Expressing uniqueness \( \exists! \), \( \iota \)**
- **Sorted logic**

Higher-order logic

- In FOL, variables can only range over objects
- HOL allows us to quantify over relations
- More expressive, but undecidable
- Example:
  - “two functions are equal iff they produce the same value for all arguments”
  - \( \forall f \forall g (f = g) \Rightarrow (\forall x f(x) = g(x)) \)
- Example:
  - \( \forall r \mathrm{transitive}( r ) \Rightarrow (\forall x \forall y \forall z \ r(x,y) \land r(y,z) \Rightarrow r(x,z)) \)

Expressing uniqueness

- Sometimes we want to say that there is a single, unique object that satisfies a certain condition
- “There exists a unique x such that king(x) is true”
  - \( \exists x \mathrm{king}(x) \land \forall y (\mathrm{king}(y) \Rightarrow x = y) \)
  - \( \exists x \mathrm{king}(x) \land \neg \exists y (\mathrm{king}(y) \land x \neq y) \)
  - \( \exists! x \mathrm{king}(x) \)
- “Every country has exactly one ruler”
  - \( \forall c \mathrm{country}(c) \Rightarrow \exists! r \mathrm{ruler}(c,r) \)
- Iota operator: “\( \iota x \mathrm{P}(x) \)” means “the unique x such that p(x) is true”
  - “The unique ruler of Freedonia is dead”
  - dead(\( \iota x \mathrm{ruler}(\text{Freedonia},x) \))
Notational differences

• Different symbols for and, or, not, implies, ...
  – ∀ ∃ ⇒ ⇔ ∧ ∨ ¬ • ⊆
  – p v (q ∧ r)
  – p + (q * r)
  – etc

• Prolog
  cat(X) :- furry(X), meows (X), has(X, claws)

• Lispy notations
  (forall ?x (implies (and (furry ?x)
                        (meows ?x)
                        (has ?x claws)))
   (cat ?x)))