First-Order Logic

Pros and Cons of Propositional Logic

- We can already do a lot with propositional logic
  - Propositional logic is declarative
  - Propositional logic allows partial/disjunctive/negated information
  - Propositional logic is compositional
  - Meaning in propositional logic is context-independent
- But it is unpleasant that we cannot access the structure of atomic sentences
  - Atomic formulas of propositional logic are too atomic
  - They are just statements which my be true or false but which have no internal structure
  - Propositional logic assumes the world contains facts

First-Order Logic: Basic Ideas

- In First Order Logic (FOL) the atomic formulas are interpreted as statements about *relationships* between objects
- FOL (like natural language) assumes the world contains:

Objects:	people, houses, numbers, colors, baseball games, wars, ...
Relations:	red, round, prime, brother of, bigger than, part of, comes between, ...
Functions:	father of, best friend, one more than, plus, ...

Predicates and Constants

- Let's consider the statements:

  Mary is female
  John is male
  Mary and John are siblings

- In propositional logic the above statements are atomic propositions:

  MaryIsFemale
  JohnIsMale
  MaryAndJohnAreSiblings

- In FOL atomic statements use predicates, with constants as argument

  Female(mary)
  Male(john)
  Siblings(mary, john)

Variables and Quantifiers

- Let's consider the statements:

  Everybody is male or female
  A male is not a female

- In FOL predicates may have variables as arguments, whose value is bounded by quantifiers

$$\forall x.\mathrm{Male}(x) \lor \mathrm{Female}(x)$$
$$\forall x.\mathrm{Male}(x) \rightarrow \neg\mathrm{Female}(x)$$

- Deduction (why?):
  - Mary is not male
  - i.e., $\neg\mathrm{Male}(mary)$

Functions

- Let's consider the statement:

  The father of a person is male

- In FOL objects of the domain may be denoted by functions applied to (other) objects:

$$\forall x.\mathrm{Male}(\mathrm{father}(x))$$

Syntax of FOL: atomic sentences

- Countably infinite supply of symbols (signature):
  - variable symbols: $x, y, z, ...$
  - n-ary function symbols: $f, g, h, ...$
  - individual constants: $a, b, c, ...$
  - n-ary predicate symbols: $P, Q, R, ...$

Term:

$$\begin{array}{rcll} t & \longrightarrow & x & \text{(variable)} \\ & | & a & \text{(constant)} \\ & | & f(t_1, ..., t_n) & \text{(function application)} \end{array}$$

Ground Term: terms that do not contain variables

Atomic Formula:

$$\alpha \quad \longrightarrow \quad P(t_1, ..., t_n) \quad \text{(atomic formula)}$$

Ground Atom: Atom that does not contain variables

Examples

Term: father(x), +(x, y)

Ground Term: father(john), +(2, 3)

Atom: Loves(john, x)

Ground Atom: Loves(john, mary)

Syntax of FOL

Formula:

$$
\begin{array}{lcll}
\alpha, \beta & \longrightarrow & P(t_1, ..., t_n) & \text{(atomic formula)} \\
& | & \bot & \text{(false)} \\
& | & \top & \text{(true)} \\
& | & \neg\alpha & \text{(negation)} \\
& | & \alpha \wedge \beta & \text{(conjunction)} \\
& | & \alpha \vee \beta & \text{(disjunction)} \\
& | & \alpha \rightarrow \beta & \text{(implication)} \\
& | & \alpha \leftrightarrow \beta & \text{(equivalence)} \\
& | & \forall x.\alpha & \text{(universal quantification)} \\
& | & \exists x.\alpha & \text{(existential quantification)}
\end{array}
$$

Ground Formula: Formula that does not contain variables

Examples:

- Everyone in Italy is smart:
  $\forall x.\mathrm{In}(x, \mathrm{italy}) \rightarrow \mathrm{Smart}(x)$
- Someone in France is smart:
  $\exists x.\mathrm{In}(x, \mathrm{france}) \wedge \mathrm{Smart}(x)$

Open, Closed and Ground Formula

- A formula with a free variable (not bounded by a quantifier) is called open

$$\forall x.[P(x,y) \leftrightarrow [\exists x.\exists z.[Q(x,y,z) \rightarrow R(x,y)]]]$$

- A formula with no free variables is called closed

$$\forall y.\forall x.[P(x,y) \leftrightarrow [\exists x.\exists z.[Q(x,y,z) \rightarrow R(x,y)]]]$$

- A formula with no variables is called ground

$$[P(a,b) \leftrightarrow [Q(a,b,c) \rightarrow R(a,b)]]$$

Semantics of FOL: intuition

- Just like in propositional logic, a (complex) FOL formula may be true (or false) with respect to a given interpretation
- An interpretation specifies referents for

constant symbols $\mapsto$ objects
function symbols $\mapsto$ functional relations
predicate symbol $\mapsto$ relations

- An atomic sentence $P(t_1, \ldots, t_n)$ is true in a given interpretation iff the objects referred to by $t_1, \ldots, t_n$ are in the relation referred to by the predicate P
- An interpretation in which a formula is true is called a *model* for the formula

Semantics of FOL: Interpretations

- Interpretation: $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$
  - Δ is an arbitrary non-empty set of objects
  - $\cdot^{\mathcal{I}}$ is a function that maps
    - any constant *a* into an object in Δ:

$$a^{\mathcal{I}} \in \Delta$$

    - any *n*-ary function symbol *f* to a function:

$$f^{\mathcal{I}} : \Delta^n \rightarrow \Delta$$

    - any *n*-ary predicate symbol *P* to a relation:

$$P^{\mathcal{I}} \subseteq \Delta^n$$

Interpretation Example

Consider

$$\forall x.\exists y.\mathrm{Loves}(x, \mathrm{friendOf}(y))$$
$$\mathrm{Loves}(a, b)$$

- Interpretation: $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$
  - $\Delta = \{\mathrm{john}, \mathrm{mary}, \mathrm{tim}, \mathrm{claudia}\}$
  - mapping of constants:

$$\begin{array}{lcl} a^{\mathcal{I}} & = & \mathrm{john} \\ b^{\mathcal{I}} & = & \mathrm{mary} \end{array}$$

  - mapping of functions:

$$friendOf^{\mathcal{I}}(d) \quad = \quad \begin{cases} \mathrm{mary} & \mathrm{if} \quad d = \mathrm{john} \\ \mathrm{claudia} & \mathrm{if} \quad d = \mathrm{mary} \\ \mathrm{john} & \mathrm{if} \quad d = \mathrm{tim} \\ \mathrm{tim} & \mathrm{if} \quad d = \mathrm{claudia} \end{cases}$$

  - mapping of predicates:

$$Loves^{\mathcal{I}} = \{\langle \mathrm{john}, \mathrm{mary} \rangle, \langle \mathrm{john}, \mathrm{claudia} \rangle, \langle \mathrm{mary}, \mathrm{tim} \rangle, \langle \mathrm{claudia}, \mathrm{tim} \rangle\}$$

Example (cont.)

The same interpretation can also be represented as:

- Interpretation: $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$
  - $\Delta = \{\text{john}, \text{mary}, \text{tim}, \text{claudia}\}$
  - mapping of constants:

$$\begin{array}{lcl} a^{\mathcal{I}} & = & \text{john} \\ b^{\mathcal{I}} & = & \text{mary} \end{array}$$

  - mapping of functions:

$$\{\text{friendOf}(\text{john}, \text{mary}), \text{friendOf}(\text{mary}, \text{claudia}),$$
$$\text{friendOf}(\text{tim}, \text{john}), \text{friendOf}(\text{claudia}, \text{tim})\}$$

  - mapping of predicates:

$$\{\text{Loves}(\text{john}, \text{mary}), \text{Loves}(\text{john}, \text{claudia}),$$
$$\text{Loves}(\text{mary}, \text{tim}), \text{Loves}(\text{claudia}, \text{tim})\}$$

Semantic of FOL: interpretation of ground terms

- Interpretation of ground terms

$$f(t_1, \ldots, t_n)^{\mathcal{I}} = f^{\mathcal{I}}({t_1}^{\mathcal{I}}, \ldots, {t_n}^{\mathcal{I}})$$

Example:

$$\begin{aligned} (\text{friendOf(a)})^{\mathcal{I}} &= \text{friendOf}^{\mathcal{I}}(\text{a}^{\mathcal{I}}) \\ &= \text{friendOf}^{\mathcal{I}}(\text{john}) \\ &= \text{mary} \end{aligned}$$

Semantic of FOL: Satisfaction (Model)

- Satisfaction (model of) of ground atoms $P(t_1, \ldots, t_n)$

$$\mathcal{I} \models P(t_1, \ldots, t_n) \quad \text{iff} \quad \langle t_1{}^{\mathcal{I}}, \ldots, t_n{}^{\mathcal{I}} \rangle \in P^{\mathcal{I}}$$

Example:

  - $\mathcal{I} \models \mathrm{Loves}(\mathrm{a}, \mathrm{b})$

$$\begin{aligned} \mathcal{I} \models \mathrm{Loves}(\mathrm{a}, \mathrm{b}) \quad & \text{iff} \quad \langle \mathrm{a}^{\mathcal{I}}, \mathrm{b}^{\mathcal{I}} \rangle \in \mathrm{Loves}^{\mathcal{I}} \\ & \text{iff} \quad \langle \mathrm{john}, \mathrm{mary} \rangle \in \mathrm{Loves}^{\mathcal{I}} \end{aligned}$$

  - $\mathcal{I} \not\models \mathrm{Loves}(\mathrm{b}, \mathrm{a})$

$$\begin{aligned} \mathcal{I} \not\models \mathrm{Loves}(\mathrm{b}, \mathrm{a}) \quad & \text{iff} \quad \langle \mathrm{b}^{\mathcal{I}}, \mathrm{a}^{\mathcal{I}} \rangle \notin \mathrm{Loves}^{\mathcal{I}} \\ & \text{iff} \quad \langle \mathrm{mary}, \mathrm{john} \rangle \notin \mathrm{Loves}^{\mathcal{I}} \end{aligned}$$

Semantic of FOL: Variable Assignments

- An Interpretation $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$ maps
  - any variable x into an object in Δ:

$$x^{\mathcal{I}} \in \Delta$$

- Let x be a variable and let $d \in \Delta$ be an object. Then

$$\mathcal{I}_x^d$$

is an interpretation, which is as $\mathcal{I}$, except that x is mapped into d: i.e.

$$\begin{array}{rcl} z^{\mathcal{I}_x^d} & = & \begin{cases} z^{\mathcal{I}} & \text{if} \quad z \neq x \\ d & \text{if} \quad z = x \end{cases} \\ a^{\mathcal{I}_x^d} & = & a^{\mathcal{I}} \\ f^{\mathcal{I}_x^d} & = & f^{\mathcal{I}} \\ P^{\mathcal{I}_x^d} & = & P^{\mathcal{I}} \end{array}$$

Interpretation Example

Consider: Loves(x, y)

- Interpretation: $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$
  - $\Delta = \{\text{john}, \text{mary}, \text{tim}, \text{claudia}\}$
  - mapping of variables:

$$\begin{aligned} x^{\mathcal{I}} &= \text{john} \\ y^{\mathcal{I}} &= \text{mary} \end{aligned}$$

  - mapping of predicates:

$$\text{Loves}^{\mathcal{I}} = \{\langle \text{john}, \text{mary} \rangle, \langle \text{mary}, \text{tim} \rangle\}$$

- $\mathcal{I} \models \text{Loves}(x, y)$

$$\begin{aligned} \mathcal{I} \models \text{Loves}(x, y) \quad & \text{iff} \quad \langle x^{\mathcal{I}}, y^{\mathcal{I}} \rangle \in \text{Loves}^{\mathcal{I}} \\ & \text{iff} \quad \langle \text{john}, \text{mary} \rangle \in \text{Loves}^{\mathcal{I}} \end{aligned}$$

- $\mathcal{I}_y^{\text{claudia}} \not\models \text{Loves}(x, y)$

$$\begin{aligned} \mathcal{I}_y^{\text{claudia}} \not\models \text{Loves}(x, y) \quad & \text{iff} \quad \langle x^{\mathcal{I}_y^{\text{claudia}}}, y^{\mathcal{I}_y^{\text{claudia}}} \rangle \notin \text{Loves}^{\mathcal{I}_y^{\text{claudia}}} \\ & \text{iff} \quad \langle x^{\mathcal{I}}, \text{claudia} \rangle \notin \text{Loves}^{\mathcal{I}} \\ & \text{iff} \quad \langle \text{john}, \text{claudia} \rangle \notin \text{Loves}^{\mathcal{I}} \end{aligned}$$

Semantics of FOL: Satisfiability of formulae

- An interpretation $\mathcal{I}$ satisfies (is a model of) a formula α (α is true in $\mathcal{I}$), denoted

  $$\mathcal{I} \models \alpha$$

  iff:

  $$\begin{array}{rcl}
  \mathcal{I} \models P(t_1, \ldots, t_n) & \text{iff} & \langle {t_1}^{\mathcal{I}}, \ldots, {t_n}^{\mathcal{I}} \rangle \in P^{\mathcal{I}} \\
  \mathcal{I} \models \neg\alpha & \text{iff} & \mathcal{I} \not\models \alpha \\
  \mathcal{I} \models \alpha \wedge \beta & \text{iff} & \mathcal{I} \models \alpha \text{ and } \mathcal{I} \models \beta \\
  \mathcal{I} \models \alpha \vee \beta & \text{iff} & \mathcal{I} \models \alpha \text{ or } \mathcal{I} \models \beta \\
  \mathcal{I} \models \alpha \rightarrow \beta & \text{iff} & \mathcal{I} \models \neg\alpha \vee \beta \\
  \mathcal{I} \models \alpha \leftrightarrow \beta & \text{iff} & \mathcal{I} \models (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)
  \end{array}$$

Semantics of FOL: Satisfiability of formulae (cont.)

$$\mathcal{I} \models \forall x.\alpha \quad \text{iff} \quad \text{for all } d \in \Delta, \mathcal{I}_x^d \models \alpha$$

$$\mathcal{I} \models \exists x.\alpha \quad \text{iff} \quad \text{for some } d \in \Delta, \mathcal{I}_x^d \models \alpha$$

- $\mathcal{I}$ satisfies (is a model of) a set of formulae *KB* (denoted $\mathcal{I} \models KB$) iff for each $\alpha \in KB$, $\mathcal{I} \models \alpha$

Example

Interpretation $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$ with

$$\begin{array}{rcl} \Delta & = & \{d_1, \ldots, d_n\} \text{ with } n > 1 \\ \mathrm{x}^{\mathcal{I}} & = & d_1 \quad \mathrm{y}^{\mathcal{I}} \; = \; d_2 \\ \mathrm{a}^{\mathcal{I}} & = & d_1 \quad \mathrm{b}^{\mathcal{I}} \; = \; d_1 \\ \mathrm{Block}^{\mathcal{I}} & = & \{d_1\} \\ \mathrm{Red}^{\mathcal{I}} & = & \Delta \end{array}$$

1. $\mathcal{I} \models \mathrm{Block(a)} \lor \neg\mathrm{Block(a)}$?
2. $\mathcal{I} \models \mathrm{Block(x)} \rightarrow \neg\mathrm{Block(y)}$?
3. $\mathcal{I} \models \forall x.\exists y.[\mathrm{Block(x)} \rightarrow \mathrm{Red(y)}]$?
4. For $KB = \{\mathrm{Block(a)}, \mathrm{Block(b)}, \forall \mathrm{x}.[\mathrm{Block(x)} \rightarrow \mathrm{Red(x)}]\}$

$$\mathcal{I} \models KB \text{ ?}$$

Satisfiability and Validity

Similarly as in propositional logic, a formula α can be satisfiable, unsatisfiable, falsifiable or valid

- α is satisfiable iff there is some model $\mathcal{I}$ of α
- α is unsatisfiable iff there is no model $\mathcal{I}$ of α
- α is falsifiable iff there is some $\mathcal{I}$ not satisfying α
- α is valid (i.e., a tautology) iff every interpretation $\mathcal{I}$ is a model of α

Equivalence

Analogously, two formulas are logically equivalent (denoted $\alpha \equiv \beta$) if for all $\mathcal{I}$ we have

$$\mathcal{I} \models \alpha \quad \text{iff} \quad \mathcal{I} \models \beta$$

Note that P(x) $\not\equiv$ P(y).
Indeed, consider Interpretation $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$ with

$$\begin{array}{rcl} \Delta & = & \{d_1, d_2\} \\ \mathsf{x}^{\mathcal{I}} & = & d_1 \quad \mathsf{y}^{\mathcal{I}} \; = \; d_2 \\ \mathsf{P}^{\mathcal{I}} & = & \{d_1\} \end{array}$$

Entailment

Entailment is defined similarly as in propositional logic.

- A formula α entails a formula β (denoted $\alpha \models \beta$) iff β is true in all models of α
- A set *KB* of formulae entails a formula α (denoted $KB \models \alpha$) iff α is true in all models of *KB*

Proposition: $KB \models \alpha$ iff $KB \cup \{\neg\alpha\}$ is not satisfiable.

Example

$$\begin{aligned} \mathit{KB} &= \{ && \text{Human(socrates)}, \\ &&& \forall x.[\text{Human(x)} \rightarrow \text{Mortal(x)}] \ \} \end{aligned}$$

$$\mathit{KB} \models \text{Mortal(socrates)} \ ?$$

Yes.

- Consider a model $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$ of *KB*
- Then $\mathcal{I} \models$ Human(socrates), i.e. socrates$^{\mathcal{I}} \in$ Human$^{\mathcal{I}}$
- Then $\mathcal{I} \models \forall x.$[Human(x) $\rightarrow$ Mortal(x)], i.e. Human$^{\mathcal{I}} \subseteq$ Mortal$^{\mathcal{I}}$
- As a consequence, socrates$^{\mathcal{I}} \in$ Mortal$^{\mathcal{I}}$,
  i.e. $\mathcal{I} \models$ Mortal(socrates)
- Therefore, $\mathcal{I} \models$ Mortal(socrates) in any model $\mathcal{I}$ of *KB*,
  i.e. *KB* $\models$ Mortal(socrates)

Example

$$\begin{aligned} KB &= \{ \quad \mathrm{Block(a)}, \mathrm{Block(b)}, \\ &\qquad \forall x.\exists y.[\mathrm{Block(x)} \rightarrow \mathrm{Red(y)}] \ \} \end{aligned}$$

$$KB \quad \models \quad \mathrm{Red(b)}\ ?$$

No.
Consider $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$ with

$$\begin{aligned} \Delta &= \{d_1, d_2\} \\ \mathrm{a}^{\mathcal{I}} &= d_1 \quad \mathrm{b}^{\mathcal{I}} = d_2 \\ \mathrm{Block}^{\mathcal{I}} &= \{d_1, d_2\} \\ \mathrm{Red}^{\mathcal{I}} &= \emptyset \end{aligned}$$

Then $\mathcal{I} \models KB$, but $\mathcal{I} \not\models \mathrm{Red(b)}$.

More Examples

- $\models \forall x.[P(x) \vee \neg P(x)]$
- $P(a) \models \exists x.P(x)$
- $\exists x.[P(x) \wedge [P(x) \rightarrow Q(x)]] \models \exists x.Q(x)$

Equality

- Equality is a special predicate
- Syntax: $t_1 = t_2$, for terms t_1 and t_2
- Semantics: $\mathcal{I} \models t_1 = t_2$ iff ${t_1}^{\mathcal{I}} = {t_2}^{\mathcal{I}}$, i.e., t_1 and t_2 refer to the same object
- Example: two humans are siblings iff they have the same parents

$$\begin{aligned} \forall \mathrm{x}.\forall \mathrm{y}.[\mathrm{Sibling}(\mathrm{x},\mathrm{y}) \quad \leftrightarrow \quad & [\neg(\mathrm{x} = \mathrm{y}) \wedge \\ & \exists \mathrm{m}.\exists \mathrm{f}.[\neg(\mathrm{m} = \mathrm{f}) \wedge \mathrm{Parent}(\mathrm{m},\mathrm{x}) \wedge \mathrm{Parent}(\mathrm{f},\mathrm{x}) \wedge \\ & \mathrm{Parent}(\mathrm{m},\mathrm{y}) \wedge \mathrm{Parent}(\mathrm{f},\mathrm{y})]]] \end{aligned}$$

Notes on Universal Quantification

- "Everyone in Italy is smart":

$$\forall \mathrm{x}.[\mathrm{In}(\mathrm{x}, \mathrm{italy}) \rightarrow \mathrm{Smart}(\mathrm{x})]$$

- Typically, → is the main connective with $\forall$
- Common mistake: using $\wedge$ as the main connective with $\forall$

$$\forall \mathrm{x}.[\mathrm{In}(\mathrm{x}, \mathrm{italy}) \wedge \mathrm{Smart}(\mathrm{x})]$$

means "Everyone is in Italy and everyone is smart"

Notes on Existential Quantification

- "Someone in France is smart":

$$\exists x.[\mathrm{In}(x, \mathrm{france}) \wedge \mathrm{Smart}(x)]$$

- Typically, $\wedge$ is the main connective with $\exists$
- Common mistake: using $\rightarrow$ as the main connective with $\exists$

$$\exists x.[\mathrm{In}(x, \mathrm{france}) \rightarrow \mathrm{Smart}(x)]$$

  is true if "there is no one in France"

Properties of quantifiers

- $\forall x.\forall y.\alpha$ is the same as $\forall y.\forall x.\alpha$ (why?)
- $\exists x.\exists y.\alpha$ is the same as $\exists y.\exists x.\alpha$ (why?)
- $\exists x.\forall y.\alpha$ is not the same as $\forall y.\exists x.\alpha$ (why?)
  - $\exists \mathrm{x}.\forall \mathrm{y}.\mathrm{Loves}(\mathrm{x}, \mathrm{y})$

    "There is a person who loves everyone in the world"
  - $\forall \mathrm{y}.\exists \mathrm{x}.\mathrm{Loves}(\mathrm{x}, \mathrm{y})$

    "Everyone in the world is loved by at least one person" (not necessarily the same)
- Quantifier duality

$$
\begin{aligned}
\forall \mathrm{x}.\mathrm{Loves}(\mathrm{x}, \mathrm{beer}) &\equiv \neg\exists \mathrm{x}.\neg\mathrm{Loves}(\mathrm{x}, \mathrm{beer}) \\
\exists \mathrm{x}.\mathrm{Loves}(\mathrm{x}, \mathrm{spinach}) &\equiv \neg\forall \mathrm{x}.\neg\mathrm{Loves}(\mathrm{x}, \mathrm{spinach})
\end{aligned}
$$

Equivalences

All propositional equivalences $+$

$$\begin{aligned}
(\forall x.\alpha) \wedge \beta &\equiv \forall x.(\alpha \wedge \beta) \text{ if } x \text{ not free in } \beta \\
(\forall x.\alpha) \vee \beta &\equiv \forall x.(\alpha \vee \beta) \text{ if } x \text{ not free in } \beta \\
(\exists x.\alpha) \wedge \beta &\equiv \exists x.(\alpha \wedge \beta) \text{ if } x \text{ not free in } \beta \\
(\exists x.\alpha) \vee \beta &\equiv \exists x.(\alpha \vee \beta) \text{ if } x \text{ not free in } \beta \\
(\forall x.\alpha) \wedge (\forall x.\beta) &\equiv \forall x.(\alpha \wedge \beta) \\
(\exists x.\alpha) \vee (\exists x.\beta) &\equiv \exists x.(\alpha \vee \beta) \\
\neg\forall x.\alpha &\equiv \exists x.\neg\alpha \\
\neg\exists x.\alpha &\equiv \forall x.\neg\alpha
\end{aligned}$$

Note:

$$\begin{aligned}
(\forall x.\alpha) \vee (\forall x.\beta) &\not\equiv \forall x.(\alpha \vee \beta) \\
(\exists x.\alpha) \wedge (\exists x.\beta) &\not\equiv \exists x.(\alpha \wedge \beta)
\end{aligned}$$

Equivalences (cont.)

- Let β_x^t denote the formula obtained from β by replacing all free occurrences of x with the term t
- Let $Q_i \in \{\forall, \exists\}$

$$\begin{aligned}(Q_1 x.\alpha) \vee (Q_2 x.\beta) &\equiv Q_1 x.Q_2 y.(\alpha \vee \beta_x^y) \text{ for new variable } y \\ (Q_1 x.\alpha) \wedge (Q_2 x.\beta) &\equiv Q_1 x.Q_2 y.(\alpha \wedge \beta_x^y) \text{ for new variable } y\end{aligned}$$

For instance,

$$\begin{aligned}(\forall x.p(x)) \vee (\forall x.q(x)) &\equiv \forall x.\forall y.(p(x) \vee q(y)) \\ (\exists x.p(x)) \wedge (\exists x.q(x)) &\equiv \exists x.\exists y.(p(x) \wedge q(y))\end{aligned}$$

The Prenex Normal Form

Quantifier prefix + (quantifier free) matrix

$$Q_1 x_1 . Q_2 x_2 . \ldots . Q_n x_n . \alpha$$

where $Q_i \in \{\forall, \exists\}$ and α does not contain any quantifier

1. Elimination of $\leftrightarrow$ and $\rightarrow$
2. Push $\neg$ inwards
3. Pull quantifiers outwards

For instance

$$\begin{aligned}
\neg\forall x.((\forall y.\exists z.P(x,y,z)) \rightarrow \exists x.Q(x)) &\mapsto \neg\forall x.(\neg(\forall y.\exists z.P(x,y,z)) \vee \exists x.Q(x)) \\
&\mapsto \exists x.(\neg(\neg(\forall y.\exists z.P(x,y,z)) \vee \exists x.Q(x))) \\
&\mapsto \exists x.(\neg\neg(\forall y.\exists z.P(x,y,z)) \wedge \neg\exists x.Q(x)) \\
&\mapsto \exists x.((\forall y.\exists z.P(x,y,z)) \wedge \forall x.\neg Q(x)) \\
&\mapsto \exists x.\forall y.\exists z.(P(x,y,z) \wedge \forall x.\neg Q(x)) \\
&\mapsto \exists x.\forall y.\exists z.\forall u.(P(x,y,z) \wedge \neg Q(u))
\end{aligned}$$

Skolemization

Elimination of $\exists$ in a prenex normal form

$$\begin{aligned} \exists x.\alpha &\mapsto \alpha_x^c \text{ for new constant } c \\ \forall x \exists y.\alpha &\mapsto \forall x.\alpha_y^{f(x)} \text{ for new function symbol } f \end{aligned}$$

For instance,

$$\begin{aligned} \exists \mathrm{x}.\forall \mathrm{y}.\exists \mathrm{z}.\forall \mathrm{u}.(\mathrm{P}(\mathrm{x},\mathrm{y},\mathrm{z}) \wedge \neg \mathrm{Q}(\mathrm{u})) &\mapsto \forall \mathrm{y}.\exists \mathrm{z}.\forall \mathrm{u}.(\mathrm{P}(\mathrm{c},\mathrm{y},\mathrm{z}) \wedge \neg \mathrm{Q}(\mathrm{u})) \\ \forall \mathrm{y}.\exists \mathrm{z}.\forall \mathrm{u}.(\mathrm{P}(\mathrm{c},\mathrm{y},\mathrm{z}) \wedge \neg \mathrm{Q}(\mathrm{u})) &\mapsto \forall \mathrm{y}.\forall \mathrm{u}.(\mathrm{P}(\mathrm{c},\mathrm{y},\mathrm{f}(\mathrm{y})) \wedge \neg \mathrm{Q}(\mathrm{u})) \end{aligned}$$

Proposition: Let α be a proposition in prenex normal form and let sk(α) its skolemization. Then α is satisfiable iff sk(α) is satisfiable.

Hence any formula can be transformed into a satisfiability preserving form (α quantifier free):

$$\forall x_1.\forall x_2.\ldots.\forall x_n.\alpha$$

Herbrand Interpretation

Consider a formula $\beta := \forall x_1.\forall x_2.\ldots.\forall x_n.\alpha$, where α is quantifier free.

Hebrand universe: the smallest set U_β of terms inductively defined as:

- if c is a constant that occurs in α then $c \in U_\beta$. If no constant occurs in α then $c \in U_\beta$ for a new constant c
- if f is an n-ary function symbol occurring in α and $t_1, \ldots, t_n \in U_\beta$, then $f(t_1, \ldots, t_n) \in U_\beta$

Hebrand base: the set B_β of ground atoms such that

- if P is an n-ary predicate symbol occurring in α and $t_1, \ldots, t_n \in U_\beta$, then $P(t_1, \ldots, t_n) \in B_\beta$

Hebrand Interpretation: any subset $\mathcal{I}$ of B_β ($A \in \mathcal{I}$ means that A is true in $\mathcal{I}$)

Herbrand Models

A Herbrand model is a Herbrand interpretion that is a model.
For instance, given

$$\beta := \forall x.\forall y.P(f(x)) \vee Q(g(y)) \vee P(a)$$

Hebrand universe: $U_\beta = \{a, f(a), g(a), f(f(a)), f(g(a)), g(f(a)), \ldots\}$

Hebrand base: $B_\beta = \{P(a), Q(a), P(f(a)), P(g(a)), Q(f(a)), Q(g(a)), \ldots\}$

Hebrand Interpretation: Examples,

$$\begin{array}{rcl} \mathcal{I}_1 & = & \{P(a)\} \\ \mathcal{I}_2 & = & \{P(g(a)), Q(f(a))\} \end{array}$$

Hebrand models: Examples,

$$\begin{array}{rcl} \mathcal{I}_1 & \models & \beta \\ \mathcal{I}_2 & \not\models & \beta \end{array}$$

Proposition: $\forall x_1.\forall x_2.\ldots\forall x_n.\alpha$ (α quantifier free) is satisfiable iff it has a Hebrand model. Hence, any formula is satisfiable iff it has a Herbrand model.

The Conjunctive Normal Form

$\forall$ prefix + (quantifier free) matrix

$$\forall x_1.\forall x_2.\ldots.\forall x_n.(C_1 \land C_2 \land \ldots \land C_k)$$

where each C_j (clause) is a disjunction of literals

Proposition: Any formula can be transformed into a satisfiability preserving Conjunctive Normal Form.

1. Transform the formula into a prenex normal form
2. Apply skolemization
3. Transform the quantifier free matrix into conjunctive normal form in a similar way as for propositional logic

Excercise

$$
\begin{aligned}
KB &= \{Person(john), Person(andrea), Female(susan), Male(bill)\} \\
&\cup \{Loves(andrea, bill), Loves(susan, andrea), HasFriend(john, susan), HasFriend(john, andrea)\} \\
&\cup \{\forall x. Person(x) \leftrightarrow (Male(x) \lor Female(x), \neg\exists x. Male(x) \land Female(x)\}
\end{aligned}
$$

$$KB \models \exists y \exists z. HasFriend(john, y) \land Female(y) \land Loves(y, z) \land Male(z) \ \ ?$$

Checking KB Satisfiability using Resolution

1. Put a KB into CNF, i.e., all formulae F in the KB are of the form $\forall \mathbf{x}.C_1 \wedge \ldots \wedge C_n$ and each $C_i = \{L_{i_1} \vee \ldots \vee L_{i_k}\}$ is a disjunction of literals L_{i_j}
2. Transform then all $F \in KB$ into a set of clauses, $C_1, \ldots, C_n$, where each clause C_i is a set of literals $C_i = \{L_{i_1}, \ldots L_{i_k}\}$
3. Apply interactively the resolution rule to two clauses to generate a new clause until either no new clause can be generated or the empty clause has been generated
4. Then the KB is unsatisfiable iff the empty clause can be generated

We next illustrate the resolution rule

The resolution rule relies on the notion of most general unifier, *mgu*:

Most general unifier

- a substitution σ is a set

  $$\sigma = \{v_1/t_1, \ldots v_n/t_n\} ,$$

  where v_i are variables, t_i are terms, all v_i are distinct and t_i is not v_i
  - Example: $\sigma = \{y/f(a), x/a, z/a\}$
- Let S be set of atoms. Then $S\sigma$ is obtained from S by replacing all variables v_i occurring in S with t_i
  - Example: for $S = \{P(f(x), z), P(y, a)\}$, $S\sigma = \{P(f(a), a)\}$
- A substitution σ is an unifier of S iff $|S\sigma| = 1$
  - Example: $\theta = \{y/f(x), z/a\}$ is an unifier of S as $S\theta = \{P(f(x), a)\}$

- Given substitutions $\sigma = \{v_1/t_1, \ldots v_n/t_n\}$ and $\theta = \{u_1/s_1, \ldots u_n/s_m\}$, then the composition $\theta\sigma$ is

  $$\{u_1/s_1\sigma, \ldots u_n/s_m\sigma, v_1/t_1, \ldots v_n/t_n\} \ ,$$

  from which we remove v_i/t_i if $v_i \in \{u_1, \ldots u_n\}$ and we remove bindings of the form $u_j/s_j\sigma$ if $u_j = s_j\sigma$
  - Example: for $\delta = \{x/a\}$, $\theta\delta = \{y/f(a), x/a, z/a\} = \sigma$
- θ is a most general unifier (mgu) of S iff
  1. θ is an unifier of S
  2. and for any unifier σ there is a substitution δ such that $\sigma = \theta\delta$

  - Example: θ is a mgu for S

- Given S, the *disagreement set* of S
  - Locate the leftmost symbol position at which not all atoms in S agree and extract from each atom the subexpression beginning with that symbol
  - The set of these subexpressions is the disagreement set D
  - Example: for $S = \{P(f(x), z), P(y, a)\}$, $D = \{f(x), y\}$
- Unification Algorithm
  1. put $k = 0, \sigma_0 = \emptyset$
  2. If $|S\sigma_k| = 1$ stop. σ_k is mgu of S. Otherwise, determine disagreement set of D_k of $S\sigma_k$
  3. If there is v and t in D_k such that v variable not occurring in t then put $\sigma_{k+1} = \sigma_k\{v/t\}$, increment k and go to step 2. Otherwise, stop; S is not unifiable

Example

1. $S = \{P(f(a), g(x)), P(y, y)\}$
2. $\sigma_0 = \emptyset$
3. $D_0 = \{f(a), y\}$, $\sigma_1 = \{y/f(a)\}$,
   $S\sigma_1 = \{P(f(a), g(x)), P(f(a), f(a))\}$
4. $D_1 = \{g(x), f(a)\}$, stop. S not unifiable

Example

1. $S = \{P(f(x), y), P(y, f(g(b)))\}$
2. $\sigma_0 = \emptyset$
3. $D_0 = \{f(x), y\}$, $\sigma_1 = \{y/f(x)\}$,
   $S\sigma_1 = \{P(f(x), f(x)), P(f(x), f(g(b)))\}$
4. $D_1 = \{x, g(b)\}$, $\sigma_2 = \{x/g(b)\}$,
   $S\sigma_2 = \{P(f(g(b)), f(g(b)))\}$, stop. σ_1 is mgu of S

Resolution rule

- Given clause $C_1 = \{\ldots, A, \ldots\}$
- Given clause $C_2 = \{\ldots, \neg B, \ldots\}$
- Given mgu σ of $S = \{A, B\}$, generate the new clause

$$C = (C_1 \cup C_2)\sigma \setminus \{A, \neg B\}\sigma$$

- We say that $Res(C_1, C_2) = C$

Example

- Consider

$$KB = \{\text{Human(socrates)}, \forall x.[\text{Human(x)} \implies \text{Mortal(x)}]\ \}$$

- Then

$$KB \models \text{Mortal(socrates) iff UNSAT } KB \cup \{\neg\text{Mortal(socrates)}\}$$

- Let us show that we get the empty clause

$$\begin{aligned} C_1 &= \{\text{Human(socrates)}\} \\ C_2 &= \{\neg\text{Human(x)}, \text{Mortal(x)}\} \\ C_3 &= \{\neg\text{Mortal(socrates)}\} \\ \\ C_4 &= \{\neg\text{Human(socrates)}\} = Res(C_3, C_2) \\ C_5 &= \emptyset = Res(C_1, C_4) \end{aligned}$$

- Therefore

$$KB \models \text{Mortal(socrates)}$$

Exercises: FOL Logic

Using a FOL Solver

- A FOL solver is a program that determines whether a set of formulae in is satisfiable or not, or check if a formula is logically entailed by a KB
- Usually, a FOL solver may provide also a proof

Some FOL Solvers

- For the exercises, we use the FOL solver SPASS.

  SPASS: `http://spass.mpi-sb.mpg.de/`

  Vapire: `http://en.wikipedia.org/wiki/Vampire_theorem_prover`

  List of Theorem Provers: `http://en.wikipedia.org/wiki/Automated_theorem_proving`

  System Competition: `http://www.cs.miami.edu/~tptp/CASC/J3/`

Simple Example

$$KB = \{\text{Human(socrates)}, \forall x.[\text{Human(x)} \implies \text{Mortal(x)}]\ \}$$
$$KB \models \text{Mortal(socrates)}\ ?$$

```
begin_problem(Sokrates1).
list_of_descriptions.
  name({*Sokrates*}).
  author({*Christoph Weidenbach*}).
  status(unsatisfiable).
  description({*Sokrates is mortal and since all humans are mortal, he is mortal too. *}).
end_of_list.
list_of_symbols.
  functions[(sokrates,0)].
  predicates[(Human,1),(Mortal,1)].
end_of_list.
list_of_formulae(axioms).
  formula(Human(sokrates),1).
  formula(forall([x],implies(Human(x),Mortal(x))),2).
end_of_list.
list_of_formulae(conjectures).
  formula(Mortal(sokrates),3).
end_of_list.
end_problem.
```

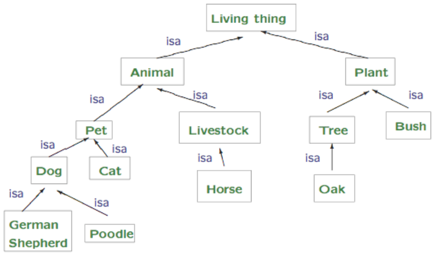
Simple Example (cont.)

$$KB = \{\text{Human(socrates)}, \forall x.[\text{Human(x)} \implies \text{Mortal(x)}]\ \}$$
$$KB \models \text{Mortal(socrates) ?}$$

$$KB \models \text{Mortal(socrates) iff unsatisfiable } KB \cup \{\neg\text{Mortal(socrates)}\}$$

Chiamata del solver: `./SPASS -DocProof ./examples/socrates.dfg`

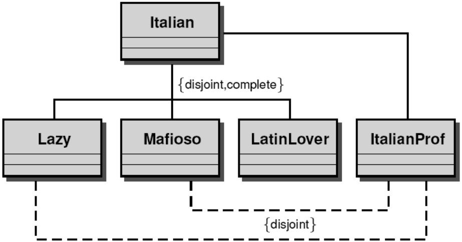
Example ISA Hierarchy



- Encode the hierarchy and
  - check if “a dog is an animal”
  - check if “a dog is a plant”
  - check if “a dog is not a plant”
- Note:

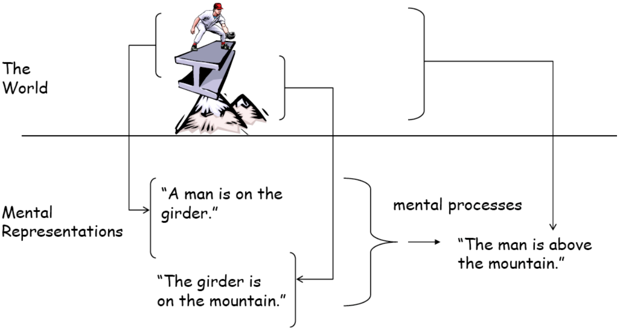
$KB \models \forall x.\mathrm{Dog}(x) \implies \mathrm{Animal}(x)$ iff unsatisfiable $KB \cup \{\neg(\forall x.\mathrm{Dog}(x) \implies \mathrm{Animal}(x))\}$
iff unsatisfiable $KB \cup \{\exists x.\neg(\mathrm{Dog}(x) \implies \mathrm{Animal}(x))\}$
iff unsatisfiable $KB \cup \{\exists x.\neg(\neg\mathrm{Dog}(x) \vee \mathrm{Animal}(x))\}$
iff unsatisfiable $KB \cup \{\exists x.(\mathrm{Dog}(x) \wedge \neg\mathrm{Animal}(x))\}$
iff unsatisfiable $KB \cup \{\mathrm{Dog}(a) \wedge \neg\mathrm{Animal}(a)\}$ (Skolemization)
iff unsatisfiable $KB \cup \{\mathrm{Dog}(a), \neg\mathrm{Animal}(a)\}$

Example "Latin Lover"



- Encode the example
  - check if "an ItalianProf is a LatinLover"

Example “A man on the girder”



The
World

Mental
Representations

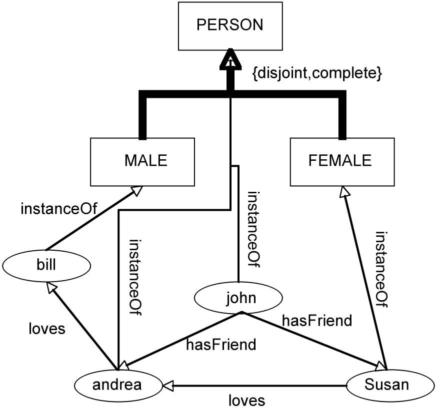
“A man is on the
girder.”

“The girder is
on the mountain.”

mental processes

“The man is above
the mountain.”

Example:reasoning at the individual level

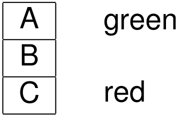


Verify that

> *“John has a female friend loving a male person”*

Example “Blocks”

Three blocks stacked
Top one is green
Bottom one is red



Is there a green block directly on top of a non-green block?.