2.1 Syntax of Predicate Logic

Freitag, 10. April 2015 10:00

2.1: Syntax
2.2: Semantics } of Predicate Logic

2.1. Syntax of Predicate Logic

Syntax: determines which symbols constitute the words of a language and in which order these symbols may occur

First: define alphabet for formulas of predicate logic

Def 2.1.1 (Signature)

A signature (Σ, Δ) is a pair with $\Sigma = \bigcup_{n \in \mathbb{N}} \Sigma_n$ and $\Delta = \bigcup_{n \in \mathbb{N}} \Delta_n$. The sets Σ_n and Δ_n are pairwise disjoint. Every $f \in \Sigma_n$ is a function symbol of arity n, every $p \in \Delta_n$ is a predicate symbol of arity n.
The elements of Σ_0 are also called constants. We always require $\Sigma_0 \neq \emptyset$.

Ex. 2.1.2: Signature (Σ, Δ) for the logic prog. from Chapter 1. Here $\Sigma = \Sigma_0 \cup \Sigma_3$, $\Delta = \Delta_1 \cup \Delta_2$.
date is an additional fct. symbol of arity 3
(for dates consisting of day, month, year)

Fct symbols create objects (terms)

Pred symbols create statements (formulas)

Def 2.13 (Terms)

Let (Σ, Δ) be a signature, let $\mathcal{V}$ be a set of variables with $\Sigma \cap \mathcal{V} = \emptyset$. Then $\mathcal{T}(\Sigma, \mathcal{V})$ is the set of all terms over Σ and $\mathcal{V}$. $\mathcal{T}(\Sigma, \mathcal{V})$ is the smallest set such that:

- $\mathcal{V} \subseteq \mathcal{T}(\Sigma, \mathcal{V})$
- $f(t_1, \dots, t_n) \in \mathcal{T}(\Sigma, \mathcal{V})$ if $f \in \Sigma_n$ and $t_1, \dots, t_n \in \mathcal{T}(\Sigma, \mathcal{V})$ for some $n \in \mathbb{N}$.

$\mathcal{T}(\Sigma)$ stands for $\mathcal{T}(\Sigma, \emptyset)$, i.e., the set of ground terms (terms without variables)

For any term t, let $\mathcal{V}(t)$ be the set of all variables in t.

Ex 2.14 Let Σ be as in Ex 2.1.2, let $\mathcal{V} = \{X, Y, Z, \dots\}$.

Terms in $\mathcal{T}(\Sigma, \mathcal{V})$: X, monika, 42, date(10, 4, 2015), date(X, monika, date(10, 4, 2015)), ...

Def 2.15 (Formulas)

Let (Σ, Δ) be a signature and $\mathcal{V}$ be a set of variables. The set of atomic formulas over (Σ, Δ) and $\mathcal{V}$ is defined as $At(\Sigma, \Delta, \mathcal{V}) = \{p(t_1, \dots, t_n) \mid p \in \Delta_n,\ t_1, \dots, t_n \in \mathcal{T}(\Sigma, \mathcal{V})\}$.

$\mathcal{F}(\Sigma, \Delta, \mathcal{V})$ is the set of all formulas over (Σ, Δ) and $\mathcal{V}$. $\mathcal{F}(\Sigma, \Delta, \mathcal{V})$ is the smallest set such that

- $At(\Sigma, \Delta, \mathcal{V}) \subseteq \mathcal{F}(\Sigma, \Delta, \mathcal{V})$
- if $\varphi \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$ then $\neg \varphi \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$

"not φ"

"or"

"implies"

- if $\varphi \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$, then $\neg\varphi \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$
- if $\varphi_1, \varphi_2 \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$, then $(\varphi_1 \wedge \varphi_2)$, $(\varphi_1 \vee \varphi_2)$, $(\varphi_1 \rightarrow \varphi_2)$, $(\varphi_1 \leftrightarrow \varphi_2) \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$

"or" "plies"

"and"

"is equivalent to"

- if $X \in \mathcal{V}$ and $\varphi \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$, then $(\forall X\ \varphi)$, $(\exists X\ \varphi) \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$

"for all" "exists"

For a formula φ, $\mathcal{V}(\varphi)$ is the set of variables occurring in φ.

A variable X occurs <u>free</u> in a formula φ iff — "if and only if"

- φ is an atomic formula and $X \in \mathcal{V}(\varphi)$ or
- $\varphi = \neg\varphi_1$ and X occurs free in φ_1 or
- $\varphi = (\varphi_1 \circ \varphi_2)$ with $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ and X occurs free in φ_1 or in φ_2 or
- $\varphi = (QY\ \varphi_1)$ with $Q \in \{\forall, \exists\}$, X occurs free in φ_1, and $X \neq Y$.

A formula is <u>closed</u> iff it does not contain free variables.

A formula is <u>quantifier-free</u> iff it does not contain $\forall$ or $\exists$.

We usually omit (...) whenever possible.

<u>Ex 2.16</u> We use the signature of Ex. 2.12.

$female(monika) \in At(\Sigma, \Delta, \mathcal{V})$

$motherOf(X, susanne) \in At(\Sigma, \Delta, \mathcal{V})$

$born(monika, date(15, 10, 1966)) \in At(\Sigma, \Delta, \mathcal{V})$

$\forall W\,(married(gerd, W) \wedge motherOf(W, C)) \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$

gerd is married with all women W and they all are the mother of C

only free variable: C

$married(gerd, W) \wedge \neg(\forall W\; motherOf(W, C)) \in \mathcal{F}(\Sigma, \Delta, \mathcal{V})$

free variables: W, C

We abbreviate $\quad \forall X_1 (\ldots (\forall X_n \;\; \varphi) \ldots)$ by $\forall X_1, \ldots, X_n \;\; \varphi$

$\qquad\qquad\qquad\;\; \exists X_1 (\ldots (\exists X_n \;\; \varphi) \ldots)$ by $\exists X_1, \ldots, X_n \;\; \varphi$

To distinguish variables from fct. and pred. symbols:

Variables start with upper-case letters

fct. + pred. symbols start with lower-case letters

Ex 2.1.7 Every logic program stands for a set of formulas. Here, the variables are universally quantified (i.e., with $\forall$).

Variables in terms and formulas stand for arbitrary objects

$\Rightarrow$ they can be substituted by objects (i.e., by terms).

Def 2.1.8 (Substitution)

A mapping $\sigma : \mathcal{V} \to \mathcal{T}(\Sigma, \mathcal{V})$ is a substitution iff $\sigma(X) \neq X$ holds for finitely many $X \in \mathcal{V}$. $\mathcal{DOM}(\sigma) = \{X \in \mathcal{V} \mid \sigma(X) \neq X\}$ is the domain of σ and

$\mathcal{RANGE}(\sigma) = \{\sigma(X) \mid X \in \mathcal{DOM}(\sigma)\}$ is the range of σ.

A substitution can be denoted as $\{X / \sigma(X) \mid X \in \mathcal{DOM}(\sigma)\}$.

A subst. σ is a ground substitution :⟺ $\sigma(X)$ ...

A subst. σ is a <u>ground substitution</u> iff $\sigma(X)$ contains no variables for all $X \in DOM(\sigma)$.

A substitution σ is a <u>variable renaming</u> iff it is injective and $\sigma(X) \in \mathcal{V}$ for all $X \in \mathcal{V}$.

Ex: $\sigma = \{X/Y, Y/Z, Z/X\}$.

$\sigma(X) = Y$
$\sigma(Y) = Z$
$\sigma(Z) = X$
$\sigma(U) = U$
$\vdots$

Substitutions can be extended to terms, i.e., $\sigma : \mathcal{T}(\Sigma, \mathcal{V}) \to \mathcal{T}(\Sigma, \mathcal{V})$:

$$\sigma(f(t_1, \ldots, t_n)) = f(\sigma(t_1), \ldots, \sigma(t_n))$$

Ex:

$\sigma(date(X, monika, Y)) =$
$date(\sigma(X), monika, \sigma(Y)) =$
$date(Y, monika, Z)$

Substitutions can also be extended to formulas:

- $\sigma(p(t_1, \ldots, t_n)) = p(\sigma(t_1), \ldots, \sigma(t_n))$
- $\sigma(\neg \varphi_1) = \neg \sigma(\varphi_1)$
- $\sigma(\varphi_1 \circ \varphi_2) = \sigma(\varphi_1) \circ \sigma(\varphi_2)$ for $\circ \in \{\wedge, \vee, \to, \leftrightarrow\}$
- $\sigma(QX \ \varphi_1) = QX \ \sigma(\varphi_1)$, for $Q \in \{\forall, \exists\}$ if $X \notin DOM(\sigma) \cup \underbrace{\mathcal{V}(RANGE(\sigma))}_{\text{Variables occurring in the RANGE of } \sigma}$

Reason: $\forall X \ human(X)$
and $\forall Y \ human(Y)$
should be treated in the same way.
$\Rightarrow$ if σ modifies X or if the application of σ

should be treated in the same way.
$\Rightarrow$ if σ modifies X or if the application of σ introduces X, then first rename the bound var. X to a fresh variable

- $\sigma(QX\ \varphi_1) = Q X'\ \sigma(\delta(\varphi_1))$ for $Q \in \{\forall, \exists\}$, $X \in DOM(\sigma) \cup \mathcal{V}(RANGE(\sigma))$.
  Here, X' is a fresh variable with
  $X' \notin DOM(\sigma) \cup \mathcal{V}(RANGE(\sigma)) \cup \mathcal{V}(\varphi_1)$ and
  $\delta = \{X/X'\}$.

Ex. 2.19 $\sigma = \{X/date(X,Y,Z),\ Y/monika,\ Z/date(Z,Z,Z)\}$

$\sigma(date(X,Y,Z)) = date(date(X,Y,Z), monika, date(Z,Z,Z))$

$\sigma(\forall Y\ married(X,Y)) =$

$\sigma(\forall Y'\ married(X,Y')) =$

$\forall Y'\ married(date(X,Y,Z), Y')$

Problem 1: $\sigma(X)$ contains Y

Problem 2: $Y \in DOM(\sigma)$

An instance $\sigma(t)$ or $\sigma(\varphi)$ of a term t (resp. a $\underbrace{\text{formula}}^{\text{quantifier-free}}$ φ) is a ground instance iff it doesn't contain variables.