Computational Logic Lecture 6

Relational Logic Semantics

~~Michael Genesereth~~ David Haley Autumn 2007

Propositional Logic Semantics

A Propositional logic *interpretation* is an association between the propositional constants in a propositional language and the truth values T or F.

Relational Logic Semantics

The *big question*: what is a relational logic interpretation? There are no propositional constants, just object constants, function constants, and relation constants. To what do they refer?

Universe of Discourse

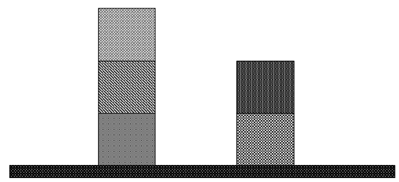
The *Universe of Discourse* is the set of all objects about which we want to say something.

Primitive: *a quark*
Composite: *an engine*, *this class*

Real: *Sun*, *Mike*
Imaginary: *a unicorn*, *Sherlock Holmes*

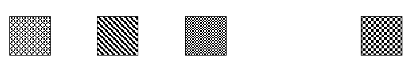
Physical: *Earth*, *Moon*, *Sun*
Abstract: *Justice*

Blocks World



10/9/07 5

Universe of Discourse



10/9/07 6

Relations

A *relation* is a set of objects or tuples of objects each of which manifest a particular property or relationship.

clear - set of all blocks with no blocks on top.
table - set of all blocks on the table.

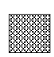
on - set of pairs of blocks in which first is on the second.
above - all pairs in which first block is above the second.
below - all pairs in which first is below the second.

stack - set of triples of blocks arranged in a stack.

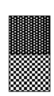
Arity

Each relation has an *arity* that determines the number of objects that can participate in an instance of the relation.

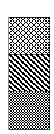
1-ary (unary): *clear*



2-ary (binary): *on*



3-ary (ternary): *stack*



Relations as Tables

on

a	*b*
b	*c*
d	*e*

clear

a
d

stack

a	*b*	*c*

Set Representation of Relations

Each row in a table with *n* columns can be represented as a single *n*-tuple. The table can be represented as the set of such tuples.

a	*b*
b	*c*
d	*e*

$\{\langle a,b\rangle, \langle b,c\rangle, \langle d,e\rangle\}$

a
d

$\{\langle a\rangle, \langle d\rangle\}$ or $\{a,d\}$

a	*b*	*c*

$\{\langle a,b,c\rangle\}$

Counting

Assume a universe of discourse with 5 objects.
Number of 2-tuples: $5^2=25$
Number of binary tables: 2^{25}

Assume a universe of discourse with *n* objects.
Number of *k*-tuples: n^k
Number of *k*-ary tables: $2\wedge n^k$

10/9/07 11

Functions

An *n*-ary function is a relation associating each combination of *n* objects in a universe of discourse (called the *arguments*) with a single object (called the *value*).

Numerical Examples:
Unary: *sqrt*, *log*
Binary: +, -, *, /

Symbolic Examples:
Unary: *mother*, *father*
Binary: *grade*

10/9/07 12

Functions

Functions are *total* and *single-valued* - one and exactly one value for each combination of arguments.

Partial - not defined for some combination of arguments

Multivalued - more than value for some argument combination

NB: We ignore partial and multi-valued functions.

Unary Function

$$
\begin{array}{ccc}
a & \rightarrow & b \\
b & \rightarrow & a \\
c & \rightarrow & d \\
d & \rightarrow & c \\
e & \rightarrow & e
\end{array}
$$

Binary Function

a	a	$\rightarrow$	b
a	b	$\rightarrow$	a
a	c	$\rightarrow$	c
b	a	$\rightarrow$	c
b	b	$\rightarrow$	b
b	c	$\rightarrow$	a
c	a	$\rightarrow$	b
c	b	$\rightarrow$	b
c	c	$\rightarrow$	b

Unary Functions as Binary Relations

a	$\rightarrow$	b
b	$\rightarrow$	a
c	$\rightarrow$	d
d	$\rightarrow$	c
e	$\rightarrow$	e

a	b
b	a
c	d
d	c
e	e

An n-ary function can always be viewed as an n+1-ary relation.

Binary Functions as Ternary Relations

a	*a*	→	*b*
a	*b*	→	*a*
a	*c*	→	*c*
b	*a*	→	*c*
b	*b*	→	*b*
b	*c*	→	*a*
c	*a*	→	*b*
c	*b*	→	*b*
c	*c*	→	*b*

a	*a*	*b*
a	*b*	*a*
a	*c*	*c*
b	*a*	*c*
b	*b*	*b*
b	*c*	*a*
c	*a*	*b*
c	*b*	*b*
c	*c*	*b*

Set Representation of Functions

A function can be represented as a set of associations of arguments and values.

a	*b*
b	*a*
c	*d*
d	*c*
e	*e*

$\{\langle a \to b\rangle, \langle b \to a\rangle, \langle c \to d\rangle, \langle d \to c\rangle, \langle e \to e\rangle\}$

Same as representation of tables except for the use of arrows as a reminder that the table is a function.

Counting

Assume a universe of discourse with 5 objects.

Number of 1-tuples: 5

Number of unary functions: 5^5=3125

Number of binary relations: 2^{25}=33554432

Assume a universe of discourse with *n* objects.

Number of *k*-tuples: n^k

Number of *k*-ary functions: n^n^k=2^(n^k *log n)*

Number of *k*+1-ary relations: 2^n^{k+1}=2^($n^k n$)

10/9/07 19

Role of Logic

Incomplete Information

Block *a* is on block *b or* it is on block *c*.

Block *a* is *not* on block *b*.

Integrity

A block may not be *on* itself.

A block may be *on* only one block at a time.

Definitions

A block is *under* another iff the second is *on* the first.

A block is *clear* iff there is no block *on* it.

A block is on the *table* iff there is no block *under* it.

10/9/07 20

Our World

○ ●

Conceptualization

Universe of Discourse - a set U of objects.

{○, ●}

Functional Basis Set - set $\{f_1,\ldots,f_m\}$ of functions on U.

$$f_i: U^k \rightarrow U$$

Relational Basis Set - set $\{r_1,\ldots,r_n\}$ of relations on U.

$$r_i \subseteq U^k$$

Interpretations

An *interpretation* is a mapping from the constants of a language into elements of a conceptualization $\langle U,F,R \rangle$.

$$i: objconst \rightarrow U$$
$$i: funconst \rightarrow F$$
$$i: relconst \rightarrow R$$

The arity of the function and relation constants must match the arity of their interpretations.

Example

$|i| = U = \{○, ●\}$

$i(a) = ○$
$i(b) = ●$
$i(c) = ●$

$i(f_1) = \{○ \rightarrow ●, ● \rightarrow ○\}$

$i(q_1) = \{\langle ○ \rangle\}$
$i(r_2) = \{\langle ○, ● \rangle, \langle ○, ○ \rangle, \langle ●, ● \rangle\}$
$i(s_3) = \{\langle ○, ●, ○ \rangle, \langle ●, ●, ○ \rangle\}$

Ground Value Assignments

A *ground value assignment* s_i based on interpretation i is a mapping from the ground terms of the language into the universe of discourse that agrees with i on constants and that, for functional terms, yields the result of applying the interpretation of the functional constant to the values assigned to the argument terms.

$$s_i(\sigma) = i(\sigma)$$
$$s_i(\pi(\tau_1,\ldots,\tau_n)) = i(\pi)(s_i(\tau_1),\ldots, s_i(\tau_n))$$

Example

Interpretation:

$i(a) = ○$
$i(b) = ●$
$i(f) = \{○\rightarrow●, ●\rightarrow○\}$
$i(r) = \{\langle ○, ●\rangle, \langle ●, ●\rangle\}$

Ground Value Assignment:

$s_i(a) = i(a) = ○$
$s_i(f(a)) = i(f)(s_i(a)) = i(f)(○) = ●$

Ground Truth Assignments

A *ground truth assignment* t_i based on interpretation i is a mapping from the ground sentences of the language into {*true*, *false*}.

$$t_i\text{: } \textit{ground sentences} \rightarrow \{\textit{true}, \textit{false}\}$$

The details of the definition are given on the following slides.

Relational Sentences

A ground truth assignment *satisfies* a ground relational sentence if and only if the tuple of objects denoted by the arguments is a member of the relation denoted by the relation constant.

$$\begin{aligned} t_i(\rho(\tau_1,\ldots,\tau_n)) &= \textit{true} \text{ if } \langle i(\tau_1),\ldots, i(\tau_1)\rangle \in i(\rho) \\ &= \textit{false} \text{ otherwise} \end{aligned}$$

Example

Interpretation:

$i(a) = ○$

$i(b) = ●$

$i(f) = \{○\rightarrow●, ●\rightarrow○\}$

$i(r) = \{\langle ○, ●\rangle, \langle ●, ●\rangle\}$

Example:

$t_{iv}(r(a,b)) = \mathit{true}$ since $\langle ○, ●\rangle \in i(r)$

$t_{iv}(r(b,a)) = \mathit{false}$ since $\langle ○, ●\rangle \notin i(r)$

Logical Sentences

$t_i(\neg\varphi) = \mathit{true}$ iff $t_i(\varphi) = \mathit{false}$

$t_i(\varphi \wedge \psi) = \mathit{true}$ iff $t_i(\varphi) = \mathit{true}$ and $t_i(\psi) = \mathit{true}$

$t_i(\varphi \vee \psi) = \mathit{true}$ iff $t_i(\varphi) = \mathit{true}$ or $t_i(\psi) = \mathit{true}$

$t_i(\varphi \Rightarrow \psi) = \mathit{true}$ iff $t_i(\varphi) = \mathit{false}$ or $t_i(\psi) = \mathit{true}$

$t_i(\varphi \Leftarrow \psi) = \mathit{true}$ iff $t_i(\varphi) = \mathit{true}$ or $t_i(\psi) = \mathit{false}$

$t_i(\varphi \Leftrightarrow \psi) = \mathit{true}$ iff $t_i(\varphi) = t_i(\psi)$

Quantified Sentences

Intuitively, a universally quantified sentence is true if and only if it is true no matter what *value* we assign to the universally quantified variable.

Intuitively, an existentially quantified sentence is true if and only if it is true for some *value* of the existentially quantified variable.

Stating these definitions precisely is a little tricky due to the possibility of nested quantifiers.

$$\forall x.(\exists y.r(x,y) \Rightarrow \forall x.r(x,x))$$

Variable Assignments

An *variable assignment* for a conceptualization $\langle U,F,R \rangle$ is a mapping of variables into U.

$$v\text{: } variable \rightarrow U$$

Universe of Discourse:
U={○, ●}

Example:
$v(x)$ = ○
$v(y)$ = ●
$v(z)$ = ●

Example:
$v(x)$ = ●
$v(y)$ = ●
$v(z)$ = ●

Value Assignments

A *value assignment* s_{iv} based on interpretation i and variable assignment v is a mapping from the terms of the language into the universe of discourse that agrees with i on constants, that agrees with v on variables, and that, for functional terms, yields the result of applying the interpretation of the functional constant to the values assigned to the argument terms.

$$s_{iv}(\sigma)=i(\sigma)$$
$$s_{iv}(\upsilon)=v(\upsilon)$$
$$s_{iv}(\pi(\tau_1,\ldots,\tau_n)=i(\pi)(s_{iv}(\tau_1),\ldots, s_{iv}(\tau_n))$$

Truth Assignments

A *truth assignment* t_{iv} based on interpretation i and variable assignment v is a mapping from the sentences of the language into $\{true, false\}$.

$$t_{iv}: sentence \rightarrow \{true, false\}$$

The details of the definition are given on the following slides.

Relational Sentences

A truth assignment satisfies a relational sentence if and only if the tuple of objects denoted by the arguments is a member of the relation denoted by the relation constant.

$$t_{iv}(\rho(\tau_1,\ldots,\tau_n)) = \mathit{true} \;\text{ if } \langle s_{iv}(\tau_1),\ldots, s_{iv}(\tau_1)\rangle \in i(\rho)$$
$$= \mathit{false} \text{ otherwise}$$

Logical Sentences

$t_{iv}(\neg \varphi) = \mathit{true}$ iff $t_{iv}(\varphi) = \mathit{false}$

$t_{iv}(\varphi \wedge \psi) = \mathit{true}$ iff $t_{iv}(\varphi) = \mathit{true}$ and $t_{iv}(\psi) = \mathit{true}$

$t_{iv}(\varphi \vee \psi) = \mathit{true}$ iff $t_{iv}(\varphi) = \mathit{true}$ or $t_{iv}(\psi) = \mathit{true}$

$t_{iv}(\varphi \Rightarrow \psi) = \mathit{true}$ iff $t_{iv}(\varphi) = \mathit{false}$ or $t_{iv}(\psi) = \mathit{true}$

$t_{iv}(\varphi \Leftarrow \psi) = \mathit{true}$ iff $t_{iv}(\varphi) = \mathit{true}$ or $t_{iv}(\psi) = \mathit{false}$

$t_{iv}(\varphi \Leftrightarrow \psi) = \mathit{true}$ iff $t_{iv}(\varphi) = t_{iv}(\psi)$

Versions

A *version* $w[\nu \leftarrow x]$ of a variable assignment w is the variable assignment that agrees with w on all variables except ν, which is assigned the value x.

$$w[\nu \leftarrow x](\mu) = w(\mu)$$
$$w[\nu \leftarrow x](\nu) = x$$

10/9/07 37

Examples

Interpretation:
|*i*|={○, ●}

Variable Assignment:
$v(x)$=○
$v(y)$=●

Version 1:
$v[x \leftarrow$ ○$](x)$ = ○
$v[x \leftarrow$ ○$](y)$ = ●

Version 2:
$v[x \leftarrow$ ●$](x)$ = ●
$v[x \leftarrow$ ●$](y)$ = ●

Version 3:
$v[y \leftarrow$ ○$](x)$ = ○
$v[y \leftarrow$ ○$](y)$ = ○

Version 4:
$v[y \leftarrow$ ●$](x)$ = ○
$v[y \leftarrow$ ●$](y)$ = ●

10/9/07 38

Quantified Sentences

A universally quantified sentence is true in interpretation *I* and variable assignment *v* if and only if the scope is true for *I* and every version of *v*.

$t_{iw}(\forall v.\varphi) = true$ iff $t_{iw[v \leftarrow x]}(\varphi) = true$ for all $x \in |i|$.

An existentially quantified sentence is true in interpretation *I* and variable assignment *v* if and only if the scope is true for *I* and some version of *v*.

$t_{iw}(\exists v.\varphi) = true$ iff $t_{iw[v \leftarrow x]}(\varphi) = true$ for some $x \in |i|$.

Examples

Interpretation:

$i(a) = \bigcirc$

$i(b) = \bullet$

$i(r) = \{\langle \bigcirc, \bullet \rangle, \langle \bullet, \bullet \rangle\}$

Variable Assignment:

$v(x) = \bigcirc$

$v(y) = \bullet$

$t_{iv}(\forall x.r(x,x)) = false$

$t_{iv[x \leftarrow \bigcirc]}(r(x,x)) = false$

$t_{iv[x \leftarrow \bullet]}(r(x,x)) = true$

$t_{iv}(\exists y.r(x,y)) = true$

$t_{iy[y \leftarrow \bigcirc]}(r(x,y)) = false$

$t_{iy[y \leftarrow \bullet]}(r(x,y)) = true$

$t_{iv}(\exists x.r(x,x)) = false$

$t_{iv[x \leftarrow \bigcirc]}(r(x,x)) = false$

$t_{iv[x \leftarrow \bullet]}(r(x,x)) = true$

$t_{iv}(\forall x.\exists y.r(x,y)) = true$

$t_{iv[x \leftarrow \bigcirc]}(\exists y.r(x,y)) = true$

$t_{iv[x \leftarrow \bullet]}(\exists y.r(x,y)) = true$

Models

A universally quantified sentence is *true in interpretation I and a particular variable assignment v* if and only if the scope is true for *I* and every version of *v*.

A universally quantified sentence is *true in interpretation I* if and only if the scope is true for *I* and every variable assignment *v*.

An interpretation of a sentence is a *model* if and only if the sentence is true in that interpretation.

Examples

Interpretation

$i(a) = \bigcirc$

$i(b) = \bullet$

$i(r) = \{\langle \bigcirc, \bullet \rangle, \langle \bullet, \bullet \rangle\}$

$t_i(\forall x.r(x,x)) = false$

$t_{i[x \leftarrow \bigcirc]}(r(x,x)) = false$

$t_{i[x \leftarrow \bullet]}(r(x,x)) = true$

$t_i(\exists y.r(x,y)) = true$

$t_{i[y \leftarrow \bigcirc]}(r(x,y)) = false$

$t_{i[y \leftarrow \bullet]}(r(x,y)) = true$

$t_i(\exists x.r(x,x)) = false$

$t_{i[x \leftarrow \bigcirc]}(r(x,x)) = false$

$t_{i[x \leftarrow \bullet]}(r(x,x)) = true$

$t_i(\forall x.\exists y.r(x,y)) = true$

$t_{i[x \leftarrow \bigcirc]}(\exists y.r(x,y)) = true$

$t_{i[x \leftarrow \bullet]}(\exists y.r(x,y)) = true$

Properties of Sentences

A sentence is *valid* if and only if every interpretation is a model.

A sentence is *satisfiable* if and only some interpretation is a model.

A sentence is *unsatisfiable* if and only if no intepretation is a model.

Logical Entailment

A set of premises Δ *logically entails* a conclusion φ (written $\Delta \models \varphi$) if and only if every model of Δ is a model of φ.

NB: *Some* authors say that Δ *logically entails* a conclusion φ (written $\Delta \models \varphi$) if and only if every interpretation and variable assignment that satisfy Δ also satisfy φ. This leads to slightly different properties. (*Exercise: find such a difference*.) However, all important results are the same (e.g. expressiveness, computability, and so forth).