Mathematics for linguists

Gerhard Jäger

University of Tübingen

November 9, 2010

Predicate logic: Introduction

- Extension of statement logic
- syntactic structure of PL is partially inspired by structure of natural languages
- essential innovations:
  - decomposition of atomic formulas into **predicates** (cf. verbs) and **arguments** (cf. subject, object)
  - **quantification**: counterparts to English words like *all, some*

Predicate logic: atomic formulas

Syntax

- *jo*, *bertie*, *ethel*, *the-cake* ... are individual constants (names)
- *Run*, *Laugh*, *Howl*, *Sing*, ... are one-place predicates
- *Rain*, *Snow*, ... are zero-place predicates
- *Eat*, *Like*, *Loath* ... are two-place predicates
- *Give* is a three-place predicate

Predicate logic: atomic formulas

Syntax

Definition

1. *There are infinitely many individual constants.*
2. *For each natural number n there are infinitely many n-place predicates.*
3. *If P is an n-place predicate and $c_1, \ldots, c_n$ are individual constants, then $P(c_1, \ldots, c_n)$ is an atomic formula.*
4. *Every atomic formula is a formula.*
5. *If φ and ψ are formulas, then $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$ and $\varphi \leftrightarrow \psi$ are also formulas.*

Predicate logic: atomic formulas

- remarks on the notational conventions (for these slides; different authors use different conventions):
  - atomic statements of statement logic can be considered as 0-place predicates
  - predicates and individual constants are written in italic latin letters
  - predicate names start with uppercase letters and individual constants with lowercase letters
  - predicates and individual constants need not resemple words from a natural language — it is just convenient to do so

Predicate logic: atomic formulas

- names that just consist of a single letter are also common, e.g.:

$$P(a) \wedge R(d, e) \rightarrow Q(a, d, e)$$

- parantheses around arguments and commas between arguments are sometimes omitted.

$$Pa \wedge Rde \rightarrow Qade$$

Predicate logic: atomic formulas

Examples

- $Run(jo)$
- $Rain$
- $Like(bertie, ethel)$
- $Give(bertie, ethel, the\text{-}cake)$
- $Run(jo) \to Give(bertie, ethel, the\text{-}cake)$
- $Rain \vee \neg Rain$
- $\neg Like(ethel, bertie) \to \neg Like(bertie, ethel)$

Translation: atomic statements

- rule of thumb for translation:
  - atomic clauses are translated as atomic formulas
  - proper nouns and definite descriptions are translated as individual constants
  - 0-place verbs (like *to rain*) are translated as 0-place predicates
  - intransitive verbs and predicateive adjectives are translated as 1-place predicates
  - transitive verbs are translated as 2-place predicates
  - ditransitive verbs are translated as 3-place predicates

Translation: atomic statements

(1)
a. Chester ran.
b. Chester ate the cake.
c. Ethel gave the cake to Jo.
d. Ethel gave Chester the cake.
e. It rained.

(2)
a. $Run(chester)$
b. $Eat(chester, the\text{-}cake)$
c. $Give(ethel, the\text{-}cake, jo)$
d. $Give(ethel, the\text{-}cake, jo)$
e. $Rain$

Predicate logic: atomic statements

- Semantics
  - model M, consists of
    - individual domain E and
    - interpretation function F
  - interpretation function maps
    - individual constants to elements of E, and
    - n-place predicates to n-place relations over E.
  - extension to formulas:

$$\begin{array}{rcl} [P(c_1, \ldots, c_n)]^M = 1 & \text{iff} & \langle F(c_1), \ldots, F(c_n)\rangle \in F(P) \\ [c_1 = c_2]^M = 1 & \text{iff} & F(c_1) = F(c_2) \end{array}$$

Predicate logic: an example

$$
\begin{aligned}
M &= \langle E, F\rangle \\
E &= \{\textbf{DOG}, \textbf{CAT}, \textbf{MAN}_1, \textbf{MAN}_2, \textbf{WOMAN}_1, \\
&\qquad \textbf{WOMAN}_2, \textbf{CAKE}\} \\
F(\textit{jo}) &= \textbf{MAN}_1 \\
F(\textit{bertie}) &= \textbf{MAN}_2 \\
F(\textit{ethel}) &= \textbf{WOMAN}_1 \\
F(\textit{fiona}) &= \textbf{WOMAN}_2 \\
F(\textit{chester}) &= \textbf{DOG} \\
F(\textit{prudence}) &= \textbf{CAT} \\
F(\textit{the-student}) &= \textbf{MAN}_1 \\
F(\textit{the-cat}) &= \textbf{CAT} \\
F(\textit{the-cake}) &= \textbf{CAKE}
\end{aligned}
$$

Predicate logic: an example

$$
\begin{aligned}
F(\textit{Run}) &= \{\textbf{DOG}, \textbf{CAT}\} \\
F(\textit{Laugh}) &= \{\textbf{MAN}_1, \textbf{WOMAN}_1\} \\
F(\textit{Howl}) &= \{\textbf{DOG}\} \\
F(\textit{Sing}) &= \{\textbf{WOMAN}_2\} \\
F(\textit{Scream}) &= \emptyset \\
F(\textit{Crazy}) &= \emptyset \\
F(\textit{Disgusting}) &= \{\textbf{CAKE}\} \\
F(\textit{Wealthy}) &= \{\textbf{MAN}_2\} \\
F(\textit{Happy}) &= \{\textbf{MAN}_1, \textbf{MAN}_2, \textbf{WOMAN}_1\} \\
F(\textit{Messy}) &= \emptyset
\end{aligned}
$$

Predicate logic: an example

$$
\begin{aligned}
F(\textit{Like}) \quad = \quad & \{\langle \textbf{MAN}_1, \textbf{WOMAN}_1\rangle \\
& \langle \textbf{MAN}_1, \textbf{MAN}_2\rangle, \\
& \langle \textbf{MAN}_1, \textbf{WOMAN}_2\rangle, \\
& \langle \textbf{MAN}_1, \textbf{MAN}_1\rangle, \\
& \langle \textbf{WOMAN}_1, \textbf{WOMAN}_2\rangle, \\
& \langle \textbf{WOMAN}_1, \textbf{MAN}_1\rangle, \\
& \langle \textbf{WOMAN}_1, \textbf{CAT}\rangle, \\
& \langle \textbf{WOMAN}_2, \textbf{WOMAN}_1\rangle\}
\end{aligned}
$$

Predicate logic: an example

$$
\begin{array}{rcl}
F(\mathit{Loathe}) & = & \{\langle \textbf{MAN}_1, \textbf{DOG}\rangle \\
& & \langle \textbf{MAN}_2, \textbf{DOG}\rangle, \\
& & \langle \textbf{WOMAN}_1, \textbf{DOG}\rangle, \\
& & \langle \textbf{WOMAN}_2, \textbf{DOG}\rangle, \\
& & \langle \textbf{WOMAN}_2, \textbf{MAN}_1\rangle, \\
& & \langle \textbf{CAT}, \textbf{DOG}\rangle\} \\
F(\mathit{Poison}) & = & \{\langle \textbf{CAKE}, \textbf{DOG}\rangle\} \\
F(\mathit{Eat}) & = & \{\langle \textbf{DOG}, \textbf{CAKE}\rangle\} \\
F(\mathit{Read}) & = & \{\langle \textbf{WOMAN}_1, \textbf{BOOK}\rangle, \\
& & \langle \textbf{MAN}_2, \textbf{BOOK}\rangle\} \\
F(\mathit{Kick}) & = & \emptyset
\end{array}
$$

Predicate logic: an example

$$
\begin{array}{rcl}
F(\mathit{Give}) & = & \{\langle \textbf{WOMAN}_2\textbf{, CAKE, MAN}_1\rangle, \\
 & & \langle \textbf{MAN}_1, \textbf{CAKE}, \textbf{DOG}\rangle, \\
 & & \langle \textbf{MAN}_1, \textbf{BOOK}, \textbf{MAN}_2\rangle, \\
 & & \langle \textbf{MAN}_2, \textbf{BOOK}, \textbf{WOMAN}_1\rangle, \\
 & & \langle \textbf{MAN}_1, \textbf{CAT}, \textbf{WOMAN}_2\rangle\} \\
F(\mathit{Rain}) & = & 1 \\
F(\mathit{Snow}) & = & 0
\end{array}
$$

Predicate logic: an example

(3)
a. Ethel was happy and laughed.
b. Fiona sang and was happy.
c. Bertie was wealthy.
d. The dog ran and howled.
e. The cat ran.
f. The cake was disgusting.

Variables

- instead of names one can use pronouns
- statements with pronouns usually don't have a definite truth value, even if the model is known

(4)
  a. He ran.
  b. Chester ate it.
  c. She gave it to him.
  d. She gave him the cake.

- truth value depends on what the pronoun refers to
- pronouns are translated as **variables** in predicate logic

Variables

- conventions:
  - variables are written as italic latin lowercase letters from the end of the alphabet, sometimes augmented with indices or primes

$$x, y, z', w_3, ...$$

  - variables with the same name refere to the same ("unknown") object
  - variables with different names can refer to different things (but need not)

Variables

- translation conventions
  - pronouns are translated as variables if and only if they are not co-referent (in the particular context) with a name/constant
  - if two pronouns refer to the same object, they are translated as the same variable
  - if two pronouns do not (or not necessarily) refer to the same object, they are translated as different variables

(5)
a. He walks.
b. Peter knows him.
c. If Peter walks, he sings.
d. If Peter walks, she things.
e. Hans shaves himself.
f. He shaves himself.
g. He shaves him.

Variables

- translation conventions
  - pronouns are translated as variables if and only if they are not co-referent (in the particular context) with a name/constant
  - if two pronouns refer to the same object, they are translated as the same variable
  - if two pronouns do not (or not necessarily) refer to the same object, they are translated as different variables

(5)
a. He walks. $\rightsquigarrow Wx$
b. Peter knows him. $\rightsquigarrow Kpy$
c. If Peter walks, he sings. $\rightsquigarrow Wp \rightarrow Sp$
d. If Peter walks, she things. $\rightsquigarrow Wp \rightarrow Sx$
e. Hans shaves himself. $\rightsquigarrow Shh$
f. He shaves himself. $\rightsquigarrow Sww$
g. He shaves him. $\rightsquigarrow Sxy$

Variables

- natural language: pronouns can be ambiguous

  He saw him, and he enjoyed it (he$\rightsquigarrow$ seer/seen)

- predicate logic: ambiguity is resolved by choice of variable names
- Statements from predicate logic (and from almost any other logical language) are never ambiguous!

$$Sxy \wedge Ex \text{ or } Sxy \wedge Ey$$

Interpretation of variables

- variables refer, just like individual constants, to individuals, i.e. elements of E
- therefore: official name is **individual variables**
- unlike individual constants, there reference is not fixed by the model

  Knowing the model amounts to being omniscient; if you know the model, you know all the relevant facts. You know the interpretation of all predicates and constants, and thus the truth values of all statements, to the degree that they have a definite truth value. But even if you know all the facts, you may not know what object a speaker has in mind when she uses a third person personal pronoun/a variable.

Interpretation of variables

- interpretation of variables is not entirely arbitrary
- different occurrences of the same variable refer to the same object
- some formulas are true/false independent of the reference of the variables occurring in them

$$Px \vee \neg Px$$
$$\mathit{Loathe}(x, x)$$
$$\mathit{Messy}(w)$$

- therefore: interpretation of variables is fixed via an **assignment function**; the function "assigns" a value to each variable

Variables: syntax

Definition (Syntax of predicate logic, second version)

1. *There are infinitely many individual constants.*
2. *There are infinitely many individual variables.*
3. *Every individual constants and every individual variable is a term.*
4. *For each natural number n there are infinitely many n-place predicates.*
5. *If P is an n-place predicate and $t_1, \ldots, t_n$ terms, then $P(t_1, \ldots, t_n)$ is an atomic formula.*
6. *Every atomic formula is a formula.*
7. *If φ and ψ are formulas, then $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \rightarrow \psi$ and $\varphi \leftrightarrow \psi$ are also formulas.*

Variables: semantics

Definition (assignment function)

An assignment function g for a model $M = \langle E, F \rangle$ is a function from the set of variables into the individual domain E.

Variables: semantics

Definition (Semantics of predicate logic (provisional))

Let $M = \langle E, F\rangle$ be a model, and g an assignment function for M.

1. $[c]_g^M = F(c)$, *if c is an individual constant.*
2. $[v]_g^M = g(v)$, *if v is an individual variable.*
3. $[P(t_1, \ldots, t_n)]_g^M = 1$ *iff* $\langle [t_1]_g^M, \ldots, [t_n]_g^M\rangle \in F(P)$
4. $[\neg\varphi]_g^M = 1 - [\varphi]_g^M$
5. $[\varphi \wedge \psi]_g^M = \min([\varphi]_g^M, [\psi]_g^M)$
6. $[\varphi \vee \psi]_g^M = \max([\varphi]_g^M, [\psi]_g^M)$
7. $[\varphi \rightarrow \psi]_g^M = \max(1 - [\varphi]_g^M, [\psi]_g^M)$
8. $[\varphi \leftrightarrow \psi]_g^M = 1 - ([\varphi]_g^M - [\psi]_g^M)^2$