EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Faculty of Philosophy
General Linguistics

Semantics & Pragmatics SoSe 2023

Lecture 6: Formal Semantics III
(Second-Order Logic)

11/05/2023, Christian Bentz

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Overview

Section 1: Recap of Lecture 5

The Vocabulary

Similar as for propositional logic, we can define a **language** *L* **for predicate logic**. In this case, the "vocabulary" of *L* consits of

- a (potentially infinite) supply of **constant symbols** (e.g. a, b, c, etc.),
- a (potentially infinite) supply of **variable symbols** representing the constants (e.g. x, y, z, etc.),
- a (potentially infinite) supply of **predicate symbols** (e.g. A, B, C, etc.),
- the **connectives** (e.g. $\neg$, $\wedge$, $\vee$, $\rightarrow$, etc.),
- the **quantifiers** $\forall$ and $\exists$,
- as well as the round brackets '(' and ')'.
- (The equal sign '='.)

English sentences:

(1) Socrates admires someone.

(2) Socrates is admired by someone.

(3) All teachers are friendly.

(4) Some teachers are friendly.

(5) Some friendly people are teachers.

(6) All teachers are unfriendly.

(7) Some teachers are unfriendly.

Translations:

(1) $\exists$yAsy

(2) $\exists$xAxs

(3) $\forall$x(Tx $\rightarrow$ Fx)

(4) $\exists$x(Tx $\wedge$ Fx)

(5) $\exists$x(Fx $\wedge$ Tx)

(6) $\forall$x(Tx $\rightarrow$ $\neg$Fx)

(7) $\exists$x(Tx $\wedge\neg$Fx)

Notes:
We have to add *Tx: x is a teacher* to the key.

Due to the so-called commutatitivity of $\wedge$, i.e. $\phi \wedge \psi \equiv \psi \wedge \phi$, we have that the predicate logic expressions in (4) and (5) are seen as equivalent too. However, the asymmetry might be seen as actually relevant in the natural language examples.

Generally, we have that $\forall$x$\neg\phi \equiv \neg\exists$x$\phi$.

The Syntax: Recursive Definition

Given the vocabulary of L we then define the following clauses to create formulas of L.

(i) If A is an n-ary predicate letter in the vocabulary of L, and each of $t_1, \ldots, t_n$ is a constant or a variable in the vocabulary of L, then $At_1, \ldots, t_n$ is a formula in L.

(ii) If ϕ is a formula in L, then $\neg\phi$ is too.

(iii) If ϕ and ψ are formulas in L, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ are too.

(iv) If ϕ is a formula in L and x is a variable, then $\forall x\phi$ and $\exists x\phi$ are formulas in L.

(v) Only that which can be generated by the clauses (i)-(iv) in a finite number of steps is a formula in L.

Gamut, L.T.F (1991). Volume 1, p. 75.

Clarification: the number *n*

"In general, n-ary predicates may be introduced for any whole number n larger than zero."

Gamut (1991), Volume 1, p. 67.

"More precisely, a lexicon of predicate logic is a function assigning to any natural number $n \geq 0$ a set $\text{PRED}_{n,L}$ of (primitive) symbols [...] For $n = 0$, PRED_n contains the individual constants of L; and if $n \geq 1$, the members of PRED_n are called *n*-place predicates (of L)."

Zimmermann and Sternefeld (2013), p. 245.

Predicates: $n > 0$
Functions: $n \geq 0$

Examples of Valid and Invalid Formulas

Formula	Rule Applied
Aa ✓	(i)
Ax ✓	(i)
Aab ✓	(i)
Axy ✓	(i)
$\neg$Axy ✓	(i) and (ii)
Aa $\rightarrow$ Axy ✓	(i) and (iii)
$\forall$x(Aa $\rightarrow$ Axy) ✓	(i),(iii), and (iv)
$\forall$xAa $\rightarrow$ Axy ✓	(i),(iii), and (iv)
a x	–
A x	–
$\forall$ x	–
$\forall$(Axy) x	–

Definition: *Formula* vs. *Sentence*

There is a further distinction between **formulas** and **sentences** in predicate logic. Namely, sentences are a subset of formulas for which it holds that: "A sentence is a formula in *L* which **lacks free variables**."[1]

Gamut, L.T.F (1991). Volume 1, p. 77.

Sentence	**Not a Sentence (but Formula)**
Aa	Ax
$\forall$x(Fx)	Fx
$\forall$x(Ax $\rightarrow$ $\exists$yBy)	Ax $\rightarrow$ $\exists$yBy

[1] Free variables, in turn, are precisely defined by Gamut (1991), p.77 in their Definition 3. We will simply state here that a variable is free if it is not within the scope of a quantifier.

Model Theory

"In order to develop and test a set of interpretive rules [...] it is important to provide very explicit descriptions for the test situations. As stated above, this kind of **description of a situation is called a model**, and must include two types of information: (i) **the domain**, i.e., the set of all individual entities in the situation; and (ii) the **denotation sets for the basic vocabulary items** [constant symbols, predicates] in the expressions being analyzed."

Kroeger (2019). Analyzing meaning, p. 240.

Interpretation Functions

"The interpretation of the constants in L will therefore be an attribution of some entity in D to each of them, that is, a function with the set of constants in L as its domain and D as its range. Such functions are called **interpretation functions**."

$$I(c) = e. \tag{1}$$

"$I(c)$ is called the *interpretation of a constant c*, or its *reference* or its *denotation*, and if e is the entity in D such that $I(c) = e$, then c is said to be one of e's names (e may have several different names)."

Gamut, L.T.F (1991). Volume 1, p. 88.

Example

$I = \{\langle m, e_1\rangle, \langle s, e_1\rangle, \langle v, e_1\rangle\}$
$I(m) = e_1$
$I(s) = e_1$
$I(v) = e_1$
Translation key: m: morning star; s: evening star; v: venus.

Valuation Example

Given a Model of the world **M**, consisting of D and I, and some formula ϕ which adheres to predicate logic syntax (and which consists of atomic sentences and or quantifications with bound variables), we can then evaluate the truth of ϕ as follows.

Model **M**

$D = \{e_1, e_2, e_3\}$
$I = \{\langle j, e_1\rangle, \langle p, e_2\rangle, \langle m, e_3\rangle, \langle S, \{\langle I(j), I(m)\rangle, \langle I(p), I(m)\rangle\}\rangle\}$
Translation key: j: John; p: Peter; m: morning star; Sxy: x sees y.

Valuation

"John sees the morning star": V_M(Sjm) = 1 (according to (i))
"Everybody sees the morning star": V_M(∀xSxm) = 0 (according to (vii))[2]

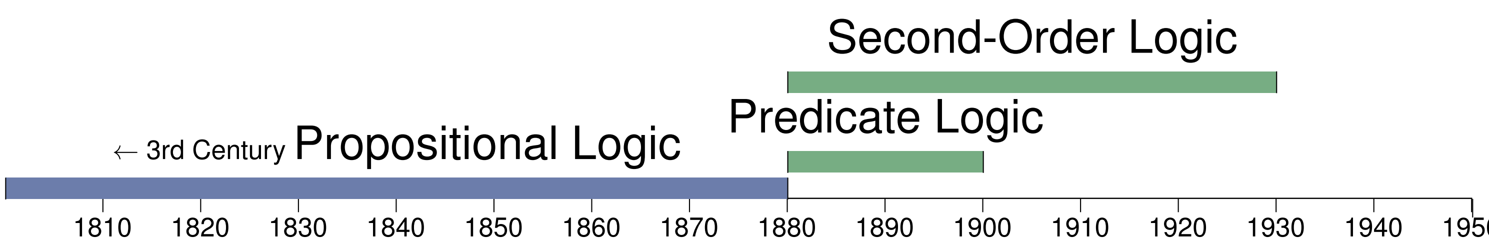
[2]This valuation gives 0 since the morning star (m) is a constant c in L, but it does not see itself, i.e. $\langle I(m), I(m)\rangle \notin S$.

Section 2: Historical Notes

Historical Background

"Second-order logic was introduced by Frege in his *Begriffsschrift* (1879) who also coined the term "second order" ("zweiter Ordnung") in (1884: paragraph 53). It was widely used in logic until the 1930s, when set theory started to take over as a foundation of mathematics."

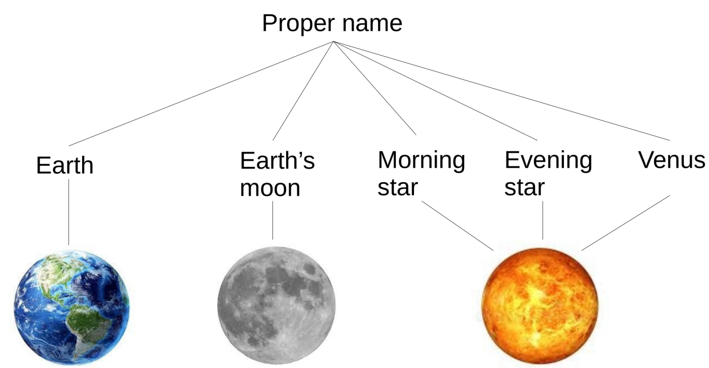
https://plato.stanford.edu/entries/logic-higher-order/



Frege’s Original Statement

Sprache folgend zuschreiben möchte. Wenn man z. B. alle Begriffe, unter welche nur Ein Gegenstand fällt, unter einen Begriff sammelt, so ist die Einzigkeit Merkmal dieses Begriffes. Unter ihn würde z. B. der Begriff „Erdmond,“ aber nicht der sogenannte Himmelskörper fallen. So kann man einen Begriff unter einen höhern, so zu sagen einen Begriff zweiter Ordnung fallen lassen. Dies Verhältniss ist aber nicht mit dem der Unterordnung zu verwechseln.

[...] If you collect, for instance, all notions [*Begriffe*], which have the property of denoting just a single object, under another notion, then the uniqueness is the property of this other notion. For example, the notion “earth’s moon” would fall under this other notion, but not the celestial object itself. Hence, you can have a given notion fall under another notion, a **notion of second order** so to speak. [...]



Frege (1884), p. 65 (paragraph 53).

Historical Background

"In present terminology, the thesis is that languages with **variables ranging over properties and relations** are natural extensions of languages with **variables ranging over objects**."

Shapiro (1991), p. 203.

x, y, z: variables ranging over "objects" (i.e. constants)
X, Y, Z: variables ranging over "properties" and relations (i.e. predicates)

Dispute over Second-Order Logic

It is disputed whether second-order predicates are necessarily needed in logical systems generally, and for natural language logic in particular. Some of the reasons for this include:

- There is **no completeness theorem** for second-order logic (see Gamut 1991, Volume 1, p. 171), while for first-order logic there is.
- W. V. Quine rejected the idea that **quantification over predicates** makes sense. He conceptualized predicates as an abbreviation for an incomplete sentence, e.g. F standing for "...is friendly", and such *incomplete sentences* are not to be seen as *objects* to quantify over.

See also discussion on *https://en.wikipedia.org/wiki/Second-order_logic* under *History and disputed value*.

Section 3: Beyond First-Order Logic

Beyond First-Order Logic

We have seen that predicate logic is an extension of propositional logic, by introducing predicates and quantifiers. **First-Order Predicate Logic** might itself be superseded by another logical system, called **Second-Order Predicate Logic**.

Gamut, L.T.F (1991). Volume 1, p. 168.

Take the following English sentences:

(1) Mars is red.
(2) Red is a color.
(3) Mars has a color.
(4) John has at least one thing in common with Peter.

How can we translate these into logical expressions?

The adjective “red” is a **property of individuals**. Hence, the first sentence can be straightforwardly translated into predicate logic notation as

(5) Rm (**Rx**: x is red, m: Mars)

What about the second sentence? We could stick with standard predicate notation and translate it into

(6) Cr (Cx: x is a color, **r**: red)

Note however, that now we have treated “red” once as a property of individuals in (5), and once as an individual itself in (6). In predicate logic terms it is once represented as a **predicate constant**, and once as a **constant**.

Second-Order Predicates

To circumvent this discrepancy, we can construe the predicate *x is a color* not as a property, but as a **property of properties**. $\mathcal{C}$ then represents a so-called **second-order property**, i.e. a **second-order predicate** over the first-order predicate *x is red*.

Instead of

(7) Cr (Cx: x is a color, r: red),

we then get

(8) $\mathcal{C}$R ($\mathcal{C}$X: X is a predicate with the property of being a color, Rx: x is red)

Note: We introduce **two** new sets of symbols here compared to standard predicate logic, a) the set of *second-order predicates*, and b) the set of *first-order predicate variables*. See details below.

First-Order and Second-Order Logic

A **second-order logic** language L' is then an extension to a standard predicate logic language L by adding second-order predicates to L. The original language L is then sometimes referred to as **first-order logic** language.

Further Examples:

(9) $\exists X(\mathcal{C}X \wedge Xm)$ (English sentence: "Mars has a color.")

(10) $\exists X(Xj \wedge Xp)$ (English sentence: "John has at least one thing in common with Peter.")

(11) $\exists \mathcal{X}(\mathcal{X}R \wedge \mathcal{X}G)$ (English sentence: "Red has something (a property) in common with green.")

Section 3: The Vocabulary

Vocabulary (shared with First-Order Logic)

The vocabulary of a second-order logic language *L* consists of symbols which are *shared with first-order logic languages*, and some which need to be introduced especially to fit the *second-order properties*. The once **shared with first-order logic** languages are:

- A (potentially infinite) supply of constant symbols (e.g. a, b, c, etc.),
- a (potentially infinite) supply of variable symbols representing the constants (e.g. x, y, z, etc.),
- a (potentially infinite) supply of **first-order predicate constants** (e.g. A, B, C, etc.),
- the connectives (e.g. $\neg$, $\wedge$, $\vee$, $\rightarrow$, etc.),
- the quantifiers $\forall$ and $\exists$,
- as well as the round brackets '(' and ')'.

Vocabulary (special to Second-Order Logic)

The vocabulary extensions to fit **second-order logic requirements** are:

- A (potentially infinite) supply of **first-order predicate variables** (e.g. X, Y, Z, etc.), which are necessary to quantify over first-order predicates,
- a (potentially infinite) supply of **second-order predicate constants** (e.g. $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, etc.).

If we wanted to take it to a higher-order level we could also have:

- a (potentially infinite) supply of **second-order predicate variables** (e.g. $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$, etc.) to stand in for second-order predicates.

Example of a Translation Key

Constants	First-Order Pred.	Second-Order Pred.
j: Jumbo	B_1x: x is a bee	$\mathcal{A}X$: X is a property with the property of being an animal
s: Simba	Ex: x is an elephant	$\mathcal{C}X$: X is a property with the property of being a color
b: Bambi	Lx: x is a lion	
m: Maya	Dx: x is a deer	
	B_2: x has big ears	
	Fx: x is fast	
	Gx: x is gray	
	Yx: x is yellow	
	B_3: x is brown	
	Cxy: x chases y	

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Faculty of Philosophy
General Linguistics

Section 4: The Syntax of Second-Order Logic

The Syntax: Recursive Definition

Given the vocabulary of L we then define the following clauses to create formulas of *L*:

(i) If A is an n-ary **first-order** predicate letter/constant in *L*, and $t_1, \ldots, t_n$ are individual terms in *L*, then $At_1, \ldots, t_n$ is an (atomic) formula in *L*;

(ii) If X is a [**first-order**] predicate variable and t is an individual term in *L*, then Xt is an atomic formula in *L*;

(iii) If $\mathcal{A}$ is an n-ary **second-order** predicate letter/constant in *L*, and $T_1, \ldots, T_n$ are **first-order unary** predicate constants, or predicate variables, in *L*, then $\mathcal{A}T_1, \ldots, T_n$ is an (atomic) formula in *L*;

(iv) If ϕ is a formula in *L*, then $\neg\phi$ is too;

(v) If ϕ and ψ are formulas in *L*, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ are too.

The Syntax: Recursive Definition

(vi) If *x* is an individual variable ϕ is a formula in *L*, then $\forall x\phi$ and $\exists x\phi$ are also formulas in *L*;

(vii) If *X* is a [first-order] predicate variable, and ϕ is a formula in *L*, then $\forall X\phi$ and $\exists X\phi$ are also formulas in *L*;

(viii) Only that which can be generated by the clauses (i)-(vii) in a finite number of steps is a formula in *L*.

Gamut, L.T.F (1991). Volume 1, p. 170.

Note: In the above clauses (i) and (ii), the word "term" includes both constants and variables (of constants), i.e. a, b, c, etc. and x, y, z, etc.

Note on Clause (ii)

(ii) If X is a [**first-order**] predicate variable and t is an individual term in *L*, then Xt is an atomic formula in *L*;

Note: About the question of why we have a single term *t* here as the argument of *X*, Gamut (1991), Volume I, p. 169): "The variable X is a variable over properties. We here shall disregard variables over relations between entities, since they complicate everything without introducing anything really new. (In the logic of types with lambda abstraction there is another and better approach; see vol. 2)."

Discussion Point

Can you come up with an example in natural language for which we would need a *variable for relations between entities*?

First-Order Predicate Logic

(i) If A is an n-ary predicate letter in the vocabulary of L, and each of $t_1, \ldots, t_n$ is a constant or a variable in the vocabulary of L, then $At_1, \ldots, t_n$ is a formula in L.

(ii) If ϕ is a formula in L, then $\neg\phi$ is too.

(iii) If ϕ and ψ are formulas in L, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ are too.

(iv) If ϕ is a formula in L and x is a variable, then $\forall x\phi$ and $\exists x\phi$ are formulas in L.

(v) Only that which can be generated by the clauses (i)-(iv) in a finite number of steps is a formula in L.

Second-Order Predicate Logic

(i) If A is an n-ary **first-order** predicate letter/constant in L, and $t_1, \ldots, t_n$ are individual terms in L, then $At_1, \ldots, t_n$ is an (atomic) formula in L;

(ii) If X is a [**first-order**] predicate variable and t is an individual term in L, then Xt is an atomic formula in L;

(iii) If $\mathcal{A}$ is an n-ary **second-order** predicate letter/constant in L, and $T_1, \ldots, T_n$ are **first-order unary** predicate constants, or predicate variables, in L, then $\mathcal{A}T_1, \ldots, T_n$ is an (atomic) formula in L;

(iv) If ϕ is a formula in L, then $\neg\phi$ is too;

(v) If ϕ and ψ are formulas in L, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ are too.

(vi) If x is an individual variable ϕ is a formula in L, then $\forall x\phi$ and $\exists x\phi$ are also formulas in L;

(vii) If X is a [first-order] predicate variable, and ϕ is a formula in L, then $\forall X\phi$ and $\exists X\phi$ are also formulas in L;

(viii) Only that which [...].

Examples of Valid and Invalid Formulas

Formula	Rule Applied
Aa ✓	(i)
Ax ✓	(i)
Axy ✓	(i)
Xa ✓	(ii)
Xx ✓	(ii)
$\mathcal{A}A$ ✓	(iii)
$Xa \rightarrow \neg Xb$ ✓	(ii), (iv) and (v)
$\forall X \forall x(Xa \rightarrow Axy)$ ✓	(i),(ii), (v), (vi), and (vii)
x x	–
X x	–
Xab x	–
$\forall(Xa)$ x	–

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Faculty of Philosophy
General Linguistics

Section 5: The Semantics of Second-Order Logic

The Semantics of Second-Order Logic

Similar as for the syntax of second-order logic, its semantics can also be defined based on what has been defined for first-order logic before.

For instance, just as a **first-order predicate** denotes a **set of entities**, a **second-order predicate** denotes a **set of a set of entities**.

However, since the formal definitions of valuation functions get increasingly more complex, and the interpretation with regards to natural language examples more abstract, we will not further delve into the issue here.

Gamut, L.T.F (1991). Volume 1, p. 173-174.

Summary

Summary

- **Second-order predicate logic** goes beyond first-order predicate logic by, firstly, introducing **predicate variables**, which allow to quantify over first-order predicates, and secondly, by introducing **second order predicates**, which are to be seen as properties of properties, i.e. predicates over predicates.
- These changes lead to **adjustments in the formal definitions** of the syntax and semantics of the logical language L.
- These adjustments enable the translation of a wider array of natural language sentences, although there are still natural language phenomena not captured appropriately.

References

References

Gamut, L.T.F (1991). *Logic, Language, and Meaning. Volume 1: Introduction to Logic*. Chicago: University of Chicago Press.

Kroeger, Paul R. (2019). *Analyzing meaning. An introduction to semantics and pragmatics.* Second corrected and slightly revised version. Berlin: Language Science Press.

Frege, Gottlob (1884). *Die Grundlagen der Arithmetik. Eine logisch mathematische Untersuchung über den Begriff der Zahl.* Verlag von Wilhelm Koebner, Breslau.

Shapiro, Stewart (1991). *Foundations without foundationalism. A case for Second-Order Logic*. Oxford: Clarendon Press.

Zimmermann, Thomas E. & Sternefeld, Wolfgang (2013). *Introduction to semantics. An essential guide to the composition of meaning.* Mouton de Gruyter.

Thank You.

Contact:

Faculty of Philosophy
General Linguistics
Dr. Christian Bentz
SFS Wilhelmstraße 19-23, Room 1.15
chris@christianbentz.de
Office hours:
During term: Wednesdays 10-11am
Out of term: arrange via e-mail