Elizabeth Coppock
coppock@phil.hhu.de
Wed. October 12th, 2011
Time: 14:30–16:00

Compositional Semantics
Heinrich Heine University
Winter Semester 2011/12
Room: 25.22-U1.72

Introduction to Compositional Semantics

1 Semantics

Semantics: The study of meaning.

What is meaning? Clue: The meanings of the English word *means* all have to do with consequences:

- Epistemic consequences: Smoke means fire (Grice 1957)
- Practical consequences: This means we're going to have to move
- Logical consequences: Being a bachelor means being unmarried

How can you tell whether somebody or something understands? Does Google understand language?

An argument that it does not: Google can't do *inferences*:

(1) Obama was born in 1961 $\vDash$ Obama was born in the 1960s

(2) JFK was assassinated $\vDash$ JFK is dead

The left sentences *imply* the right sentences.
The right sentences *follow from* the left sentences.
The right sentences are *consequences* of the left sentences.
The right sentences can be *inferred* from the left sentences.

A hallmark of a system or agent that understands language / grasps meaning is that it can infer the right sentences from the left sentences.

Example of a computer system designed to be able to do inferences: Cyc (Symbolic Artificial Intelligence): Reasoning + commonsense/world knowledge. Recent trend in computer science: Recognizing Textual Entailments.

Doing semantics you can think of yourself as a software engineer, building a machine that is capable of inferring sentences from other sentences.

Kinds of inferences:

1. Entailment (domain of semantics)
2. Presupposition (semantics/pragmatics)
3. Implicature (pragmatics)

In this course, we focus primarily on entailments. What are entailments? Chierchia and McConnell-Ginet (1990) give four alternative definitions for 'A entails B':

- Whenever A is true, B is true
- The information that B conveys is contained in the information that A conveys
- A situation describable by A must also be a situation describable by B
- *A and not B* is contradictory (can't be true in any situation)

To understand why these all mean basically the same thing, we have to understand a little bit of logic.

Logic also helps us understand the view that "To know the meaning of a sentence is to know its truth-conditions" (Heim and Kratzer 1998). What is the relationship between truth conditions and semantic entailment?

2 Montague semantics

Formal semantics uses tools from logic to capture the meanings of sentences (primarily entailments, but also presuppositions). It is inspired primarily by the work of Richard Montague, who said:

> I reject the contention that an important theoretical difference exists between formal and natural languages. ... In the present paper I shall accordingly present a precise treatment, culminating in a theory of truth, of a formal language that I believe may reasonably be regarded as a fragment of ordinary English. ... The treatment given here will be found to resemble the usual syntax and model theory (or semantics) [due to Tarski] of the predicate calculus, but leans rather heavily on the intuitive aspects of certain recent developments in intensional logic [due to Montague himself]. (Montague 1970b, p.188 in Montague 1974)

What does he mean, syntax and model theory of the predicate calculus?

Predicate calculus is a formal language; a logic. Logics have a syntax and a semantics.

Syntax: specifies which formulas of the logic are well-formed.

Semantics: specifies which objects (given a model) the formulas correspond to.

Two examples of statements in predicate calculus:

$$\text{LOVE}(\text{JOHN}, \text{MARY})$$

$$\forall x[\text{LOVE}(\text{MARY}, x) \Rightarrow \text{HAPPY}(x)]$$

Syntactic categories. Formulas are built up from the following expressions of the following syntactic categories:

- individual constants
- variables
- predicate constants
- logical connectives
- quantifiers

Semantic types. Each expression belongs to a certain semantic type. The types of PC are:

- individuals
- relations
- truth values (True or False)

For our examples we have:

Expressions	Syntactic categories	Semantic Type
JOHN,MARY	(individual) constant	individual
x	variable	individual
HAPPY	unary predicate constant	unary relation
LOVE	binary predicate constant	binary relation
$\text{LOVE}(\text{JOHN}, \text{MARY})$	formula	truth value
$\text{HAPPY}(x)$	formula	truth value
$\forall x[\text{LOVE}(\text{MARY}, x) \Rightarrow \text{HAPPY}(x)]$	formula	truth value

Montague's idea: English as a formal language. Montague's idea is to give a semantics for a fragment of English in the same way that we define semantics for formal logics like predicate calculus.

Example. Lexical entries:[1]

- ⟦Kim⟧ = Kim
- ⟦run⟧ $= \{x | x$ runs$\}$

[1]Taken from a handout by Chris Kennedy.

- $[\![\text{smoke}]\!] = \{x | x \text{ smokes}\}$
- $[\![\text{not}]\!] =$ set complementation; $[\![\text{not}]\!](A) = \{x | x \notin A\}$
- $[\![\text{and}]\!] =$ set intersection; $[\![\text{and}]\!](A, B) = \{x | x \in A \text{ and } x \in B\}$

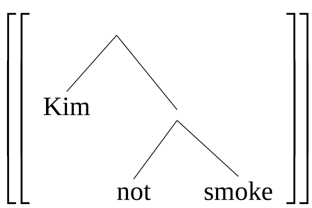
Modes of Composition.

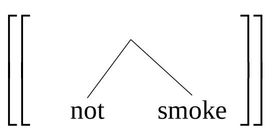
- **Predication**
  If α is a constituent whose immediate subconstituents are β and γ, and if $[\![\beta]\!]$ is an individual and $[\![\gamma]\!]$ a set of individuals, then $[\![\alpha]\!] = 1$ if $[\![\beta]\!] \in [\![\gamma]\!]$ (i.e., if $[\![\beta]\!]$ is a member of $[\![\gamma]\!]$), and 0 otherwise.

- **1-place application**
  If α is a constituent whose immediate subconstituents are β and γ, and if $[\![\beta]\!]$ is a 1-place set-theoretic operation (e.g. set complementation) and $[\![\gamma]\!]$ is a set of individuals, then $[\![\alpha]\!] = [\![\beta]\!]([\![\gamma]\!])$.

- **2-place application**
  If α is a constituent whose immediate subconstituents are βx, γ and δ, and if $[\![\beta]\!]$ is a 2-place set-theoretic operation (e.g. set intersection) and $[\![\gamma]\!]$ and $[\![\delta]\!]$ are sets of individuals, then $[\![\alpha]\!] = [\![\beta]\!]([\![\gamma]\!], [\![\delta]\!])$. ($[\![\alpha]\!]$ is the result return by applying $[\![\beta]\!]$ to the pair $[\![\gamma]\!], [\![\delta]\!]$.)

Computing truth conditions

1. 

$= 1$ if and only if...
2. $[\![\text{Kim}]\!] \in$ 

[**Predication**]
3. $[\![\text{Kim}]\!] \in [\![\text{not}]\!]([\![\text{smoke}]\!])$ [**1-pl app.**]
4. $[\![\text{Kim}]\!] \in \{x | x \notin [\![\text{smoke}]\!]\}$
5. $[\![\text{Kim}]\!] \in \{x | x \notin \{y | y \text{ smokes}\}$
6. $[\![\text{Kim}]\!] \notin \{x | \text{smokes}\}$