LING 564, Formal Semantics, week of January 18 - 20
Discussion of de Swart, ch. 1-2, and Heim & Kratzer, ch. 1

0. Agenda

Three conceptual areas:

(1) Semantic values, truth values, and truth conditions
(2) object language and metalanguage
(3) recursion and compositionality

Two technical areas:

(1) Set theory and relations between sets
(2) functions and function application

I. Set Theory (see separate handout on sets)

II. Relations and Functions

Relation: Given two sets A and B, a relation between them is a pairing of elements of A with elements of B. This can be one-to-one, one-to-many, or many-to-one, or many-to-many. (Construct examples in class).

Formally, a relation is a subset of the Cartesian Product of A and B, written and defined as

(1) $A \times B := \{ <x, y> \mid x \in A \text{ and } y \in B\}$

So, a relation is a set of ordered pairs, with the first member of each ordered pair drawn from the first set, and the second member from the second set.

Function: a specific type of relation, with the constraint that each member of A is paired with at most one member of B.

Partial function: Some As have no B paired with them.
Total function: Each A has a B paired with it.

We'll almost always be talking about total functions.

We call the first set the domain of the function, and the second set the range (or co-domain) of the function. We can state this succinctly as the domain restriction of the function, written as:

(2) $f: A \rightarrow B$

The domain restriction tells us what set the arguments (inputs) of the function are drawn from, and what set the values (outputs) of the function are drawn from.

Since, for each element of the domain, there is a unique element of the range paired with it, we can talk about what we get when we feed the function a specific element.

(3) f(x) := the unique y such that < x, y > is in f.

We call y the <u>value of the function, as applied to x</u> .

III. How to specify a function

First Way: list it in a table

(5) A = {Adam, Bill, Charlie}
B = {Donna, Ellen, Francine}

(6) f_1: Adam → Donna
Bill → Donna
Charlie → Francine

Second way: list as a set of ordered pairs:

(7) f_1 = { < Adam, Donna > , < Bill, , Donna > , < Charlie ,Francine > }

This works for finite functions. However, suppose we have a function involving infinite sets.

In set theory, we can describe an infinite set with a general statement:

(8) { x | x is an integer} = {… -2, -1, 0, 1, 2, …}

Similarly, we can describe a function with a general statement:

(9) { < x, y > | x is an integer and y is an integer and y = x + 1}

This is equivalent to (where I = set of integers):

(10) f: I → I (coarse-grained **domain restriction**)
for all x ∈ I, f(x) = x + 1 (fine-grained **value condition**)

In general, you can think of functions as input-output 'machines'.

IV. Truth Values, Recursion, and Compositionality

- In phonology, syntax, morphology: we talk about structures, well-formedness of structures, and relations between them.

- In semantics, we add in a connection to the world around us: sentences express assertions/statements about the world (and our mental/psychological/emotional lives as well).

Truth value: One of {True, False}. S is True iff it accurately describes the world, False otherwise.

(8) **Tarski:** "snow is white" is True iff snow is white.
"beavers build dams" is True iff beavers build dams.

("T-sentences", mnemonic for both Tarski and Truth-conditional.)

Object language: what we are analyzing (stuff in quotes above).

Metalanguage: way we as scientists talk about the object language (stuff not in quotes, above).

Davidson: Each T-sentence by itself is banal. However, when you add in a crucial factor, it stops being banal. That factor is **recursion plus compositionality**.

Recursion: property a formal grammar has if it can embed a structure within a structure of same category. As a result, infinite outputs.

Compositionality: working hypothesis that the meaning of a structure X is a function of (a) the meaning of its subparts, and (b) the ways the subparts are put together.

So… if

- there are an infinite number of syntactic sentences, and
- each has a meaning derived in a systematic and compositional way from its structure, then

- we need a **recursive theory of truth**. Developing that is the goal of semantic theory.

V. Truth Conditions, Truth Values, and Semantic Values

truth value: labels a sentence True or False.

<u>truth conditions</u>: way the world would have to be in order for S to be true.
<u>semantic value/denotation</u> (for a constituent X): the item that it denotes (i.e. picks out, refers to, is connected with through interpretation).

The interpretation function: maps any expression of object-language to it semantic value/denotation.

Denotation of a name: the thing it names.

(9) "Ann" denotes Ann.
Ann is the semantic value/denotation of "Ann".

(10) "The Eiffel Tower" denotes the Eiffel Tower.
The Eiffel Tower is the semantic value/denotation of "The Eiffel Tower".

The interpretation function has a special notation: **[| |]**

We write the argument of the function (the piece of object language) inside the brackets:

[| "the Eiffel Tower" |] is read as "the interpretation of "The Eiffel Tower"

[| "the Eiffel Tower" |] = The Eiffel Tower is read as "the interpretation of "The Eiffel Tower" is the Eiffel Tower

(the argument of the function needs to be written out so as to orthographically distinguish it from the metalanguage description; H&K use boldface, I use quotation marks or boldface).

Denotation of a sentence: its truth value.
We identify Truth with 1, and False with 0.

So:

(11) [| "Andy Barss is teaching 564 this semester" |] = 1

(12) [| "It snowed in Tucson on May 5th, 2004" |] = 0

But … it's often the case that we don't know the truth value of a sentence. But the sentence still carries meaning. Consider:

(13) Chomsky's most recent paper is 65 pages long.

We don't know what the truth value is. But we do know something about the sentence and its connection to the world:

(14) [|”Chomsky’s most recent paper is 65 pages long.” |] = 1 iff Chomsky’s most recent paper is 65 pages long.

That is, we know *what the world would have to be like* for the sentence to be true. I.e. we can always compute the truth condition for any sentence.

The interplay of truth conditions and truth values plays a vitsal role in learning new information. Let’s consider this scenario:

> Andy has a copy of Chomsky’s new paper.
> You don’t, and know nothing about it.

I tell you:

(16) “Chomsky’s most recent paper is 65 pages long.”

Your semantic system computes:

(17) [|”Chomsky’s most recent paper is 65 pages long.” |] = 1 iff Chomsky’s most recent paper is 65 pages long.

IF you then make the pragmatic assumption that I am telling the truth, you may then conclude

(18) [|”Chomsky’s most recent paper is 65 pages long.” |] = 1

And by substitution of equivalents (SE), you may finally conclude:

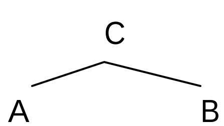
(19) Chomsky’s most recent paper is 65 pages long.

VI. Function Application and Frege’s conjecture. (Heim & Kratzer chapter 2, section 1)

Goal of a semantic theory:

(20) (a) Provide rules/statements/principles that give every tree and subtree (down to lexical items) a denotation.

(b) Provide rules/statements/principles that tell us how to calculate the denotation of C as a function of the denotations of A and B, in

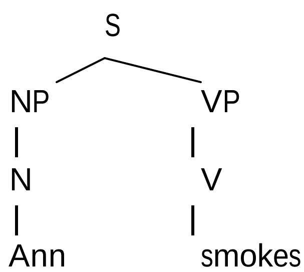


Frege's conjecture: (b) reduces to one general rule: function application. One of A, B is a function; the other its argument.

What is the denotation of the VP in a typical sentence? Here's a partial proof.

Consider the tree

(21)



(22) Nonbranching Node Rule: In any nonbranching structure

α
|
β

[| α |] = [| β |]

(23) by (22):

So, [| "NP" |] = [| "N" |] = [| "Ann" |]
with NP dominating N, and N dominating Ann

(24) Since we know that [| "Ann" |] = Ann, by SE we now have

(25) [| "NP" |] = Ann
with NP dominating N, and N dominating Ann

Look back at 21. We know that the denotation of S is a truth value. We know that the denotation of the NP subject is a thing/entity.

(26) Entities are not functions.Thus, the denotation of VP has to be a function, and the denotation of NP is its argument.

(27) Specifically, [| VP |] must be a function from entities to truth values. (This is forced by Frege's conjecture).

Since VP is nonbranching, the denotation of V, and of "smokes", is the same function.

So, we know this much (where D = set of entities):

(28) [| "smokes" |] = f: D → {0, 1}

But, specifically, "smokes" denotes a function from entities to truth values, such that the value of the function is 1 iff the argument smokes. More succinctly:

(29) [| "smokes" |] = f: D → {0, 1} (*domain restriction*)
f(x) = 1 iff x smokes (*value condition*)

What do we do now?

By compositionality: the interpretation of S is a matter of (a) the interpretation of NP and VP, and

(b) how we put them together.

Hypothesis: (b) is always function application.

How do we put (25) and (29) together?

(29) [| "smokes" |] = f: D → {0, 1} (DOMAIN DESCRIPTION)
f(x) = 1 iff x smokes (value restriction)

(25) [| "NP" |] = Ann
|
N
|
Ann

Well, we apply the function in (29) to the argument in (25), and what bwe get is the value of that function (the one mapping those who smoke to 1, and everyone else to 0), for that argument (Ann). Namely, we get:

(30) [| “ S “ |] = f: D → {0, 1}
for all x ∈ D, f(x) = 1 iff x smokes (Ann)

NP VP

N V

Ann smokes

Looking at the right hand side only, we can now see that

(31) f: D → {0, 1}
for all x ∈ D, f(x) = 1 iff x smokes (Ann) = 1 iff Ann smokes

An now we have precisely and compositionally derived the truth conditions for “Ann smokes” using:

a) lexical specification of the values for “Ann” and “smokes”
b) the nonbranching node rule (which just seems empirically true)
c) substitution of equivalents (a property of any logical deduction system, not just the theory of semantics presented here)
d) function application

As we proceed to more rich and complex structures, the main point of discussion will be (a), the lexical interpretations that we assign to items, together with a continuing discussion of the merits of considering function application as the sole means of meaning combination.