Quantification and scope

1 Class updates

- **Schedule updates:** The syllabus is updated online. Next week we'll discuss ellipsis.

- **Final papers** are due April 15. "Should be approximately 10 pages. The paper should identify an original puzzle, in a language you speak or in another language by working with a native speaker consultant. Use the skills developed in class to carefully diagnose and describe the issue, and sketch a possible solution."

  Advice for finding a topic: Look around your language for functional morphology or constructions whose meanings are not immediately obvious. Using the Principle of Compositionality, figure out what its semantic contribution is.

  A sample outline:

  1. Introduction: I am studying X and I will propose that it means X.
  2. Some basic data: Comparing minimal pairs of sentences with X and without X, we see that X must contribute Y meaning. X is grammatical in these sentences but not those others. A generalization for X's meaning and/or distribution is Z.
  3. Proposal: I propose X's denotation is $[\![X]\!]$. Here are trees and computations for a couple examples above, showing that my proposed denotation yields the desired meaning.
  4. Conclusion / open questions / problems with this analysis

  This is just one sample; your paper does not have to follow it closely.

  Advice for writing: Follow the advice in this short set of guidelines to writing Linguistics papers: `https://mitcho.com/teaching/newmeyer1988.pdf`.

  If you want to work on another language, through elicitation: I would suggest looking at expressions for universal quantifiers (*every student*) or words like 'only,' 'also,' 'again.'

  Talk to me or email me about your topic by March 15 and I can give you some comments and/or references.

- **Extra session?** I'd like to schedule an (optional) extra session to answer any technical questions and go over problems from the problem sets. Is this better sooner? Or later, before the final?

2 Subject quantifiers

The DPs we have studied so far have generally been of type e. Let's now consider subject DPs like *everyone*, *no one*,[1] and *someone*.

(1) Everyone sleeps.

Option 1: Include "plurals" in D_e, including a symbol that refers to 'nothing,' ϵ. *Everyone* is type e, the sum of all individuals.

(2) a. $D_e = \left\{ \begin{array}{c} \epsilon\text{, John, Mary, Kara,} \\ \text{John + Mary, John + Kara, Mary + Kara,} \\ \text{John + Mary + Kara} \end{array} \right\}$

b. $[\![\text{everyone}]\!]$ = John + Mary + Kara (type e)

c. $[\![\text{everyone sleeps}]\!]$ = 1 iff (John + Mary + Kara) sleeps

This sort of works for *everyone*, but it does not work for *no one* and *someone*. Why?

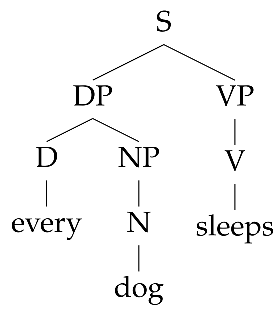
Option 2: *Everyone* is not type e.

(3) a. $[\![\text{everyone}]\!] = \lambda Q_{\langle e,t \rangle} \,.\, \forall x \in D_e\ [x \text{ is animate} \rightarrow Q(x)]$[2]

b. $[\![\text{everyone sleeps}]\!] = 1$ iff $\forall x \in D_e\ [x \text{ is animate} \rightarrow x \text{ sleeps}]$

3 Quantificational determiners

Let's now consider quantificational determiners:



(4) **Example quantifiers as set-relations, from Week 2:**

a. $every(A)(B) = 1$ iff $A \subseteq B$

b. $a/some(A)(B) = 1$ iff $A \cap B \neq \emptyset$

c. $more\text{-}than\text{-}two(A)(B) = 1$ iff $|A \cap B| > 2$

Because we normally work with truth conditions and functions, not sets, we have to translate (4a) into non-set terms:

(5) $[\![\text{every dog sleeps}]\!] = \{x : x \text{ is a dog}\} \subseteq \{y : y \text{ sleeps}\}$

$\Leftrightarrow \forall z \in \{x : x \text{ is a dog}\}\ [z \in \{y : y \text{ sleeps}\}]$

$\Leftrightarrow \forall z \in D_e\ [\ \underbrace{z \text{ is a dog}}_{\text{D's first argument}} \rightarrow \underbrace{z \text{ sleeps}}_{\text{D's second argument}}\ ]$

(6) $[\![\text{every}]\!] = \lambda P_{\langle e,t \rangle} \,.\, \lambda Q_{\langle e,t \rangle} \,.\, \forall x[P(x) \rightarrow Q(x)]$

[1]Although we spell this as two words, "no one," we will treat it as one word, just like *nothing*.
[2]Think of "→" as *if...then*.

4 Quantifier scope

(7) Everyone does not sleep (during class).

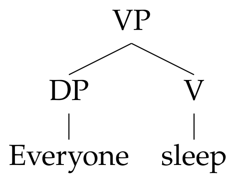
a. $$1 \text{ iff } \forall x \in D_e \underbrace{\left[x \text{ is animate} \rightarrow \text{it's not that } \underbrace{[x \text{ sleeps (during class)}]}_{\text{scope of } not} \right]}_{\text{scope of } \forall} \qquad (\forall > not)$$

b. $$1 \text{ iff it's not that } \underbrace{\left[\forall x \in D_e \ \underbrace{[x \text{ is animate} \rightarrow x \text{ sleeps (during class)}]}_{\text{scope of } \forall} \right]}_{\text{scope of } not} \qquad (not > \forall)$$

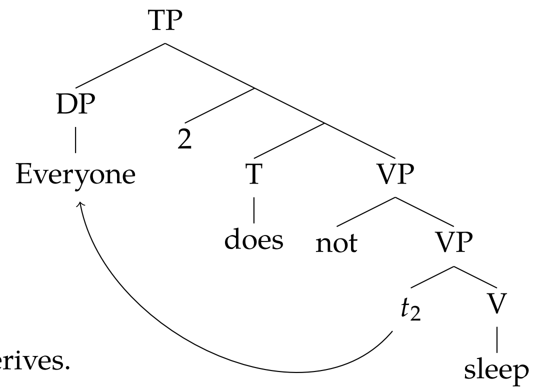
The two readings in (7) represent a *scope ambiguity*. There are two operators that "take scope"—∀ and negation—and one scope contains the other. We say ∀ in (7a) takes *wider* scope, and write ∀ > *not* to indicate this.

Recall from PS5 that there are advantages to adopting a VP-internal subject, interpreted through movement. We will adopt this here.

Step 1: Build subject in Spec,VP



Step 2: Add *not* + T, move subject DP to Spec,TP



Exercise: Let's see what meaning this tree derives.

Tools repeated from last time:

(8) **The interpretation of movement:**

Pick an arbitrary index i.

a. The base position of movement is replaced with a *trace* with index i: t_i.

b. A *binder index i* is adjoined right under the target position of the movement chain.

(9) **Traces and Pronouns Rule (T&P):**

If α is a pronoun or trace, g is a variable assignment, and $g(i)$ is defined, then $[\![\alpha_i]\!]^g = g(i)$.

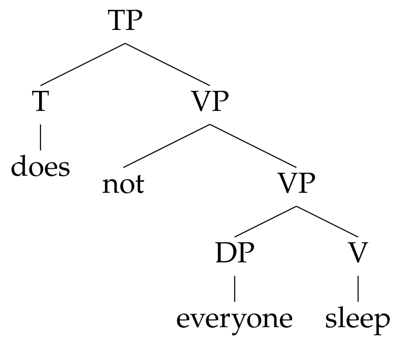
(10) **Predicate Abstraction (PA):** (H&K p. 186 version)

Let α be a branching node with daughters β and γ, where β dominates only a numerical index i. Then, for any assignment g, $[\![\alpha]\!]^g = \lambda x\ .\ [\![\gamma]\!]^{[i \mapsto x]]g}$.

We call the meaning that is reflected on the surface form—here, (7a)—a *surface scope* reading.

How do we get reading (7b)? One option: *pretend the movement didn't take place*.

At Logical Form (LF): Pretend the movement didn't happen

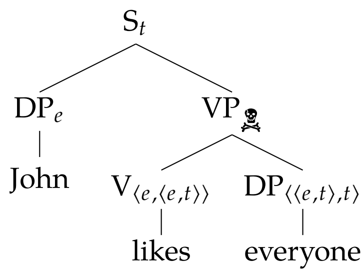


Exercise: Interpret this tree.

We call this the *inverse scope* interpretation. The process of "ignoring" movement at LF is called *syntactic reconstruction*.

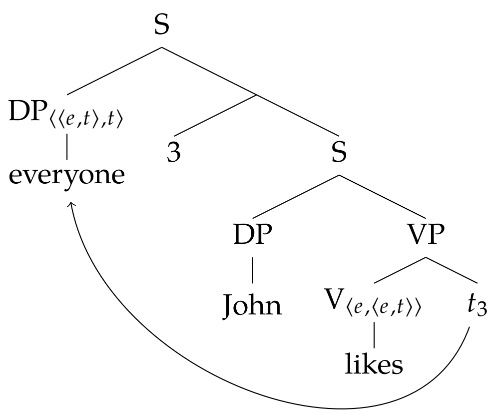
5 Quantifiers in object position

(11) John likes everyone.



Recall from PS5 that DPs of type $\langle\langle e,t\rangle,t\rangle$ can be interpreted easily if they are moved:

(12) Everyone, John likes ____.



Exercise: Make sure this works.

A solution to the problem of quantifiers in object position, like (11), is to *pretend this movement happened anyway*. The arrow is dashed because it's a *covert* movement, not reflected in pronunciation.

(13) LF for (11): everyone, John likes ____.

We call this movement *Quantifier Raising* (QR) (May, 1977). QR is required for quantifiers that are not in subject position, in order to avoid the type problem in (11).

6 Logical Form

In the past two sections we've proposed a lot of "pretending"... pretending movement happened or pretending movement didn't happen. We have opened up the possibility that *what we pronounce* is different than *what we interpret*.

(14) Structure is built in Syntax. Syntax has two outputs:

a. Phonological Form (PF): what is pronounced

b. Logical Form (PF): what is interpreted

Additional operators may take place at these "interfaces"—in particular, covert movement (like QR) and reconstruction may take place at LF.

A hypothesis developed by May (1977); Huang (1982) and others is that operations at LF are *syntactic* operations, (generally) subject to the same constraints as visible syntax. For example:

(15) **The Coordinate Structure Constraint (Ross, 1967):**

a. Which professor does John like ___?

b. * Which professor does John [[like ___] and [hate the dean]]?

(16) **Coordination and scope:** (examples from (Fox, 2003))

a. A (different) student likes every professor. $\exists > \forall, \forall > \exists$

b. A (#different) student [[likes every professor] and [hates the dean]]. $\exists > \forall, {*}\forall > \exists$

References

Fox, Danny. 2003. On logical form. In *Minimalist syntax*, 82–123. Blackwell.

Huang, Cheng-Teh James. 1982. Logical relations in Chinese and the theory of grammar. Doctoral Dissertation, Massachusetts Institute of Technology.

May, Robert Carlen. 1977. The grammar of quantification. Doctoral Dissertation, Massachusetts Institute of Technology.

Ross, John Robert. 1967. Constraints on variables in syntax. Doctoral Dissertation, Massachusetts Institute of Technology.