Semantics

Greg Kobele

June 25, 2018

Review

Summary

1. we can interpret sentences with movement
2. *structure* $\rightarrow$ *formula* $\rightarrow$ *truth conditions*
3. types govern how meanings are assembled

Semantic Interpretation Rules

$$\left[\!\!\left[\begin{array}{ccc} & \bullet & \\ / & & \backslash \\ \alpha & & \beta \end{array} \right]\!\!\right] = [\![\alpha]\!] \otimes [\![\beta]\!]$$

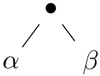
$$\left[\!\!\left[\begin{array}{c} \bullet \\ | \\ \alpha \end{array} \right]\!\!\right] = [\![\alpha]\!]$$

$$\left[\!\!\left[\begin{array}{ccc} & \bullet & \\ / & & \backslash \\ \alpha_i & & \beta \end{array} \right]\!\!\right] = [\![\alpha]\!] \, (\lambda x_i . \, [\![\beta]\!])$$

$$[\![t_i]\!] = x_i$$

Constraints on types

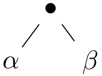
binary branching (no movement)



- one must have type (*ab*), and the other type *a*

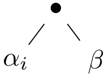
Constraints on types

binary branching (no movement)



- one must have type (*ab*), and the other type *a*

movement



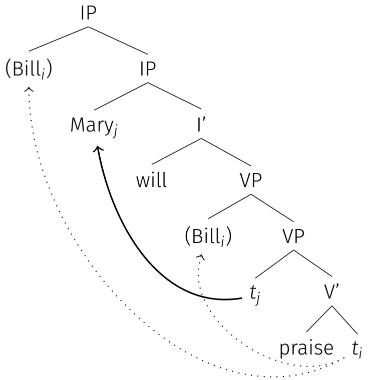
- α must have type (*ab*)*c*
- β must have type *b*

Transitivity

Where should objects move?

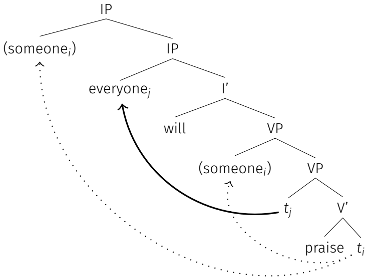
answer:
at least to a place where the interpretation of the sister is of type *t*

VP internal subjects



Ambiguity with QNPs

Change DPs



Different structures, different meanings

1. $\text{EVERY}(1)(\lambda e.\text{SOME}(1)(\lambda s.\text{PRAISE}(s)(e)))$
2. $\text{SOME}(1)(\lambda s.\text{EVERY}(1)(\lambda e.\text{PRAISE}(s)(e)))$

Different meanings, different predictions

$\textsc{every}(1)(\lambda e.\textsc{some}(1)(\lambda s.\textsc{praise}(s)(e)))$
for every thing, *e*, there is some thing, *s*, such that *e* praised *s*

$\textsc{some}(1)(\lambda s.\textsc{every}(1)(\lambda e.\textsc{praise}(s)(e)))$
there is some thing *s*, such that for every thing, *e*, *e* praised *s*

in English
sentences with multiple quantifiers are often ambiguous

S V O

Subject wide scope (SWS)

$$S(\lambda s.O(\lambda o.V\ o\ s))$$

Object wide scope (OWS)

$$O(\lambda o.S(\lambda s.V\ o\ s))$$

Scope

Scope

A scopes over B
iff B is (inside of) an argument to A

$A(\dots B \dots)$

- corresponds roughly to c-command

S V O

Subject wide scope (SWS)

$$S(\lambda s.O(\lambda o.V\ o\ s))$$

Object wide scope (OWS)

$$O(\lambda o.S(\lambda s.V\ o\ s))$$

Some more examples

not and *and*

- *It is not raining and snowing*

every and *not*

- *all that glisters is not gold*

Some non-examples

John and *every*

- *John praised every girl*

every and *every*

- *every boy praised every girl*

Why the exceptions?

Semantic justification

- individuals commute with GQs!
- *every* and *some* commute with themselves

A principle

If a sentence has two meanings, it should have two structures

If a sentence has two structures, it may have two meanings

A principled exception

Pragmatics
(reasoning about)* communicative intentions

1. Can you open the window?
2. I have two children.

Forcing scope

Ellipsis can force readings
The chickens are ready to eat, and the children are too.

- structure of `ellipsis` site must be identical to antecedent

But consider:
Every doctor praised a nurse, and John did too.

SWS	bad
OWS	ok!

Contextual forcing

context/expectations can force readings
A flag was hanging in front of every building

DPs in DPs

[one [apple [in every basket]]] was rotten

in	eet
basket,apple	et
one,every	(et)t
be	(et)et
rotten	et

compare with:
one apple which was in every basket was rotten

Modification

The meaning of adjectives

***Red* denotes a property (et)**
of being red

- This cherry is red

***Red* denotes a function ($(et)et$)**
from properties to properties

- This red cherry is tasty

The function of *red*

Red **denotes a function $((et)et)$**
which one? Let R be the set of red things

- $\text{RED}(p)(x) = 1$ iff $p(x) \wedge x \in R$

Predicate Modification

- since coordination $\wedge$ is used
- in the definition of adjectives ((*et*)*et*)
- to let them combine with NP meanings (*et*)
- simplify meaning, and introduce new semantic rule

$$\left[\!\!\left[\begin{array}{c} \bullet \\ / \quad \backslash \\ \alpha \qquad \beta \end{array} \right]\!\!\right] = [\![\alpha]\!] \wedge [\![\beta]\!]$$

- here the types of α and β must
  - be the same
  - be boolean

New operations

- Not every binary branching structure can be interpreted
- must have compatible types
  - with just function application (($\alpha\beta$) and α)
  - with PM too (boolean)
- Adding more operations
  - lets us interpret more structures

Not all adjectives

Absolute
$f(x) = x \wedge f(1)$

- male, female, odd, even

John is an f x $\vdash$ John is f and John is x

Restrictive
$f(x) \leq x$

- skillful, tall

John is an f x $\vdash$ John is an x

Non-restrictive
(no restriction)

- fake

John is an f x $\vdash$???