Word order domain

Jean Mark Gawron
Linguistics
San Diego State University
gawron@mail.sdsu.edu
http://www.rohan.sdsu.edu/∼gawron

Declarativeness

$$\left\langle \text{here}, \begin{bmatrix} \text{HEAD} & \textit{prep} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

$$\left\langle \text{there}, \begin{bmatrix} \text{HEAD} & \textit{prep} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

(a) Statements of constraints, relationships.

(b) Rather than specifications of processes.

(c) Mutilayered representations rather than derivations.

Concatenation

The semantics of 'concatenate'

(a) $\circ(\epsilon,\ \sigma,\ \sigma)$

(b) $\circ(x \circ \sigma_1,\ \sigma_2,\ x \circ \sigma_3) \quad \leftrightarrow \quad \circ(\sigma_1,\ \sigma_2,\ \sigma_3)$

To show: $\circ(ab, cd, abcd)$

(1)	$\circ(ab, cd, abcd)$	$\leftrightarrow$	$\circ(b, cd, bcd)$	(b)
(2)	$\circ(b, cd, bcd)$	$\leftrightarrow$	$\circ(\epsilon, cd, cd)$	(b)
(3)	$\circ(\epsilon, cd, cd)$			(a)

To find X s/t $\circ(ab, cd, X)$:

(1)	$\circ(ab, cd, X)$	$\leftrightarrow$	$\circ(b, cd, Y)$	$(b), X = aY$	$\therefore$ X = abcd
(2)	$\circ(b, cd, Y)$	$\leftrightarrow$	$\circ(\epsilon, cd, Z)$	$(b), Y = bZ$	
(3)	$\circ(\epsilon, cd, cd)$			$(a), Z = cd$	

Backward

Find X, Y s/t $\circ(X, Y, abcd)$:

$X = ab, Y = cd$

(1)	$\circ(X, Y, abcd)$	$\leftrightarrow$	$\circ(X1, Y, bcd)$	$(b), X = aX1$
(2)	$\circ(X1, Y, bcd)$	$\leftrightarrow$	$\circ(X2, Y, cd)$	$(b), X1 = bX2$
(3)	$\circ(\epsilon, cd, cd)$			$(a), X2 = \epsilon,\ Y = cd$

$X = abc, Y = d$

(1)	$\circ(X, Y, abcd)$	$\leftrightarrow$	$\circ(X1, Y, bcd)$	$(b), X = aX1$
(2)	$\circ(X1, Y, bcd)$	$\leftrightarrow$	$\circ(X2, Y, cd)$	$(b), X1 = bX2$
(3)	$\circ(X2, Y, cd)$	$\leftrightarrow$	$\circ(X3, Y, d)$	$(b), X2 = cX3$
(4)	$\circ(\epsilon, cd, cd)$			$(a), X3 = \epsilon,\ Y = d$

$X = abcd, Y = \epsilon$

(1)	$\circ(X, Y, abcd)$	$\leftrightarrow$	$\circ(X1, Y, bcd)$	$(b), X = aX1$
(2)	$\circ(X1, Y, bcd)$	$\leftrightarrow$	$\circ(X2, Y, cd)$	$(b), X1 = bX2$
(3)	$\circ(X2, Y, cd)$	$\leftrightarrow$	$\circ(X3, Y, d)$	$(b), X2 = cX3$
(4)	$\circ(X2, Y, cd)$	$\leftrightarrow$	$\circ(X4, Y, \epsilon)$	$(b), X2 = dX4$

Further Back

Find X, Y s/t $\circ(X, Y, abcd)$:

$X = a, Y = bcd$

(1)	$\circ(X, Y, abcd)$	$\leftrightarrow$	$\circ(X1, Y, bcd)$	$(b), X = aX1$
(3)	$\circ(\epsilon, bcd, bcd)$			$(a), X1 = \epsilon,\ Y = bcd$

$X = \epsilon, Y = abcd$

(1)	$\circ(\epsilon, abcd, abcd)$	$(a), X = \epsilon,\ Y = abcd$

X	Y
$abcd$	ϵ
abc	d
ab	cd
a	bcd
ϵ	$abcd$

Shuffling

The semantics of 'shuffle':

(a)	$\bigcirc(\epsilon,\ \sigma,\ \epsilon)$		
(b)	$\bigcirc(x \circ \sigma_1,\ \sigma_2,\ x \circ \sigma_3)$	$\leftrightarrow$	$\bigcirc(\sigma_1,\ \sigma_2,\ \sigma_3)$
(c)	$\bigcirc(\sigma_1,\ x \circ \sigma_2,\ x \circ \sigma_3)$	$\leftrightarrow$	$\bigcirc(\sigma_1,\ \sigma_2,\ \sigma_3)$

To show: $\bigcirc(ab, cd, cdab)$

(1)	$\circ(ab, cd, cdab)$	$\leftrightarrow$	$\circ(ab, d, dab)$	(c)
(2)	$\circ(ab, d, dab)$	$\leftrightarrow$	$\circ(ab, \epsilon, ab)$	(c)
(3)	$\circ(ab, \epsilon, ab)$	$\leftrightarrow$	$\circ(b, \epsilon, b)$	(b)
(4)	$\circ(b, \epsilon, b)$	$\leftrightarrow$	$\circ(\epsilon, \epsilon, \epsilon)$	(b)
(5)	$\circ(\epsilon, \epsilon, \epsilon)$			(a)

Reape's approach

Two levels of structure

- Syntactic-semantic functor-argument structure (FAS)
- Word-order domains (WOD)
- The grammar defines the WOD of a constituent in terms of its daughters or the WODs of its daughters

Pollard's problem

- Partial VP frontings allowed by allowing partial VP constituents via "Xtreme raising".
- Many trees now provided for analyzing the same string.
- Each tree provides a partial VP necessary because it CAN be fronted, and therefore that VP must be a constituent *in situ* to "feed" the long distance dependency machinery. (36), p.302

(1) a. [Das Has bauen] wird Hans.

b. Wird Hans das Haus bauen?

c. Wird das Haus Hans bauen?

d.

Hat	er	seiner Tochter	ein Märchen	erzaählen	können
Has	he	his daughter	a fairy tale	tell	be able

Has he been able to tell his daughter a fairy tale? (6 structures)

The conclusion

Pollard (1996)

> It seems to follow from these considerations that my analysis of German is not really phrase structure grammar at all. Somehow, between generalizing Schema B to license partial VPs and relax auxiliary subcategorizations to allow raising of non-subject complements, I have passed beyond the frontier of phrase-structure grammar... [!]

> ... The way I think about the problem is this: The kind of structural analysis I have been doing here... is too fine-grained, is the sense that it produces distinct structures like those in (36) that do not reflect genuine lingusitic differences. What is needed, then, is a coarser-grained level of representation than phrase structure, some canonical method of dividing up phrase-structural analyses into equivalence classes such (36b) and (36c) end up in the same equivalence classes.

Reape's soln

- One functor argument structure corresponds to many distinct linearizations of constituents.
- FAS: coarse-grained
- Constituent linearizations: fine-grained
- But does Reape's grammar allow fronting of partial VPs? [No, but ...]