An Introduction to Minimalist Grammars

Meaghan Fowlie
mfowlie@coli.uni-saarland.de
meaghanfowlie.com

Computer Linguistik, Universität des Saarlandes

Minimalism Seminar
May 30, 2017

Minimalist Grammars

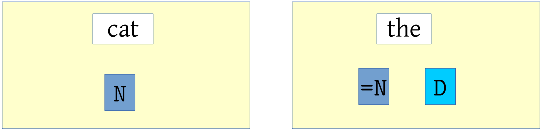
- Ed Stabler's (1997 etc.) formalisation of Chomsky (1995) Minimalism
- Two operations: Merge and Move (or External and Internal Merge)
- Merge & Move both say "put 2 things together when their features match"
- Merge: get second element from Lexicon/Numeration
- Move: get second element from Mover List

Features

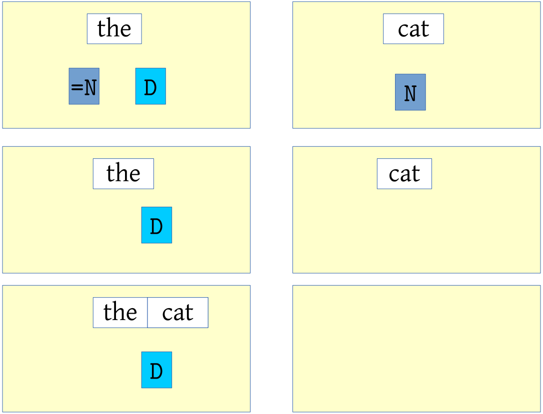
	Positive				Negative			
Merge	=N	=D	=V	=C	N	D	V	C
Move	+wh	+foc	+top		-wh	-foc	-top	

Features

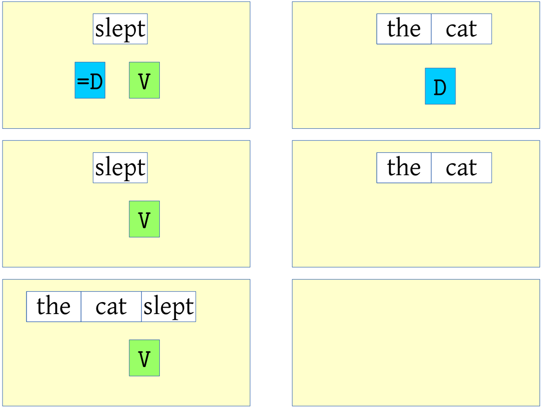
	Positive				Negative			
Merge	=N	=D	=V	=C	N	D	V	C
Move	+wh	+foc	+top		-wh	-foc	-top	



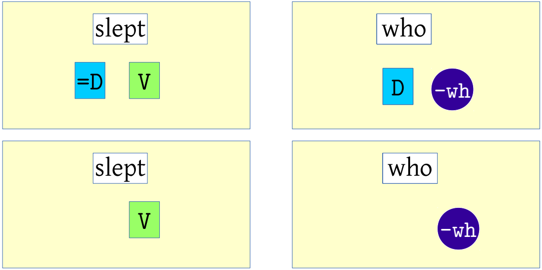
Merge



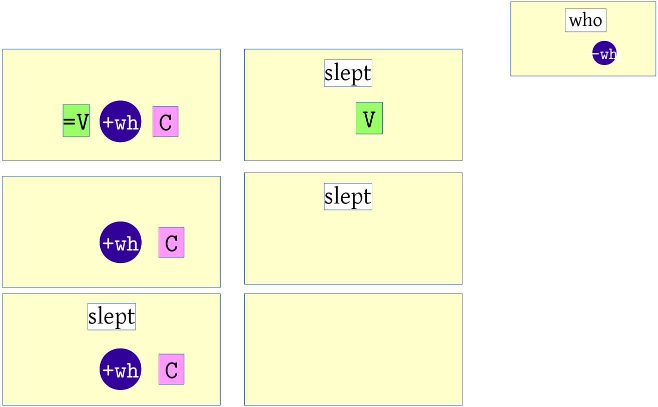
Merge



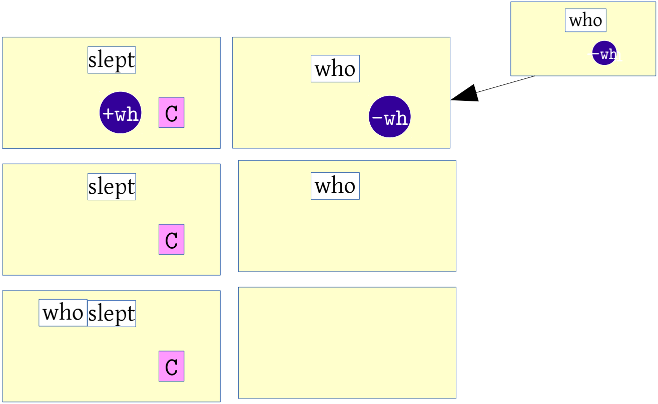
Merge a Mover



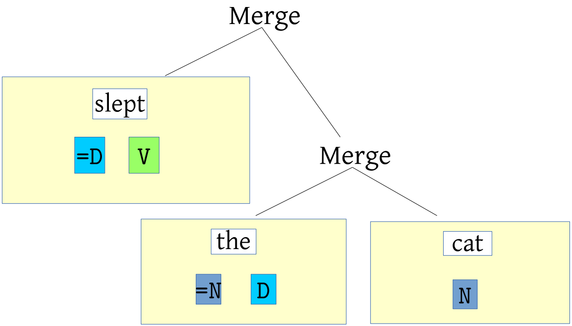
Merge (with waiting Mover)



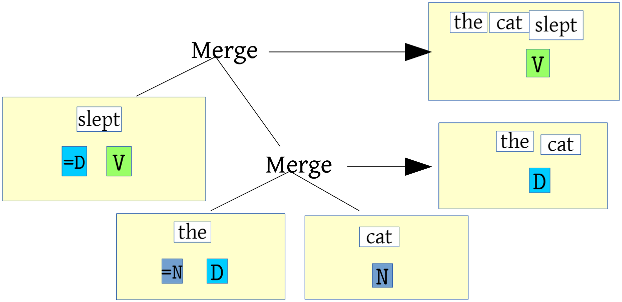
Move



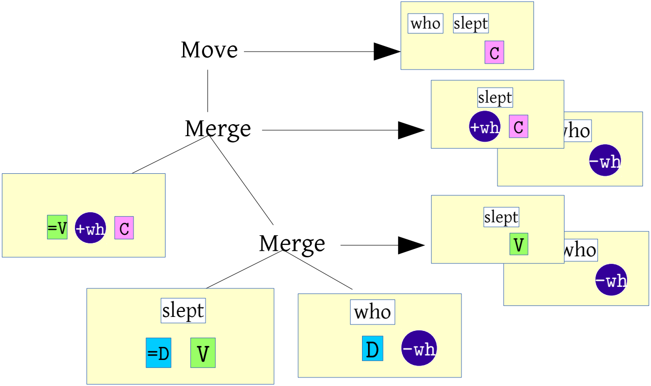
Derivation Trees



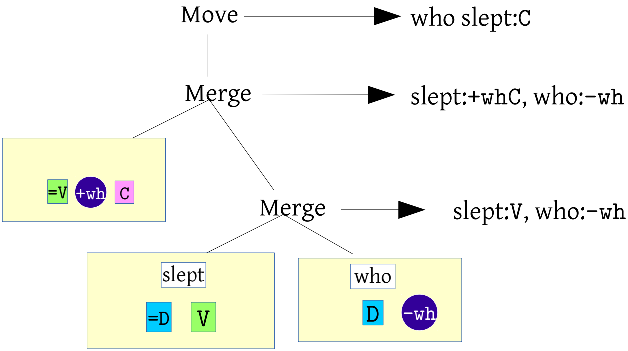
Derivation Trees



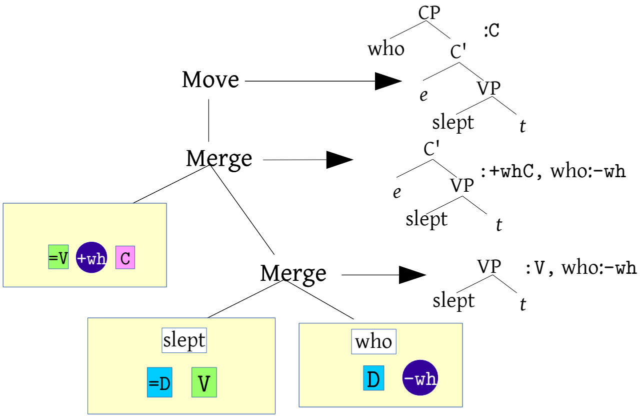
Derivation Trees



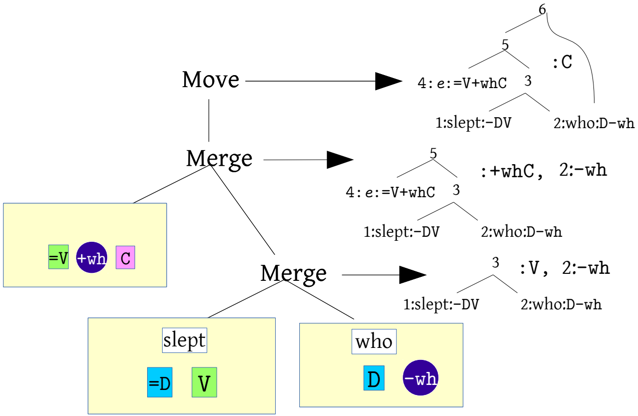
Derivation Trees



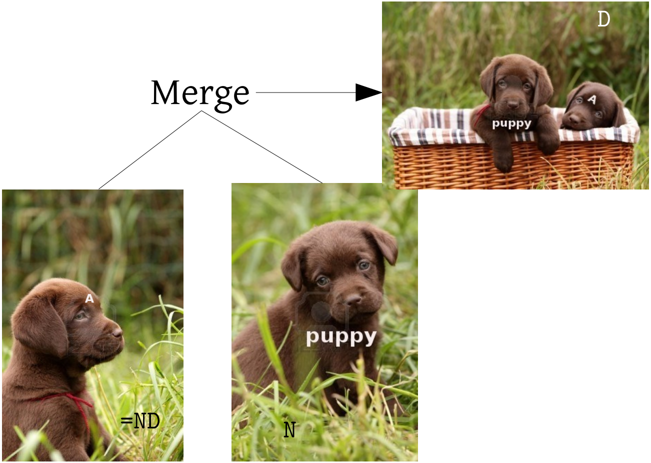
Derivation Trees



Derivation Trees



Derivation Trees



MG algebra

MG is itself interpreted in an algebra $\langle \Sigma, \frown^{(2)} \rangle$, usually string concatenation or tree formation.

Let **sel** and **lic** be sets, $|\mathbf{lic}| = k$.

Let F=$\{$`+f`,`-f`,`=X`,`X`|`f`$\in$ **lic**, `X` $\in$ **sel** $\}$ be the *features*.

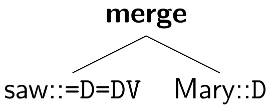
$MG = \langle (T_\Sigma \times F^*)^*, \mathbf{merge}^{(2)}, \mathbf{move}^{(1)} \rangle$ is a partial algebra where:

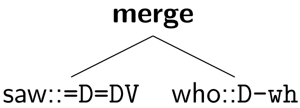
merge **merge** : $(T_\Sigma \times F^*)^* \times (T_\Sigma \times F^*)^* \to (T_\Sigma \times F^*)^{k+1}$

Let $\alpha, \beta, \gamma \in F^*$, let $\mathtt{X} \in$ **sel**, and let $\mathtt{f} \in$ **lic**.

$$[\![\mathbf{merge}((s, \mathtt{=X} :: \alpha) :: m_s, (t, \mathtt{X} :: \beta) :: m_t)]\!] =$$

$$\begin{cases} (s^\frown t, \alpha) :: (m_s \cdot m_t) & \text{if } \beta = \epsilon \\ (s, \alpha) :: ((t, \beta) :: (m_s \cdot m_t)) & \text{if } \beta = \mathtt{-f} :: \gamma \end{cases}$$





((*saw Mary*,=DV)) ((*saw*,=DV),(*who*,-wh))

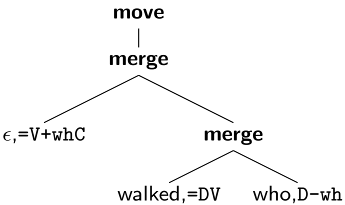
move $\textbf{move} : (T_\Sigma \times F^*)^* \to (T_\Sigma \times F^*)^*$

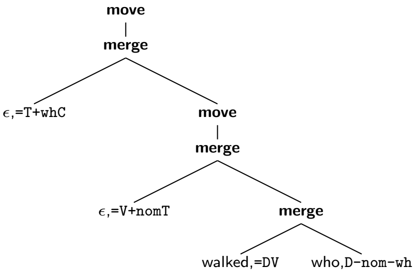
Let $\alpha, \beta, \gamma \in F^*$, and let $\mathtt{f}, \mathtt{g} \in \textbf{lic}$

Let $m_f = (t, -\mathtt{f} :: \beta)$ be the unique element of m headed by -f.

$$[\![\textbf{move}((s, +\mathtt{f} :: \alpha) :: m)]\!] =$$

$$\begin{cases} (t^\frown s, \alpha) :: m - m_f) & \text{if } \beta = \epsilon \\ (s, \alpha) :: ((t, \beta) :: m - m_f)) & \text{if } \beta = \mathtt{-g} :: \gamma \end{cases}$$





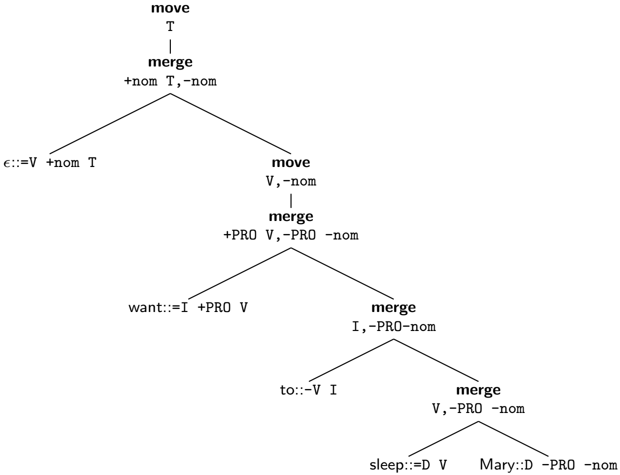
The language

The language generated by an MG: for some set of final categories $F \subseteq \textbf{sel}$, if a derivation generates a one-tuple $((\texttt{t},\texttt{F}))$ for some $\texttt{F} \in F$, the interpretation of $\texttt{t}$ in the interpretation algebra is in the language generated by the MG.

eg: $((\frown(\textit{Mary},\frown(\textit{ saw, John})),\texttt{T}))$
$\texttt{T}$ is a final category, so $[\![\frown(\textit{Mary},\frown(\textit{ saw, John}))]\!]= \textit{Mary saw John}$ $\in L(MG)$

Control as movement

Mary wants to sleep



References

Chomsky, Noam. 1995. *The minimalist program*. Cambridge, MA: MIT Press.

Stabler, Edward. 1997. Derivational minimalism. *Logical Aspects of Computational Linguistics* 68–95.