Computational Tools for Logic-Based Grammar Formalisms

Minimalist Grammar

Willemijn Vermaat – UiL-OTS, Utrecht

4th August 2002

Contents

1 The problem 3

2 Example 6

3 Minimalist Grammar 8

3.1 Declarative sentence 10

3.2 Wh-phrase 12

1. The problem

Syntax

Generative syntax seeks answers to linguistic phenomena.

Provide an abstract theory that:

- captures the data descriptively
- can be applied cross-linguistically
- generalizes to similar phenomena in one language
- can be build in one bigger framework

Formal frameworks

Formal frameworks, such as Type Logical Grammar and Minimalist Grammar, might provide the basis for such an abstract theory.

Similarities

The basic machinery of the three frameworks are similar:

- basic operations: Merge and Move,
- lexicon
- important role played by features (properties of words)

Differences

The implementation and level of formality differs.

Plan of action

- analyze empirical data
- implement data in both formal frameworks
- compare the three analysis:
  - enhance the two formal reasoning systems
  - provide answers for generative syntax

Parsers

Parsers such as Grail and MGCKY help:

- to proofcheck the analysis that you made
- to look into the computational complexity

2. Example

Wh-movement in English

(1) Willem loves Maxima

(2) Does Willem love Maxima?

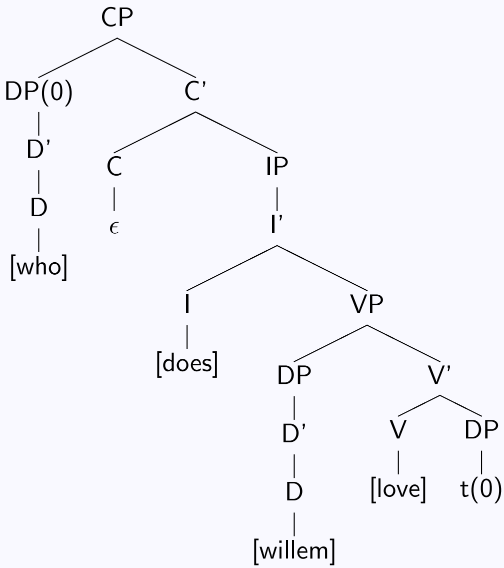
(3) Who does Willem love t?

Data shows:

- The object of the sentence, '*Maxima*' is base-generated as a complement of the verb '*love*'
- English needs *do-support* for negative sentences, yes/no-questions and wh-phrases
- The wh-object, in wh-phrases, is base-generated in object position and then moved to the front of the sentence to precede the verb phrase.

Puzzle: How is the wh-phrase moved to the front of the sentence?

Syntactic analysis



3. Minimalist Grammar

A minimalist grammar $MG = (\sum, F, Types, Lex, \mathcal{F})$

Features F:

$$\begin{array}{rcl}
\textbf{base } B & = & \{v, n, np, case, wh, \ldots\} \\
\textbf{selectors } S & = & \{\texttt{=f} | \texttt{f} \in B\} \\
\textbf{licensees } M & = & \{\texttt{-f} | \texttt{f} \in B\} \\
\textbf{licensors } N & = & \{\texttt{+f} | \texttt{f} \in B\} \\
\textbf{features } F & = & B \cup S \cup M \cup N
\end{array}$$

Grammar

$$\begin{array}{lrcl}
\textbf{Lexical types} & LT & = & \Sigma^* :: F^* \\
\textbf{Derived types} & DT & = & \Sigma^* : F^* \\
 & \cdot & \in & \{::, :\} \\
\textbf{Lexicon} & Lex & \subset & LT^+ \\
\textbf{Minimalist grammar} & G & = & Lex
\end{array}$$

Operations

Merge $:(E \times E) \rightarrow E$
where $\mathtt{t} = (\mathtt{t}_s \mathtt{t}_h \mathtt{t}_c)$

[r1] *if* s *is lexical, and* t *has one* [f]

$$\frac{\mathtt{s} :: \mathtt{=f}\gamma \quad \mathtt{t}_s,\ \mathtt{t}_h,\ \mathtt{t}_c \cdot \mathtt{f}}{\epsilon,\ \mathtt{s},\ \mathtt{t} : \gamma}\ r1$$

[r2] *if* s *is derived, and* t *has one* [f]

$$\frac{\mathtt{s}_s,\ \mathtt{s}_h,\ \mathtt{s}_c : \mathtt{=f}\gamma \quad \mathtt{t}_s,\ \mathtt{t}_h,\ \mathtt{t}_c \cdot \mathtt{f}}{\mathtt{ts}_s,\ \mathtt{s}_h,\ \mathtt{s}_c : \gamma}\ r2$$

[r3] *if* s *is lexical or derived, and* t *has one* [f] *and a set of (licensee) features* δ

$$\frac{\mathtt{s}_s,\ \mathtt{s}_h,\ \mathtt{s}_c \cdot \mathtt{=f}\gamma \quad \mathtt{t}_s,\ \mathtt{t}_h,\ \mathtt{t}_c \cdot \mathtt{f}\delta}{\mathtt{s}_s,\ \mathtt{s}_h,\ \mathtt{s}_c : \gamma,\ \mathtt{t} : \delta}\ r3$$

3.1. Declarative sentence

Lexicon:

Lexical:	*Functional:*
willem :: `d` maxima :: `d` loves :: `=d =d vp`	ϵ :: `=vp c`

willem loves maxima : c

Move $: E \to E$

[m1] *if* s *is derived, and* t *in the chain is the only element (SMC) with one* $[-\mathtt{f}]$

$$\frac{\mathsf{s}_s,\ \mathsf{s}_h,\ \mathsf{s}_c\ :\ \mathtt{+f}\gamma,\ \Gamma[\mathsf{t}_s,\ \mathsf{t}_h,\ \mathsf{t}_c\ :\ \mathtt{-f}]}{\mathsf{ts}_s,\ \mathsf{s}_h,\ \mathsf{s}_c\ :\ \gamma,\ \Gamma}\ m1$$

[m2] *if* s *is derived, and* t *in the chain is the only element (SMC) with a* $[-\mathtt{f}]$ *followed by a non-empty set of features* δ

$$\frac{\mathsf{s}_s,\ \mathsf{s}_h,\ \mathsf{s}_c\ :\ \mathtt{+f}\gamma,\ \Gamma[\mathsf{t}_s,\mathsf{t}_h,\mathsf{t}_c\ :\ \mathtt{-f}\delta]}{\mathsf{s}_s,\ \mathsf{s}_h,\ \mathsf{s}_c\ :\ \gamma,\ \Gamma[\ \mathsf{t}_s,\mathsf{t}_h,\mathsf{t}_c\ :\ \delta]}\ m2$$

3.2. Wh-phrase

Lexicon:

Lexical:	*Functional:*
willem :: d maxima :: d loves :: =d =d vp love :: =d =d V	 ϵ :: =vp c ϵ :: =i c does :: =v i

Question:

who :: ?	ϵ :: ?

Grammar: wh.pl

who does willem love $: c$