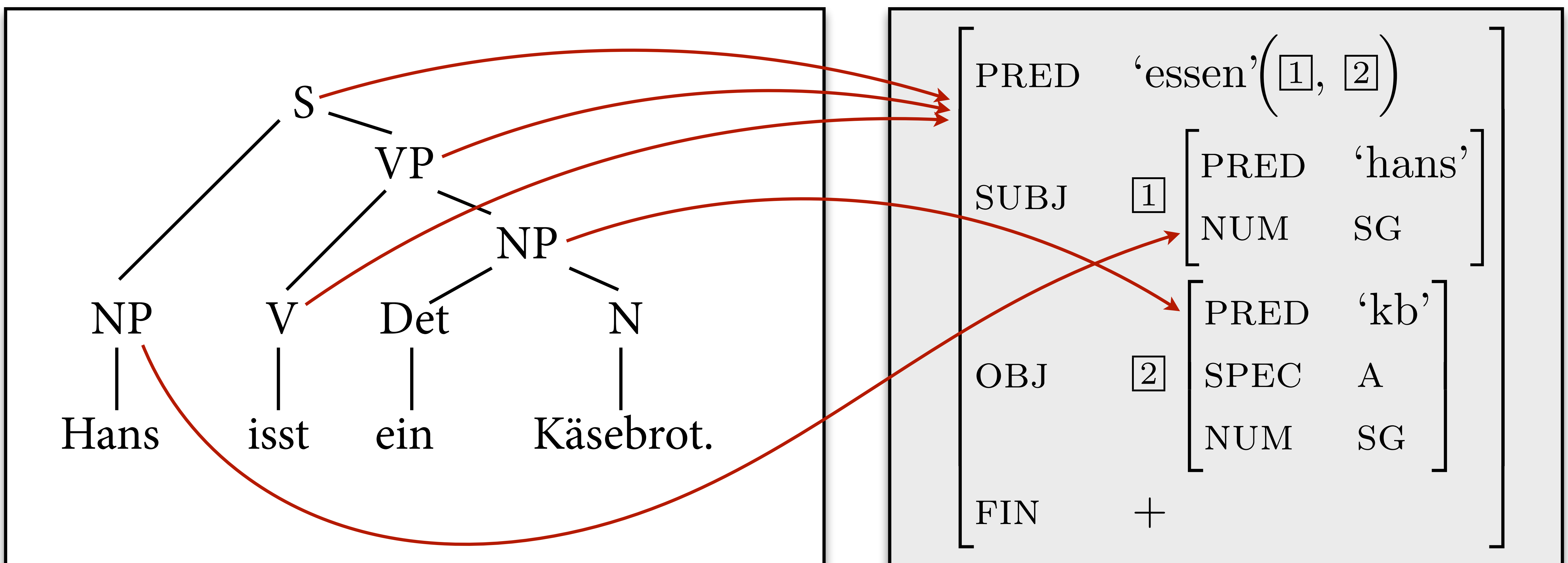
Lexikalisch-Funktionale Grammatik (LFG), Teil 2

Vorlesung “Grammatikformalismen”

Alexander Koller

23. Juni 2017

Beispiel



c-Struktur

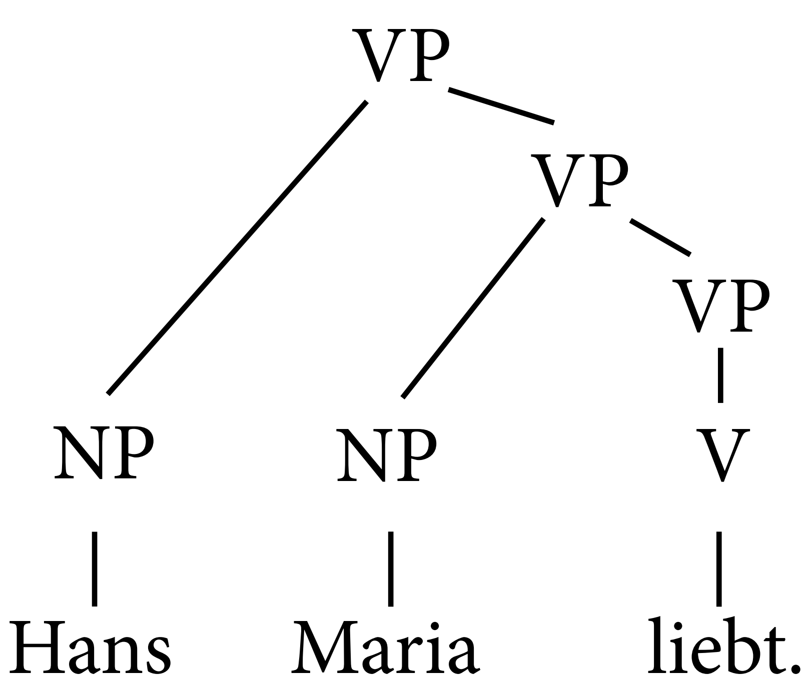
f-Struktur

Abbildung ϕ

Verbletztstellung

$$
\begin{array}{llcc}
\mathrm{VP} & \rightarrow & \underset{\uparrow=\downarrow}{\mathrm{V}} & \\[2ex]
\mathrm{VP} & \rightarrow & \underset{(\uparrow\textsc{subj})\,=\,\downarrow}{\mathrm{NP}} & \underset{\uparrow=\downarrow}{\mathrm{VP}} \\[2ex]
\mathrm{VP} & \rightarrow & \underset{(\uparrow\textsc{obj})\,=\,\downarrow}{\mathrm{NP}} & \underset{\uparrow=\downarrow}{\mathrm{VP}}
\end{array}
$$

$$
\begin{array}{lll}
\text{Hans} & \mathrm{NP} & (\uparrow \textsc{pred}) = \text{‘Hans’} \\
& & (\uparrow\textsc{agr case}) = \textsc{nom} \\[1ex]
\text{Maria} & \mathrm{NP} & (\uparrow \textsc{pred}) = \text{‘Maria’} \\
& & (\uparrow\textsc{agr case}) = \textsc{acc} \\[1ex]
\text{liebt} & \mathrm{V} & (\uparrow \textsc{pred}) = \text{‘lieben’}\langle(\uparrow\textsc{subj}), (\uparrow\textsc{obj})\rangle \\
& & (\uparrow\textsc{subj agr case}) = \textsc{nom} \\
& & (\uparrow\textsc{obj agr case}) = \textsc{acc}
\end{array}
$$



$$
\begin{bmatrix}
\textsc{pred} & \text{‘lieben’}\Big\langle\Big(\uparrow\textsc{subj}\Big), \Big(\uparrow\textsc{obj}\Big)\Big\rangle \\
\textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘hans’} \\ \textsc{agr} & \begin{bmatrix} \textsc{case} & \textsc{nom} \end{bmatrix} \end{bmatrix} \\
\textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{‘maria’} \\ \textsc{agr} & \begin{bmatrix} \textsc{case} & \textsc{acc} \end{bmatrix} \end{bmatrix}
\end{bmatrix}
$$

Heute

- Fernabhängigigkeiten / Functional Uncertainty.
- Expressivität von LFG.
- Parsing von LFG.
- Wahrscheinlichkeitsmodelle für LFG.

Fernabhängigkeiten in LFG

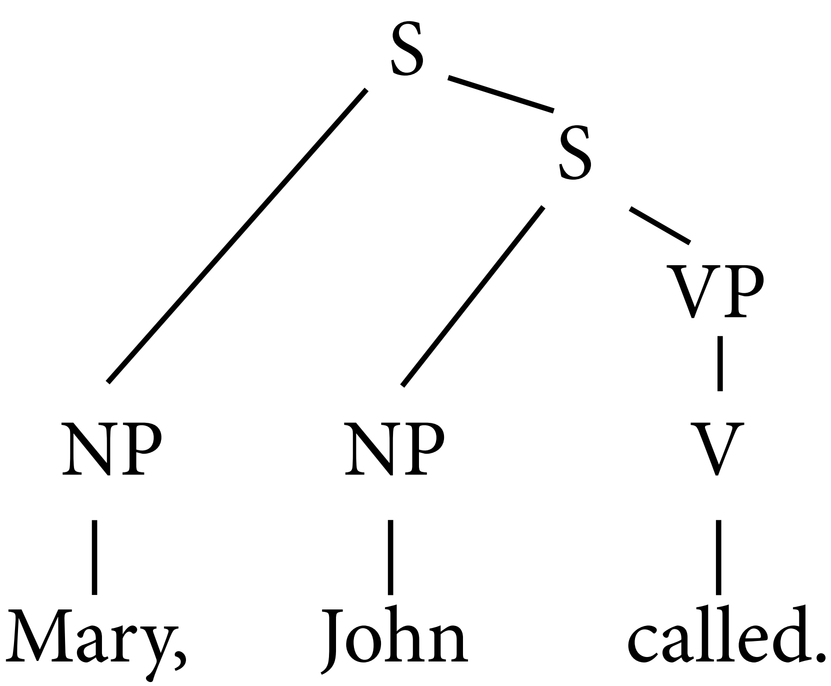
- Grammatikalität:
  - c-Struktur: korrekter Parsebaum der kfG.
  - f-Struktur: eindeutig, kohärent, vollständig und erfüllt alle f-Struktur-Constraints aus der Grammatik
- Präd-Arg-Struktur muss in der *f-Struktur* richtig herauskommen; c-Struktur eher Mittel zum Zweck.
- Fernabhängigkeiten: Unifiziere zwei f-Knoten, die in der c-Struktur weit auseinanderstehen.

Fernabhängigkeiten

$$
\begin{array}{lccc}
\mathrm{S} & \rightarrow & \underset{\substack{(\uparrow\textsc{topic}) = \downarrow \\ (\uparrow\textsc{topic}) = (\uparrow\textsc{obj})}}{\mathrm{NP}} & \underset{\uparrow = \downarrow}{\mathrm{S}}
\end{array}
\qquad
\text{called} \quad \mathrm{V} \quad (\uparrow \textsc{pred}) = \text{‘call’}\langle(\uparrow\textsc{subj}), (\uparrow\textsc{obj})\rangle
$$

$$
\begin{array}{lccc}
\mathrm{S} & \rightarrow & \underset{(\uparrow\textsc{subj}) = \downarrow}{\mathrm{NP}} & \underset{\uparrow=\downarrow}{\mathrm{VP}}
\end{array}
$$

$$
\begin{array}{lccc}
\mathrm{VP} & \rightarrow & \underset{\uparrow=\downarrow}{\mathrm{V}} & \left(\underset{(\uparrow\textsc{obj}) = \downarrow)}{\mathrm{NP}} \right)
\end{array}
$$



$$
\begin{bmatrix}
\textsc{pred} & \text{‘call’}\left\langle \left(\uparrow\textsc{subj}\right), \left(\uparrow\textsc{obj}\right) \right\rangle \\
\textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘john’} \end{bmatrix} \\
\textsc{obj} & \boxed{1} \\
\textsc{topic} & \boxed{1} \begin{bmatrix} \textsc{pred} & \text{‘mary’} \end{bmatrix}
\end{bmatrix}
$$

Fernabhängigigkeiten

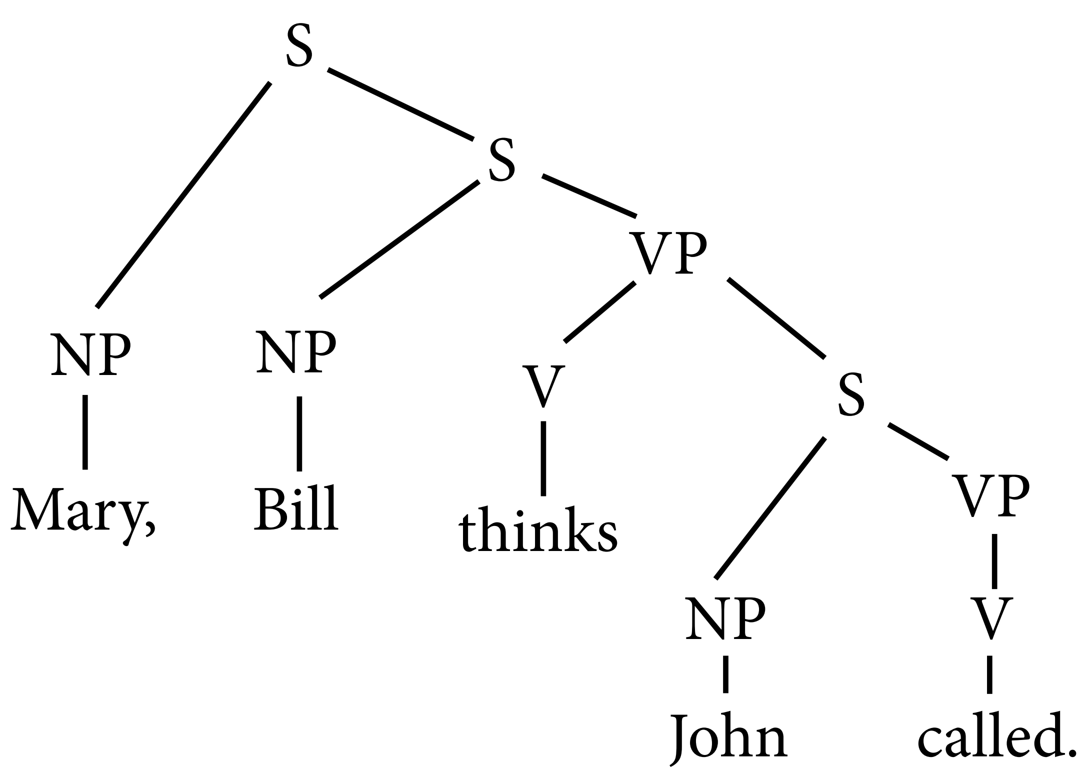
$$S \rightarrow \underset{\substack{(\uparrow\text{TOPIC}) = \downarrow \\ (\uparrow\text{TOPIC}) = (\uparrow\text{COMP OBJ})}}{NP} \quad \underset{\uparrow = \downarrow}{S}$$

$$S \rightarrow \underset{(\uparrow\text{SUBJ}) = \downarrow}{NP} \quad \underset{\uparrow=\downarrow}{VP}$$

$$VP \rightarrow \underset{\uparrow=\downarrow}{V} \left(\underset{(\uparrow\text{OBJ}) = \downarrow)}{NP} \right) \left(\underset{(\uparrow\text{COMP}) = \downarrow}{S} \right)$$

called V $(\uparrow \text{PRED}) = \text{‘call’}\langle(\uparrow\text{SUBJ}), (\uparrow\text{OBJ})\rangle$

thinks V $(\uparrow \text{PRED}) = \text{‘think’}\langle(\uparrow\text{SUBJ}), (\uparrow\text{COMP})\rangle$



$$\begin{bmatrix} \text{PRED} & \text{‘think’}\Big\langle\Big(\uparrow\text{SUBJ}\Big), \Big(\uparrow\text{COMP}\Big)\Big\rangle \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘bill’} \end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{‘call’}\Big\langle\Big(\uparrow\text{SUBJ}\Big), \Big(\uparrow\text{OBJ}\Big)\Big\rangle \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘john’} \end{bmatrix} \\ \text{OBJ} & \boxed{1} \end{bmatrix} \\ \text{TOPIC} & \boxed{1} \begin{bmatrix} \text{PRED} & \text{‘mary’} \end{bmatrix} \end{bmatrix}$$

Fernabhängigkeiten

- Problem: für jede Einbettungstiefe braucht man separate Regel.

  $(\uparrow\textsc{topic}) = (\uparrow\textsc{obj})$

  $(\uparrow\textsc{topic}) = (\uparrow\textsc{comp obj})$

  $(\uparrow\textsc{topic}) = (\uparrow\textsc{comp comp obj})\ \ldots$

- Lösung: *Functional uncertainty*
  = regulärer Ausdruck über Pfade
  = Disjunktion über alle möglichen Pfadlängen.

  $$(\uparrow\textsc{topic}) = (\uparrow\textsc{comp}^{*}\ \textsc{obj})$$

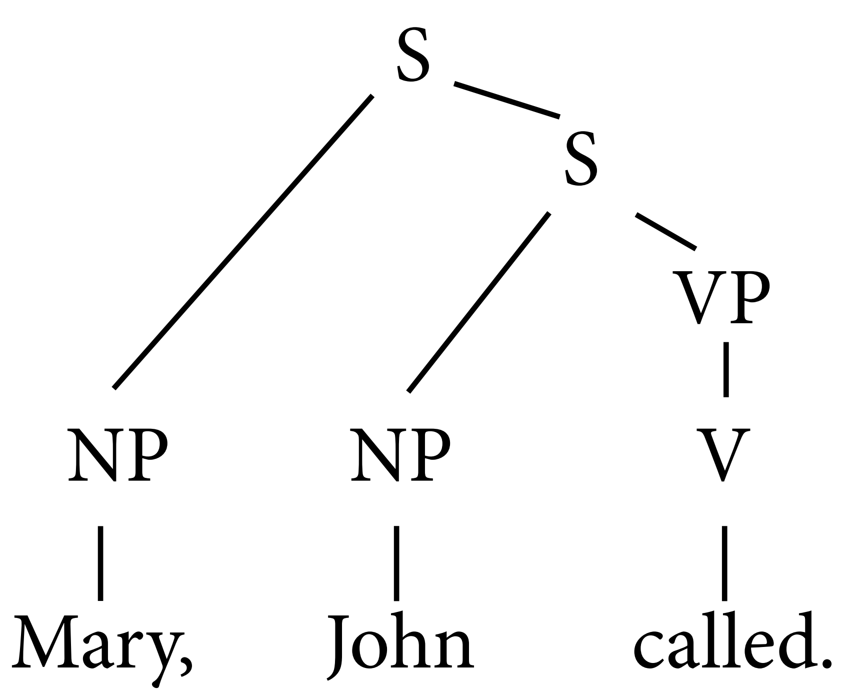
Fernabhängigkeiten

$$
\begin{array}{lll}
S & \rightarrow & \begin{array}{c} NP \\ (\uparrow \textsc{topic}) = \downarrow \\ \boxed{(\uparrow \textsc{topic}) = (\uparrow \textsc{comp}^{*}\ \textsc{obj})} \end{array} \quad \begin{array}{c} S \\ \uparrow = \downarrow \end{array}
\end{array}
$$

$$
\begin{array}{lll}
S & \rightarrow & \begin{array}{c} NP \\ (\uparrow \textsc{subj}) = \downarrow \end{array} \quad \begin{array}{c} VP \\ \uparrow = \downarrow \end{array}
\end{array}
$$

$$
\begin{array}{lll}
VP & \rightarrow & \begin{array}{c} V \\ \uparrow = \downarrow \end{array} \left(\begin{array}{c} NP \\ (\uparrow \textsc{obj}) = \downarrow) \end{array} \right) \left(\begin{array}{c} S \\ (\uparrow \textsc{comp}) = \downarrow \end{array} \right)
\end{array}
$$

$$
\begin{array}{lll}
\text{called} & V & (\uparrow \textsc{pred}) = \text{‘call’}\langle(\uparrow \textsc{subj}), (\uparrow \textsc{obj})\rangle \\
\text{thinks} & V & (\uparrow \textsc{pred}) = \text{‘think’}\langle(\uparrow \textsc{subj}), (\uparrow \textsc{comp})\rangle
\end{array}
$$



$$
\begin{bmatrix}
\textsc{pred} & \text{‘call’}\Big\langle\Big(\uparrow \textsc{subj}\Big), \Big(\uparrow \textsc{obj}\Big)\Big\rangle \\
\textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘john’} \end{bmatrix} \\
\textsc{obj} & \boxed{1} \\
\textsc{topic} & \boxed{1} \begin{bmatrix} \textsc{pred} & \text{‘mary’} \end{bmatrix}
\end{bmatrix}
$$

Fernabhängigkeiten

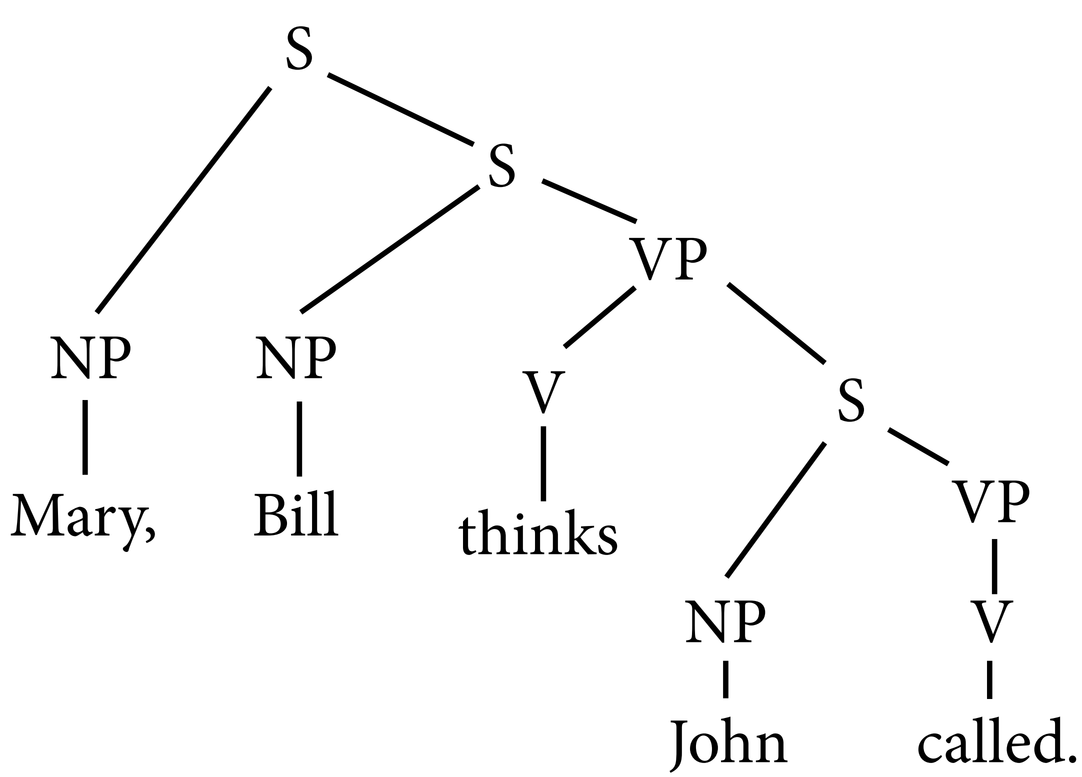
$$\text{S} \rightarrow \begin{array}{c} \text{NP} \\ (\uparrow\textsc{topic}) = \downarrow \\ \boxed{(\uparrow\textsc{topic}) = (\uparrow\textsc{comp}^{*}\ \textsc{obj})} \end{array} \quad \begin{array}{c} \text{S} \\ \uparrow=\downarrow \end{array}$$

$$\text{S} \rightarrow \begin{array}{c} \text{NP} \\ (\uparrow\textsc{subj}) = \downarrow \end{array} \quad \begin{array}{c} \text{VP} \\ \uparrow=\downarrow \end{array}$$

$$\text{VP} \rightarrow \begin{array}{c} \text{V} \\ \uparrow=\downarrow \end{array} \left(\begin{array}{c} \text{NP} \\ (\uparrow\textsc{obj}) = \downarrow) \end{array} \right) \left(\begin{array}{c} \text{S} \\ (\uparrow\textsc{comp}) = \downarrow \end{array} \right)$$

$$\text{called} \quad \text{V} \quad (\uparrow \textsc{pred}) = \text{`call'}\langle(\uparrow\textsc{subj}), (\uparrow\textsc{obj})\rangle$$

$$\text{thinks} \quad \text{V} \quad (\uparrow \textsc{pred}) = \text{`think'}\langle(\uparrow\textsc{subj}), (\uparrow\textsc{comp})\rangle$$



$$\begin{bmatrix} \textsc{pred} & \text{`think'}\Big\langle\Big(\uparrow\textsc{subj}\Big), \Big(\uparrow\textsc{comp}\Big)\Big\rangle \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{`bill'} \end{bmatrix} \\ \textsc{comp} & \begin{bmatrix} \textsc{pred} & \text{`call'}\Big\langle\Big(\uparrow\textsc{subj}\Big), \Big(\uparrow\textsc{obj}\Big)\Big\rangle \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{`john'} \end{bmatrix} \\ \textsc{obj} & \boxed{1} \end{bmatrix} \\ \textsc{topic} & \boxed{1} \begin{bmatrix} \textsc{pred} & \text{`mary'} \end{bmatrix} \end{bmatrix}$$

Expressivität von LFG

- Ist LFG ein schwach kontextsensitiver Grammatikformalismus?

- Definition schwach kontextsensitiv:
  - enthält kontextfreie Sprachen
  - cross-serial dependencies / Copy-Sprache
  - constant growth
  - polynomielles Parsingproblem

Expressivität von LFG

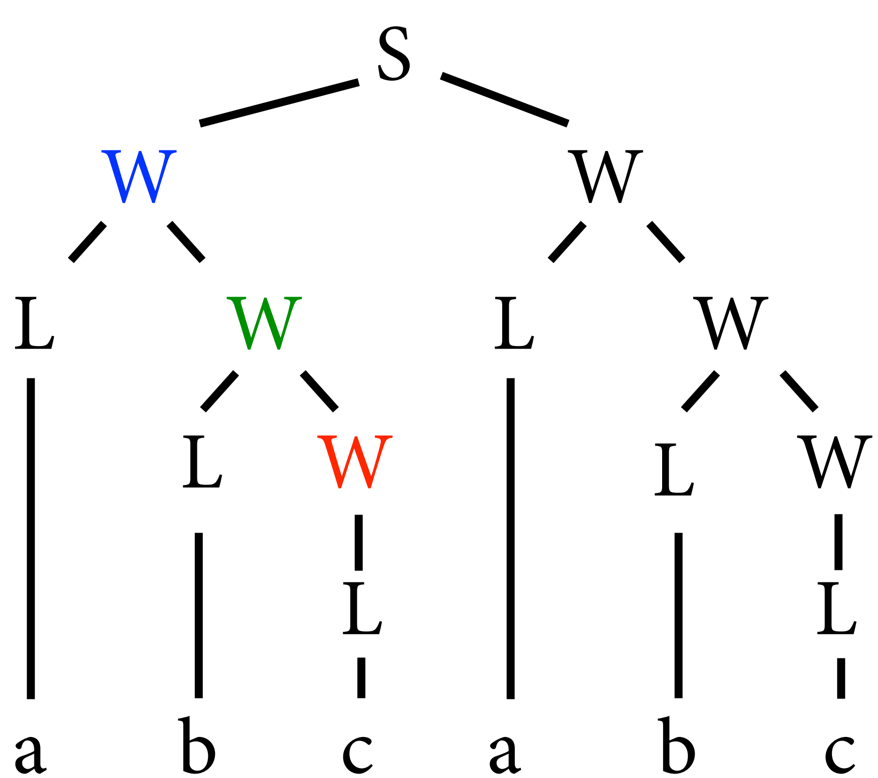
- Ist LFG ein schwach kontextsensitiver Grammatikformalismus?

- Definition schwach kontextsensitiv:
  - enthält kontextfreie Sprachen ✔
  - cross-serial dependencies / Copy-Sprache
  - constant growth
  - polynomielles Parsingproblem

Copy-Sprache in LFG

$$\begin{array}{lccc} W & \rightarrow & \underset{\uparrow=\downarrow}{L} & \left(\underset{(\uparrow W)\, =\, \downarrow}{W} \right) \\[2em] S & \rightarrow & \underset{\uparrow=\downarrow}{W} & \underset{\uparrow=\downarrow}{W} \end{array}$$

$$\begin{array}{lll} a & L & (\uparrow L) = a \\ b & L & (\uparrow L) = b \\ c & L & (\uparrow L) = c \end{array}$$



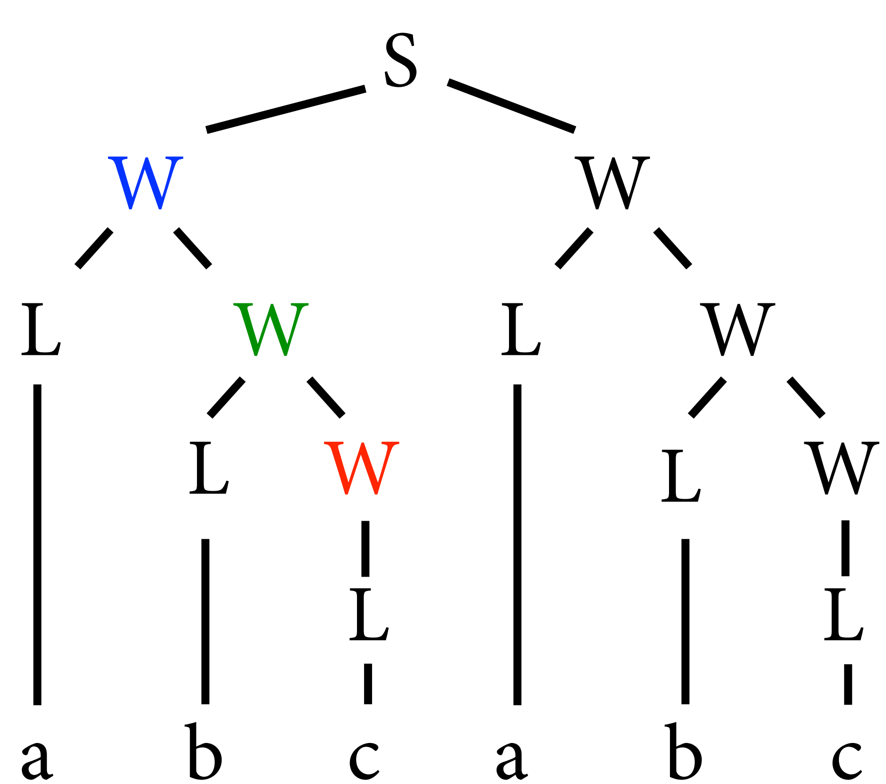
$$\begin{bmatrix} L & a \\ W & \begin{bmatrix} L & b \\ W & \begin{bmatrix} L & c \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

(Kaplan & Bresnan 82)

Copy-Sprache in LFG

$$
\begin{array}{lclc}
\mathrm{W} & \rightarrow & \underset{\uparrow=\downarrow}{\mathrm{L}} & \left(\underset{(\uparrow \mathrm{W}) = \downarrow}{\mathrm{W}} \right) \\[2em]
\mathrm{S} & \rightarrow & \underset{\uparrow=\downarrow}{\mathrm{W}} & \underset{\uparrow=\downarrow}{\mathrm{W}}
\end{array}
$$

$$
\begin{array}{lll}
\mathrm{a} & \mathrm{L} & (\uparrow L) = \mathrm{a} \\
\mathrm{b} & \mathrm{L} & (\uparrow L) = \mathrm{b} \\
\mathrm{c} & \mathrm{L} & (\uparrow L) = \mathrm{c}
\end{array}
$$



$$
\begin{bmatrix}
\mathrm{L} & \mathrm{a} \\
\mathrm{W} & \begin{bmatrix} \mathrm{L} & \mathrm{b} \\ \mathrm{W} & \boxed{\begin{bmatrix} \mathrm{L} & \mathrm{c} \end{bmatrix}} \end{bmatrix}
\end{bmatrix}
$$

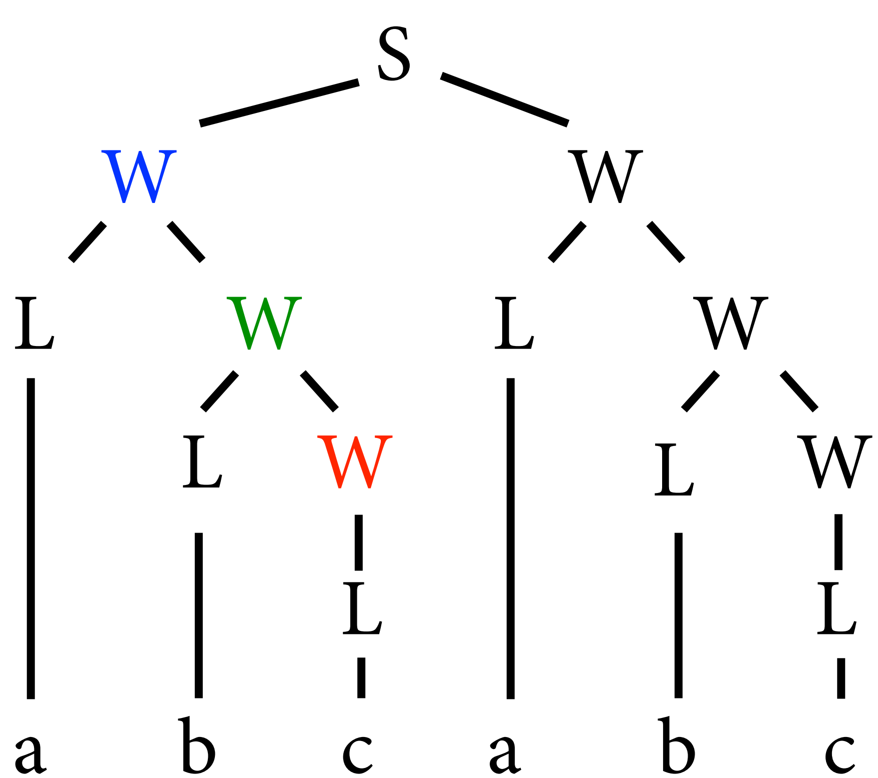
(Kaplan & Bresnan 82)

Copy-Sprache in LFG

$$\mathrm{W} \rightarrow \underset{\uparrow=\downarrow}{\mathrm{L}} \left(\underset{(\uparrow \mathrm{W}) = \downarrow}{\mathrm{W}} \right)$$

$$\mathrm{S} \rightarrow \underset{\uparrow=\downarrow}{\mathrm{W}} \quad \underset{\uparrow=\downarrow}{\mathrm{W}}$$

a	L	$(\uparrow L) = \mathrm{a}$
b	L	$(\uparrow L) = \mathrm{b}$
c	L	$(\uparrow L) = \mathrm{c}$

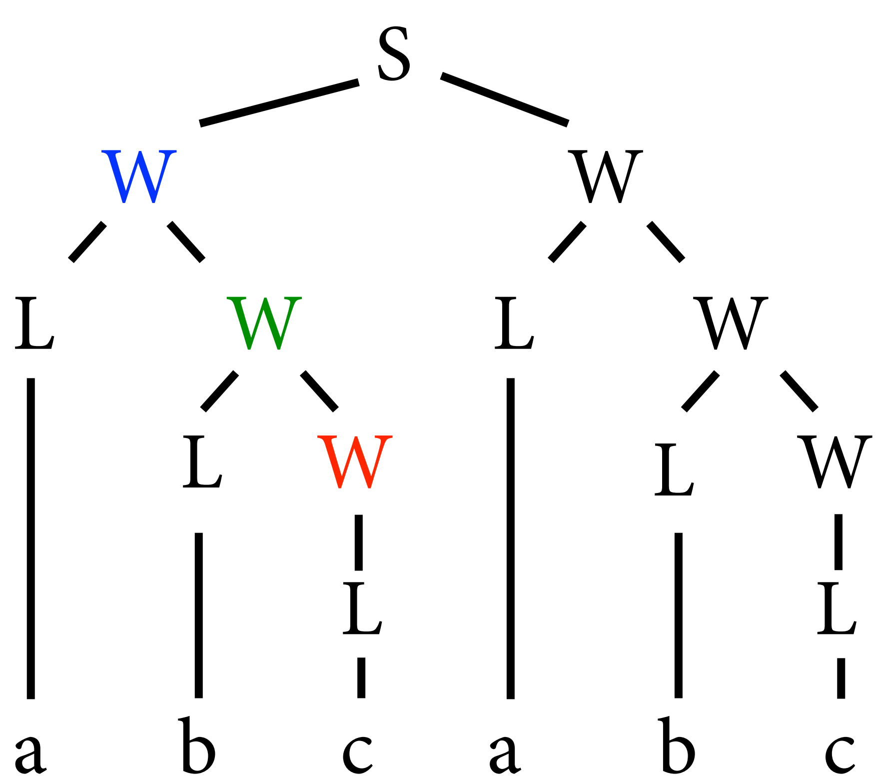


$$\begin{bmatrix} \mathrm{L} & \mathrm{a} \\ \mathrm{W} & \boxed{\begin{bmatrix} \mathrm{L} & \mathrm{b} \\ \mathrm{W} & \boxed{\begin{bmatrix} \mathrm{L} & \mathrm{c} \end{bmatrix}} \end{bmatrix}} \end{bmatrix}$$

(Kaplan & Bresnan 82)

Copy-Sprache in LFG

$$
\begin{array}{llll}
W & \rightarrow & \underset{\uparrow=\downarrow}{L} & \left(\underset{(\uparrow W)\,=\,\downarrow}{W} \right) \\[2ex]
S & \rightarrow & \underset{\uparrow=\downarrow}{W} & \underset{\uparrow=\downarrow}{W}
\end{array}
\qquad\qquad
\begin{array}{lll}
a & L & (\uparrow L) = a \\
b & L & (\uparrow L) = b \\
c & L & (\uparrow L) = c
\end{array}
$$



$$
\boxed{\begin{bmatrix} L & a \\ W & \boxed{\begin{bmatrix} L & b \\ W & \boxed{\begin{bmatrix} L & c \end{bmatrix}} \end{bmatrix}} \end{bmatrix}}
$$

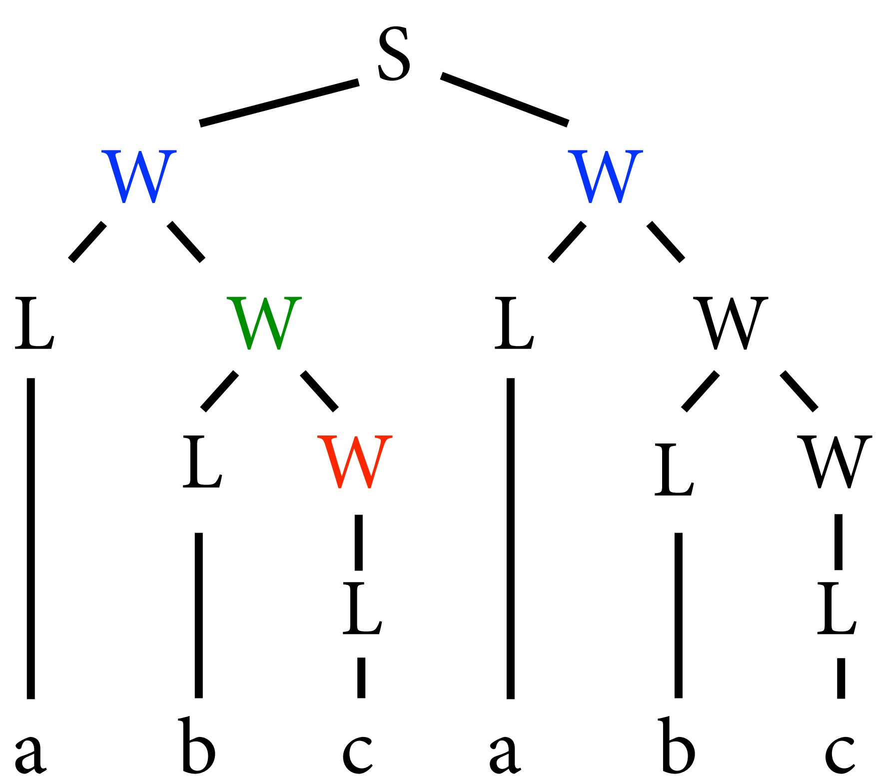
(Kaplan & Bresnan 82)

Copy-Sprache in LFG

$$W \rightarrow \underset{\uparrow=\downarrow}{L} \left(\underset{(\uparrow W) = \downarrow}{W} \right)$$

$$S \rightarrow \underset{\uparrow=\downarrow}{W} \quad \underset{\uparrow=\downarrow}{W}$$

$$\begin{array}{lll} a & L & (\uparrow L) = a \\ b & L & (\uparrow L) = b \\ c & L & (\uparrow L) = c \end{array}$$



$$\boxed{\begin{bmatrix} L & a \\ W & \boxed{\begin{bmatrix} L & b \\ W & \boxed{\begin{bmatrix} L & c \end{bmatrix}} \end{bmatrix}} \end{bmatrix}}$$

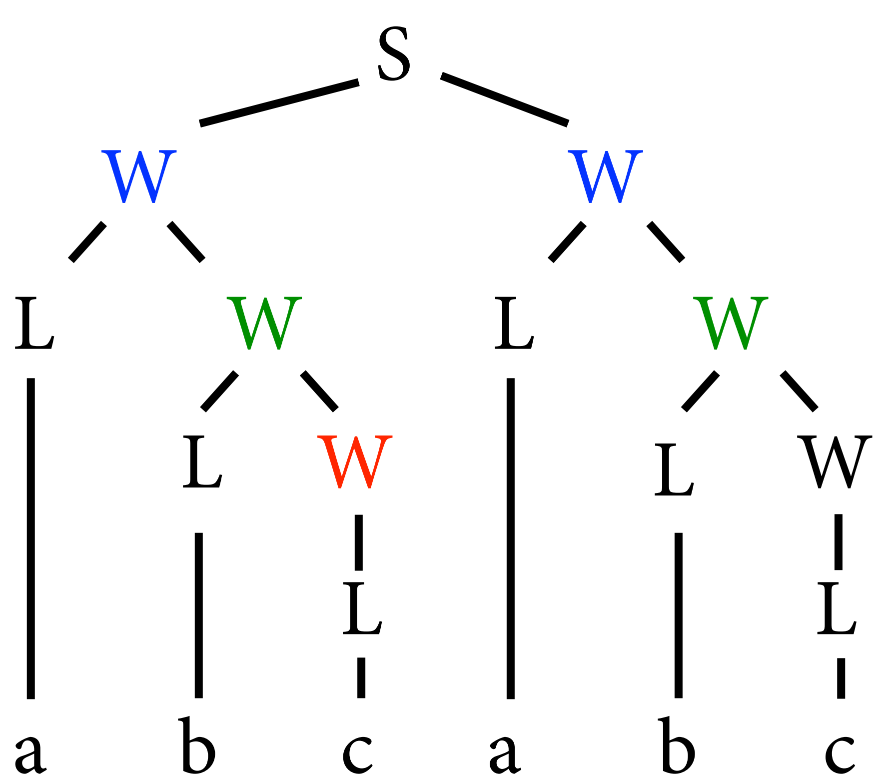
(Kaplan & Bresnan 82)

Copy-Sprache in LFG

$$\mathrm{W} \;\rightarrow\; \underset{\uparrow=\downarrow}{\mathrm{L}} \;\left(\underset{(\uparrow \mathrm{W}) \,=\, \downarrow}{\mathrm{W}} \right)$$

$$\mathrm{S} \;\rightarrow\; \underset{\uparrow=\downarrow}{\mathrm{W}} \quad \underset{\uparrow=\downarrow}{\mathrm{W}}$$

$$\begin{array}{lll} \mathrm{a} & \mathrm{L} & (\uparrow L) = \mathrm{a} \\ \mathrm{b} & \mathrm{L} & (\uparrow L) = \mathrm{b} \\ \mathrm{c} & \mathrm{L} & (\uparrow L) = \mathrm{c} \end{array}$$



$$\boxed{\begin{bmatrix} \mathrm{L} & \mathrm{a} \\ \mathrm{W} & \boxed{\begin{bmatrix} \mathrm{L} & \mathrm{b} \\ \mathrm{W} & \boxed{\begin{bmatrix} \mathrm{L} & \mathrm{c} \end{bmatrix}} \end{bmatrix}} \end{bmatrix}}$$

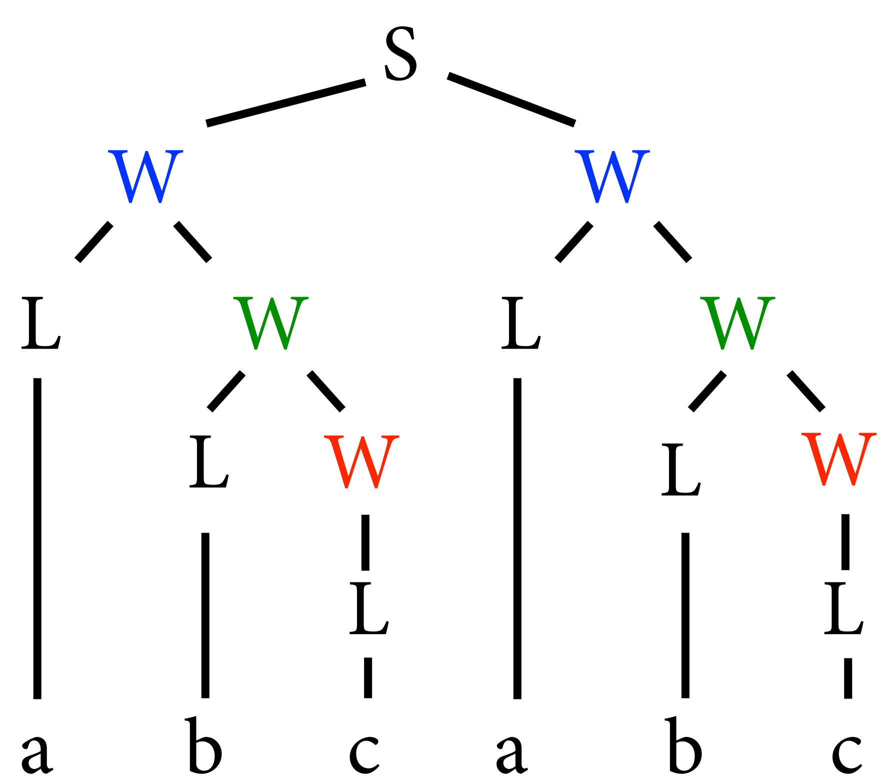
(Kaplan & Bresnan 82)

Copy-Sprache in LFG

$$W \rightarrow \underset{\uparrow=\downarrow}{L} \left(\underset{(\uparrow W) = \downarrow}{W} \right)$$

$$S \rightarrow \underset{\uparrow=\downarrow}{W} \quad \underset{\uparrow=\downarrow}{W}$$

$$\begin{array}{lll} a & L & (\uparrow L) = a \\ b & L & (\uparrow L) = b \\ c & L & (\uparrow L) = c \end{array}$$



$$\begin{bmatrix} L & a \\ W & \begin{bmatrix} L & b \\ W & \begin{bmatrix} L & c \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

(Kaplan & Bresnan 82)

Expressivität von LFG

- Ist LFG ein schwach kontextsensitiver Grammatikformalismus?

- Definition schwach kontextsensitiv:
  - enthält kontextfreie Sprachen ✔
  - cross-serial dependencies / Copy-Sprache
  - constant growth
  - polynomielles Parsingproblem

Expressivität von LFG

- Ist LFG ein schwach kontextsensitiver Grammatikformalismus?

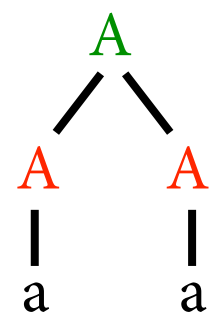
- Definition schwach kontextsensitiv:
    - enthält kontextfreie Sprachen ✔
    - cross-serial dependencies / Copy-Sprache ✔
    - constant growth
    - polynomielles Parsingproblem

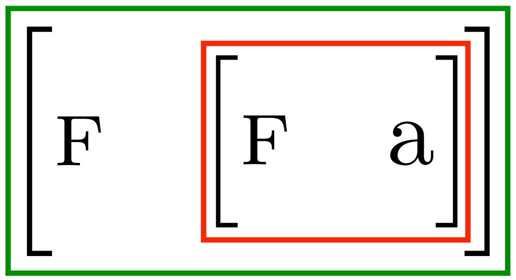
L_2 in LFG

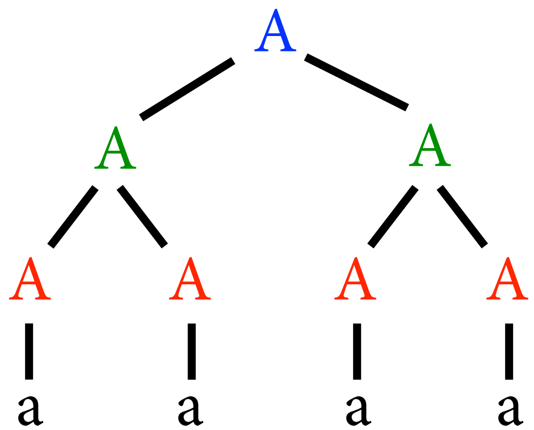
$$L_2 = \{a^{2^n} \mid n \geq 1\}$$

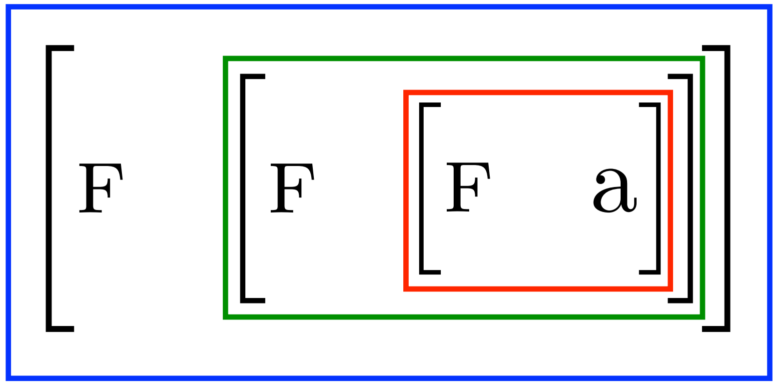
$$A \rightarrow \underset{(\uparrow f) = \downarrow}{A} \quad \underset{(\uparrow f) = \downarrow}{A}$$

$$a \quad A \quad (\uparrow f) = a$$









(Berwick 84)

Status von LFG

- LFG ist *nicht* schwach kontextsensitiv:
  hat die constant-growth-Eigenschaft nicht. ✘

- Immerhin sind alle LFG-Sprachen kontextsensitiv.

- Parsingkomplexität?

Parsing von LFG

- Naiver Algorithmus:
  - Normales Chart-Parsing mit c-Struktur-Grammatik.
  - Zähle alle möglichen c-Strukturen auf.
  - Berechne für jede einzelne c-Struktur die möglichen f-Strukturen.
- Schritte 1 und 3 sind relativ klar:
  - Schritt 1 = Chartparsing
  - Schritt 3 = Lösen von Feature-Constraints.
- Wie ist es mit Schritt 2?

Aufzählen von c-Strukturen

- Problem: String kann unendlich viele c-Strukturen haben.
  - von denen jede eine andere f-Struktur hat und deshalb vielleicht die kleinste grammatisch korrekte ist
  - insbesondere mit *Kettenregeln*, d.h. A → B und B → A
  - Kettenregeln können beliebig oft angewendet werden, ohne im Parsing Fortschritt zu machen.
- Lösung: Beim Berechnen von c-Strukturen wiederholte Anwendung von Kettenregeln verbieten.

Aufzählen von c-Strukturen

- Problem: Jetzt terminiert c-Struktur-Parser, aber wir müssen immer noch alle Bäume aufzählen.

- String kann exponentiell viele kf. Parses haben.
  - A → A A: Parsebäume = alle binären Bäume der Länge n
  - Anzahl ist n-te Catalan-Zahl $C_n = \frac{1}{n+1}\binom{2n}{n}$

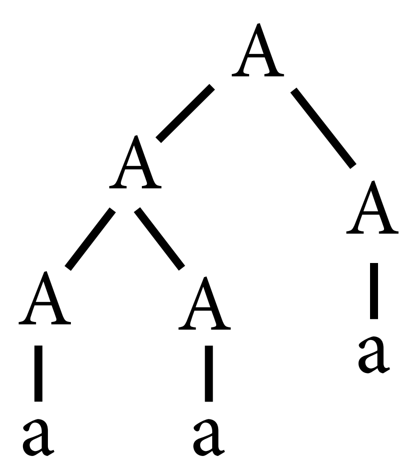
- Grundprinzip: Aufzählen von Parsebäumen ist fast immer eine ganz schlechte Idee.

Unifikation auf der Chart

- f-Struktur-Constraints schon beim Berechnen der c-Struktur-Chart auswerten.
- Manchmal schlägt Unifikation fehl. Dann Item gar nicht erst in Chart eintragen.
- Parsing immer noch worst-case exponentiell, aber Chart ist viel kleiner als Menge der c-Strukturen.

Unifikation auf der Chart

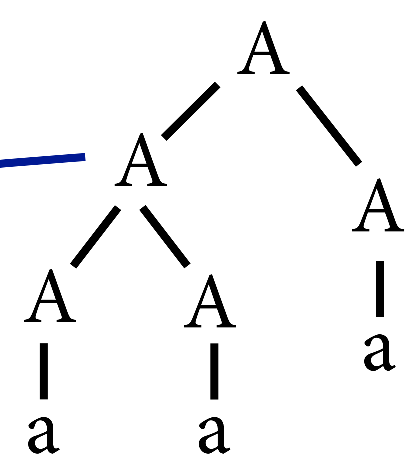
$$A \rightarrow \underset{(\uparrow f) = \downarrow}{A} \quad \underset{(\uparrow f) = \downarrow}{A} \qquad\qquad \text{a} \quad A \quad (\uparrow f) = \text{a}$$



Unifikation auf der Chart

$$A \rightarrow \underset{(\uparrow f) = \downarrow}{A} \quad \underset{(\uparrow f) = \downarrow}{A}$$

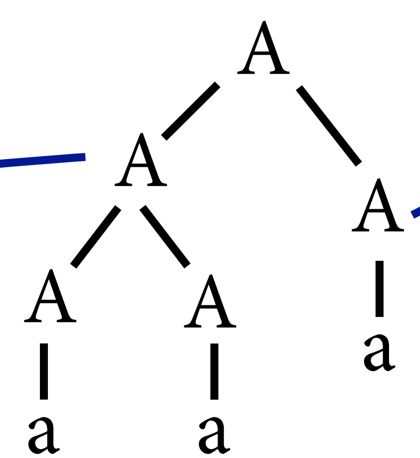
$$\text{a} \quad A \quad (\uparrow f) = \text{a}$$



Item: (A,0,2) $\begin{bmatrix} \text{F} & \begin{bmatrix} \text{F} & \text{a} \end{bmatrix} \end{bmatrix}$

Unifikation auf der Chart

A	$\rightarrow$	A	A	a	A	$(\uparrow f) = \text{a}$
		$(\uparrow f) = \downarrow$	$(\uparrow f) = \downarrow$			



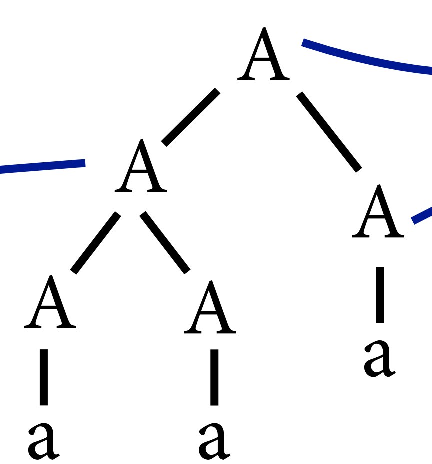
Item: (A,0,2) $\begin{bmatrix} \text{F} & \begin{bmatrix} \text{F} & \text{a} \end{bmatrix} \end{bmatrix}$

Item: (A,2,3) $\begin{bmatrix} \text{F} & \text{a} \end{bmatrix}$

Unifikation auf der Chart

$$A \quad \rightarrow \quad \underset{(\uparrow f) =\downarrow}{A} \quad \underset{(\uparrow f) =\downarrow}{A} \qquad\qquad a \quad A \quad (\uparrow f) = a$$

Unifikation schlägt fehl:
keine f-Struktur für (A,0,3) abgeleitet.



Item: (A,2,3) $\begin{bmatrix} \text{F} & \text{a} \end{bmatrix}$

Item: (A,0,2) $\begin{bmatrix} \text{F} & \begin{bmatrix} \text{F} & \text{a} \end{bmatrix} \end{bmatrix}$

LFG-Parsing

- LFG-spezifische Optimierung: Aus c-Chart einen f-Struktur-Constraint mit *Disjunktionen* berechnen und diesen lösen (Maxwell & Kaplan 95).

- Beste bekannte Algorithmen für LFG-Parsing sind worst-case exponentiell. Optimierungen reduzieren *typische* Laufzeit mit echten Grammatiken.

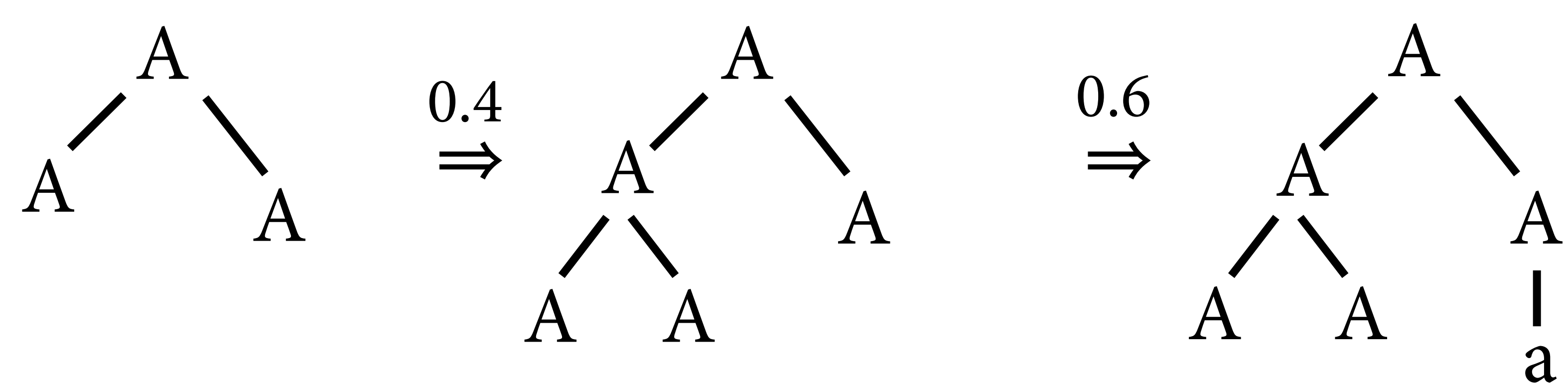
- Praxis: Parser im XLE-System extrem effizient.

Wahrscheinlichkeitsmodell

- LFG-Grammatiken enthalten kfGen. Naheliegender Gedanke: PCFGen auf LFG verallgemeinern.

- PCFG: Expansion verschiedener Knoten sind statistisch unabhängige Ereignisse.

$A \rightarrow A \ \ A$	[0.4]
$A \rightarrow a$	[0.6]

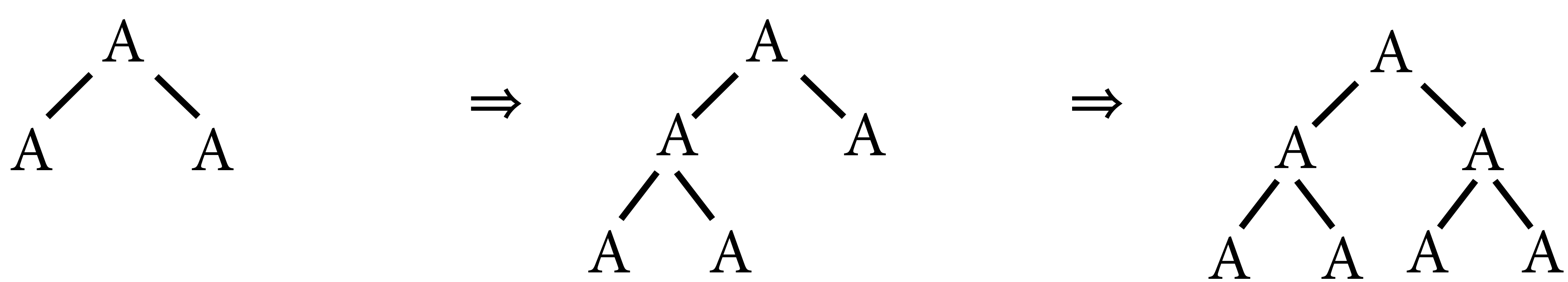


Statistische Abhängigkeiten

- Problem: In LFG schränken f-Constraints ein, wie verschiedene c-Knoten expandiert werden dürfen.

- Expansionen *nicht statistisch unabhängig*.

$$\begin{array}{llll}
A & \rightarrow & \underset{(\uparrow f)=\downarrow}{A} & \underset{(\uparrow f)=\downarrow}{A} \\
a & & A & (\uparrow f) = a
\end{array}$$



Maximum-Entropy-Modell

- Typische Lösung: Verwende ein Maximum-Entropy-Modell für LFG-Ableitungen.

$$P(t|w) = \frac{e^{\theta \cdot f(t,w)}}{\sum_{t'} e^{\theta \cdot f(t',w)}}$$

- Feature-Funktionen $f_i(t,w)$; Gewichte θ_i.

- MaxEnt-Modelle sind robust gegenüber statistischen Abhängigkeiten der Features.

- Maximum Entropy = log-linear (siehe CCG).

(Riezler et al. 02)

MaxEnt für LFG: Features

- Features über c-Struktur
  - wie oft wurde jede c-Regel verwendet?
- Features über f-Struktur
  - wie oft wird jede grammatische Funktion verwendet?
  - von wem wird sie gefüllt?
- Lexikon-Features:
  - wie oft wird Wort x mit Lexikoneintrag L verwendet?
- Und noch viele mehr.

(Riezler et al. 02)

Evaluation

- Penn Treebank hat keine LFG-Annotationen, aber kann PTB in LFG-Baumbank “konvertieren”:
  - Originaltext mit großer LFG-Grammatik parsen
  - nur LFG-Ableitungen behalten, deren c-Struktur mit Annotation übereinstimmt

- F-Score
(aus Precision & Recall von f-Struktur-Einträgen):
  - lower bound (zufällige Auswahl aus Parses): 75.5
  - upper bound (Parse mit bestem Match): 84.1
  - trainiertes MaxEnt-Modell: 78.6

(Riezler et al. 02)

Zusammenfassung

- Fernabhängigkeiten gehen prima,
  mit Functional Uncertainty.

- Expressivität und Parsing:
  - kontextsensitiv, aber nicht schwach kontextsensitiv
  - worst-case-Komplexität ist exponentiell
  - durch geschickten Umgang mit Unifikationen
    in der Praxis effizient parsbar.

- Standard-W.modell: Maximum Entropy.

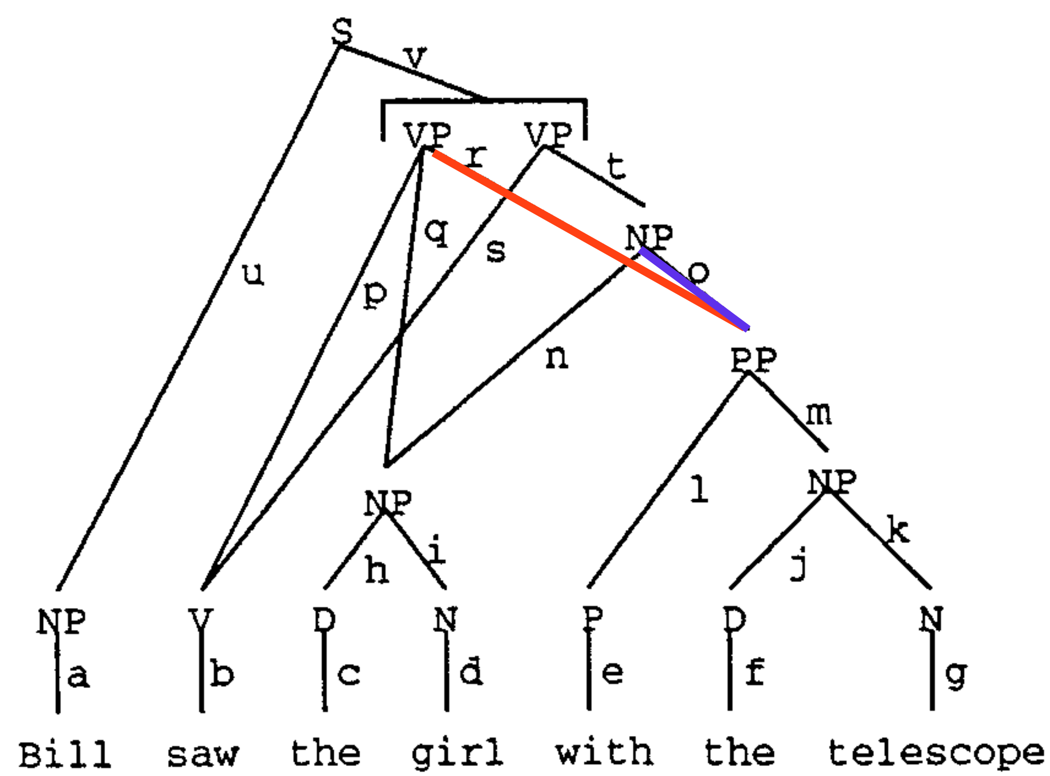
Globale f-Constraints

- Jede Regel in der Chart trägt f-Constraints bei.
  Wir sammeln alle auf und lösen sie am Schluss alle auf einmal.

- Eintrag (A, i, k) in Chart mit Regeln
  $A \rightarrow r_1, \ldots, \; A \rightarrow r_n$ expandiert:
  Constraint für (A,i,k) ist *Disjunktion* der Constraints für $r_1, \ldots, r_n$.

(Maxwell & Kaplan 95)

Globale f-Constraints



$$a \wedge u \wedge b \wedge c \wedge h \wedge d \wedge i \wedge e \wedge l \wedge f \wedge j \wedge g \wedge k \wedge m \wedge p \wedge q \wedge (r \vee o) \wedge v$$

f-Struktur 1: $a \wedge u \wedge b \wedge c \wedge h \wedge d \wedge i \wedge e \wedge l \wedge f \wedge j \wedge g \wedge k \wedge m \wedge p \wedge q \wedge r \wedge v$

f-Struktur 2: $a \wedge u \wedge b \wedge c \wedge h \wedge d \wedge i \wedge e \wedge l \wedge f \wedge j \wedge g \wedge k \wedge m \wedge p \wedge q \wedge o \wedge v$