Minimalist Grammar Transition-Based Parsing

Miloš Stanojević
m.stanojevic@uva.nl

Institute for Logic, Language and Computation
University of Amsterdam

Logical Aspects of Computational Linguistics 2016

Motivation

Types of parsers

- Chart-based : full search space
- Transition-based : partial search space, no guarantees

Motivation

Types of parsers

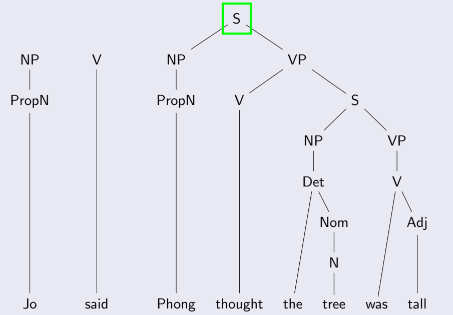
- Chart-based : full search space
- Transition-based : partial search space, no guarantees

Why use Transition-Based parser instead of Chart-Based one?

- Speed: $O(n^{4m+4})$ vs $O(n^2)$
- Accuracy: local vs non-local conditioning

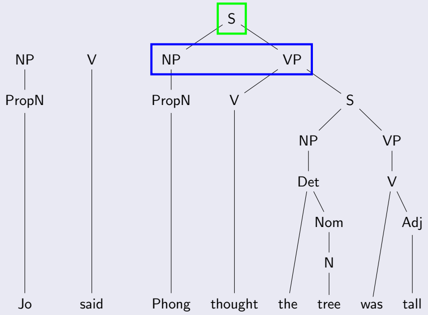
Accuracy – Locally vs Non-Locally Conditioned Models

Example sentence



Accuracy – Locally vs Non-Locally Conditioned Models

Example sentence

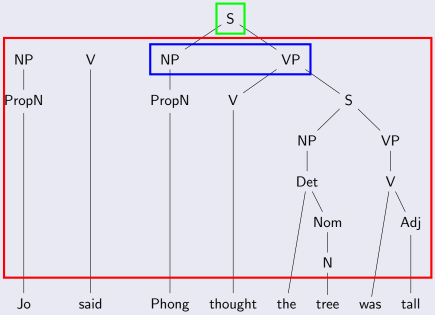


Locally conditioned model

$$P(\boxed{\text{S}} \mid \boxed{\text{NP} \quad \text{VP}})$$

Accuracy – Locally vs Non-Locally Conditioned Models

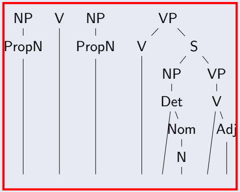
Example sentence



Locally conditioned model

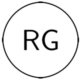
$$P(\boxed{\text{S}} \mid \boxed{\text{NP} \quad \text{VP}})$$

Non-Locally conditioned model

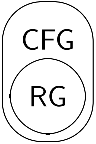
$P(\boxed{\text{S}} \mid$  $)$

Very successfull for CFG, CCG and dependency parsing.

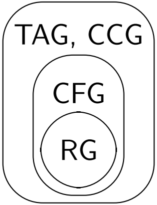
Speed



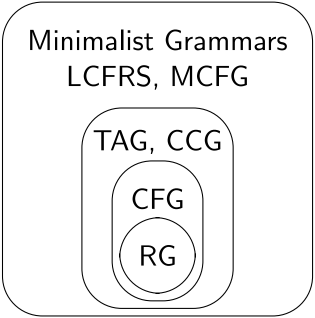
Speed



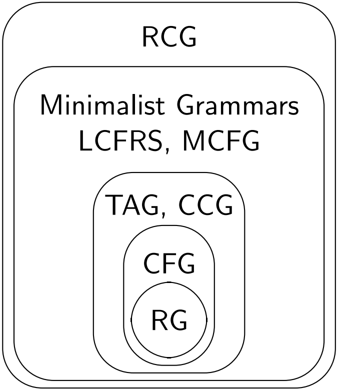
Speed



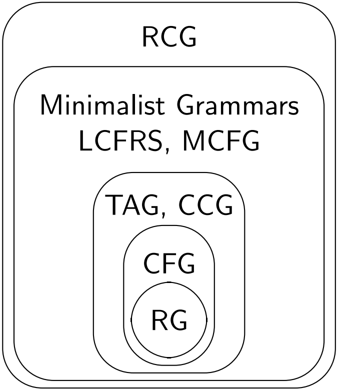
Speed



Speed



Speed



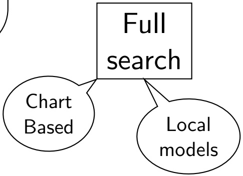
RCG

Minimalist Grammars
LCFRS, MCFG

TAG, CCG

CFG

RG

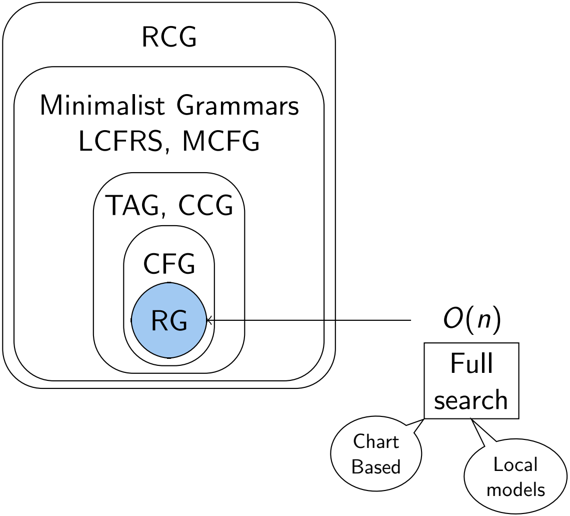


Full search

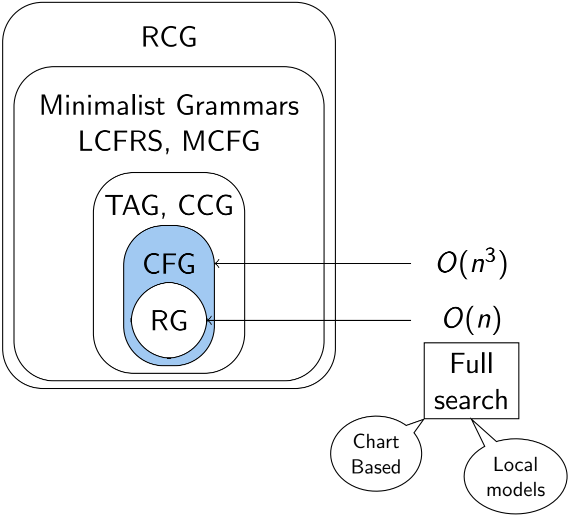
Chart Based

Local models

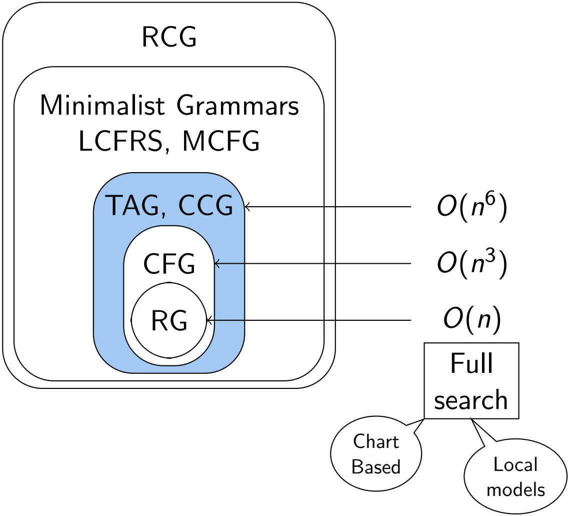
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n)$
Full search
Chart Based
Local models

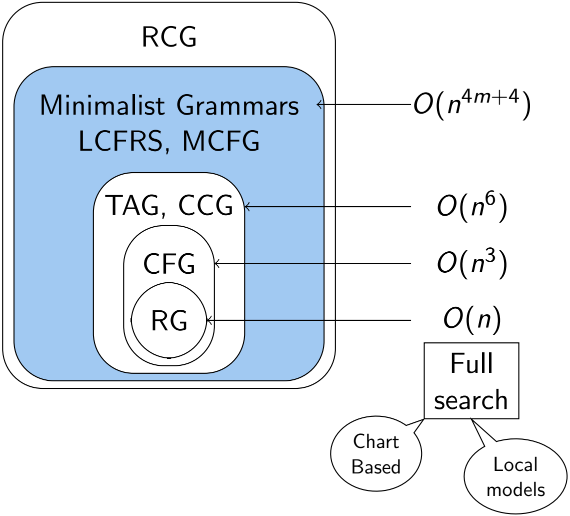
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^3)$
$O(n)$
Full search
Chart Based
Local models

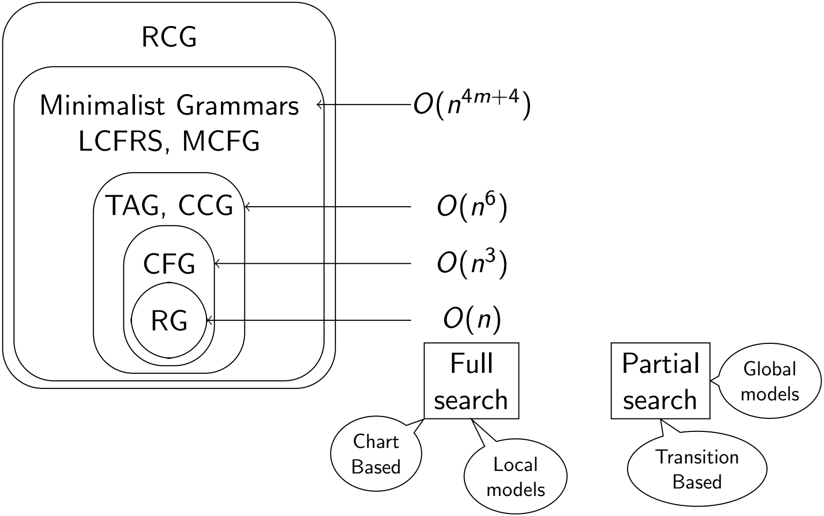
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^6)$
$O(n^3)$
$O(n)$
Full search
Chart Based
Local models

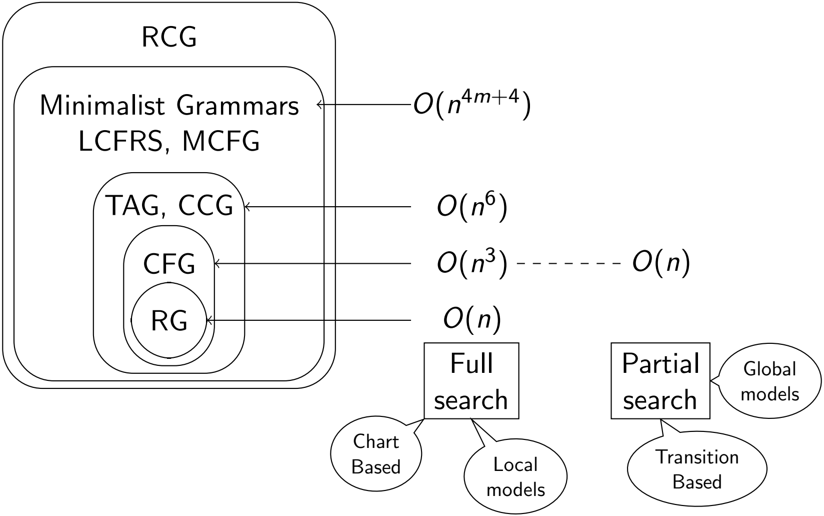
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
$O(n^6)$
$O(n^3)$
$O(n)$
Full search
Chart Based
Local models

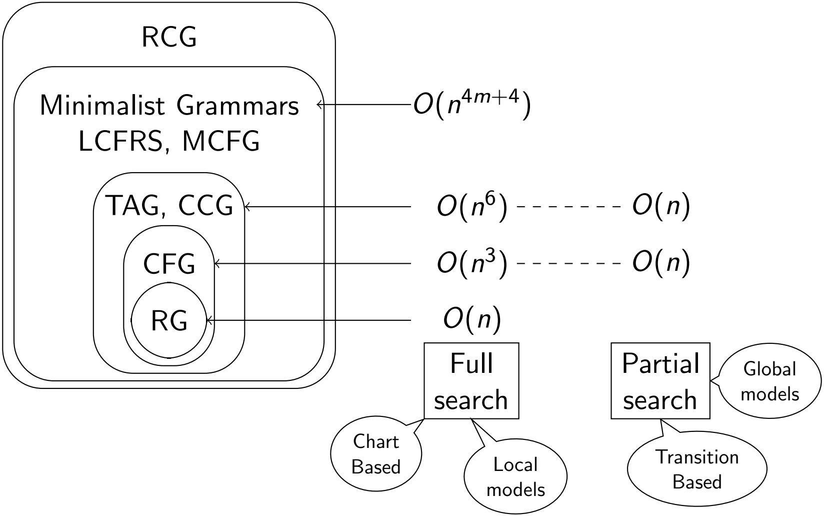
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
$O(n^6)$
$O(n^3)$
$O(n)$
Full search
Chart Based
Local models
Partial search
Global models
Transition Based

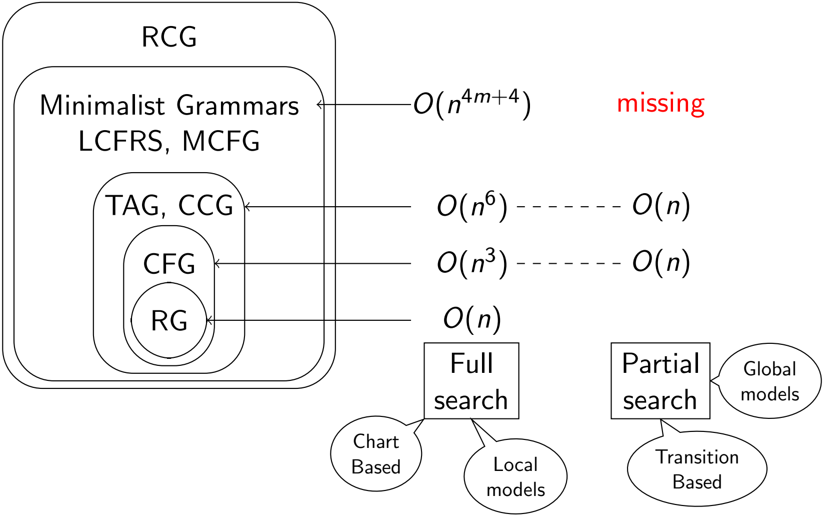
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
$O(n^6)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Partial search
Chart Based
Local models
Global models
Transition Based

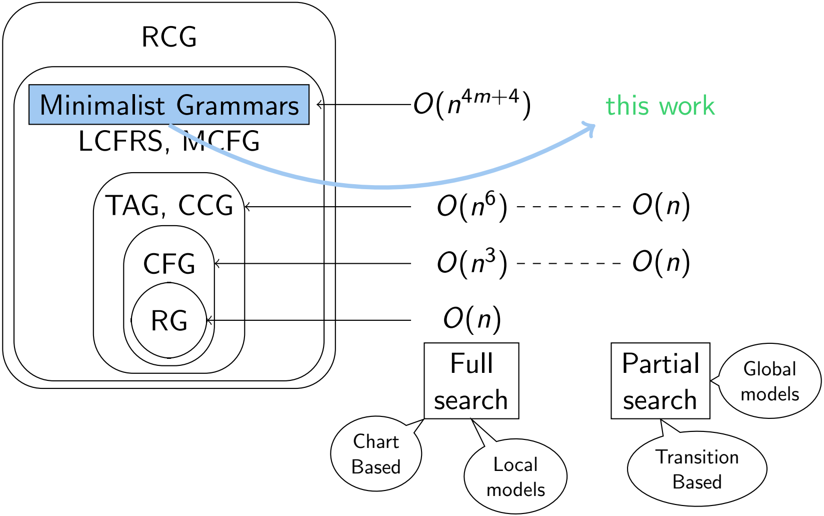
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
$O(n^6)$ - - - - - - - - $O(n)$
$O(n^3)$ - - - - - - - - $O(n)$
$O(n)$
Full search
Partial search
Global models
Chart Based
Local models
Transition Based

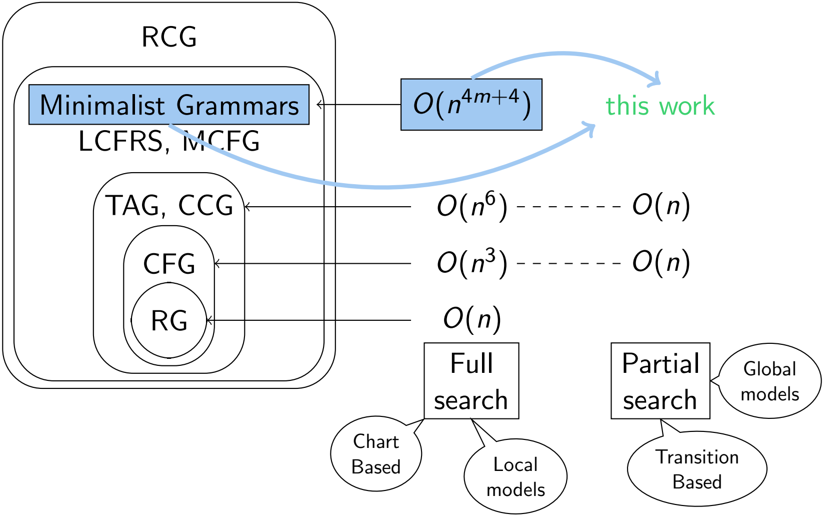
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
missing
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Partial search
Global models
Chart Based
Local models
Transition Based

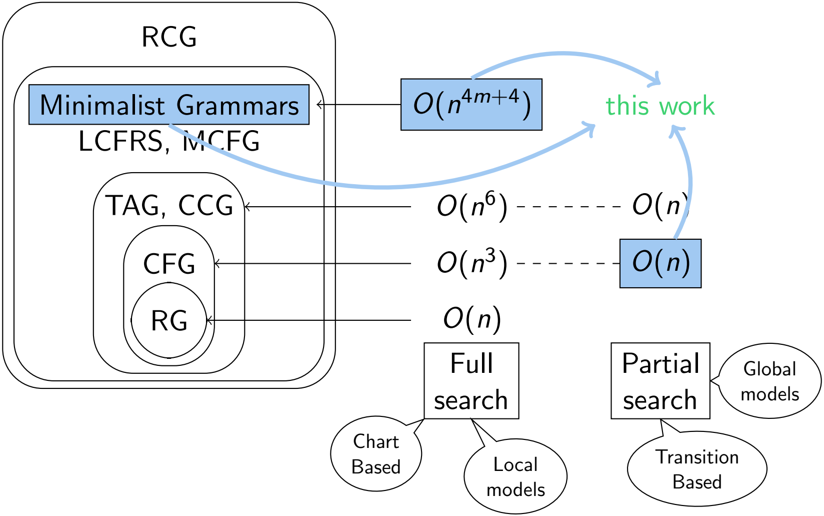
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
this work
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Partial search
Global models
Chart Based
Local models
Transition Based

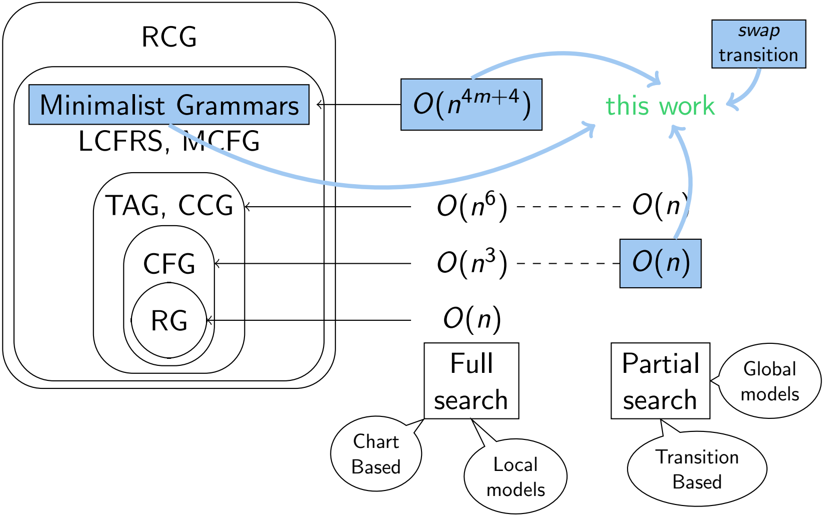
Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
this work
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Partial search
Global models
Chart Based
Local models
Transition Based

Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
this work
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Partial search
Global models
Chart Based
Local models
Transition Based

Speed


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
this work
swap transition
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Partial search
Global models
Chart Based
Local models
Transition Based

Minimalist Grammars

Composition Functions

- *merge*
- *move*

Features

- features for *merge*:
  - selectees: v, n, c, d, a
  - selectors: $=v,\ =n,\ =c,\ =d,\ =a$
- features for *move*:
  - licensees: $-wh, -case$
  - licensors: $+wh, +case$

Minimalist Grammars

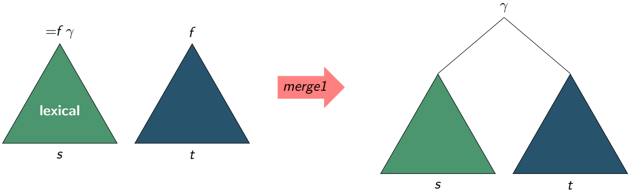
Lexicon

- likes :: $=d \;\; =d \;\; v$
- Chomsky :: d
- what :: $d \;\; -wh$
- ε :: $=v \;\; c$
- ε :: $=v \;\; +wh \; c$

Chains and Expressions

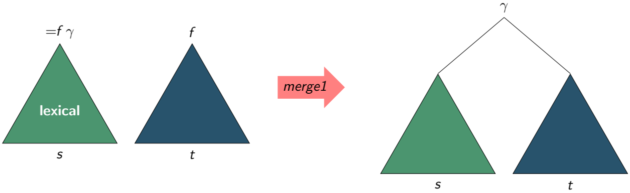
- likes Chomsky : $=d \;\; v$
- likes : $=d \;\; v$, what : $-wh$

Merge for Complements



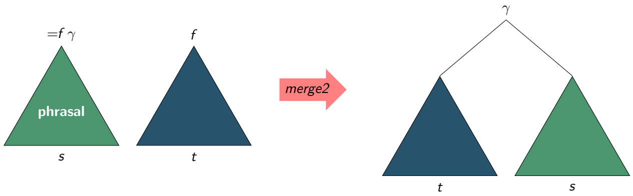
$$\textit{merge1}\ \frac{s :: =f\ \gamma \qquad\qquad t \cdot f, \alpha_1, \ldots, \alpha_k}{st : \gamma, \alpha_1, \ldots, \alpha_k}$$

Merge for Complements



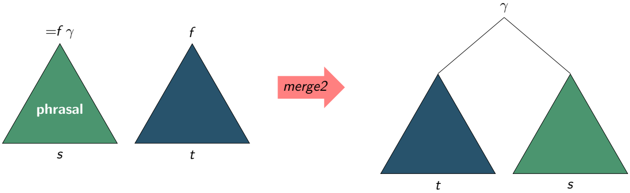
$$\textit{merge1} \quad \frac{s :: =f\gamma \qquad\qquad t \cdot f, \alpha_1, \ldots, \alpha_k}{st : \gamma, \alpha_1, \ldots, \alpha_k}$$

Merge for Specifiers



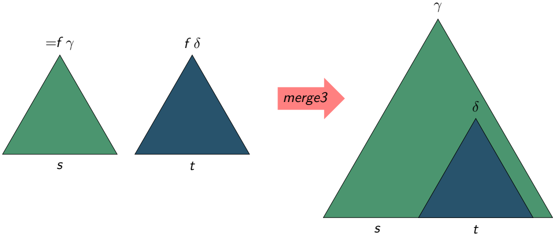
$$\textit{merge2}\ \frac{s : {=}f\ \gamma, \alpha_1, \ldots, \alpha_k \qquad\qquad t \cdot f, \iota_1, \ldots, \iota_l}{ts : \gamma, \alpha_1, \ldots, \alpha_k, \iota_1, \ldots, \iota_l}$$

Merge for Specifiers



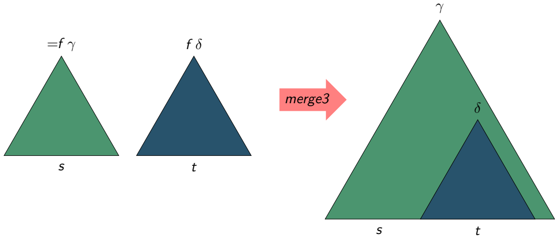
$$\textit{merge2}\ \frac{s : \boxed{=f}\boxed{\gamma, \alpha_1, \ldots, \alpha_k} \qquad\qquad t \cdot \boxed{f}, \boxed{\iota_1, \ldots, \iota_l}}{ts : \boxed{\gamma, \alpha_1, \ldots, \alpha_k}, \boxed{\iota_1, \ldots, \iota_l}}$$

Merge for Moving Constituents



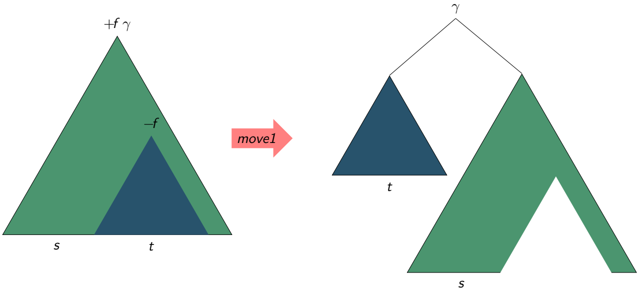
$$merge3 \quad \frac{s \cdot \;\; {=}f\,\gamma, \alpha_1, \ldots, \alpha_k \qquad\qquad t \cdot f\,\delta, \iota_1, \ldots, \iota_l}{s : \gamma, \alpha_1, \ldots, \alpha_k, t : \delta, \iota_1, \ldots, \iota_l}$$

Merge for Moving Constituents



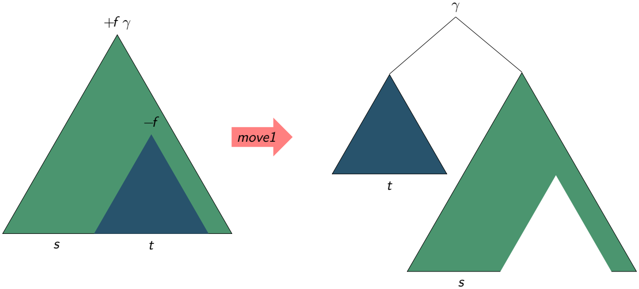
$$\textit{merge3}\quad \frac{s \cdot =f\gamma, \alpha_1, \ldots, \alpha_k \qquad t \cdot f\delta, \iota_1, \ldots, \iota_l}{s : \gamma, \alpha_1, \ldots, \alpha_k, t : \delta, \iota_1, \ldots, \iota_l}$$

Move for the Landing Constituents



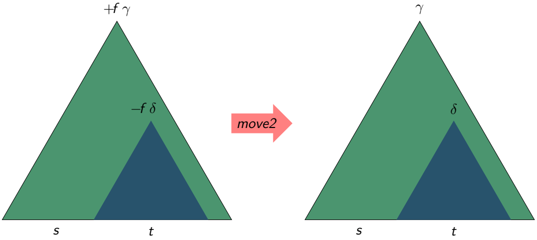
$$move1 \quad \frac{s : +f\ \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : -f, \alpha_{i+1}, \ldots, \alpha_k}{ts : \gamma, \alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_k}$$

Move for the Landing Constituents



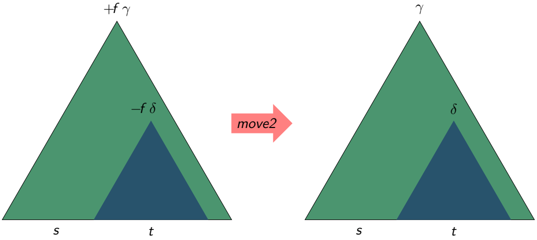
$$move1 \quad \frac{s : +f\,\gamma, \alpha_1, \ldots, \alpha_{i-1},\ t : -f, \alpha_{i+1}, \ldots, \alpha_k}{ts : \gamma, \alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_k}$$

Move for the Moving Constituents



$$\textit{move2}\quad \frac{s : +f\ \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : -f\ \delta, \alpha_{i+1}, \ldots, \alpha_k}{s : \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : \delta, \alpha_{i+1}, \ldots, \alpha_k}$$

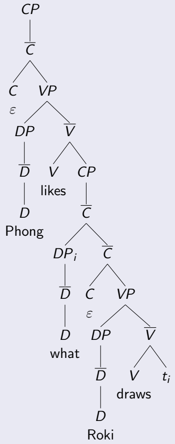
Move for the Moving Constituents



$$move2 \quad \frac{s : +f\ \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : -f\ \delta, \alpha_{i+1}, \ldots, \alpha_k}{s : \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : \delta, \alpha_{i+1}, \ldots, \alpha_k}$$

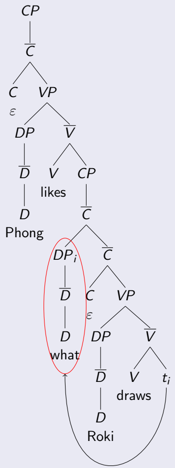
Derived and Derivation Trees

X-bar Structure



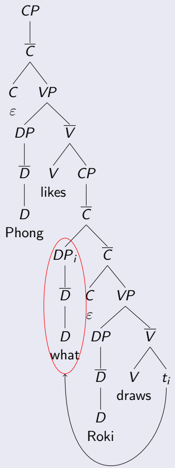
Derived and Derivation Trees

X-bar Structure

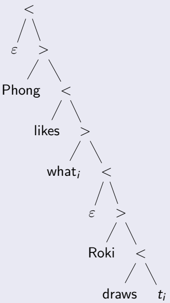


Derived and Derivation Trees

X-bar Structure

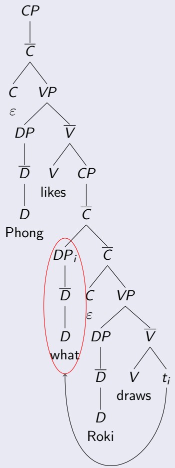


Bare Phrase Structure

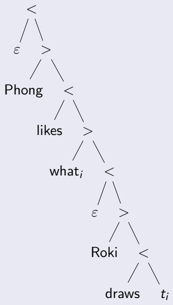


Derived and Derivation Trees

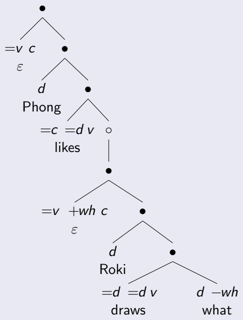
X-bar Structure



Bare Phrase Structure



Derivation Tree



Example MG derivation

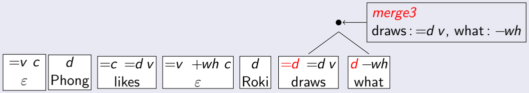
Derivation Tree

$=v\ c$	d	$=c\ =d\ v$	$=v\ +wh\ c$	d	$=d\ =d\ v$	$d\ -wh$
ε	Phong	likes	ε	Roki	draws	what

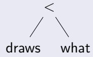
Derived Tree

Example MG derivation

Derivation Tree

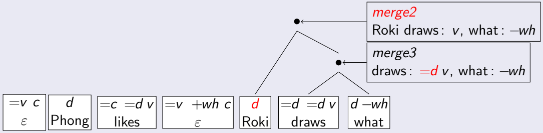


Derived Tree



Example MG derivation

Derivation Tree



merge2
Roki draws : v, what : $-wh$

merge3
draws : $=d$ v, what : $-wh$

$=v$ c
ε

d
Phong

$=c$ $=d$ v
likes

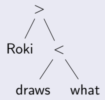
$=v$ $+wh$ c
ε

d
Roki

$=d$ $=d$ v
draws

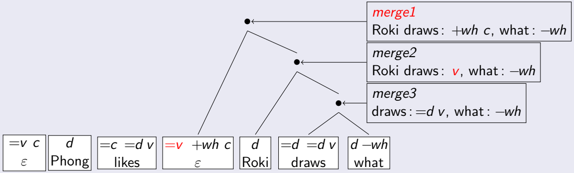
d $-wh$
what

Derived Tree



Example MG derivation

Derivation Tree



merge1
Roki draws : $+wh\ c$, what : $-wh$

merge2
Roki draws : v, what : $-wh$

merge3
draws : $=d\ v$, what : $-wh$

$=v\ c$
ε

d
Phong

$=c\ \ =d\ v$
likes

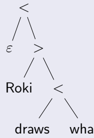
$=v\ \ +wh\ c$
ε

d
Roki

$=d\ \ =d\ v$
draws

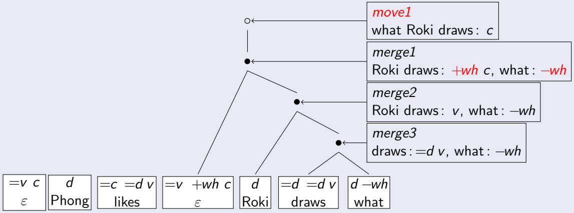
$d\ -wh$
what

Derived Tree



Example MG derivation

Derivation Tree



move1
what Roki draws : c

merge1
Roki draws : $+wh$ c, what : $-wh$

merge2
Roki draws : v, what : $-wh$

merge3
draws : $=d$ v, what : $-wh$

$=v$ c
ε

d
Phong

$=c$ $=d$ v
likes

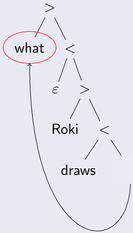
$=v$ $+wh$ c
ε

d
Roki

$=d$ $=d$ v
draws

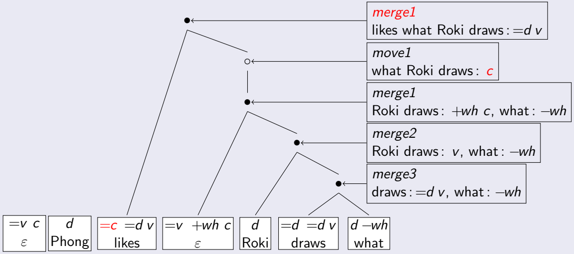
d $-wh$
what

Derived Tree

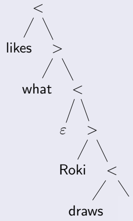


Example MG derivation

Derivation Tree

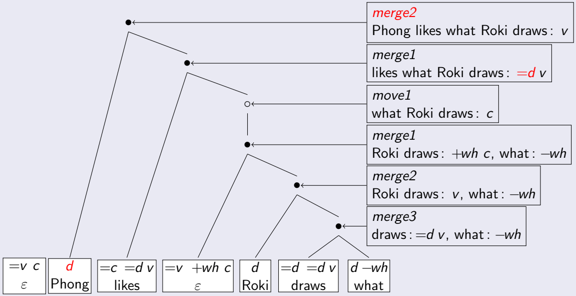


Derived Tree



Example MG derivation

Derivation Tree



merge2
Phong likes what Roki draws : v

merge1
likes what Roki draws : $=d\ v$

move1
what Roki draws : c

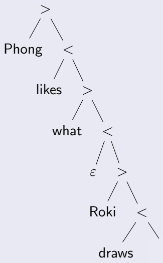
merge1
Roki draws : $+wh\ c$, what : $-wh$

merge2
Roki draws : v, what : $-wh$

merge3
draws : $=d\ v$, what : $-wh$

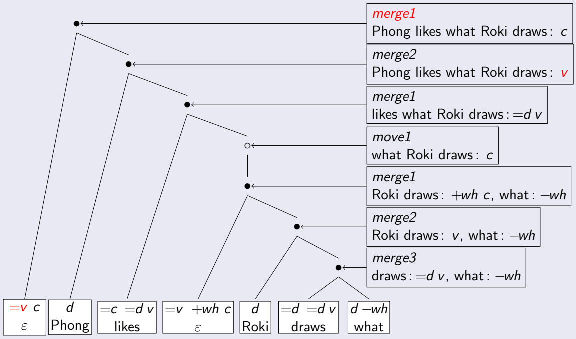
$=v\ c$	d	$=c\ =d\ v$	$=v\ +wh\ c$	d	$=d\ =d\ v$	$d\ -wh$
ε	Phong	likes	ε	Roki	draws	what

Derived Tree



Example MG derivation

Derivation Tree



merge1
Phong likes what Roki draws : c

merge2
Phong likes what Roki draws : v

merge1
likes what Roki draws : $=d\ v$

move1
what Roki draws : c

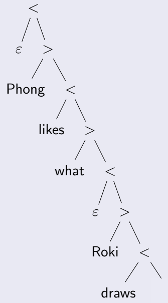
merge1
Roki draws : $+wh\ c$, what : $-wh$

merge2
Roki draws : v, what : $-wh$

merge3
draws : $=d\ v$, what : $-wh$

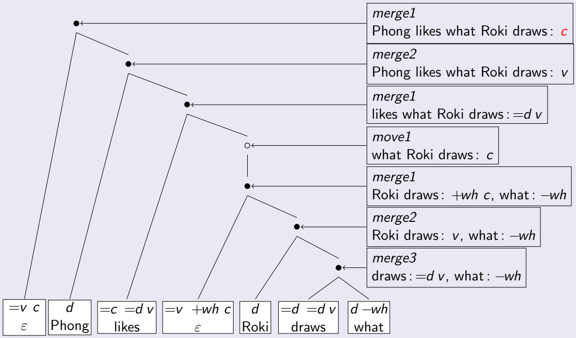
$=v\ c$	d	$=c\ =d\ v$	$=v\ +wh\ c$	d	$=d\ =d\ v$	$d\ -wh$
ε	Phong	likes	ε	Roki	draws	what

Derived Tree



Example MG derivation

Derivation Tree



merge1
Phong likes what Roki draws : c

merge2
Phong likes what Roki draws : v

merge1
likes what Roki draws : $=d\ v$

move1
what Roki draws : c

merge1
Roki draws : $+wh\ c$, what : $-wh$

merge2
Roki draws : v, what : $-wh$

merge3
draws : $=d\ v$, what : $-wh$

$=v\ c$ ε

d Phong

$=c\ =d\ v$ likes

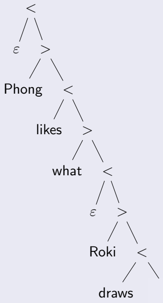
$=v\ +wh\ c$ ε

d Roki

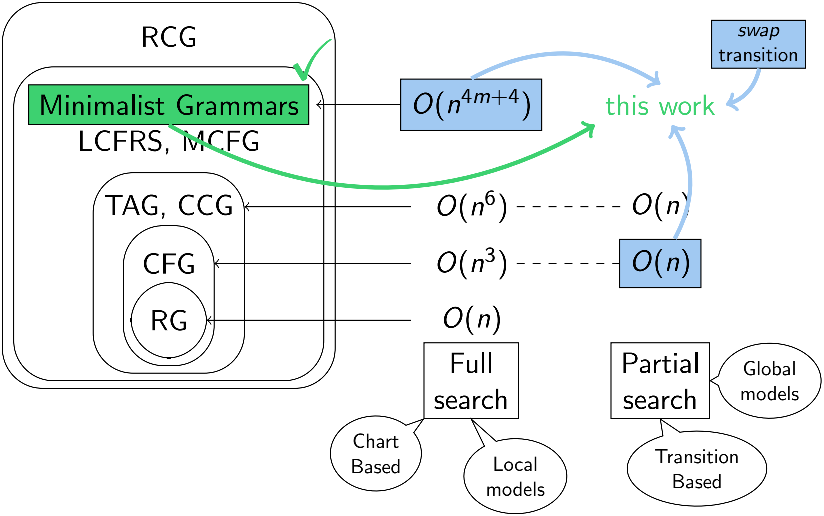
$=d\ =d\ v$ draws

$d\ -wh$ what

Derived Tree



Big Picture



RCG

Minimalist Grammars

LCFRS, MCFG

TAG, CCG

CFG

RG

$O(n^{4m+4})$

this work

swap transition

$O(n^6)$ - - - - - - - $O(n)$

$O(n^3)$ - - - - - - - $O(n)$

$O(n)$

Full search

Partial search

Chart Based

Local models

Transition Based

Global models

Chart-Based MG Parser

Harkema/Stabler MG Chart Parser

- Items $\sim$ MG expressions
- Inference rules $\sim$ MG composition functions
- axioms $\sim$ supertags
- goal $(0, n) \cdot c$
- Computational Complexity $O(n^{4m+4})$

Chart-Based MG Parser

merge1 DEFINITION

$$\frac{s :: =f\ \gamma \qquad t\ \cdot f, \alpha_1, \ldots, \alpha_k}{st : \gamma, \alpha_1, \ldots, \alpha_k}$$

merge1 INFERENCE RULE

$$\frac{(a, b) :: =f\ \gamma \qquad (b, c) \cdot f, \alpha_1, \ldots, \alpha_k}{(a, c) : \gamma, \alpha_1, \ldots, \alpha_k}$$

Chart-Based MG Parser

merge2 definition

$$\frac{s : {=}f\ \gamma, \alpha_1, \ldots, \alpha_k \qquad t \cdot f, \iota_1, \ldots, \iota_l}{ts : \gamma, \alpha_1, \ldots, \alpha_k, \iota_1, \ldots, \iota_l}$$

merge2 inference rule

$$\frac{(b, c) : {=}f\ \gamma, \alpha_1, \ldots, \alpha_k \qquad (a, b) \cdot f, \iota_1, \ldots, \iota_l}{(a, c) : \gamma, \alpha_1, \ldots, \alpha_k, \iota_1, \ldots, \iota_l}$$

Chart-Based MG Parser

merge3 Definition

$$\frac{s \quad \cdot{=}f\ \gamma, \alpha_1, \ldots, \alpha_k \qquad\qquad t \ \cdot f\ \delta, \iota_1, \ldots, \iota_l}{s : \gamma, \alpha_1, \ldots, \alpha_k, t : \delta, \iota_1, \ldots, \iota_l}$$

merge3 Inference Rule

$$\frac{(a, b) \cdot {=}f\ \gamma, \alpha_1, \ldots, \alpha_k \qquad\qquad (c, d) \cdot f\,\delta, \iota_1, \ldots, \iota_l}{(a, b) : \gamma, \alpha_1, \ldots, \alpha_k, (c, d) : \delta, \iota_1, \ldots, \iota_l}$$

Chart-Based MG Parser

move1 DEFINITION

$$\frac{s : +f\,\gamma, \alpha_1, \ldots, \alpha_{i-1}, \; t : -f, \alpha_{i+1}, \ldots, \alpha_k}{ts : \gamma, \alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_k}$$

move1 INFERENCE RULE

$$\frac{(b, c) : +f\,\gamma, \alpha_1, \ldots, \alpha_{i-1}, \; (a, b) : -f, \alpha_{i+1}, \ldots, \alpha_k}{(a, c) : \gamma, \alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_k}$$

Chart-Based MG Parser

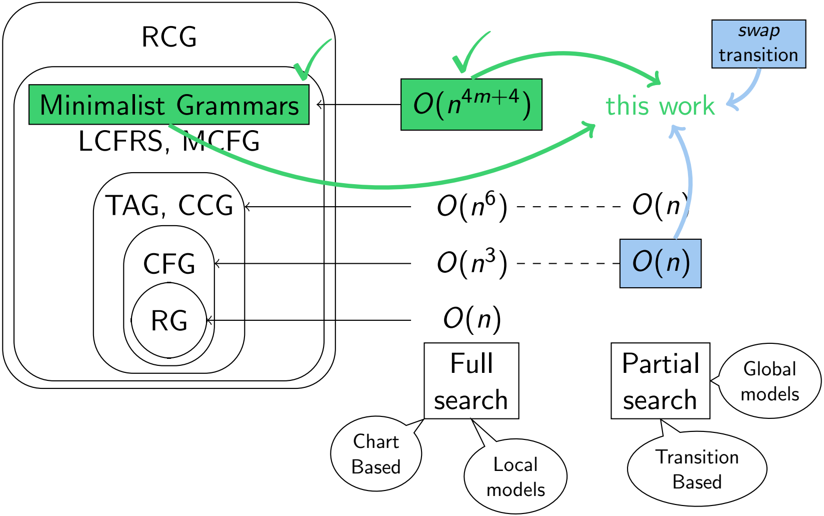
move2 DEFINITION

$$\frac{s : +f\ \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : -f\ \delta, \alpha_{i+1}, \ldots, \alpha_k}{s : \gamma, \alpha_1, \ldots, \alpha_{i-1}, t : \delta, \alpha_{i+1}, \ldots, \alpha_k}$$

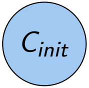
move2 INFERENCE RULE

$$\frac{(a, b) : +f\ \gamma, \alpha_1, \ldots, \alpha_{i-1}, (c, d) : -f\ \delta, \alpha_{i+1}, \ldots, \alpha_k}{(a, b) : \gamma, \alpha_1, \ldots, \alpha_{i-1}, (c, d) : \delta, \alpha_{i+1}, \ldots, \alpha_k}$$

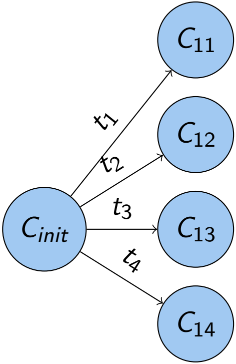
Big Picture


RCG
✓
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
✓
$O(n^{4m+4})$
this work
swap transition
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
Full search
Chart Based
Local models
Partial search
Global models
Transition Based

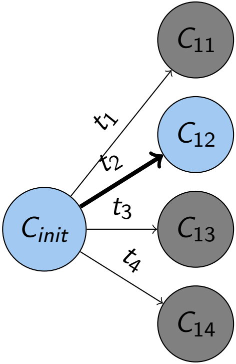
What is a Transition-Based Parser?



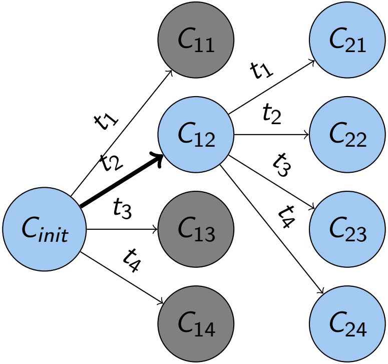
What is a Transition-Based Parser?



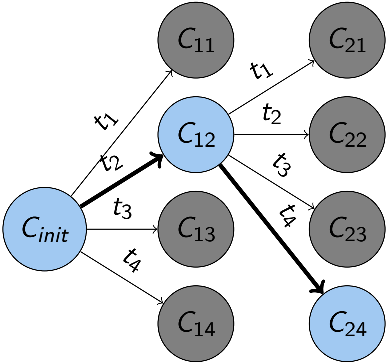
What is a Transition-Based Parser?



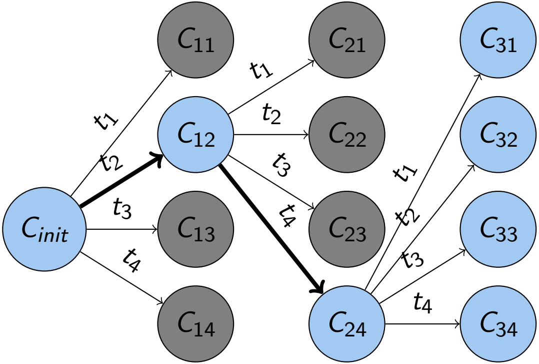
What is a Transition-Based Parser?



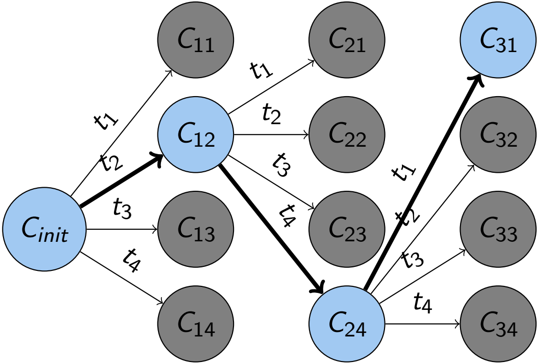
What is a Transition-Based Parser?



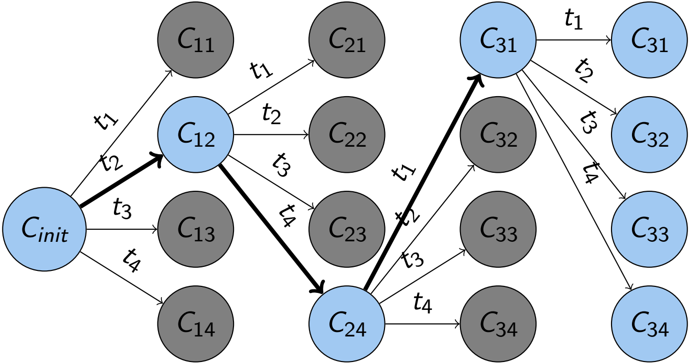
What is a Transition-Based Parser?



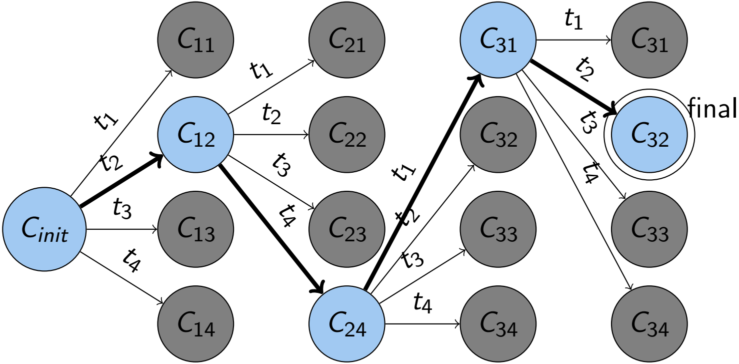
What is a Transition-Based Parser?



What is a Transition-Based Parser?



What is a Transition-Based Parser?



Transition-Based CFG Parsing

Components

- Configurations
  - buffer β
    words remaining to process
  - stack σ
    constituents built so far
- Transitions
  - Shift
  - Reduce

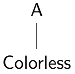
Deduction System

$$\begin{array}{rl}
\textit{axiom} & \langle [], [w_1, \ldots, w_n] \rangle \\
\textit{goal} & \langle [S], [] \rangle \\
\textit{shift}\{X\} & \dfrac{\langle \sigma, w|\beta \rangle}{\langle \sigma|X, \beta \rangle} \quad X \rightarrow w \in G \\
\textit{reduce}\{Z\} & \dfrac{\langle \sigma|X|Y, \beta \rangle}{\langle \sigma|Z, \beta \rangle} \quad Z \rightarrow XY \in G
\end{array}$$

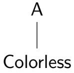
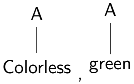
CFG Transition-Based Parsing Example

	σ	β	transition
1		Colorless , green , ideas , sleep ,furiously	*shift*$\{A\}$

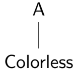
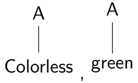
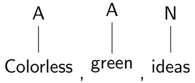
CFG Transition-Based Parsing Example

	σ	β	transition
1		Colorless , green , ideas , sleep ,furiously	*shift*$\{A\}$
2	 A Colorless	green , ideas , sleep ,furiously	*shift*$\{A\}$

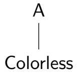
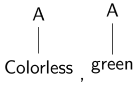
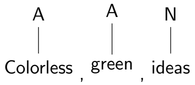
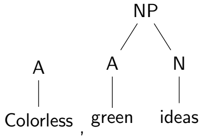
CFG Transition-Based Parsing Example

	σ	β	transition
1		Colorless , green , ideas , sleep ,furiously	*shift*$\{A\}$
2	 A Colorless	green , ideas , sleep ,furiously	*shift*$\{A\}$
3	 A Colorless , A green	ideas, sleep ,furiously	*shift*$\{N\}$

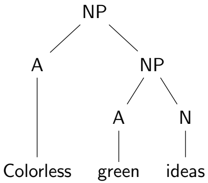
CFG Transition-Based Parsing Example

	σ	β	transition
1		Colorless , green , ideas , sleep ,furiously	*shift*$\{A\}$
2		green , ideas , sleep ,furiously	*shift*$\{A\}$
3		ideas, sleep ,furiously	*shift*$\{N\}$
4		sleep ,furiously	*reduce*$\{NP\}$

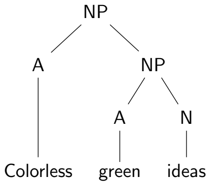
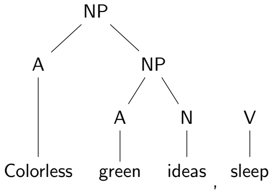
CFG Transition-Based Parsing Example

	σ	β	transition
1		Colorless , green , ideas , sleep ,furiously	*shift*$\{A\}$
2		green , ideas , sleep ,furiously	*shift*$\{A\}$
3		ideas, sleep ,furiously	*shift*$\{N\}$
4		sleep ,furiously	*reduce*$\{NP\}$
5		sleep ,furiously	*reduce*$\{NP\}$

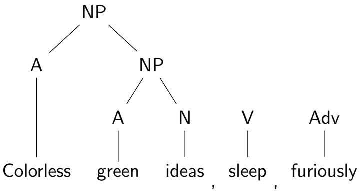
CFG Transition-Based Parsing Example

	σ	β	transition
6		sleep ,furiously	*shift*$\{V\}$

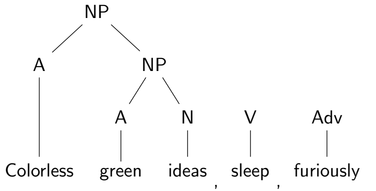
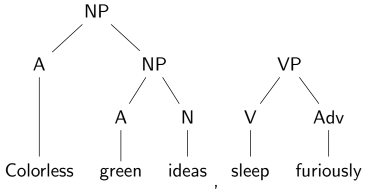
CFG Transition-Based Parsing Example

	σ	β	transition
6		sleep ,furiously	$\mathit{shift}\{V\}$
7	 NP A NP A N V Colorless green ideas , sleep	furiously	$\mathit{shift}\{Adv\}$

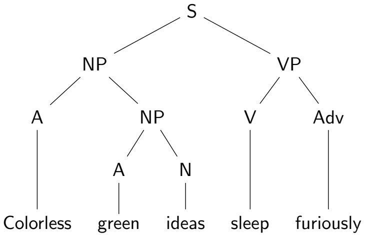
CFG Transition-Based Parsing Example

	σ	β	transition
8	 NP A NP A N V Adv Colorless green ideas , sleep , furiously		*reduce*$\{VP\}$

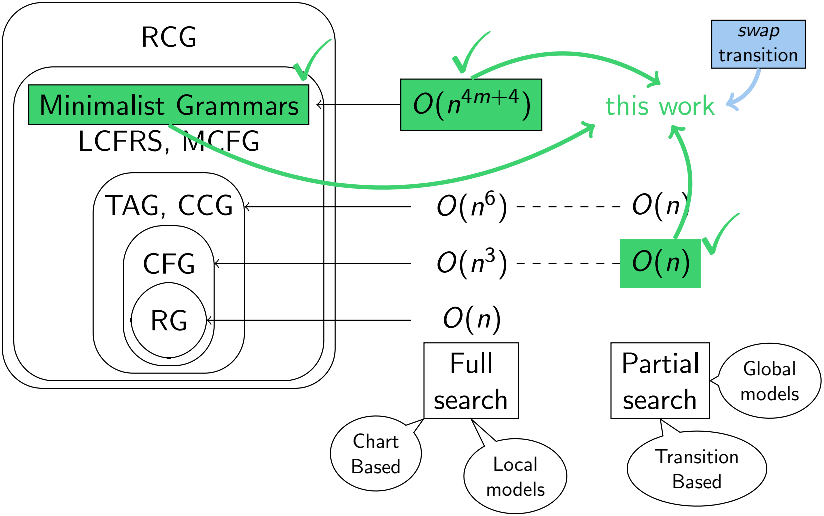
CFG Transition-Based Parsing Example

	σ	β	transition
8	 NP A NP A N V Adv Colorless green ideas , sleep , furiously		$\mathit{reduce}\{VP\}$
9	 NP A NP VP A N V Adv Colorless green ideas , sleep furiously		$\mathit{reduce}\{S\}$

CFG Transition-Based Parsing Example

	σ	β	transition
10			*goal*

Big Picture


RCG
Minimalist Grammars
LCFRS, MCFG
TAG, CCG
CFG
RG
$O(n^{4m+4})$
$O(n^6)$
$O(n)$
$O(n^3)$
$O(n)$
$O(n)$
this work
swap transition
Full search
Partial search
Chart Based
Local models
Global models
Transition Based

Minimalist Transition-Based Parser

Convert CFG Parser to MG Parser

- keep shift-reduce structure

Minimalist Transition-Based Parser

Convert CFG Parser to MG Parser

- keep shift-reduce structure
- *mini-items* instead of constituents on the stack

Minimalist Transition-Based Parser

Convert CFG Parser to MG Parser

- keep shift-reduce structure
- *mini-items* instead of constituents on the stack
- add a stack to the configuration

Minimalist Transition-Based Parser

Convert CFG Parser to MG Parser

- keep shift-reduce structure
- *mini-items* instead of constituents on the stack
- add a stack to the configuration
- add new transitions

Minimalist Transition-Based Parser

Convert CFG Parser to MG Parser

- keep shift-reduce structure
- *mini-items* instead of constituents on the stack
- add a stack to the configuration
- add new transitions

Configuration Structure

$$\langle \ \sigma_1, \quad \sigma_2, \quad \beta, \quad k \ \rangle$$

σ_1 Main stack – contains *mini-items* same as chart *items*

σ_2 Auxiliary stack

β Buffer

k Number of inserted empty strings

Transforming Transition-Based CFG to MG

CFG reuse

- axiom

Minimalist Grammar Transitions

$$\textit{axiom} \ \langle \ [\,], \quad [\,], \quad [0, \ldots, n-1], \quad 0 \ \rangle$$

Transforming Transition-Based CFG to MG

CFG reuse

- axiom
- goal

Minimalist Grammar Transitions

$$\begin{array}{rlllll} \textit{axiom} & \langle\ [\,], & [\,], & [0, \ldots, n-1], & 0 & \rangle \\ \textit{goal} & \langle\ [\{(0,n) \cdot c\}], & [\,], & [\,], & k & \rangle \end{array}$$

Transforming Transition-Based CFG to MG

CFG reuse

- axiom
- goal
- shift

Minimalist Grammar Transitions

$$\textit{axiom} \quad \langle \; [\,], \quad [\,], \quad [0, \ldots, n-1], \quad 0 \; \rangle$$

$$\textit{goal} \quad \langle \; [\{(0,n) \cdot c\}], \quad [\,], \quad [\,], \quad k \; \rangle$$

$$\textit{select}\{\gamma\} \quad \frac{\langle \; \sigma_1, \quad \sigma_2, \quad i|\beta, \quad k \; \rangle}{\langle \; \sigma_1|\{(i, i+1) :: \gamma\}, \quad \sigma_2, \quad \beta, \quad k \; \rangle} \quad w_i{::}\gamma \in \textit{Lex}$$

Transforming Transition-Based CFG to MG

CFG reuse

- axiom
- goal
- shift
- reduce

Minimalist Grammar Transitions

$$
\begin{array}{rll}
\textit{axiom} & \langle\ [\,],\quad [\,],\quad [0,\ldots,n-1],\quad 0\ \rangle & \\
\textit{goal} & \langle\ [\{(0,n)\cdot c\}],\quad [\,],\quad [\,],\quad k\ \rangle & \\
\textit{select}\{\gamma\} & \dfrac{\langle\ \sigma_1,\quad \sigma_2,\quad i|\beta,\quad k\ \rangle}{\langle\ \sigma_1|\{(i,i+1)::\gamma\},\quad \sigma_2,\quad \beta,\quad k\ \rangle} & w_i{::}\gamma \in \textit{Lex} \\
\textit{tmerge} & \dfrac{\langle\ \sigma_1|x|y,\quad \sigma_2,\quad \beta,\quad k\ \rangle}{\langle\ \sigma_1|\textit{merge}(x,y),\quad \sigma_2,\quad \beta,\quad k\ \rangle} & (x,y) \in \textit{Dom}(\textit{merge}) \\
\textit{tmove} & \dfrac{\langle\ \sigma_1|x,\quad \sigma_2,\quad \beta,\quad k\ \rangle}{\langle\ \sigma_1|\textit{move}(x),\quad \sigma_2,\quad \beta,\quad k\ \rangle} & x \in \textit{Dom}(\textit{move})
\end{array}
$$

Transforming Transition-Based CFG to MG

The Extensions Needed

- Empty strings
- Discontinuous structures

Transforming Transition-Based CFG to MG

The Extensions Needed

- Empty strings
- Discontinuous structures

Empty Strings Solution

$$\mathit{selectEpsilon}\{\gamma\}\ \frac{\langle\ \sigma_1,\quad \sigma_2,\quad \beta,\quad k\ \rangle}{\langle\ \sigma_1|\{(*,*) :: \gamma\},\quad \sigma_2,\quad \beta,\quad k+1\ \rangle}\ k < e \wedge \varepsilon{::}\gamma \in \mathit{Lex}$$

e is any linear function of sentence length *n*

Transforming Transition-Based CFG to MG

Discontinuity Solution: SWAP transition

- Reorders elements on the stack (*stupid sort*)
- Derives any permutation of words/phrases
- Minimum 0
- Maximally $O(n^2)$

Transitions

$$\textit{swap}\ \frac{\langle\ \sigma_1|x|y,\quad \sigma_2,\quad \beta,\quad k\ \rangle}{\langle\ \sigma_1|y,\quad x|\sigma_2,\quad \beta,\quad k\ \rangle}\ \textit{spanStart}(x) < \textit{spanStart}(y)$$

$$\textit{takeBack}\ \frac{\langle\ \sigma_1,\quad x|\sigma_2,\quad \beta,\quad k\ \rangle}{\langle\ \sigma_1|x,\quad \sigma_2,\quad \beta,\quad k\ \rangle}$$

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
1			0 , 1 , 2 , 3 , 4 Phong likes what Roki draws	0	*select*{***d***}

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
1			0 , 1 , 2 , 3 , 4 Phong likes what Roki draws	0	*select*$\{$***d***$\}$
2	$\{(0,1) :: d\}$ Phong		1 , 2 , 3 , 4 likes what Roki draws	0	*select*$\{$***=c =d v***$\}$

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
1			0 , 1 , 2 , 3 , 4 Phong likes what Roki draws	0	*select*{***d***}
2	$\{(0,1) :: d\}$ Phong		1 , 2 , 3 , 4 likes what Roki draws	0	*select*{***=c =d v***}
3	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$ Phong likes		2 , 3 , 4 what Roki draws	0	*select*{***d −wh***}

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
1			$\underset{\text{Phong}}{0}, \underset{\text{likes}}{1}, \underset{\text{what}}{2}, \underset{\text{Roki}}{3}, \underset{\text{draws}}{4}$	0	*select*{***d***}
2	$\left\{\underset{\text{Phong}}{(0,1) :: d}\right\}$		$\underset{\text{likes}}{1}, \underset{\text{what}}{2}, \underset{\text{Roki}}{3}, \underset{\text{draws}}{4}$	0	*select*{***=c =d v***}
3	$\left\{\underset{\text{Phong}}{(0,1) :: d}\right\}, \left\{\underset{\text{likes}}{(1,2) :: {=c}\ \ {=d}\ v}\right\}$		$\underset{\text{what}}{2}, \underset{\text{Roki}}{3}, \underset{\text{draws}}{4}$	0	*select*{***d −wh***}
4	$\left\{\underset{\text{Phong}}{(0,1) :: d}\right\}, \left\{\underset{\text{likes}}{(1,2) :: {=c}\ \ {=d}\ v}\right\}, \left\{\underset{\text{what}}{(2,3) :: d\ \ {-wh}}\right\}$		$\underset{\text{Roki}}{3}, \underset{\text{draws}}{4}$	0	*select*{***d***}

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
5	$\underset{\text{Phong}}{\{(0,1) :: d\}}, \underset{\text{likes}}{\{(1,2) :: =c \ \ =d \ \ v\}}, \underset{\text{what}}{\{(2,3) :: d \ \ -wh\}}, \underset{\text{Roki}}{\{(3,4) :: d\}}$		$\underset{\text{draws}}{4}$	0	*swap*

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
5	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(2,3) :: d\ \ -wh\}$, $\{(3,4) :: d\}$ Phong likes what Roki		4 draws	0	*swap*
6	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,4) :: d\}$ Phong likes Roki	$\{(2,3) :: d\ \ -wh\}$ what	4 draws	0	*takeBack*

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
5	$\{(0,1) :: d\}, \{(1,2) :: {=c}\ \ {=d}\ \ v\}, \{(2,3) :: d\ \ {-wh}\}, \{(3,4) :: d\}$ Phong likes what Roki		4 draws	0	*swap*
6	$\{(0,1) :: d\}, \{(1,2) :: {=c}\ \ {=d}\ \ v\}, \{(3,4) :: d\}$ Phong likes Roki	$\{(2,3) :: d\ \ {-wh}\}$ what	4 draws	0	*takeBack*
7	$\{(0,1) :: d\}, \{(1,2) :: {=c}\ \ {=d}\ \ v\}, \{(3,4) :: d\},\{(2,3) :: d\ \ {-wh}\}$ Phong likes Roki what		4 draws	0	$\mathit{select}\{\boldsymbol{{=d}\ \ {=d}\ \ v}\}$

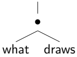
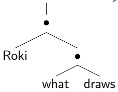
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
8	$\underset{\text{Phong}}{\{(0,1) :: d\}}, \underset{\text{likes}}{\{(1,2) :: {=}c\ \ {=}d\ v\}}, \underset{\text{Roki}}{\{(3,4) :: d\}}, \underset{\text{what}}{\{(2,3) :: \mathbf{d}\ \ {-}wh\}}, \underset{\text{draws}}{\{(4,5) :: \mathbf{{=}d}\ \ {=}d\ v\}}$			0	*tmerge*

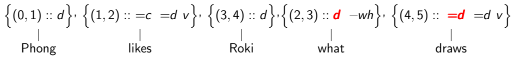
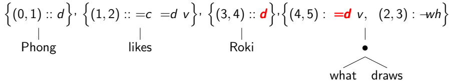
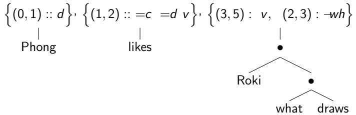
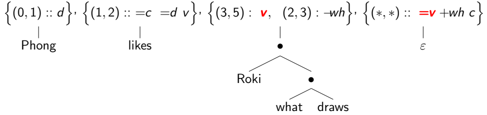
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
8	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,4) :: d\}$, $\{(2,3) :: d\ \ -wh\}$, $\{(4,5) :: =d\ \ =d\ \ v\}$ Phong likes Roki what draws			0	*tmerge*
9	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,4) :: d\}$, $\{(4,5) :\ =d\ \ v,\ \ (2,3) : -wh\}$ Phong likes Roki • what draws			0	*tmerge*

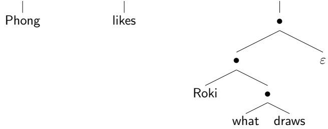
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
8	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,4) :: d\}$,$\{(2,3) :: d\ \ -wh\}$, $\{(4,5) :: =d\ \ =d\ \ v\}$ Phong likes Roki what draws			0	*tmerge*
9	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,4) :: d\}$,$\{(4,5) : =d\ \ v,\ \ (2,3) : -wh\}$ Phong likes Roki 			0	*tmerge*
10	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,5) : v,\ \ (2,3) : -wh\}$ Phong likes 			0	*selectEpsilon* $\{=v\ \ +wh\ \ c\}$

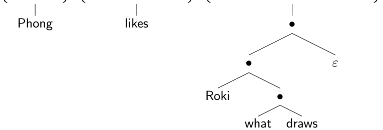
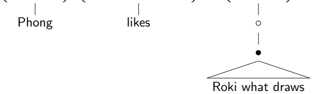
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
8	 $\{(0,1) :: d\}, \{(1,2) :: {=c}\ {=d}\ v\}, \{(3,4) :: d\}, \{(2,3) :: \mathbf{d}\ {-wh}\}, \{(4,5) :: \mathbf{=d}\ {=d}\ v\}$ Phong likes Roki what draws			0	*tmerge*
9	 $\{(0,1) :: d\}, \{(1,2) :: {=c}\ {=d}\ v\}, \{(3,4) :: \mathbf{d}\}, \{(4,5) : \mathbf{=d}\ v,\ (2,3) : {-wh}\}$ Phong likes Roki what draws			0	*tmerge*
10	 $\{(0,1) :: d\}, \{(1,2) :: {=c}\ {=d}\ v\}, \{(3,5) : v,\ (2,3) : {-wh}\}$ Phong likes Roki what draws			0	*selectEpsilon* $\{\mathbf{=v}\ \ \mathbf{+wh}\ \mathbf{c}\}$
11	 $\{(0,1) :: d\}, \{(1,2) :: {=c}\ {=d}\ v\}, \{(3,5) : \mathbf{v},\ (2,3) : {-wh}\}, \{(*,*) :: \mathbf{=v}\ {+wh}\ c\}$ Phong likes Roki what draws ε			1	*tmerge*

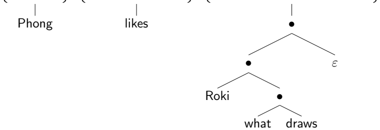
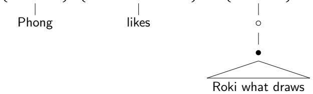
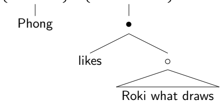
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
12	$\{(0,1) :: d\}$, $\{(1,2) :: =c\ \ =d\ \ v\}$, $\{(3,5) : +wh\ \ c,\quad (2,3) : -wh\}$  			1	*tmove*

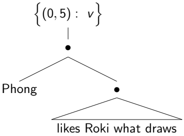
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
12	$\{(0,1) :: d\}, \{(1,2) :: =c\ \ =d\ v\}, \{(3,5) : +wh\ c,\ \ (2,3) : -wh\}$  			1	*tmove*
13	$\{(0,1) :: d\}, \{(1,2) :: =c\ \ =d\ v\}, \{(2,5) : c\}$  			1	*tmerge*

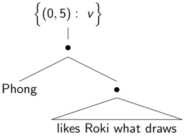
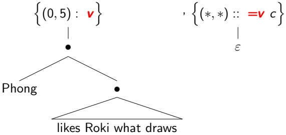
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
12	$\{(0,1) :: d\}, \{(1,2) :: =c\ \ =d\ \ v\}, \{(3,5) : +wh\ \ c,\ \ (2,3) : -wh\}$  			1	*tmove*
13	$\{(0,1) :: d\}, \{(1,2) :: =c\ \ =d\ \ v\}, \{(2,5) : c\}$  			1	*tmerge*
14	$\{(0,1) :: d\}, \{(1,5) : =d\ \ v\}$  			1	*tmerge*

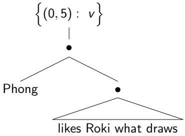
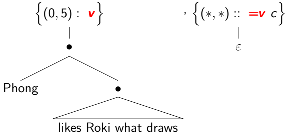
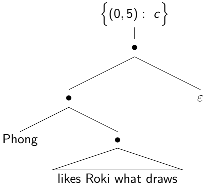
MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
15				1	*selectEpsilon*{***=v c***}

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
15				1	*selectEpsilon*{**=v c**}
16	 {(0, 5) : **v**} Phong likes Roki what draws , {(∗, ∗) :: =v c} ε			2	*tmerge*

MG Transition-Based Parsing Example

	σ_1	σ_2	β	k	transition
15				1	*selectEpsilon*{**=v c**}
16	 $\{(0,5):\ \mathbf{v}\}$ Phong likes Roki what draws , $\{(*,*) :: \mathbf{=v}\ c\}$ ε			2	*tmerge*
17	 $\{(0,5):\ c\}$ ε Phong likes Roki what draws			2	goal

Computational Complexity

operation	best case	worst case
$select\{\cdot\}$	$O(n)$	$O(n)$

Computational Complexity

operation	best case	worst case
$select\{\cdot\}$	$O(n)$	$O(n)$
$selectEpsilon\{\cdot\}$	0	$O(n)$

Computational Complexity

operation	best case	worst case
$\mathit{select}\{\cdot\}$	$O(n)$	$O(n)$
$\mathit{selectEpsilon}\{\cdot\}$	0	$O(n)$
tmerge	$O(n)$	$O(n)$

Computational Complexity

operation	best case	worst case
$select\{\cdot\}$	$O(n)$	$O(n)$
$selectEpsilon\{\cdot\}$	0	$O(n)$
$tmerge$	$O(n)$	$O(n)$
$tmove$	0	$O(n)$

Computational Complexity

operation	best case	worst case
$select\{\cdot\}$	$O(n)$	$O(n)$
$selectEpsilon\{\cdot\}$	0	$O(n)$
$tmerge$	$O(n)$	$O(n)$
$tmove$	0	$O(n)$
$swap$	0	$O(n^2)$

Computational Complexity

operation	best case	worst case
select$\{\cdot\}$	$O(n)$	$O(n)$
selectEpsilon$\{\cdot\}$	0	$O(n)$
tmerge	$O(n)$	$O(n)$
tmove	0	$O(n)$
swap	0	$O(n^2)$
takeBack	0	$O(n^2)$

Computational Complexity

operation	best case	worst case
select$\{\cdot\}$	$O(n)$	$O(n)$
selectEpsilon$\{\cdot\}$	0	$O(n)$
tmerge	$O(n)$	$O(n)$
tmove	0	$O(n)$
swap	0	$O(n^2)$
takeBack	0	$O(n^2)$
asympt. max	$O(n)$	$O(n^2)$

Conclusion

- Efficient parser: worst case $O(n^2)$ and best case $O(n)$
- Transition Parser+Learning Model+Treebank = Minimalism in NLP applications
- Technique applicable to other formalisms

Bibliography I