n-Best Parsing Revisited

Matthias Büchse Daniel Geisler
Torsten Stüber Heiko Vogler

Technische Universität Dresden

Workshop ATANLP
ACL 2010
2010-07-16

Contribution

this
work

Contribution

this work	=	Huang and Chiang, 2005

Contribution

this work	$=$	Huang and Chiang, 2005
		$+$ larger class of hypergraphs

Contribution

this work	$=$	Huang and Chiang, 2005
	$+$	larger class of hypergraphs
	$+$	confirm conjecture for cyclic case

Contribution

this work	=	Huang and Chiang, 2005

+ larger class of hypergraphs
+ confirm conjecture for cyclic case
+ more flexible weight structures

Contribution

this work	=	Huang and Chiang, 2005

+ larger class of hypergraphs
+ confirm conjecture for cyclic case
+ more flexible weight structures
+ functional specification (Haskell)

Contribution

this work	=	Huang and Chiang, 2005

\+ larger class of hypergraphs

\+ confirm conjecture for cyclic case

\+ more flexible weight structures

\+ functional specification (Haskell)

Two Scenarios

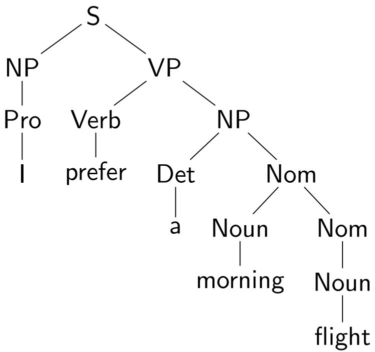
- Parsing
- Decoding

Parsing

$e =$ "I prefer a morning flight"

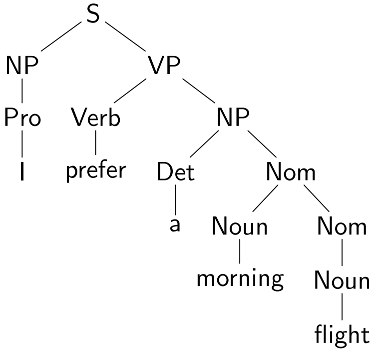
Parsing

$e =$ "I prefer a morning flight" $\Longrightarrow$



Parsing

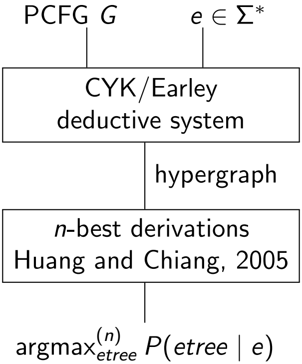
$e =$ "I prefer a morning flight" $\Longrightarrow$



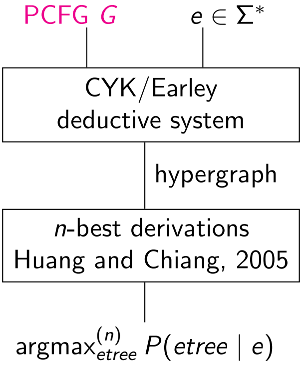
Statistical NLP:

$$\boxed{\text{argmax}_{etree}\, P(etree \mid e)}$$

PCFG Parsing Application



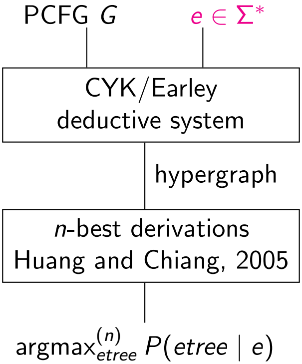
PCFG Parsing Application



S	$\rightarrow$	NP VP
NP	$\rightarrow$	Pronoun | Proper-Noun | Det Nominal
Nominal	$\rightarrow$	Noun Nominal | Noun
VP	$\rightarrow$	Verb | Verb NP | ...
PP	$\rightarrow$	Preposition NP
Noun	$\rightarrow$	flight | breeze | trip | morning | ...
Verb	$\rightarrow$	is | prefer | like | need | want | fly
Pronoun	$\rightarrow$	me | I | you | it | ...
Proper-Noun	$\rightarrow$	Alaska | Baltimore | ...
Det	$\rightarrow$	the | a | an | this | these | that | ...
Preposition	$\rightarrow$	from | to | on | near | ...

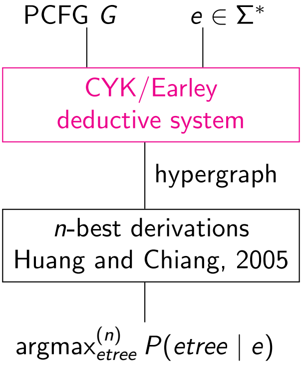
(after Jurafsky and Martin (2000))

PCFG Parsing Application



I prefer a morning flight

PCFG Parsing Application



Item form:
$[i, A \to \alpha \bullet \beta, j]$

Goal:
$[1, S' \to S \bullet, n+1]$

Rules:

$$\frac{}{[1, S' \to \bullet S, 1]}$$ Initialization

$$\frac{[i, A \to \alpha \bullet w_j \beta, j]}{[i, A \to \alpha w_j \bullet \beta, j+1]}$$ Scanning

$$\frac{R(B \to \gamma)}{[j, B \to \bullet \gamma, j]} \quad [i, A \to \alpha \bullet B\beta, j]$$ Prediction

$$\frac{[i, A \to \alpha \bullet B\beta, k] \quad [k, B \to \gamma \bullet, j]}{[i, A \to \alpha B \bullet \beta, j]}$$ Completion

(Goodman, 1999)

PCFG Parsing Application

PCFG G $e \in \Sigma^*$

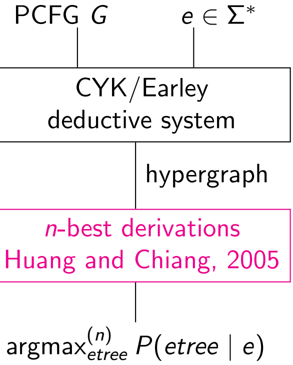
CYK/Earley
deductive system

hypergraph   "=" instantiation of the deductive system

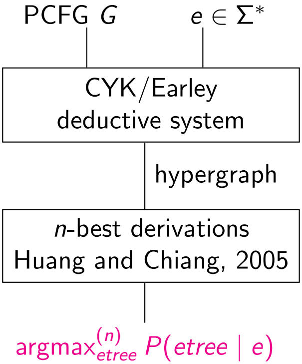
n-best derivations
Huang and Chiang, 2005

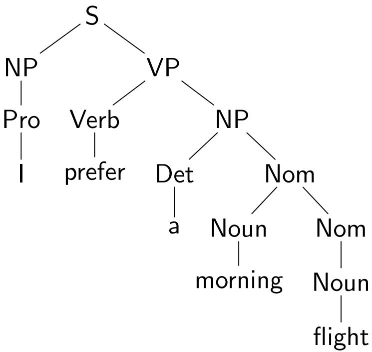
$\text{argmax}^{(n)}_{etree}\, P(etree \mid e)$

PCFG Parsing Application



PCFG Parsing Application





–LM Decoding

$f =$ "Jean aime Marie"

–LM Decoding

f = "Jean aime Marie" $\Longrightarrow$ John loves Mary

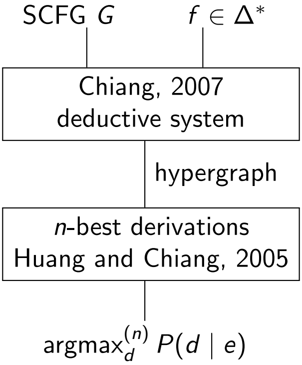
–LM Decoding

$f =$ "Jean aime Marie" $\Longrightarrow$ John loves Mary

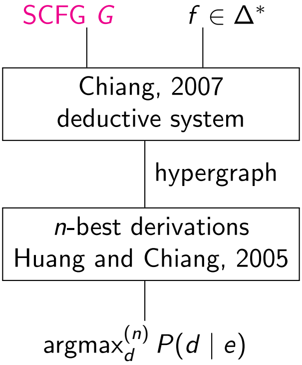
Statistical NLP:

$$\boxed{\text{argmax}_e \, P(e \mid f)}$$

SCFG Decoding Application

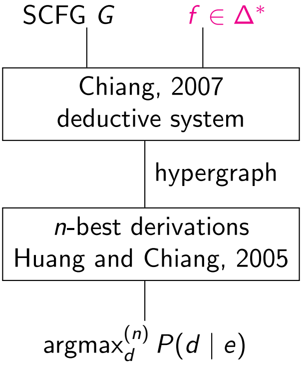


SCFG Decoding Application



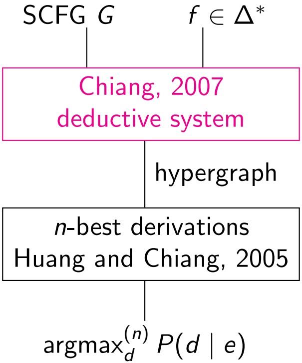
$$
\begin{aligned}
\text{S} &\rightarrow \langle \text{S}_{\boxed{1}}\ \text{X}_{\boxed{2}},\ \text{S}_{\boxed{1}}\ \text{X}_{\boxed{2}} \rangle \\
\text{S} &\rightarrow \langle \text{X}_{\boxed{1}},\ \text{X}_{\boxed{1}} \rangle \\
\text{X} &\rightarrow \langle \text{X}_{\boxed{1}}\ \text{X}_{\boxed{2}},\ \text{X}_{\boxed{2}}\ \text{X}_{\boxed{1}} \rangle \\
\text{X} &\rightarrow \langle \text{John, Jean} \rangle \\
\text{X} &\rightarrow \langle \text{loves, aime} \rangle \\
\text{X} &\rightarrow \langle \text{Mary, Marie} \rangle
\end{aligned}
$$

SCFG Decoding Application



Jean aime Marie

SCFG Decoding Application



$$\frac{}{X \to \gamma : w} \quad (X \xrightarrow{w} \langle \gamma, \alpha \rangle) \in G$$

$$\frac{X \to f_{i+1}^{j} : w}{[X, i, j] : w}$$

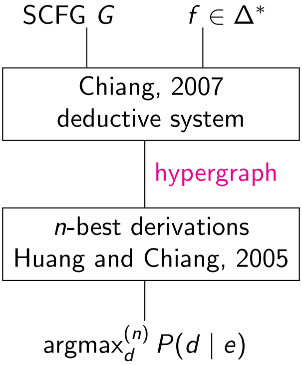
$$\frac{Z \to f_{i+1}^{i_1} X f_{j_1+1}^{j} : w \quad [X, i_1, j_1] : w_1}{[Z, i, j] : w w_1}$$

$$\frac{Z \to f_{i+1}^{i_1} X f_{j_1+1}^{i_2} Y f_{j_2+1}^{j} : w \quad [X, i_1, j_1] : w_1 \quad [Y, i_2, j_2] : w_2}{[Z, i, j] : w w_1 w_2}$$

Goal item: $[S, 0, n]$

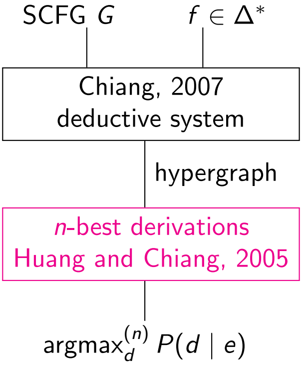
(Chiang, 2007)

SCFG Decoding Application

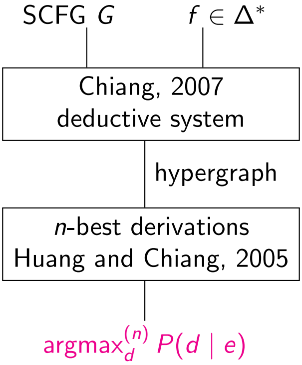


"=" instantiation of the deductive system

SCFG Decoding Application

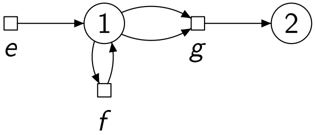


SCFG Decoding Application



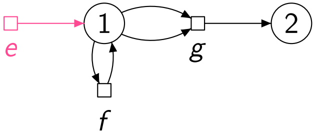
John loves Mary

Interlude: Hypergraphs and Derivations



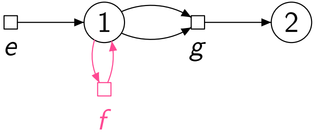
derivations for node 1	derivations for node 2

Interlude: Hypergraphs and Derivations



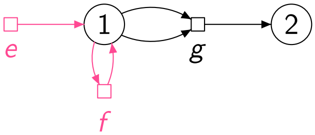
derivations for node 1	derivations for node 2
e	

Interlude: Hypergraphs and Derivations



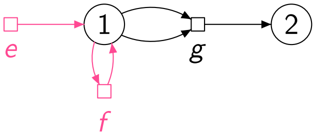
derivations for node 1	derivations for node 2
e	
$f(\ldots)$	

Interlude: Hypergraphs and Derivations



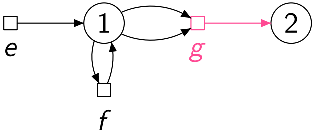
derivations for node 1	derivations for node 2
e	
$f(e)$	

Interlude: Hypergraphs and Derivations



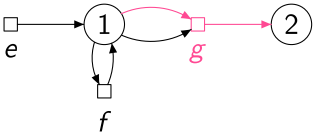
derivations for node 1	derivations for node 2
e	
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



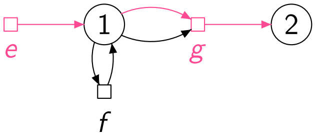
derivations for node 1	derivations for node 2
e	$g(\ldots,\ldots)$
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



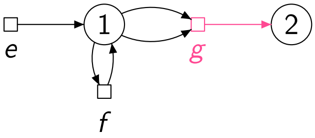
derivations for node 1	derivations for node 2
e	$g(\ldots,\ldots)$
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



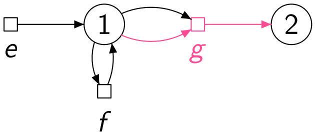
derivations for node 1	derivations for node 2
e	$g(e, \dots)$
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



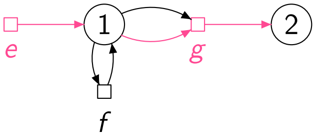
derivations for node 1	derivations for node 2
e	$g(e, \dots)$
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



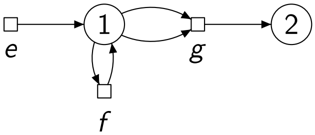
derivations for node 1	derivations for node 2
e	$g(e, \dots)$
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



derivations for node 1	derivations for node 2
e	$g(e, e)$
$f(e)$	
$f(f(e))$	
$f(f(f(e)))$	
$\vdots$	

Interlude: Hypergraphs and Derivations



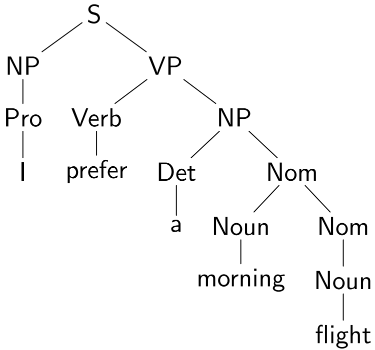
derivations for node 1	derivations for node 2
e	$g(e, e)$
$f(e)$	$g(e, f(e))$
$f(f(e))$	$g(f(e), f(e))$
$f(f(f(e)))$	$g(f(e), e)$
$\vdots$	$\vdots$

Contribution

this work	=	Huang and Chiang, 2005
	+	larger class of hypergraphs

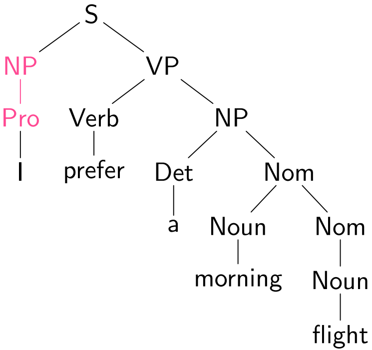
PCFG Parsing Revisited

Subject vs Object NPs



PCFG Parsing Revisited

Subject vs Object NPs

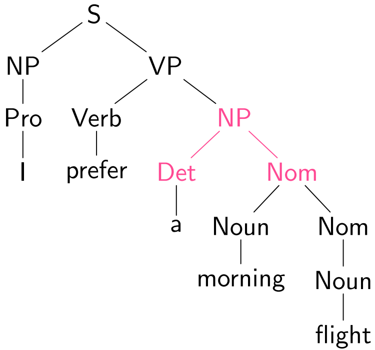


	subject
pronoun	91 %
non-pronoun	9 %

(Jurafsky and Martin, 2009)

PCFG Parsing Revisited

Subject vs Object NPs

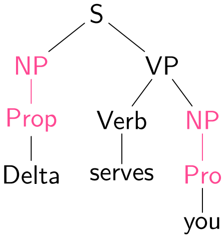


	subject	object
pronoun	91 %	34 %
non-pronoun	9 %	66 %

(Jurafsky and Martin, 2009)

PCFG Parsing Revisited

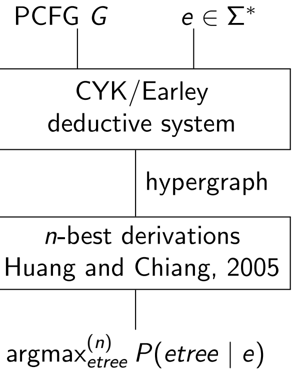
Subject vs Object NPs



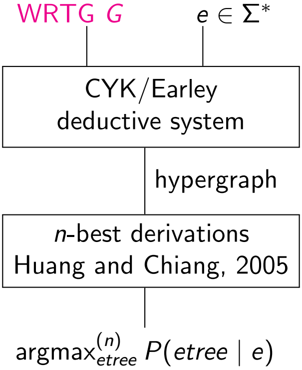
	subject	object
pronoun	91 %	34 %
non-pronoun	9 %	66 %

(Jurafsky and Martin, 2009)

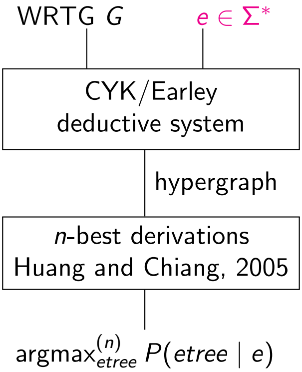
From PCFG to WRTG



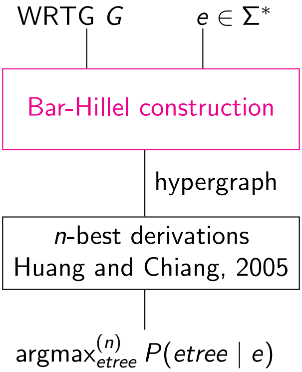
From PCFG to WRTG



From PCFG to WRTG

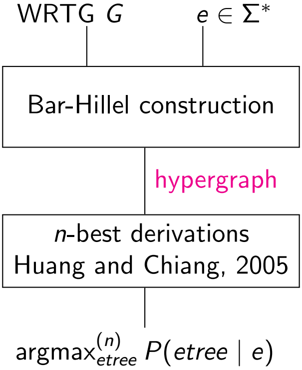


From PCFG to WRTG



(Maletti and Satta, 2009)

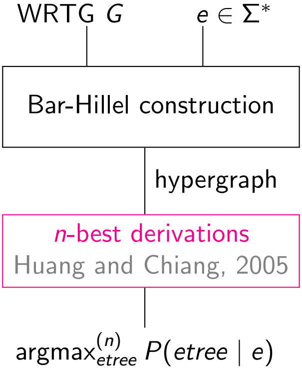
From PCFG to WRTG



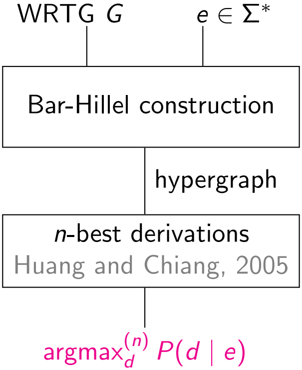
“=” G, but restricted to parse trees of e

multi-edges!

From PCFG to WRTG



From PCFG to WRTG

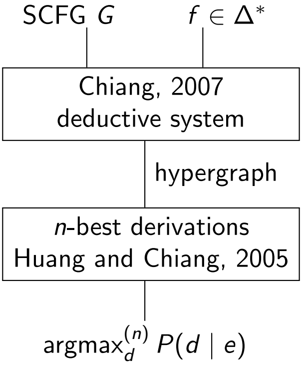


Contribution

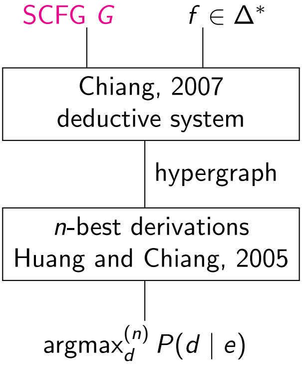
this work	=	Huang and Chiang, 2005

+ larger class of hypergraphs
+ confirm conjecture for cyclic case

SCFG Decoding Revisited

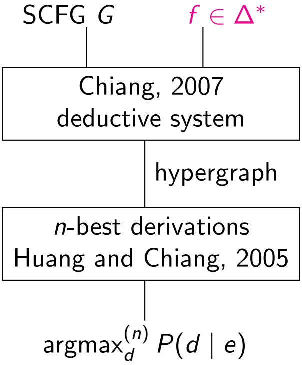


SCFG Decoding Revisited



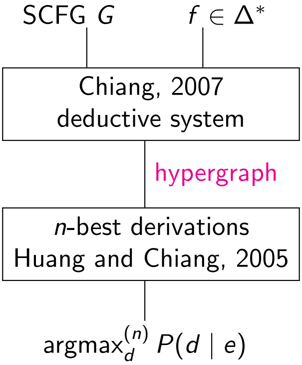
$$
\begin{aligned}
S &\rightarrow \langle S_{\boxed{1}}\ X_{\boxed{2}},\ S_{\boxed{1}}\ X_{\boxed{2}} \rangle \\
S &\rightarrow \langle X_{\boxed{1}},\ X_{\boxed{1}} \rangle \\
X &\rightarrow \langle \text{super}\ X_{\boxed{1}},\ X_{\boxed{1}} \rangle \\
X &\rightarrow \langle X_{\boxed{1}}\ X_{\boxed{2}},\ X_{\boxed{2}}\ X_{\boxed{1}} \rangle \\
X &\rightarrow \langle \text{John},\ \text{Jean} \rangle \\
X &\rightarrow \langle \text{loves},\ \text{aime} \rangle \\
X &\rightarrow \langle \text{Mary},\ \text{Marie} \rangle
\end{aligned}
$$

SCFG Decoding Revisited



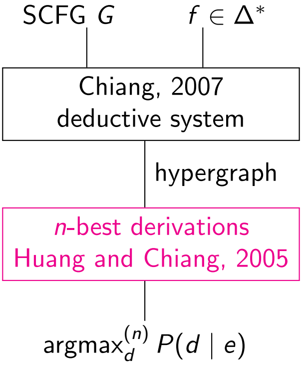
Jean aime Marie

SCFG Decoding Revisited



cyclic!

SCFG Decoding Revisited



conjecture: works in the cyclic case

Contribution

this work	=	Huang and Chiang, 2005

+ larger class of hypergraphs
+ confirm conjecture for cyclic case
+ more flexible weight structures

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}}$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}} = g^{\mathrm{I}}(e^{\mathrm{I}}(), f^{\mathrm{I}}(e^{\mathrm{I}}()))$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}} = g^{\mathrm{I}}(e^{\mathrm{I}}(), f^{\mathrm{I}}(e^{\mathrm{I}}())) = 1.3$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}} = g^{\mathrm{I}}(e^{\mathrm{I}}(), f^{\mathrm{I}}(e^{\mathrm{I}}())) = 1.3$$

3. define binary relation $\precsim$, better-than:

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}} = g^{\mathrm{I}}(e^{\mathrm{I}}(), f^{\mathrm{I}}(e^{\mathrm{I}}())) = 1.3$$

3. define binary relation $\precsim$, better-than:

$$\text{e. g.} \qquad \xi_1 \precsim \xi_2 \quad \text{iff} \quad \xi_1^{\mathrm{I}} \leq \xi_2^{\mathrm{I}}$$

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}} = g^{\mathrm{I}}(e^{\mathrm{I}}(), f^{\mathrm{I}}(e^{\mathrm{I}}())) = 1.3$$

3. define binary relation $\precsim$, better-than:

$$\text{e. g.} \qquad \xi_1 \precsim \xi_2 \quad \text{iff} \quad \xi_1^{\mathrm{I}} \leq \xi_2^{\mathrm{I}}$$

Integrate a language model!

Ranking Derivations

1. define interpretation mapping $(.)^{\mathrm{I}}$:

$$
\begin{aligned}
e^{\mathrm{I}}() &= .3 \\
f^{\mathrm{I}}(x) &= x + .1 \\
g^{\mathrm{I}}(x_1, x_2) &= x_1 + x_2 + .6
\end{aligned}
$$

2. extend to derivations:

$$[g(e, f(e))]^{\mathrm{I}} = g^{\mathrm{I}}(e^{\mathrm{I}}(), f^{\mathrm{I}}(e^{\mathrm{I}}())) = 1.3$$

3. define binary relation $\precsim$, better-than:

$$\text{e.g.} \qquad \xi_1 \precsim \xi_2 \quad \text{iff} \quad \xi_1^{\mathrm{I}} \leq \xi_2^{\mathrm{I}}$$

$\Rightarrow$ "linear pre-order"

Linear pre-order

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

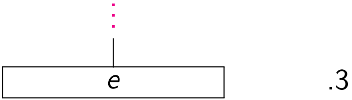
Linear pre-order

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

e .3

Linear pre-order

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

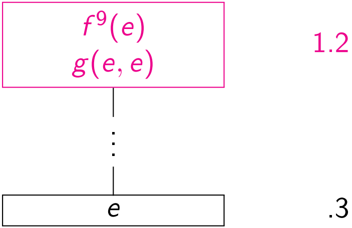


Linear pre-order

$$e^{\mathrm{I}}() = .3$$

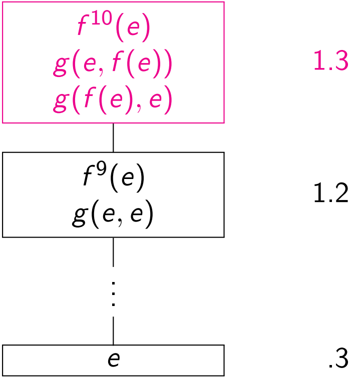
$$f^{\mathrm{I}}(x) = x + .1$$

$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$



Linear pre-order

$$e^{\mathrm{I}}() = .3$$
$$f^{\mathrm{I}}(x) = x + .1$$
$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$



Linear pre-order

$$e^{\mathrm{I}}() = .3$$

$$f^{\mathrm{I}}(x) = x + .1$$

$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$

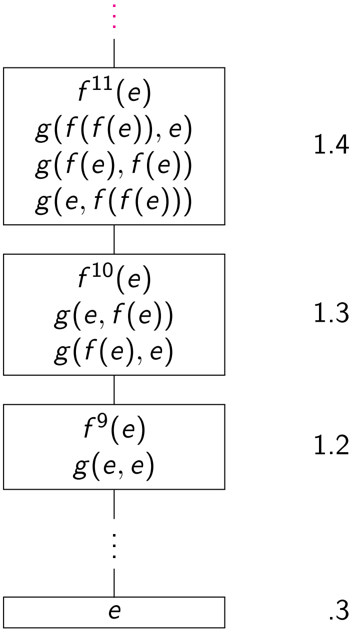
$f^{11}(e)$ $g(f(f(e)), e)$ $g(f(e), f(e))$ $g(e, f(f(e)))$	1.4
$f^{10}(e)$ $g(e, f(e))$ $g(f(e), e)$	1.3
$f^{9}(e)$ $g(e, e)$	1.2
$\vdots$	
e	.3

Linear pre-order

$$e^{\mathrm{I}}() = .3$$

$$f^{\mathrm{I}}(x) = x + .1$$

$$g^{\mathrm{I}}(x_1, x_2) = x_1 + x_2 + .6$$



Additional Requirements on the Pre-order

- Compatibility:

$$\forall i\colon \xi_i \precsim \xi_i' \quad \Longrightarrow \quad \sigma(\xi_1, \ldots, \xi_k) \precsim \sigma(\xi_1', \ldots, \xi_k')$$

Additional Requirements on the Pre-order

- Compatibility:

$$\forall i\colon \xi_i \precsim \xi_i' \quad \Longrightarrow \quad \sigma(\xi_1, \dots, \xi_k) \precsim \sigma(\xi_1', \dots, \xi_k')$$

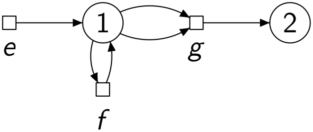
- Subtree property:

$$\forall i\colon \xi_i \precsim \sigma(\xi_1, \dots, \xi_k)$$

The n-Best Derivations Problem

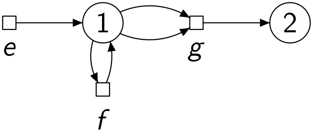
Given

The n-Best Derivations Problem

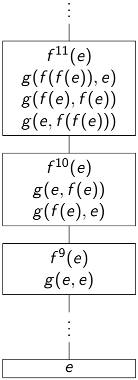


Given a hypergraph H,

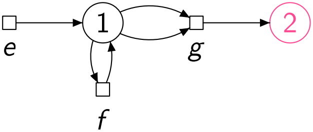
The n-Best Derivations Problem



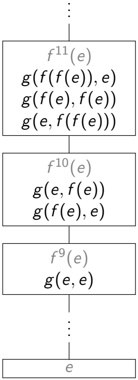
Given a hypergraph H,
a linear pre-order $\precsim$
(fulfilling our requirements),



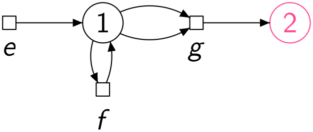
The n-Best Derivations Problem



Given a hypergraph H,
a linear pre-order $\precsim$
(fulfilling our requirements), and
a target vertex v,

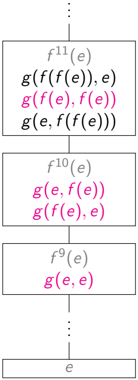


The n-Best Derivations Problem



Given a hypergraph H,
a linear pre-order $\precsim$
(fulfilling our requirements), and
a target vertex v,

compute n best derivations for v (in order).

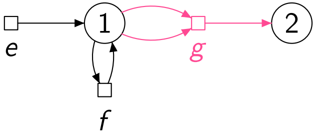


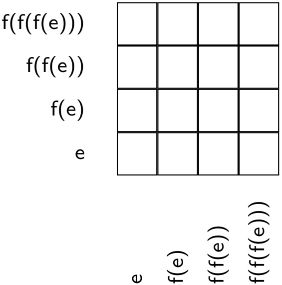
Contribution

this work	=	Huang and Chiang, 2005

+ larger class of hypergraphs
+ confirm conjecture for cyclic case
+ more flexible weight structures
+ functional specification (Haskell)

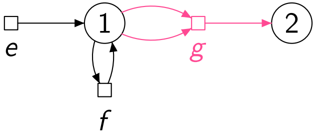
The Top Concatenation Operation

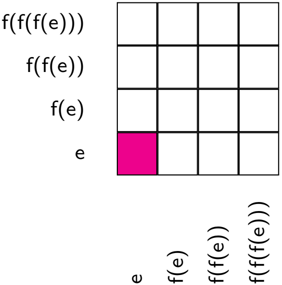




$$\textcolor{magenta}{g^{\text{tc}}(\text{l}_1,\text{l}_2)} = g(\text{head } \text{l}_1, \text{head } \text{l}_2) : \text{merge } \{g^{\text{tc}}(\text{tail } \text{l}_1, [\text{head } \text{l}_2]),\ g^{\text{tc}}(\text{l}_1, \text{tail } \text{l}_2)\}$$

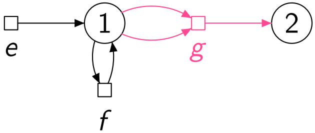
The Top Concatenation Operation

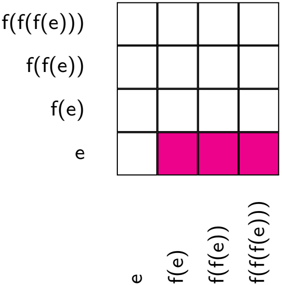




$$g^{\mathrm{tc}}(\mathsf{l}_1,\mathsf{l}_2) = g(\text{head } \mathsf{l}_1, \text{head } \mathsf{l}_2):$$
$$\text{merge } \{g^{\mathrm{tc}}(\text{tail } \mathsf{l}_1,[\text{head } \mathsf{l}_2]),\ g^{\mathrm{tc}}(\mathsf{l}_1,\text{tail } \mathsf{l}_2)\}$$

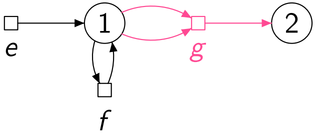
The Top Concatenation Operation

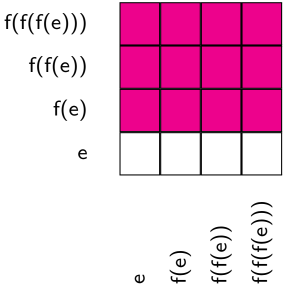




$$g^{tc}(l_1,l_2) = g(\text{head } l_1, \text{head } l_2): \text{merge } \{g^{tc}(\text{tail } l_1,[\text{head } l_2]),\ g^{tc}(l_1,\text{tail } l_2)\}$$

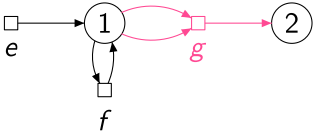
The Top Concatenation Operation

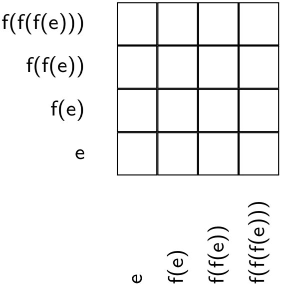




$$g^{tc}(l_1,l_2) = g(\text{head } l_1, \text{head } l_2): \text{merge } \{g^{tc}(\text{tail } l_1, [\text{head } l_2]),\ g^{tc}(l_1, \text{tail } l_2)\}$$

The Top Concatenation Operation



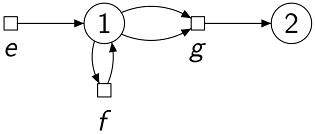


$$g^{tc}(l_1,l_2) = g(\text{head } l_1, \text{head } l_2):$$
$$\text{merge } \{g^{tc}(\text{tail } l_1,[\text{head } l_2]),\ g^{tc}(l_1,\text{tail } l_2)\}$$

The Merge Operation

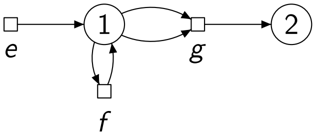
$$
\begin{aligned}
\text{merge } \{l_1, l_2\} = \; & \text{if } (\text{head } l_1) \precsim (\text{head } l_2) \\
& \text{then head } l_1 : \text{merge } \{\text{tail } l_1,\ l_2\} \\
& \text{else head } l_2 : \text{merge } \{l_1, \text{tail } l_2\}
\end{aligned}
$$

A Provisional Specification



$$\text{p } 1 = \text{merge } \{e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$\text{p } 2 = \text{merge } \{g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

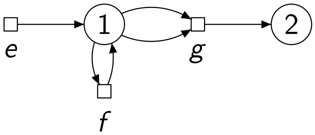
A Provisional Specification



$$\text{p } 1 = \text{merge } \{e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$\text{p } 2 = \text{merge } \{g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

does not work with cyclic hypergraphs

A Provisional Specification

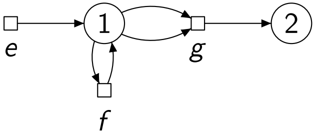


$$p\ 1 = \text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\}$$
$$p\ 2 = \text{merge}\ \{g^{tc}(p\ 1,\ p\ 1)\}$$

does not work with cyclic hypergraphs

head (p 1)

A Provisional Specification

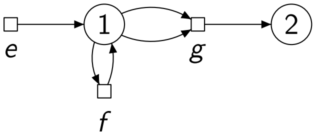


$$\text{p } 1 = \text{merge } \{e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$\text{p } 2 = \text{merge } \{g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

does not work with cyclic hypergraphs

head (p 1)
$= \text{head (merge } \{e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\})$

A Provisional Specification

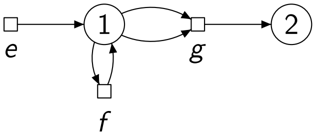


$$\mathsf{p}\ 1 = \mathsf{merge}\ \{e^{\mathsf{tc}}(), f^{\mathsf{tc}}(\mathsf{p}\ 1)\}$$
$$\mathsf{p}\ 2 = \mathsf{merge}\ \{g^{\mathsf{tc}}(\mathsf{p}\ 1,\ \mathsf{p}\ 1)\}$$

does not work with cyclic hypergraphs

$$\begin{aligned} &\mathsf{head}\ (\mathsf{p}\ 1) \\ &= \mathsf{head}\ (\mathsf{merge}\ \{e^{\mathsf{tc}}(), f^{\mathsf{tc}}(\mathsf{p}\ 1)\}) \\ &= \mathsf{if}\ (\mathsf{head}\ e^{\mathsf{tc}}()) \precsim (\mathsf{head}\ f^{\mathsf{tc}}(\mathsf{p}\ 1))\ \ldots \end{aligned}$$

A Provisional Specification

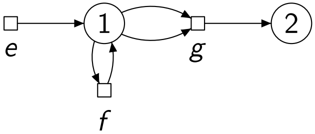


$$p\ 1 = \text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\}$$
$$p\ 2 = \text{merge}\ \{g^{tc}(p\ 1,\ p\ 1)\}$$

does not work with cyclic hypergraphs

$$\begin{aligned} &\text{head}\ (p\ 1) \\ &= \text{head}\ (\text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\}) \\ &= \text{if}\ (\text{head}\ e^{tc}())\ \precsim\ (\text{head}\ f^{tc}(p\ 1))\ \ldots \\ &= \text{if}\ e \precsim f(\text{head}\ (p\ 1))\ \ldots \end{aligned}$$

A Provisional Specification

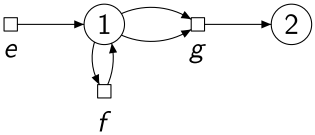


$$p\ 1 = \text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\}$$
$$p\ 2 = \text{merge}\ \{g^{tc}(p\ 1,\ p\ 1)\}$$

does not work with cyclic hypergraphs

$$\begin{aligned}
&\text{head}\ (p\ 1)\\
&= \text{head}\ (\text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\})\\
&= \text{if}\ (\text{head}\ e^{tc}()) \precsim (\text{head}\ f^{tc}(p\ 1))\ \dots\\
&= \text{if}\ e \precsim f(\text{head}\ (p\ 1))\ \dots
\end{aligned}$$

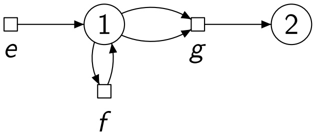
A Provisional Specification



$$p\ 1 = \text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\}$$
$$p\ 2 = \text{merge}\ \{g^{tc}(p\ 1,\ p\ 1)\}$$

works with acyclic hypergraphs

A Provisional Specification

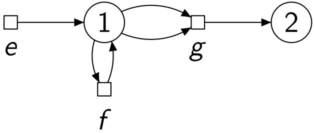


$$\text{p } 1 = \text{merge } \{e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$\text{p } 2 = \text{merge } \{g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

works with acyclic hypergraphs

- strict evaluation/call-by-value "≈" bottom-up
  "≈" Algorithm 2 of (Huang and Chiang, 2005)

A Provisional Specification

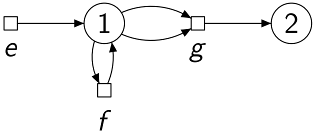


$$p\ 1 = \text{merge}\ \{e^{tc}(), f^{tc}(p\ 1)\}$$
$$p\ 2 = \text{merge}\ \{g^{tc}(p\ 1,\ p\ 1)\}$$

works with acyclic hypergraphs

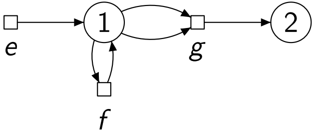
- strict evaluation/call-by-value "≈" bottom-up "≈" Algorithm 2 of (Huang and Chiang, 2005)
- lazy evaluation/call-by-need "≈" top-down "≈" Algorithm 3 of (Huang and Chiang, 2005)

Final Specification



$$p\ 1 = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(p\ 1)\}$$
$$p\ 2 = (\text{head } g^{\text{tc}}(p\ 1,\ p\ 1)):\text{merge } \{\text{tail } g^{\text{tc}}(p\ 1,\ p\ 1)\}$$

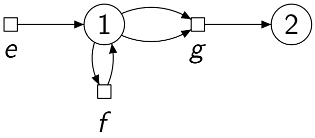
Final Specification



$$\text{p } 1 = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$\text{p } 2 = (\text{head } g^{\text{tc}}(\text{p } 1, \text{ p } 1)):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

- algorithms for solving the 1-best-derivations problem well established

Final Specification



$$\text{p } 1 = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$\text{p } 2 = (\text{head } g^{\text{tc}}(\text{p } 1, \text{p } 1)):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{p } 1)\}$$

- algorithms for solving the 1-best-derivations problem well established
- cf. (Knuth 1977)

Example Computation

$p\ 1 =$

$p\ 2 =$

Example Computation

p 1 =

p 2 = (head g^{tc}(p 1, p 1)):merge $\{$tail g^{tc}(p 1, p 1)$\}$

Example Computation

p 1 $=$

p 2 $=$ (head g^{tc}(p 1, p 1)):merge $\{$tail g^{tc}(p 1, p 1)$\}$

Example Computation

p 1 =

p 2 = (head g^{tc}(p 1, p 1)):merge {tail g^{tc}(p 1, p 1)}
= g(head (p 1), head (p 1)):merge {tail g^{tc}(p 1, p 1)}

Example Computation

p 1 =

p 2 = (head g^{tc}(p 1, p 1)):merge {tail g^{tc}(p 1, p 1)}
= g(head (p 1), head (p 1)):merge {tail g^{tc}(p 1, p 1)}

Example Computation

$$\text{p 1} = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p 1})\}$$

$$\text{p 2} = (\text{head } g^{\text{tc}}(\text{p 1, p 1})):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p 1, p 1})\}$$
$$= g(\text{head (p 1)}, \text{head (p 1)}):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p 1, p 1})\}$$

Example Computation

$$\text{p 1} = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p 1})\}$$

$$\text{p 2} = (\text{head } g^{\text{tc}}(\text{p 1, p 1})):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p 1, p 1})\}$$
$$= g(\text{head (p 1), head (p 1)}):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p 1, p 1})\}$$

Example Computation

$$\text{p } 1 = (\text{head } e^{\text{tc}}()) \text{:merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$= \text{e:merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$

$$\text{p } 2 = (\text{head } g^{\text{tc}}(\text{p } 1, \text{ p } 1)) \text{:merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$
$$= g(\text{head } (\text{p } 1), \text{ head } (\text{p } 1)) \text{:merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

Example Computation

$$\text{p } 1 = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$
$$= \text{e:merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}$$

$$\text{p } 2 = (\text{head } g^{\text{tc}}(\text{p } 1, \text{ p } 1)):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$
$$= g(\text{head } (\text{p } 1), \text{ head } (\text{p } 1)):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$
$$= g(e, e):\text{head } (\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)):\text{merge } \{\text{tail}^2 \; g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}$$

Example Computation

$$
\begin{aligned}
&\text{p 1} = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p 1})\} \\
&= \text{e:merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p 1})\}
\end{aligned}
$$

$$
\begin{aligned}
&\text{p 2} = (\text{head } g^{\text{tc}}(\text{p 1, p 1})):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p 1, p 1})\} \\
&= g(\text{head (p 1), head (p 1)}):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p 1, p 1})\} \\
&= g(e, e):\text{head } (\text{tail } g^{\text{tc}}(\text{p 1, p 1})):\text{merge } \{\text{tail}^2\ g^{\text{tc}}(\text{p 1, p 1})\}
\end{aligned}
$$

Example Computation

$$
\begin{aligned}
&\text{p } 1 = (\text{head } e^{\text{tc}}()):\text{merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\} \\
&= \text{e:merge } \{\text{tail } e^{\text{tc}}(), f^{\text{tc}}(\text{p } 1)\}
\end{aligned}
$$

$$
\begin{aligned}
&\text{p } 2 = (\text{head } g^{\text{tc}}(\text{p } 1, \text{ p } 1)):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\} \\
&= g(\text{head (p } 1), \text{ head (p } 1)):\text{merge } \{\text{tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)\} \\
&= g(e, e):\text{head (tail } g^{\text{tc}}(\text{p } 1, \text{ p } 1)):\text{merge } \{\text{tail}^2 \; g^{\text{tc}}(\text{p } 1, \text{ p } 1)\} \\
&= g(e, e):\text{head (merge } \{g^{\text{tc}}(\text{tail (p } 1), \text{ [head (p } 1)]), g^{\text{tc}}(\text{p } 1, \text{tail (p } 1))\}) \\
&\qquad\qquad :\text{merge } \{\text{tail}^2 \; g^{\text{tc}}(\text{p } 1, \text{ p } 1)\}
\end{aligned}
$$

Complexity and Runtime

Contribution

this work	=	Huang and Chiang, 2005

+ larger class of hypergraphs
+ confirm conjecture for cyclic case
+ more flexible weight structures
+ functional specification (Haskell)

References

David Chiang.
2007.
Hierarchical phrase-based translation. *Comp. Ling.*, 33(2):201–228.

Joshua Goodman.
1999.
Semiring parsing. *Comp. Ling.*, 25(4):573–605.

Liang Huang and David Chiang.
2005.
Better k-best parsing. In *Proc. 9th Int. Conf. Parsing Technologies*, pages 53–64. ACL.

Daniel Jurafsky and James H. Martin.
2009.
Speech and Language Processing – [. . .]. Prentice-Hall, 2nd edition.

Andreas Maletti and Giorgio Satta.
2009.
Parsing algorithms based on tree automata. In *Proc. 11th Int. Conf. Parsing Technologies*, pages 1–12. ACL.