Parsing with Dynamic Continuized CCG

Michael White,[a] Simon Charlow,[b] Jordan Needle,[a] Dylan Bumford[c]

4–6 September 2017, TAG+13

[a]Department of Linguistics, The Ohio State University
[b]Department of Linguistics, Rutgers University
[c]Department of Linguistics, UCLA

Joint work with

Simon
Charlow

RUTGERS

Jordan
Needle

THE OHIO STATE UNIVERSITY

Dylan
Bumford

UCLA

Introduction

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

  i.e., $\exists x.\text{relative}(x, me) \wedge [\text{dies}(x) \rightarrow \text{fortune}(me)]$

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

  i.e., $\exists x.\text{relative}(x, me) \wedge [\text{dies}(x) \rightarrow \text{fortune}(me)]$
- if <every relative of mine dies>, I'll . . . a fortune $(* \, \forall > \text{if})$

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

  i.e., $\exists x.\text{relative}(x, me) \wedge [\text{dies}(x) \rightarrow \text{fortune}(me)]$
- if <every relative of mine dies>, I'll . . . a fortune $(* \, \forall > \text{if})$

  i.e., $* \, \forall x.\text{relative}(x, me) \rightarrow [\text{dies}(x) \rightarrow \text{fortune}(me)]$

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

  i.e., $\exists x.\text{relative}(x, me) \wedge [\text{dies}(x) \rightarrow \text{fortune}(me)]$
- if <every relative of mine dies>, I'll . . . a fortune $(* \; \forall > \text{if})$

  i.e., $* \; \forall x.\text{relative}(x, me) \rightarrow [\text{dies}(x) \rightarrow \text{fortune}(me)]$

$\Rightarrow$ Explanation in terms of indefinites' discourse function a long expected result

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

  i.e., $\exists x.\text{relative}(x, me) \land [\text{dies}(x) \to \text{fortune}(me)]$
- if <every relative of mine dies>, I'll ... a fortune $(* \, \forall > \text{if})$

  i.e., $* \, \forall x.\text{relative}(x, me) \to [\text{dies}(x) \to \text{fortune}(me)]$

$\Rightarrow$ Explanation in terms of indefinites' discourse function a long expected result — arguably, Charlow (2014) first to show this satisfactorily!

A breakthrough in semantic theory

Indefinites not bothered by scope islands

Example

- if <a relative of mine dies>, I'll inherit a fortune $(\exists > \text{if})$

  i.e., $\exists x.\text{relative}(x, me) \wedge [\text{dies}(x) \rightarrow \text{fortune}(me)]$

- if <every relative of mine dies>, I'll ... a fortune $(* \, \forall > \text{if})$

  i.e., $* \, \forall x.\text{relative}(x, me) \rightarrow [\text{dies}(x) \rightarrow \text{fortune}(me)]$

⇒ Explanation in terms of indefinites' discourse function a long expected result — arguably, Charlow (2014) first to show this satisfactorily!

⇒ Can Charlow's approach be made to work computationally?

Combinatory Categorial Grammar (CCG; Steedman 2000, 2012)

- Constrained grammar formalism with linguistically motivated treatment of long-distance dependencies and coordination
- Basis for fast & accurate parsers (Hockenmaier & Steedman 2007, Clark & Curran 2007, Lee et al. 2016, . . .)

Implementing DyC^{3}G

Combinatory Categorial Grammar (CCG; Steedman 2000, 2012)

- Constrained grammar formalism with linguistically motivated treatment of long-distance dependencies and coordination
- Basis for fast & accurate parsers (Hockenmaier & Steedman 2007, Clark & Curran 2007, Lee et al. 2016, . . .)

Continuized CCG (Barker & Shan 2002, 2008, 2015)

- Quantifiers are functions on their own continuations
- Order-sensitive phenomena as linguistic side effects

Combinatory Categorial Grammar (CCG; Steedman 2000, 2012)

- Constrained grammar formalism with linguistically motivated treatment of long-distance dependencies and coordination
- Basis for fast & accurate parsers (Hockenmaier & Steedman 2007, Clark & Curran 2007, Lee et al. 2016, ...)

Continuized CCG (Barker & Shan 2002, 2008, 2015)

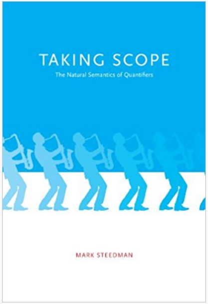
- Quantifiers are functions on their own continuations
- Order-sensitive phenomena as linguistic side effects

Dynamic Continuized CCG (Charlow 2014)

- Explains exceptional scope of indefinites by treating them as side effects in continuized grammars

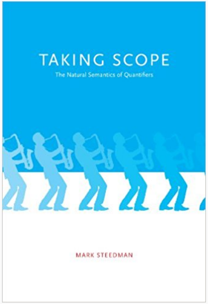
Why should we care about the scope of indefinites?

As Steedman (2012) observes, computationally implemented approaches to scope taking from Cooper storage (Cooper 1983) to underspecification (e.g. Copestake et al. 2005) and more **have not distinguished** indefinites from true quantifiers



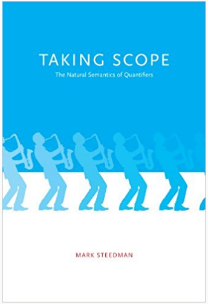
Why should we care about the scope of indefinites?

As Steedman (2012) observes, computationally implemented approaches to scope taking from Cooper storage (Cooper 1983) to underspecification (e.g. Copestake et al. 2005) and more **have not distinguished** indefinites from true quantifiers — typically resulting in vast overgeneration



Why should we care about the scope of indefinites?

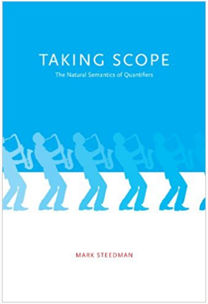
As Steedman (2012) observes, computationally implemented approaches to scope taking from Cooper storage (Cooper 1983) to underspecification (e.g. Copestake et al. 2005) and more **have not distinguished** indefinites from true quantifiers — typically resulting in vast overgeneration



While the scope possibilities for indefinites appear to be unconstrained in general, **true quantifiers** appear to have a much more limited distribution subject to constraints imposed by **scope islands**

Why should we care about the scope of indefinites?

As Steedman (2012) observes, computationally implemented approaches to scope taking from Cooper storage (Cooper 1983) to underspecification (e.g. Copestake et al. 2005) and more **have not distinguished** indefinites from true quantifiers — typically resulting in vast overgeneration



While the scope possibilities for indefinites appear to be unconstrained in general, **true quantifiers** appear to have a much more limited distribution subject to constraints imposed by **scope islands** — which is not accounted for even in implementations of **DRT** (Bos 2003)

Ok — but what about Steedman's (2012) analysis?

Steedman (2012) accounts for indefinites' exceptional scope taking by treating them as underspecified Skolem terms in a non-standard static semantics, rather than deriving this behavior from their discourse function

Ok — but what about Steedman's (2012) analysis?

Steedman (2012) accounts for indefinites' exceptional scope taking by treating them as underspecified Skolem terms in a non-standard static semantics, rather than deriving this behavior from their discourse function

. . . while true quantifiers are restricted by CCG's surface compositional combinatorics

Ok — but what about Steedman's (2012) analysis?

Steedman (2012) accounts for indefinites' exceptional scope taking by treating them as underspecified Skolem terms in a non-standard static semantics, rather than deriving this behavior from their discourse function

...while true quantifiers are restricted by CCG's surface compositional combinatorics — **but** does this suffice empirically?

Potential issues for Steedman's CCG

Steedman's CCG can't account for quantifiers taking scope from medial positions (Barker & Shan, 2015)

Potential issues for Steedman's CCG

Steedman's CCG can't account for quantifiers taking scope from medial positions (Barker & Shan, 2015)

Linear order constraints on where **negative polarity items** may appear also apparently an issue

Potential issues for Steedman's CCG

Steedman's CCG can't account for quantifiers taking scope from medial positions (Barker & Shan, 2015)

Linear order constraints on where **negative polarity items** may appear also apparently an issue

⇒ Barker & Shan's continuized grammars **generalize** Hendrik's (1993) approach to scope taking while also enabling order-sensitive analyses

This paper's contribution

Open source reference implementation[1] of a shift-reduce parser that

1. extends Barker and Shan (2014) to only invoke Charlow's (2014) monadic lifting and lowering where necessary
2. integrates Steedman's (2000) CCG for deriving basic predicate-argument structure and enriches it with a practical method of lexicalizing scope island constraints (Barker & Shan 2006)
3. takes advantage of the resulting scope islands in defining novel normal form constraints for efficient parsing

[1] `https://github.com/mwhite14850/dyc3g`

This paper's contribution

Open source reference implementation[1] of a shift-reduce parser that

1. extends Barker and Shan (2014) to only invoke Charlow's (2014) monadic lifting and lowering where necessary
2. integrates Steedman's (2000) CCG for deriving basic predicate-argument structure and enriches it with a practical method of lexicalizing scope island constraints (Barker & Shan 2006)
3. takes advantage of the resulting scope islands in defining novel normal form constraints for efficient parsing

[1] `https://github.com/mwhite14850/dyc3g`

Continuized CCG

Tower Notation

Towers provide a much more intuitive way to understand continuized grammars (Barker & Shan 2015)

$$\dfrac{\dfrac{\textit{left phrase}}{\begin{array}{c}\dfrac{C \mid D}{B/A} \\ \dfrac{g[\,]}{f}\end{array}} \qquad \dfrac{\textit{right phrase}}{\begin{array}{c}\dfrac{D \mid E}{A} \\ \dfrac{h[\,]}{x}\end{array}}}{\begin{array}{c}\dfrac{C \mid E}{B} \\ \dfrac{g[h[\,]]}{f(x)}\end{array}}\ \textbf{Comb},>$$

Lift

Generalized Type Raising (computationally: just where necessary)

$$\dfrac{\dfrac{\textit{any phrase}}{\dfrac{A}{x}}}{\dfrac{\dfrac{B \mid B}{A}}{\dfrac{[\,]}{x}}}\ \textbf{Lift}$$

$$\text{where} \quad \frac{[\,]}{x} \quad \equiv \quad \lambda k.kx$$

Lower

Needed to complete derivations, and for scope islands

$$\dfrac{\dfrac{\dfrac{\textit{any clause}}{\dfrac{A \mid S}{S}}}{\dfrac{f[\,]}{x}}}{\dfrac{A}{f[x]}}\,\textbf{Lower}$$

$$\text{where} \quad f[x] \quad \equiv \quad \frac{f[\,]}{x}(\lambda v.v)$$

Rules are defined recursively

Combine

$$
\dfrac{
\begin{array}{c} \dfrac{D \mid E}{A} \\ \dfrac{g[\,]}{a} \end{array}
\qquad
\begin{array}{c} \dfrac{E \mid F}{B} \\ \dfrac{h[\,]}{b} \end{array}
}{
\begin{array}{c} \dfrac{D \mid F}{C} \\ \dfrac{g[h[\,]]}{c} \end{array}
}\,\mathbf{C}
$$

Lift Left

$$
\dfrac{
\begin{array}{c} A \\ a \end{array}
\qquad
\begin{array}{c} \dfrac{E \mid F}{B} \\ \dfrac{h[\,]}{b} \end{array}
}{
\begin{array}{c} \dfrac{E \mid F}{C} \\ \dfrac{h[\,]}{c} \end{array}
}\,\uparrow\mathbf{L}
$$

Lift Right

$$
\dfrac{
\begin{array}{c} \dfrac{D \mid E}{A} \\ \dfrac{g[\,]}{a} \end{array}
\qquad
\begin{array}{c} B \\ b \end{array}
}{
\begin{array}{c} \dfrac{D \mid E}{C} \\ \dfrac{g[\,]}{c} \end{array}
}\,\uparrow\mathbf{R}
$$

if $\dfrac{A : a \quad B : b}{C : c}$

Linear Scope Bias

(With Steedman's CCG "on the bottom")

$$\dfrac{\dfrac{\textit{someone}}{\begin{matrix}\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{np}} \\ \dfrac{\exists x.[\,]}{x}\end{matrix}} \qquad \dfrac{\textit{loves everyone}}{\begin{matrix}\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}\backslash\mathsf{np}} \\ \dfrac{\forall y.[\,]}{\lambda z.\text{love}(z, y)}\end{matrix}}}{\dfrac{\begin{matrix}\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}} \\ \dfrac{\exists x.\forall y.[\,]}{\text{love}(x, y)}\end{matrix}}{\begin{matrix}\mathsf{s} \\ \exists x.\forall y.\text{love}(x, y)\end{matrix}}\,\textbf{Lower}}\,\textbf{Comb},<$$

Inverse Scope

External and internal lift integrated into binary step

$$
\dfrac{
\dfrac{\textit{someone}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{np}} \;\; \dfrac{\exists x.[\,]}{x}}
\qquad
\dfrac{\textit{loves everyone}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}\backslash\mathsf{np}} \;\; \dfrac{\forall y.[\,]}{\lambda z.\mathrm{love}(z, y)}}
}{
\dfrac{\mathsf{s} \mid \mathsf{s}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}}} \;\; \dfrac{\forall y.[\,]}{\dfrac{\exists x.[\,]}{\mathrm{love}(x, y)}}
}\,\textbf{LiftL},\textbf{LiftR},<
$$

$$
\dfrac{\dfrac{\mathsf{s} \mid \mathsf{s}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}}} \;\; \dfrac{\forall y.[\,]}{\dfrac{\exists x.[\,]}{\mathrm{love}(x, y)}}}{\mathsf{s} \;\; \forall y.\exists x.\mathrm{love}(x, y)}\,\textbf{Lower}
$$

CCG on the Bottom

Using Steedman's CCG on the bottom tower level enables the CCG analysis of relative clauses, right node raising, etc. — in particular, there's no need for empty string elements

$$
\dfrac{\textit{who(m)}}{\ldots}
\qquad
\dfrac{\dfrac{\textit{everyone}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}/(\mathsf{s}\backslash \mathsf{np})} \quad \dfrac{\forall x.\mathrm{person}(x) \to [\,]}{\lambda p.px}}
\qquad
\dfrac{\textit{loves}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}\backslash \mathsf{np}/\mathsf{np}} \quad \dfrac{[\,]}{\lambda zy.\mathrm{love}(z,y)}}}{\dfrac{\mathsf{s} \mid \mathsf{s}}{\mathsf{s}/\mathsf{np}} \quad \dfrac{\forall x.\mathrm{person}(x) \to [\,]}{\lambda y.\mathrm{love}(x,y)}}\ \textbf{Comb},>\textbf{B}
$$

Monadic Dynamic Semantics

Monads and Side Effects for Indefinites

Barker & Shan's tower system by itself does not adequately account for the **exceptional scope of indefinites**

Monads and Side Effects for Indefinites

Barker & Shan's tower system by itself does not adequately account for the **exceptional scope of indefinites**

Monads provide a clean way to enrich pure function application in the semantics with side effects — in particular, they provide a way to integrate a **dynamic treatment of indefinites** (Charlow 2014)

Translation to FOL similar to DRT

Example

a linguist swims

$$\lambda s.\{\langle \mathrm{swim}(x), \widehat{sx} \rangle \mid \mathrm{linguist}(x)\}$$

$$\Downarrow$$

$$\exists x.\mathrm{linguist}(x) \wedge \mathrm{swim}(x)$$

Sequencing and Sequence Reduction

Sequencing

"run m to determine v in π"

$$m_v \multimap \pi$$

Example

a linguist swims

$$(\lambda s.\{\langle x, \widehat{sx}\rangle \mid \text{linguist}(x)\})_y \multimap \lambda s.\{\langle \text{swim}(y), s\rangle\}$$

$$\Downarrow$$

$$\lambda s.\{\langle \text{swim}(x), \widehat{sx}\rangle \mid \text{linguist}(x)\}$$

The State.Set Monad

More formally:

$$\begin{array}{rcl} M\alpha & = & s \rightarrow \alpha \times s \rightarrow t \\ a^{\eta} & = & \lambda s.\{\langle a, s\rangle\} \\ m_v \multimap \pi & = & \lambda s.\bigcup_{\langle a,s'\rangle \in ms} \pi[a/v]s' \end{array}$$

Leaving States Implict

States can be left implicit for representational simplicity (cf. implicit assignments with DRT)

Example

a linguist swims

$$(\{\langle x, x\rangle \mid \mathrm{linguist}(x)\})_y \multimap \{\langle \mathrm{swim}(y), \epsilon\rangle\}$$

$$\Downarrow$$

$$\{\langle \mathrm{swim}(x), x\rangle \mid \mathrm{linguist}(x)\}$$

Dynamic Combinatory Rules

Reconceptualizing Continuized Grammars

Continuized grammars can be reconceptualized as operating over an underlying monad

Reconceptualizing Continuized Grammars

Continuized grammars can be reconceptualized as operating over an underlying monad

- Lift identified with sequencing ($\multimap$)
- Lower identified with monadic injection (η)

Monadic Lift

Sequences a continuation

$$\frac{\begin{array}{c}\dfrac{\textit{any phrase}}{A}\\ m\end{array}}{\begin{array}{c}\dfrac{S \mid S}{A}\\ \dfrac{m_v \multimap [\,]}{v}\end{array}}\,\textbf{Lift}$$

Monadic Lower

Injects meaning on tower bottom into monad

$$\dfrac{\dfrac{\textit{any clause}}{\dfrac{A \mid S}{S} \quad \dfrac{f[\,]}{a}}}{\begin{matrix} A \\ f[a^{\eta}] \end{matrix}}\ \textbf{Lower}$$

Delimit Right

$$\dfrac{\dfrac{\vdots}{\begin{array}{c}X/\langle Y\rangle\\ a\end{array}} \qquad \dfrac{\vdots}{\begin{array}{c}Y\\ b\end{array}}}{\dfrac{\vdots}{\begin{array}{c}X\\ c\end{array}}}\,\mathbf{DR} \qquad \text{if} \qquad \dfrac{\dfrac{\vdots}{\begin{array}{c}Y\\ b\end{array}}}{\dfrac{\vdots}{\begin{array}{c}Y\\ b'\end{array}}}\,{\uparrow,\downarrow} \qquad \text{and} \qquad \dfrac{\dfrac{\vdots}{\begin{array}{c}X/Y\\ a\end{array}} \qquad \dfrac{\vdots}{\begin{array}{c}Y\\ b'\end{array}}}{\dfrac{\vdots}{\begin{array}{c}X\\ c\end{array}}}$$

Conditional Scope Island

Universal forced to have narrow scope

$$
\dfrac{
\dfrac{\textit{if}}{\begin{matrix} S/\langle S\rangle/\langle S\rangle \\ \lambda xy.(x \to y)^{\eta} \end{matrix}}
\qquad
\dfrac{\textit{everyone complains}}{\begin{matrix} \dfrac{S \mid S}{S} \\ \dfrac{(\forall_x[\,])^{\eta}}{\mathrm{complain}(x)} \end{matrix}}
}{
\begin{matrix} \dfrac{S \mid S}{S/\langle S\rangle} \\ \dfrac{[\,]}{\lambda y.((\forall_x \,\mathrm{complain}(x)^{\eta})^{\eta} \to y)^{\eta}} \end{matrix}
}\ \mathbf{DR},\uparrow\mathbf{L},>^{\eta}
\qquad
\dfrac{\ldots}{\begin{matrix} S \\ \ldots \end{matrix}}
$$

Resetting a Universal

Reset closes off scope

$$\dfrac{\dfrac{(\forall_x[\,])^\eta}{\text{complain}(x)}}{(\forall_x \,\text{complain}(x)^\eta)^\eta}\downarrow$$

$$\Big\uparrow$$

$$\dfrac{[\,]}{\forall_x \,\text{complain}(x)^\eta}$$

Exceptionally Scoping Indefinite

Reset applied as before

$$
\dfrac{\dfrac{\textit{if}}{\begin{array}{c} S/\langle S\rangle/\langle S\rangle \\ \lambda xy.(x \to y)^{\eta} \end{array}} \quad \dfrac{\textit{someone complains}}{\begin{array}{c} \dfrac{S \mid S}{S} \\ \dfrac{\{\langle x, x\rangle\}_u \multimap [\,]}{\text{complain}(u)} \end{array}}}{\begin{array}{c} \dfrac{S \mid S}{S/\langle S\rangle} \\ \dfrac{\{\langle \text{complain}(x), x\rangle\}_p \multimap [\,]}{\lambda y.(p^{\eta} \to y)^{\eta}} \end{array}} \textbf{DR},\uparrow\textbf{L},>^{\eta} \qquad \dfrac{\ldots}{\begin{array}{c} S \\ \ldots \end{array}}
$$

Resetting an Indefinite

No real scope to close off, result is equivalent

$$
\dfrac{\dfrac{\{\langle x, x\rangle\}_u \multimap [\,]}{\text{complain}(u)}}{\dfrac{\{\langle x, x\rangle\}_u \multimap \{\langle \text{complain}(u), \epsilon\rangle\}}{\dfrac{\{\langle \text{complain}(x), x\rangle\}}{\dfrac{\{\langle \text{complain}(x), x\rangle\}_p \multimap [\,]}{p}}\uparrow}\equiv}\downarrow
$$

Normal Form Constraints

Normal Form Constraints

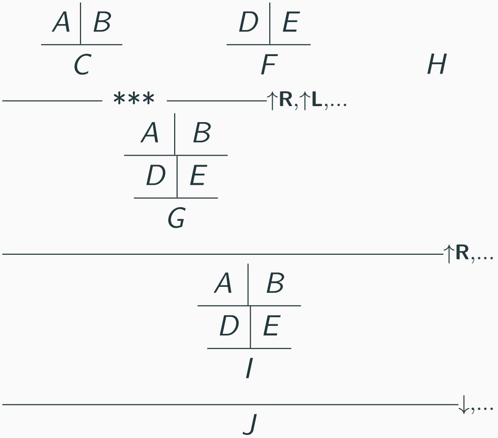
- Normal form constraints can play an important role in practical CCG parsing by eliminating derivations leading to spurious ambiguities without requiring expensive pairwise equivalence checks (Eisner, 1996; Clark and Curran, 2007; Hockenmaier and Bisk, 2010; Lewis and Steedman, 2014)

Normal Form Constraints

- Normal form constraints can play an important role in practical CCG parsing by eliminating derivations leading to spurious ambiguities without requiring expensive pairwise equivalence checks (Eisner, 1996; Clark and Curran, 2007; Hockenmaier and Bisk, 2010; Lewis and Steedman, 2014)
- The lowering operations triggered by scope islands or sentence boundaries provide an opportunity to recursively detect and eliminate non–normal form derivations beyond the base level

Non–Normal Form Derivation

Superfluous three-level tower



Initial Experiment

- Prolog implementation suitable for testing analyses
- Small test suite of 40 examples of average length 6.7 words, roughly comparable in size to Baldridge's (2002) OpenCCG test suite
- Parse time of 60ms per item in same ballpark
- Without normal form constraints, parse time jumps to 4.6s per item, two orders of magnitude slower

Discussion and Conclusions

Parsing Complexity?

- What is the complexity of parsing with Dynamic Continuized CCG?

Parsing Complexity?

- What is the complexity of parsing with Dynamic Continuized CCG?
- What if a bound is placed on tower heights?

Parsing Complexity?

- What is the complexity of parsing with Dynamic Continuized CCG?
- What if a bound is placed on tower heights?
- Recent work on parsing with neural networks has moved away from dynamic programming; Lee et al. (2016) have achieved state-of-the-art accuracy with impressive speed using global neural models and A* search

Parsing Complexity?

- What is the complexity of parsing with Dynamic Continuized CCG?
- What if a bound is placed on tower heights?
- Recent work on parsing with neural networks has moved away from dynamic programming; Lee et al. (2016) have achieved state-of-the-art accuracy with impressive speed using global neural models and A* search
- By respecting Steedman's Principle of Adjacency, such techniques become applicable to DyC^{3}G as well

Are Scope Islands Real?

- Here we've shown how scope islands can be lexicalized, thereby allowing the freedom to make conditionals and relative clauses scope islands but not *that*-complement clauses, for example

Are Scope Islands Real?

- Here we've shown how scope islands can be lexicalized, thereby allowing the freedom to make conditionals and relative clauses scope islands but not *that*-complement clauses, for example
- In principle, could instead **learn** where to prefer reset operations in derivations, rather than making them hard constraints

Are Scope Islands Real?

- Here we've shown how scope islands can be lexicalized, thereby allowing the freedom to make conditionals and relative clauses scope islands but not *that*-complement clauses, for example
- In principle, could instead **learn** where to prefer reset operations in derivations, rather than making them hard constraints
- Making indefinites indifferent to these operations would still greatly simplify the learning task

Conclusions

- First implemented method to derive the **exceptional scope of indefinites** in a principled way
- Charlow's (2014) dynamic continuized grammars can be combined with Steedman's CCG "on the bottom," retaining many of the latter's computationally attractive properties
- Initial experience with reference implementation suggests that lifting and lowering on demand together with normal form constraints just might work computationally

Future Work

- Haskell implementation
- dynamic semantics of anaphora and other order-sensitive phenomena, including negative polarity items
- Selective exceptional scope and focus alternatives
- Split-scope analyses of definites and plurals
- empirical testing with machine learned–models

Acknowledgments

Thanks to

- Mark Steedman, Carl Pollard & OSU Clippers Group
- OSU Targeted Investment in Excellence Award
- NSF IIS-1319318
- . . . you!

Extras

Type-Driven Lowering

Lower Right

$$
\begin{array}{c}
\begin{array}{cc}
\begin{array}{c} \vdots \\ \hline A \\ a : M\alpha \to \beta \end{array} &
\begin{array}{c} \vdots \\ \hline B \\ b : \gamma \end{array}
\end{array} \\ \hline
\begin{array}{c} \vdots \\ \hline C \\ c : \beta \end{array}
\end{array} {\downarrow}\mathbf{R}
\quad \text{if} \quad
\begin{array}{c} \vdots \\ \hline B \\ b : \gamma \\ \hline B' \\ b' : M\alpha \end{array} {\downarrow}
\quad \text{and} \quad
\begin{array}{c}
\begin{array}{cc}
\begin{array}{c} \vdots \\ \hline A \\ a : M\alpha \to \beta \end{array} &
\begin{array}{c} B' \\ b' : M\alpha \end{array}
\end{array} \\ \hline
\begin{array}{c} \vdots \\ \hline C \\ c : \beta \end{array}
\end{array}
$$

Narrow Scope Indefinite

Via type-driven lowering

$$\dfrac{\dfrac{\dfrac{\textit{if}}{\begin{array}{c} S/\langle S\rangle/\langle S\rangle \\ \lambda xy.(x \to y)^{\eta} \\ Mt \to Mt \to Mt \end{array}} \quad \dfrac{\textit{someone complains}}{\begin{array}{c} \frac{S \mid S}{S} \\ \frac{\{\langle x, x\rangle\}_u \multimap [\,]}{\text{complain}(u)} \\ (t \to Mt) \to Mt \end{array}}}{\begin{array}{c} S/\langle S\rangle \\ \lambda y.(\{\langle \text{complain}(x), x\rangle\} \to y)^{\eta} \\ Mt \to Mt \end{array}}\ \textbf{DR},{\downarrow}\textbf{R},> \quad \dfrac{\ldots}{\begin{array}{c} S \\ \ldots \\ t \end{array}}}{}$$

Recursive Lowering

Including case for missing arguments

	base		**recursive**
$\dfrac{\dfrac{\dfrac{A \mid S}{S}\quad \dfrac{g[\,]}{a}}{}\downarrow}{\begin{matrix}A\\ g[a^\eta]\end{matrix}}$	$\dfrac{\dfrac{S \mid S}{A}\quad \dfrac{g[\,]}{p}}{\begin{matrix}A\\ \lambda x.g[(px)^\eta]\end{matrix}}\downarrow$		$\dfrac{\dfrac{S \mid S}{A}\quad \dfrac{g[\,]}{a}}{\begin{matrix}C\\ g[c]\end{matrix}}\downarrow$
	where A is S/Y or $S\backslash Y$		if $\dfrac{A : a}{C : c}\downarrow$

Relative Clause Scope Island

Enforced by relative pronoun

$$
\begin{array}{ccc}
\textit{senator} & \textit{who} & \textit{everyone likes} \\
\hline
N & N\backslash N/\langle S/NP \rangle & \dfrac{S \mid S}{S/NP} \\
\text{senator} & \lambda qpx.px \wedge qx & \dfrac{(\forall_y\,[\,])^\eta}{\lambda x.\text{like}(y,x)}
\end{array}
$$

$$
\text{who everyone likes} \xrightarrow{\ \mathbf{DR},\uparrow\mathbf{L},>\ } \dfrac{S \mid S}{N\backslash N} \qquad \dfrac{[\,]}{\lambda px.px \wedge \forall_y\,\text{like}(y,x)^\eta}
$$

$$
\text{senator who everyone likes} \xrightarrow{\ \uparrow\mathbf{L},<\ } \dfrac{S \mid S}{N} \qquad \dfrac{[\,]}{\lambda x.\text{senator}(x) \wedge \forall_y\,\text{like}(y,x)^\eta}
$$

Scoping from Medial Positions

Inverse linking derivation in paper

Example

- [a voter in [every state]] protests ($\forall > \exists$)
- [few voters$_i$ in [every state] who$_i$ supported Trump] participated in the protests ($\forall > \mathrm{few}$)

Scoping from Medial Positions (2)

Universals sometimes invert from the subjects of sentential complements even in episodic sentences (Farkas & Giannakidou 1996, contra Fox & Sauerland 1996 and Steedman 2012)

Example

- Yesterday, a guide made sure that <[every tour to the Louvre] was fun> ($\forall > \exists$)

Linear Order and Negative Polarity Items

With Steedman's CCG, it appears to be impossible to get one without the other below

Example

- Kim gave [no$_i$ bone] [to any$_j$ dog] $(i < j)$
- * Kim gave [to any$_j$ dog] [no$_i$ bone] $(* \, j < i)$

Linear Order and Negative Polarity Items

With Steedman's CCG, it appears to be impossible to get one without the other below

Example

- Kim gave [no$_i$ bone] [to any$_j$ dog] $(i < j)$
- * Kim gave [to any$_j$ dog] [no$_i$ bone] $(* j < i)$

Cf.

Kim gave — s/pp/np; *to a dog* — s\(s/pp); *a very heavy bone* — *np*

s/pp/np + s\(s/pp) ⇒ s/np ($<\mathbf{Bx}$)

s/np + np ⇒ s ($>$)

Linear Order and Negative Polarity Items

With Steedman's CCG, it appears to be impossible to get one without the other below

Example

- Kim gave [no$_i$ bone] [to any$_j$ dog] $(i < j)$
- * Kim gave [to any$_j$ dog] [no$_i$ bone] $(* \, j < i)$

Cf.

$$\dfrac{\dfrac{\dfrac{\textit{Kim gave}}{\mathrm{s/pp/np}} \quad \dfrac{\textit{to a dog}}{\mathrm{s\backslash(s/pp)}}}{\mathrm{s/np}}{<}\mathbf{Bx} \qquad \dfrac{\textit{a very heavy bone}}{np}}{\mathrm{s}}{>}$$

⇒ Not a problem for Barker & Shan's Continuized CCG though