

Combinatory Categorical Grammars

Grammars

$$\begin{array}{c} \text{CCG} \\ \hline \text{NP} \quad \text{is} \quad \text{fun} \\ \hline \text{CCG} \quad \frac{S \backslash \text{NP} / \text{ADJ} \quad \text{ADJ}}{\lambda f. \lambda x. f(x)} \quad \lambda x. \text{fun}(x) \\ \hline \frac{S \backslash \text{NP} \quad \lambda x. \text{fun}(x)}{\text{fun}(\text{CCG})} \end{array}$$

Combinatory Categorical Grammars

- Appeared around 1980
- Categorical formalism
 - Compositional
 - Puts more information on the words
- Transparent interface between syntax and semantics
- Designed with computation in mind
- Part of a class of mildly context sensitive formalisms (e.g., TAG, HG, LIG) [Joshi et al. 1990]

CCG Categories

$ADJ : \lambda x. fun(x)$

- Basic building block
- Capture syntactic and semantic information jointly (what's the difference from CFG?)

CCG Categories

Syntax

ADJ

$\lambda x. fun(x)$

Semantics

- Basic building block
- Capture syntactic and semantic information jointly

CCG Categories

Syntax

$ADJ : \lambda x. fun(x)$

$(S \setminus NP) / ADJ : \lambda f. \lambda x. f(x)$

$NP : CCG$

- Primitive symbols: N, S, NP, ADJ and PP
- Syntactic combination operator (/,\)
- Slashes specify argument order and direction

CCG Categories

$ADJ : \lambda x. fun(x)$ Semantics

$(S \setminus NP) / ADJ : \lambda f. \lambda x. f(x)$

$NP : CCG$

- λ -calculus expression
- Syntactic type maps to semantic type

CCG Lexical Entries

$\text{fun} \vdash ADJ : \lambda x. \text{fun}(x)$

- Pair words and phrases with meaning
- Meaning captured by a CCG category

CCG Lexical Entries

fun

Natural
Language

$ADJ : \lambda x. fun(x)$

CCG Category

- Pair words and phrases with meaning
- Meaning captured by a CCG category

CCG Lexicons

$\text{fun} \vdash ADJ : \lambda x. \text{fun}(x)$

$\text{is} \vdash (S \setminus NP) / ADJ : \lambda f. \lambda x. f(x)$

$\text{CCG} \vdash NP : CCG$

- Pair words and phrases with meaning
- Meaning captured by a CCG category

Parsing with CCGs

CCG

is

fun

Use lexicon to match words and phrases with their categories

Parsing with CCGs

CCG	is	fun
<hr/>	<hr/>	<hr/>
<i>NP</i>	<i>S \ NP / ADJ</i>	<i>ADJ</i>
<i>CCG</i>	$\lambda f. \lambda x. f(x)$	$\lambda x. fun(x)$

Use lexicon to match words and phrases with their categories

CCG Operations

- Small set of operators
 - Input: 1-2 CCG categories
 - Output: A single CCG category
- Operate on syntax semantics together
- Mirror natural logic operations

CCG Operations

Application

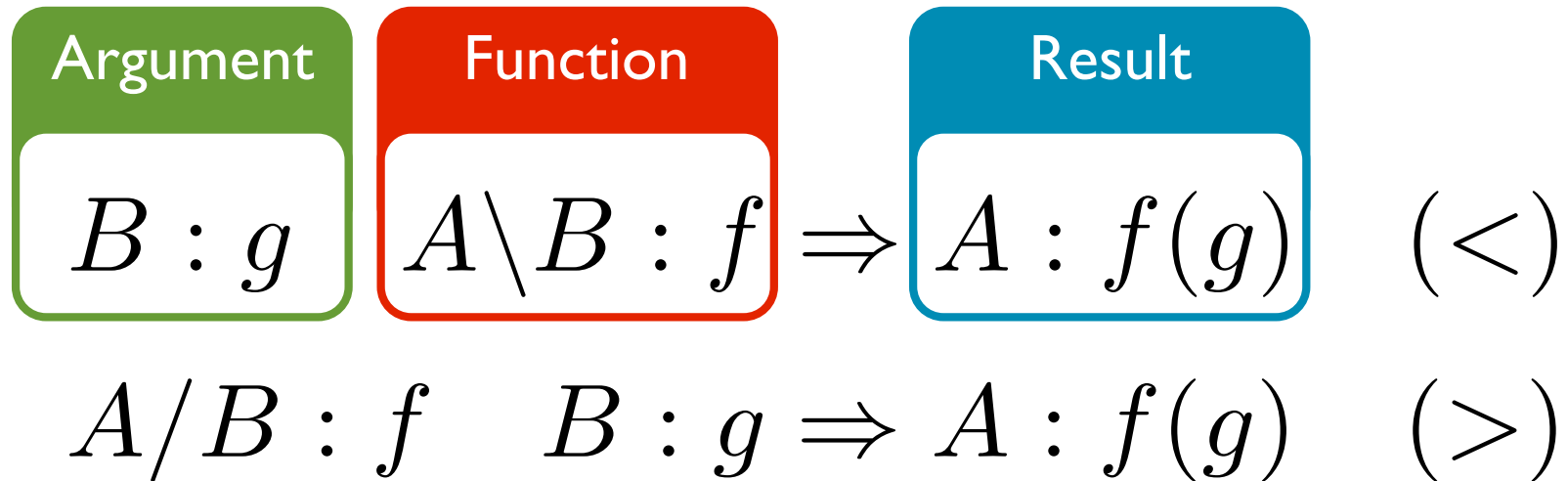
$$B : g \quad A \backslash B : f \Rightarrow A : f(g) \quad (<)$$

$$A / B : f \quad B : g \Rightarrow A : f(g) \quad (>)$$

- Equivalent to function application
- Two directions: forward and backward
 - Determined by slash direction

CCG Operations

Application



- Equivalent to function application
- Two directions: forward and backward
 - Determined by slash direction

Parsing with CCGs

CCG	is	fun
<hr/>	<hr/>	<hr/>
<i>NP</i>	<i>S \ NP / ADJ</i>	<i>ADJ</i>
<i>CCG</i>	<i>$\lambda f. \lambda x. f(x)$</i>	<i>$\lambda x. fun(x)$</i>

Use lexicon to match words and phrases with their categories

Parsing with CCGs

CCG	is	fun
<hr/>	<hr/>	<hr/>
<i>NP</i>	<i>S \ NP / ADJ</i>	<i>ADJ</i>
<i>CCG</i>	$\lambda f. \lambda x. f(x)$	$\lambda x. fun(x)$

Combine categories using operators

$$A/B : f \quad B : g \Rightarrow A : f(g) \quad (>)$$

Parsing with CCGs

CCG	is	fun
<i>NP</i>	<i>S \ NP / ADJ</i>	<i>ADJ</i>
<i>CCG</i>	<i>$\lambda f. \lambda x. f(x)$</i>	<i>$\lambda x. fun(x)$</i>
	<i>S \ NP</i>	
	<i>$\lambda x. fun(x)$</i>	

Combine categories using operators

$$A/B : f \quad B : g \Rightarrow A : f(g) \quad (>)$$

Parsing with CCGs

$$\begin{array}{c}
 \text{CCG} \\
 \hline
 NP \\
 \text{CCG}
 \end{array}
 \quad
 \begin{array}{c}
 \text{is} \\
 \hline
 S \backslash NP / ADJ \\
 \lambda f. \lambda x. f(x)
 \end{array}
 \quad
 \begin{array}{c}
 \text{fun} \\
 \hline
 ADJ \\
 \lambda x. fun(x)
 \end{array}
 \xrightarrow{\quad}
 \begin{array}{c}
 S \backslash NP \\
 \lambda x. fun(x)
 \end{array}$$

Combine categories using operators

$$B : g \quad A \backslash B : f \Rightarrow A : f(g) \quad (<)$$

Parsing with CCGs

CCG	is	fun
<i>NP</i>	$S \backslash NP / ADJ$	ADJ
<i>CCG</i>	$\lambda f. \lambda x. f(x)$	$\lambda x. fun(x)$
	>	
	$S \backslash NP$	
	$\lambda x. fun(x)$	
	<	
S		
$fun(CCG)$		

Combine categories using operators

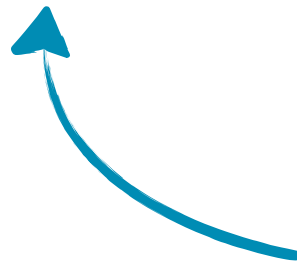
$$B : g \quad A \backslash B : f \Rightarrow A : f(g) \quad (<)$$

Parsing with CCGs

Composed
adjectives



square blue or round yellow pillow



Non-standard
coordination

CCG Operations

Composition

$$A/B : f \quad B/C : g \Rightarrow A/C : \lambda x.f(g(x)) \quad (> B)$$

$$B \setminus C : g \quad A \setminus B : f \Rightarrow A \setminus C : \lambda x.f(g(x)) \quad (< B)$$

- Equivalent to function composition
- Two directions: forward and backward

CCG Operations

Composition

$$\begin{array}{l} \boxed{f} \quad \boxed{g} \Rightarrow \boxed{f \circ g} \quad (> B) \\ A/B : f \quad B/C : g \Rightarrow A/C : \lambda x. f(g(x)) \\ B \setminus C : g \quad A \setminus B : f \Rightarrow A \setminus C : \lambda x. f(g(x)) \quad (< B) \end{array}$$

- Equivalent to function composition
- Two directions: forward and backward

CCG Operations

Type Shifting

$$ADJ : \lambda x.g(x) \Rightarrow N/N : \lambda f.\lambda x.f(x) \wedge g(x)$$

$$PP : \lambda x.g(x) \Rightarrow N \setminus N : \lambda f.\lambda x.f(x) \wedge g(x)$$

$$AP : \lambda e.g(e) \Rightarrow S \setminus S : \lambda f.\lambda e.f(e) \wedge g(e)$$

$$AP : \lambda e.g(e) \Rightarrow S/S : \lambda f.\lambda e.f(e) \wedge g(e)$$

- Category-specific unary operations
- Modify category type to take an argument
- Helps in keeping a compact lexicon

CCG Operations

Type Shifting

Input	Output
$ADJ : \lambda x.g(x)$	$\Rightarrow N/N : \lambda f.\lambda x.f(x) \wedge g(x)$
$PP : \lambda x.g(x)$	$\Rightarrow N \setminus N : \lambda f.\lambda x.f(x) \wedge g(x)$
$AP : \lambda e.g(e)$	$\Rightarrow S \setminus S : \lambda f.\lambda e.f(e) \wedge g(e)$
$AP : \lambda e.g(e)$	$\Rightarrow S/S : \lambda f.\lambda e.f(e) \wedge g(e)$

- Category-specific unary operations
- Modify category type to take an argument
- Helps in keeping a compact lexicon

CCG Operations

Type Shifting

Input	Output
$ADJ : \lambda x.g(x)$	$N/N : \lambda f.\lambda x.f(x) \wedge g(x)$
$PP : \lambda x.g(x)$	$N \setminus N : \lambda f.\lambda x.f(x) \wedge g(x)$
$AP : \lambda e.g(e)$	$S \setminus S : \lambda f.\lambda e.f(e) \wedge g(e)$
Topicalization $AP : \lambda e.g(e)$	$S/S : \lambda f.\lambda e.f(e) \wedge g(e)$

- Category-specific unary operations
- Modify category type to take an argument
- Helps in keeping a compact lexicon

CCG Operations

Coordination

and $\vdash C : conj$

or $\vdash C : disj$

- Coordination is special cased
 - Specific rules perform coordination
 - Coordinating operators are marked with special lexical entries

Parsing with CCGs

square

blue

or

round

yellow

pillow

Parsing with CCGs

square

blue

or

round

yellow

pillow

Use lexicon to match words and phrases with their categories

Parsing with CCGs

square	blue	or	round	yellow	pillow
<hr/>	<hr/>	<hr/>	<hr/>	<hr/>	<hr/>
<i>ADJ</i>	<i>ADJ</i>	<i>C</i>	<i>ADJ</i>	<i>ADJ</i>	<i>N</i>
$\lambda x.square(x)$	$\lambda x.blue(x)$	<i>disj</i>	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$

Use lexicon to match words and phrases with their categories

Parsing with CCGs

square	blue	or	round	yellow	pillow
<hr/>	<hr/>	<hr/>	<hr/>	<hr/>	<hr/>
<i>ADJ</i>	<i>ADJ</i>	<i>C</i>	<i>ADJ</i>	<i>ADJ</i>	<i>N</i>
$\lambda x.square(x)$	$\lambda x.blue(x)$	<i>disj</i>	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$

Shift adjectives to combine

$$ADJ : \lambda x.g(x) \Rightarrow N/N : \lambda f.\lambda x.f(x) \wedge g(x)$$

Parsing with CCGs

square	blue	or	round	yellow	pillow
ADJ	ADJ	C	ADJ	ADJ	N
$\lambda x.square(x)$	$\lambda x.blue(x)$	$disj$	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$
N/N					
$\lambda f.\lambda x.f(x) \wedge square(x)$					

Shift adjectives to combine

$$ADJ : \lambda x.g(x) \Rightarrow N/N : \lambda f.\lambda x.f(x) \wedge g(x)$$

Parsing with CCGs

square	blue	or	round	yellow	pillow
<i>ADJ</i> $\lambda x.square(x)$	<i>ADJ</i> $\lambda x.blue(x)$	<i>C</i> <i>disj</i>	<i>ADJ</i> $\lambda x.round(x)$	<i>ADJ</i> $\lambda x.yellow(x)$	<i>N</i> $\lambda x.pillow(x)$
<i>N/N</i> $\lambda f.\lambda x.f(x) \wedge square(x)$	<i>N/N</i> $\lambda f.\lambda x.f(x) \wedge blue(x)$		<i>N/N</i> $\lambda f.\lambda x.f(x) \wedge round(x)$	<i>N/N</i> $\lambda f.\lambda x.f(x) \wedge yellow(x)$	

Shift adjectives to combine

$$ADJ : \lambda x.g(x) \Rightarrow N/N : \lambda f.\lambda x.f(x) \wedge g(x)$$

Parsing with CCGs

square	blue	or	round	yellow	pillow
<i>ADJ</i>	<i>ADJ</i>	<i>C</i>	<i>ADJ</i>	<i>ADJ</i>	<i>N</i>
$\lambda x.square(x)$	$\lambda x.blue(x)$	<i>disj</i>	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$
<i>N/N</i>	<i>N/N</i>		<i>N/N</i>	<i>N/N</i>	
$\lambda f.\lambda x.f(x) \wedge square(x)$	$\lambda f.\lambda x.f(x) \wedge blue(x)$		$\lambda f.\lambda x.f(x) \wedge round(x)$	$\lambda f.\lambda x.f(x) \wedge yellow(x)$	

Compose pairs of adjectives

$$A/B : f \quad B/C : g \Rightarrow A/C : \lambda x.f(g(x)) \quad (> B)$$

Parsing with CCGs

square	blue	or	round	yellow	pillow
ADJ	ADJ	C	ADJ	ADJ	N
$\lambda x.square(x)$	$\lambda x.blue(x)$	$disj$	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$
N/N	N/N		N/N	N/N	
$\lambda f.\lambda x.f(x) \wedge square(x)$	$\lambda f.\lambda x.f(x) \wedge blue(x)$		$\lambda f.\lambda x.f(x) \wedge round(x)$	$\lambda f.\lambda x.f(x) \wedge yellow(x)$	
N/N			N/N		
$\lambda f.\lambda x.f(x) \wedge square(x) \wedge blue(x)$			$\lambda f.\lambda x.f(x) \wedge round(x) \wedge yellow(x)$		

Compose pairs of adjectives

$$A/B : f \quad B/C : g \Rightarrow A/C : \lambda x.f(g(x)) \quad (> B)$$

Parsing with CCGs

square	blue	or	round	yellow	pillow
<i>ADJ</i>	<i>ADJ</i>	<i>C</i>	<i>ADJ</i>	<i>ADJ</i>	<i>N</i>
$\lambda x.square(x)$	$\lambda x.blue(x)$	<i>disj</i>	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$
<i>N/N</i>	<i>N/N</i>		<i>N/N</i>	<i>N/N</i>	
$\lambda f.\lambda x.f(x) \wedge square(x)$	$\lambda f.\lambda x.f(x) \wedge blue(x)$		$\lambda f.\lambda x.f(x) \wedge round(x)$	$\lambda f.\lambda x.f(x) \wedge yellow(x)$	
<i>N/N</i>			<i>N/N</i>		
$\lambda f.\lambda x.f(x) \wedge square(x) \wedge blue(x)$			$\lambda f.\lambda x.f(x) \wedge round(x) \wedge yellow(x)$		
<i>N/N</i>					
$\lambda f.\lambda x.f(x) \wedge ((square(x) \wedge blue(x)) \vee (round(x) \wedge yellow(x)))$					

Coordinate composed adjectives

Parsing with CCGs

square	blue	or	round	yellow	pillow
<i>ADJ</i>	<i>ADJ</i>	<i>C</i>	<i>ADJ</i>	<i>ADJ</i>	<i>N</i>
$\lambda x.square(x)$	$\lambda x.blue(x)$	<i>disj</i>	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$
<i>N/N</i>	<i>N/N</i>		<i>N/N</i>	<i>N/N</i>	
$\lambda f.\lambda x.f(x) \wedge square(x)$	$\lambda f.\lambda x.f(x) \wedge blue(x)$		$\lambda f.\lambda x.f(x) \wedge round(x)$	$\lambda f.\lambda x.f(x) \wedge yellow(x)$	
$\lambda f.\lambda x.f(x) \wedge square(x) \wedge blue(x)$			$\lambda f.\lambda x.f(x) \wedge round(x) \wedge yellow(x)$		
<i>N/N</i>			<i>N/N</i>		
$\lambda f.\lambda x.f(x) \wedge square(x) \wedge blue(x)$			$\lambda f.\lambda x.f(x) \wedge round(x) \wedge yellow(x)$		
$\lambda f.\lambda x.f(x) \wedge ((square(x) \wedge blue(x)) \vee (round(x) \wedge yellow(x)))$					

Apply coordinated adjectives to noun

$$A/B : f \quad B : g \Rightarrow A : f(g) \quad (>)$$

Parsing with CCGs

square	blue	or	round	yellow	pillow
<i>ADJ</i>	<i>ADJ</i>	<i>C</i>	<i>ADJ</i>	<i>ADJ</i>	<i>N</i>
$\lambda x.square(x)$	$\lambda x.blue(x)$	<i>disj</i>	$\lambda x.round(x)$	$\lambda x.yellow(x)$	$\lambda x.pillow(x)$
<i>N/N</i>	<i>N/N</i>		<i>N/N</i>	<i>N/N</i>	
$\lambda f.\lambda x.f(x) \wedge square(x)$	$\lambda f.\lambda x.f(x) \wedge blue(x)$		$\lambda f.\lambda x.f(x) \wedge round(x)$	$\lambda f.\lambda x.f(x) \wedge yellow(x)$	
$\lambda f.\lambda x.f(x) \wedge square(x) \wedge blue(x)$			$\lambda f.\lambda x.f(x) \wedge round(x) \wedge yellow(x)$		
<i>N/N</i>			<i>N/N</i>		
$\lambda f.\lambda x.f(x) \wedge square(x) \wedge blue(x)$			$\lambda f.\lambda x.f(x) \wedge round(x) \wedge yellow(x)$		
$\lambda f.\lambda x.f(x) \wedge ((square(x) \wedge blue(x)) \vee (round(x) \wedge yellow(x)))$					
<i>N</i>					
$\lambda x.pillow(x) \wedge ((square(x) \wedge blue(x)) \vee (round(x) \wedge yellow(x)))$					

Apply coordinated adjectives to noun

$$A/B : f \quad B : g \Rightarrow A : f(g) \quad (>)$$

Between CCGs and CFGs

	CFGs	CCGs
Combination operations		
Parse tree nodes		
Syntactic symbols		
Paired with words		

Between CCGs and CFGs

	CFGs	CCGs
Combination operations	Many	Few
Parse tree nodes	Non-terminals	Categories
Syntactic symbols	Few dozen	Handful, but can combine
Paired with words	POS tags	Categories