

Introduction to Syntax and Parsing
ACS 2015/16
Stephen Clark
L5: Categorical Grammar



UNIVERSITY OF
CAMBRIDGE

Categorial Grammar (CG)

- Main responsibility for defining syntactic form is in the **lexicon**
- Hence CG is a **lexicalized** theory of grammar
 - along with other theories of grammar such as HPSG, TAG, LFG, ...
- Attractive linguistically because all language-dependent properties reside in the lexicon
 - small number of combination rules are language-invariant
- Also attractive computationally; e.g. *supertagging* for Categorial Grammar leads to highly efficient parsing (Clark and Curran, 2007)

Connection with Semantics

- Categorical Grammar has a strong commitment to Frege's **Principle of Compositionality** (along with Montague from the 70s):
- *The meaning of a phrase is a function of the meanings of the parts of the phrase and how those parts are put together*

Contrast with Phrase-Structure Rules

- Early Chomskian approach and much work in Generative Grammar uses rewrite rules or productions (as in a Context Free Grammar):

$$S \rightarrow NP VP$$
$$VP \rightarrow TV NP$$
$$TV \rightarrow \{likes, sees, \dots\}$$

- Categorical Grammar captures the same information by assigning a functional type, or *category*, to grammatical entities
- Has roots in early work by Polish mathematician Ajdukiewicz (1935) and even earlier in Russel's theory of types

Lexical Categories

- An elementary syntactic structure – a **lexical category** – is assigned to each word in a sentence, eg:

walked: $S \backslash NP$ ‘give me an NP to my left and I return a sentence’

- Think of the lexical category for a verb as a *function*: NP is the argument, S the result, and the slash indicates the direction of the argument

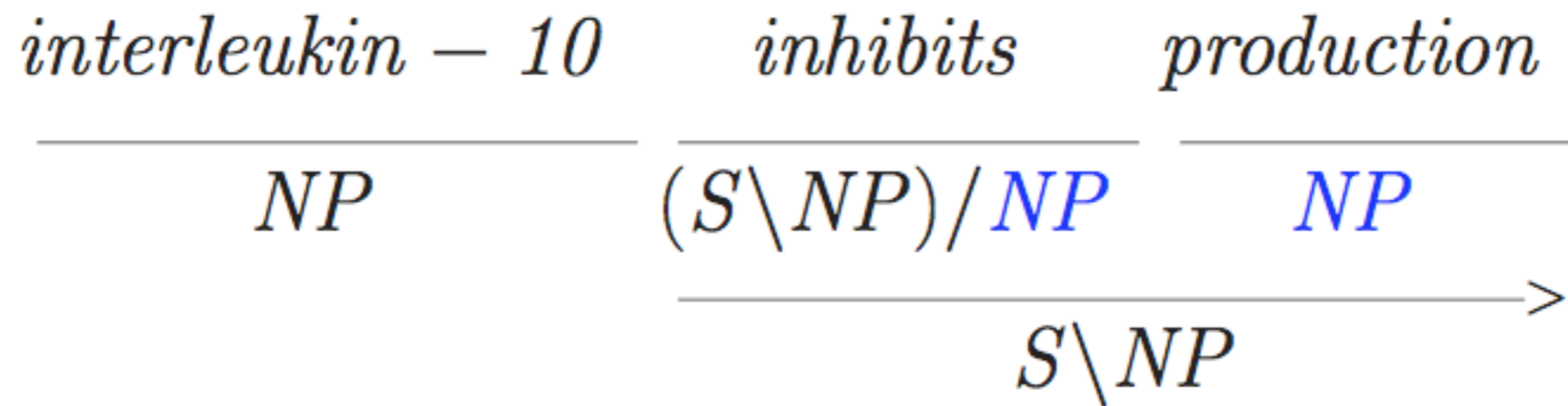
Lexical Categories

- Atomic categories: S, N, NP, PP, \dots (not many more)
- Complex categories are built recursively from atomic categories and slashes
- Example complex categories for verbs:
 - intransitive verb: $S \backslash NP$ *walked*
 - transitive verb: $(S \backslash NP) / NP$ *respected*
 - ditransitive verb: $((S \backslash NP) / NP) / NP$ *gave*

A Simple CG Derivation

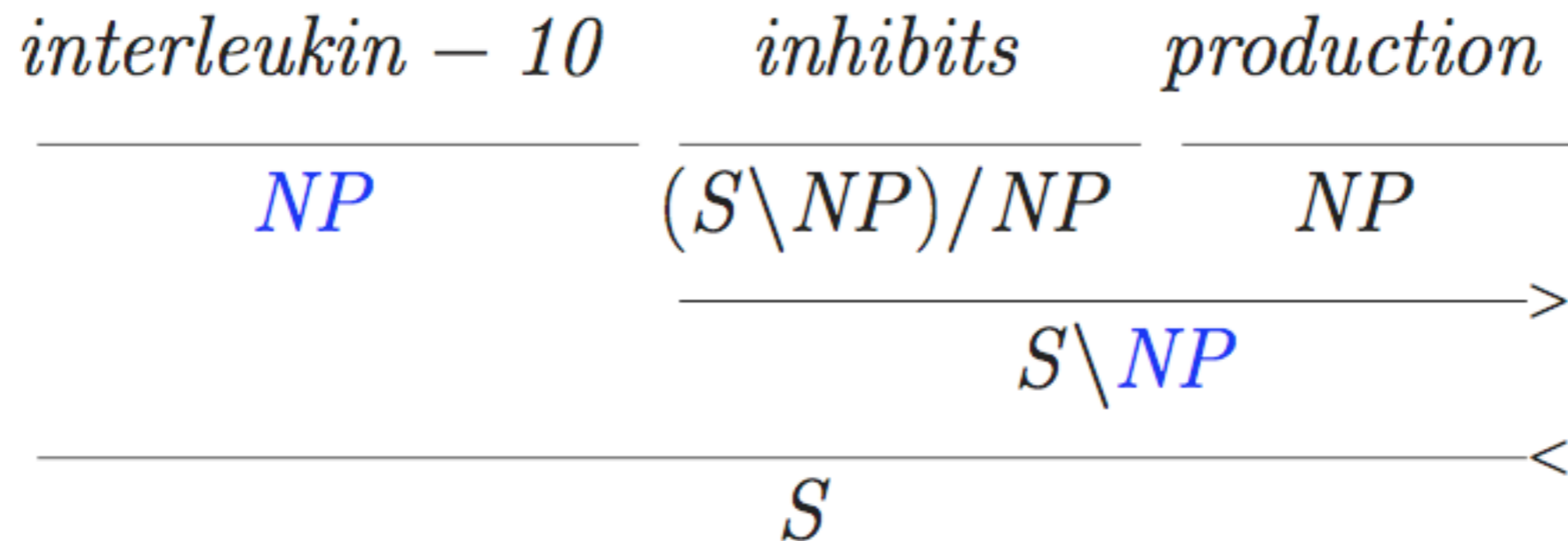
<i>interleukin – 10</i>	<i>inhibits</i>	<i>production</i>
_____	_____	_____
<i>NP</i>	<i>(S \ NP) / NP</i>	<i>NP</i>

A Simple CG Derivation



Forward application

A Simple CG Derivation



Backward application

Combination Rules in CG

- Can think of the categories in blue as “cancelling”
 - early work in CG talks about “cancellation rules”
- Also looks a bit like multiplication and division
- But fundamentally the lexical category for the verb is a function which is applied to its argument

Classical Categorical Grammar

- ‘Classical’ Categorical Grammar only has application rules
- Classical Categorical Grammar is context free
- So what is different to CFG?
 - lexicalisation means that the information in CFG rewrite rules has been pushed down to the leaves of the derivation