COMP 3225

Natural Language Processing

Constituency Grammars

Stuart E. Middleton

sem03@soton.ac.uk

University of Southampton

Overview

- Constituency
- Context Free Grammar
- Grammar Rules for English

- <break - discussion point>

- Treebanks and Head Finding
- Lexicalized Grammars

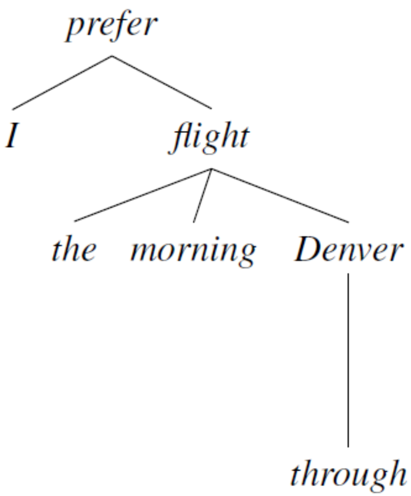
Constituency

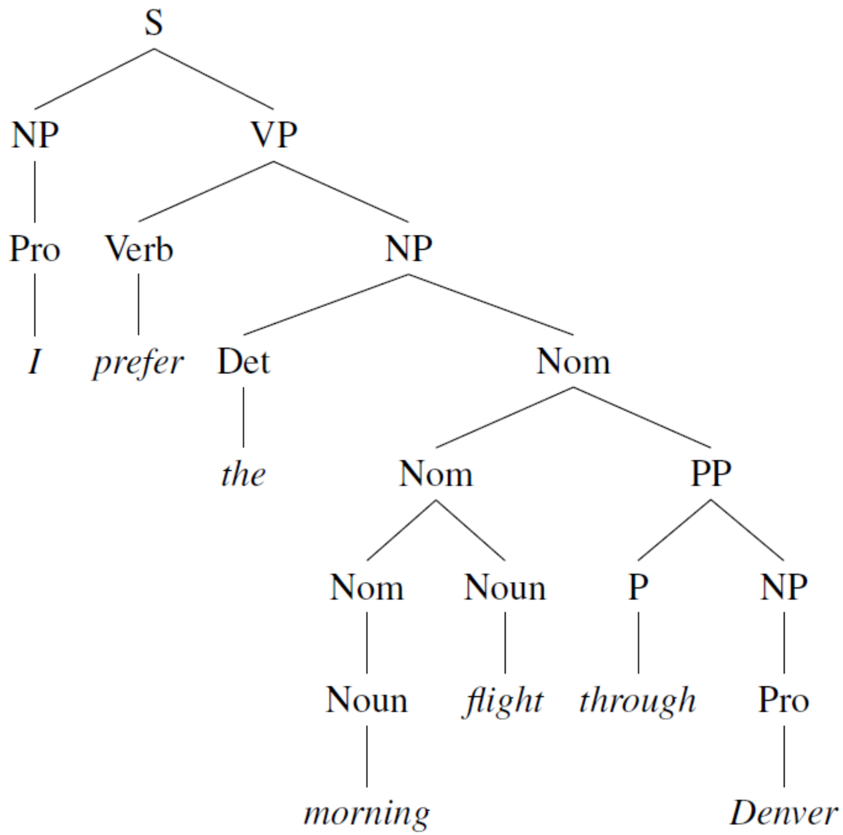
- Syntax is the way words are arranged together
- Syntactic constituency is the idea that words can be grouped into single units (e.g. Noun Phrase)
- We use evidence from the context of the sentence to group words and form constituents
  - A constituent is word (or group of words) that function as a single unit
  - Evidence can be encoded in rules or grammars
- Different grammar types will produce different syntactic structures
  - Context-Free Grammar (also called Phrase-Structure Grammar)
    - Rules based on phrasal constituents + phrase-structure
    - Word order very important
    - Head terms are embedded into trees making it harder to find
  - Dependency Structure Grammar
    - Rules based on grammatical dependencies between words
    - Word order flexible
    - (Head -> Dependent) approximates the semantic relationship between predicates and arguments

Constituency

- Dependency Grammar (left)
- Context-free Grammar (right)

I prefer the morning flight through Denver





Context Free Grammar

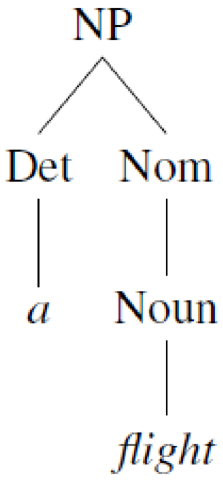
- A Context-Free Grammar (CFG) models constituent structure
- A CFG has a lexicon (of words and symbols) and a set of rules (or productions) on how these will be grouped and ordered
- Rules can be hierarchically embedded, allowing rules to trigger other rules
    - CFG rules are written in form equivalent to Backus-Naur Form (BNF), which is a generative metalanguage originating from IBM in the 1960's

Context Free Grammar

- Example productions (rules) for Noun Phrase

$$\begin{array}{rcl} NP & \rightarrow & Det\ Nominal \\ NP & \rightarrow & ProperNoun \\ Nominal & \rightarrow & Noun \mid Nominal\ Noun \end{array} \qquad \begin{array}{rcl} Det & \rightarrow & a \\ Det & \rightarrow & the \\ Noun & \rightarrow & flight \end{array}$$

- Given <left symbol> generate <right set of symbols>
- NP >> Det Nominal >> Det Noun >> a flight = one derivation
- Derivations are usually represented as a parse tree

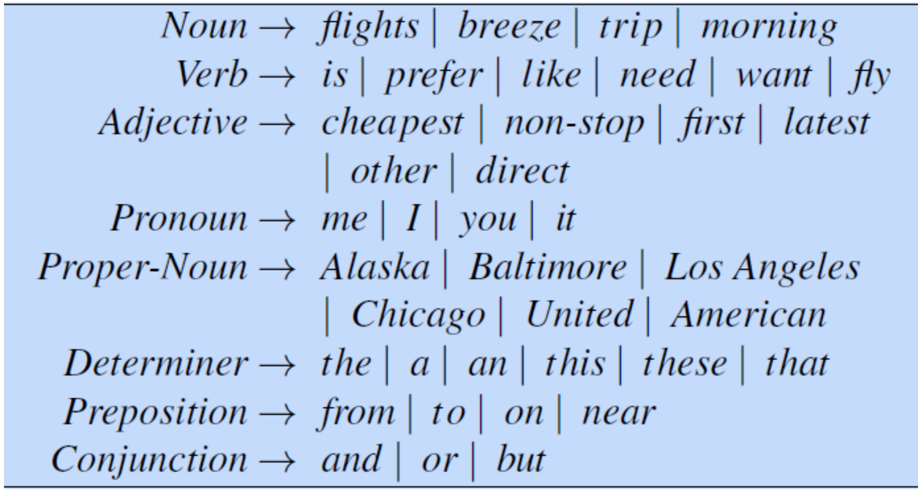


- Leaf nodes are terminal nodes (words from lexicon)
- Non-terminal nodes define lexical categories (POS)
- A node is said to dominate its child nodes
- The root node is the start symbol (usually 'S')

Context Free Grammar

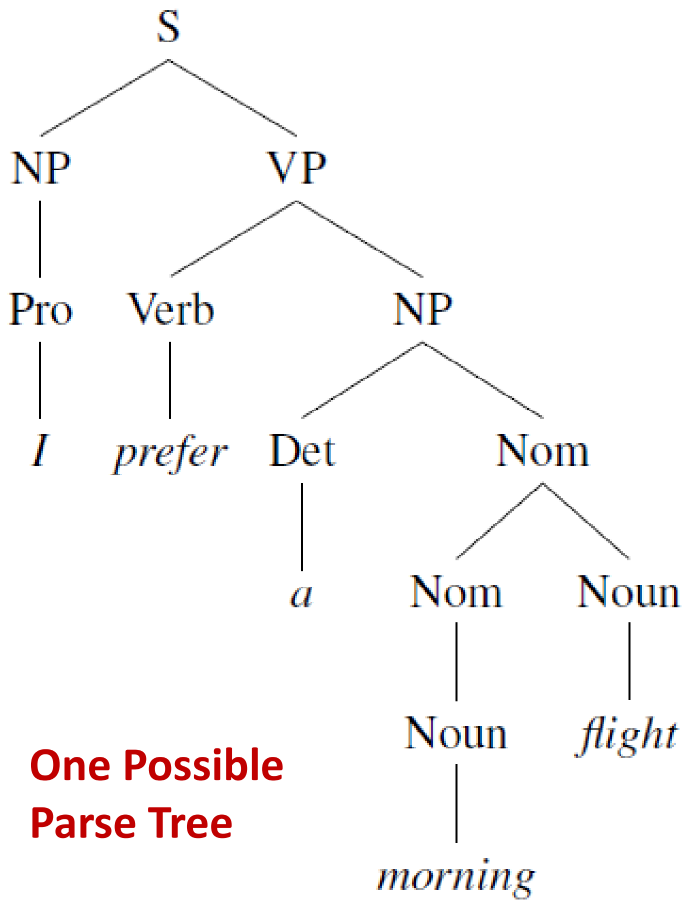
- Example CFG for talking about flights

I prefer the morning flight through Denver

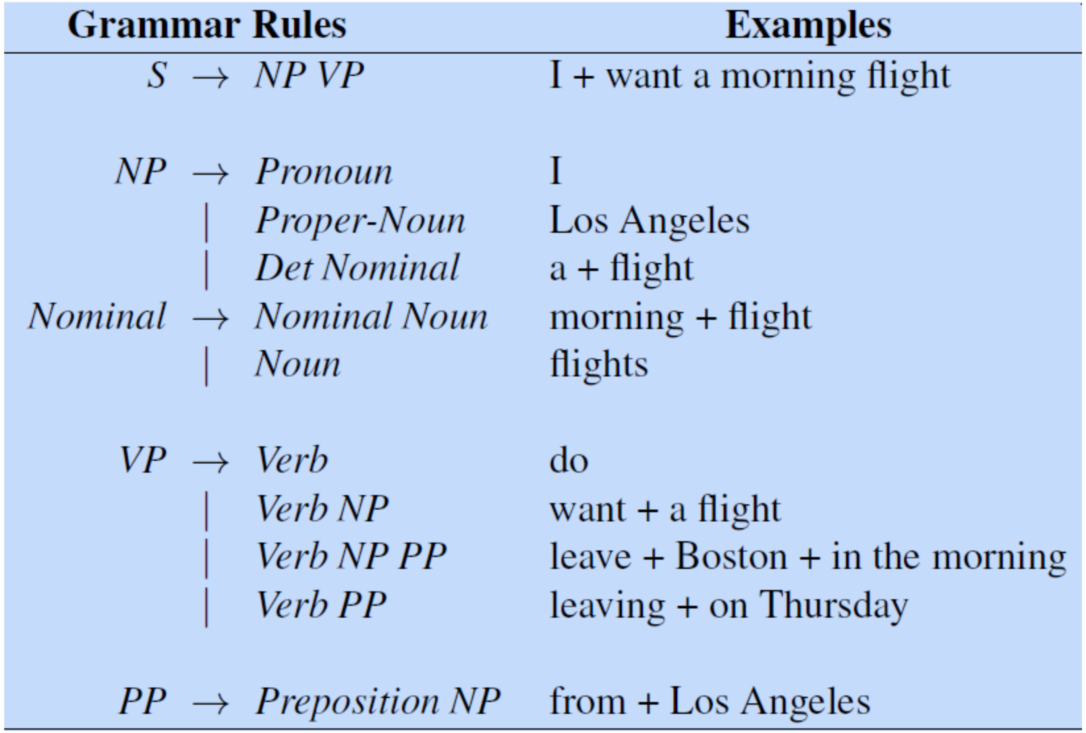


Noun → *flights* | *breeze* | *trip* | *morning*
Verb → *is* | *prefer* | *like* | *need* | *want* | *fly*
Adjective → *cheapest* | *non-stop* | *first* | *latest*
| *other* | *direct*
Pronoun → *me* | *I* | *you* | *it*
Proper-Noun → *Alaska* | *Baltimore* | *Los Angeles*
| *Chicago* | *United* | *American*
Determiner → *the* | *a* | *an* | *this* | *these* | *that*
Preposition → *from* | *to* | *on* | *near*
Conjunction → *and* | *or* | *but*

Lexicon



One Possible Parse Tree



Grammar Rules	**Examples**
S → *NP VP*	I + want a morning flight
NP → *Pronoun*	I
| *Proper-Noun*	Los Angeles
| *Det Nominal*	a + flight
Nominal → *Nominal Noun*	morning + flight
| *Noun*	flights
VP → *Verb*	do
| *Verb NP*	want + a flight
| *Verb NP PP*	leave + Boston + in the morning
| *Verb PP*	leaving + on Thursday
PP → *Preposition NP*	from + Los Angeles

Productions

Context Free Grammar

- Sentences which can be derived from a CFG are grammatical
- Sentences which cannot are ungrammatical
- A CFG is a generative grammar since the language is defined by the possible sentences it can generate
- The problem of mapping sentences to parse trees is called syntactic parsing

Grammar Rules for English

- Sentence-level constructions for English structure
- Declarative - subject NP followed by a VP

  S → NP VP

  The flight   should leave at 6pm

- Imperative - VP with no subject

  S → VP

  Show me the flight at 6pm

- yes-no question - Auxilary verb followed by subject NP and a VP

  S → Aux NP VP

  Are   any flights   available today?

Grammar Rules for English

- wh-subject-question - same as declarative but with a wh-word

  S → Wh-NP VP

  What flight should leave at 6pm?
- wh-non-subject-question - wh-phrase is not the subject

  S → Wh-NP Aux NP VP

  What flights do you have at 6pm?
- The wh-non-subject-question is an example of a long-distance dependency
  - The Wh-NP is far away from the semantically relevant main VP

Grammar Rules for English

- Sentences can consist of one or more clauses
- A clause represents a 'complete thought'
- A clause is made up of two or more of the following components
  - Subject - what the clause is about
  - Verb
  - Object - person, place, thing or idea (which is not the subject)
  - (Subject|Object) Complement - extra info which completes the phrase
  - Adverbial - adjunct (additional info), conjunct (linking), disjunct (comment)
- Clauses are critical for applications such as relation extraction
- Useful book with definitions of English grammar
  - John Seely, Oxford A-Z of Grammar and Punctuation, Oxford Press

Grammar Rules for English

- Noun phrase - pronoun, proper noun, determiner nominal
- Noun phrases consist of a head noun and various modifiers

  NP → Det Nominal

  The flight was cancelled
- The determiner can be a simple lexical term (a, the, this ...)

  Det → a|the|this ...
- Or a more complex expression with a possessive marker ('s)

  Det → NP 's

  London's mayor's flight was cancelled
- The nominal is a head noun and optional noun modifiers, which can occur before or after the head noun

  Nominal → Noun

  Nominal → NUM Nominal

  Nominal → Nominal PP

  Nominal → (who|what) VP ... and more

Grammar Rules for English

- Verb phrase - VP plus a number of other constituents

  VP → Verb VP → Verb NP

  VP → Verb NP PP VP → Verb PP

  ... leaving on Thursday
- Sequential complements - VP followed by an embedded sentence

  VP → Verb S

  You said [you had a lot of money]

  Traditional grammars subcategorize verbs into a few categories
  - Transitive verbs - object e.g. they hit the bar
  - Intransitive verbs - no object e.g. they just ran
  - Distransitive verbs - direct and indirect object e.g. she told me^1 the story^2
  - Linking verbs - links clause subject with complement e.g. could be right
- Modern grammars can have up to 100 subcategories
  - Sets of complements are called the subcategorization frame for the verb
  - You can think of a verb as a predicate
  - Verb(Arg, Arg ...) e.g. FIND(I, a flight)

Grammar Rules for English

- Coordination - conjunctions (and, or, but)
- Coordinate two or more NP's

  VP → NP and NP

  Please repeat the flights^1 and the costs^2

  Nominal → Nominal and Nominal

  Please repeat the flights^1 and costs^2

- Conjunction involving VP's and S's

  S → S and S

  VP → VP and VP

  What flights do you have leaving London and arriving in USA?

Break

- Panopto Quiz - discussion point

- Sentence: The/DT cow/NN jumped/VBD over/IN the/DT moon/NN
- Context-Free Grammar: NP → DT NOM; NOM → NN; VP → VB*; VP → VB* IN
- Which parse tree is the grammatical one?

(S NP(The cow) VP(jumped) NP(over the moon))
(S NP(The cow) VP(jumped over) NP(the moon))
(S NP(The cow jumped) NP(over the moon))
(S NP(cow) VP(jumped over) NP(moon))

Break

- Panopto Quiz - discussion point

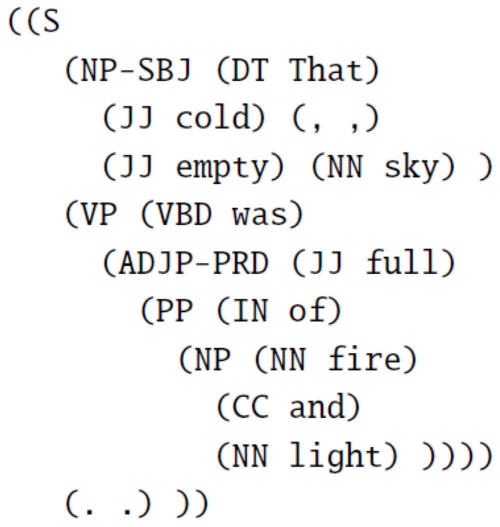
- Sentence: The/DT cow/NN jumped/VBD over/IN the/DT moon/NN
- Context-Free Grammar: NP → DT NOM; NOM → NN; VP → VB*; VP → VB* IN
- Which parse tree is the grammatical one?

(S NP(The cow) VP(jumped) NP(over the moon))
(S NP(The cow) VP(jumped over) NP(the moon))
(S NP(The cow jumped) NP(over the moon))
(S NP(cow) VP(jumped over) NP(moon))

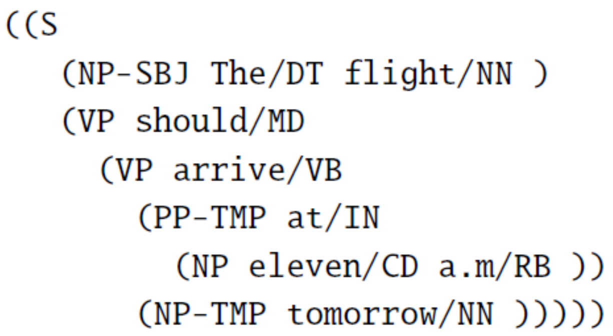
NP → DT NOM → DT NN	>> The cow
	>> the moon
VP → VB* IN → VBN IN	>> jumped over

Treebanks and Head Finding

- A treebank is a syntactically annotated corpus
- Treebanks commonly have different tagsets based on linguistic annotation choices from authoring project
- Penn Treebank 3
  - Corpus - Newswire and Transcribed Speech
  - Annotations - sentences, POS tags, syntactic parse trees
  - https://catalog.ldc.upenn.edu/LDC99T42
  - LDC datasets also available from University of Southampton Library
  - http://edshare.soton.ac.uk/20520/



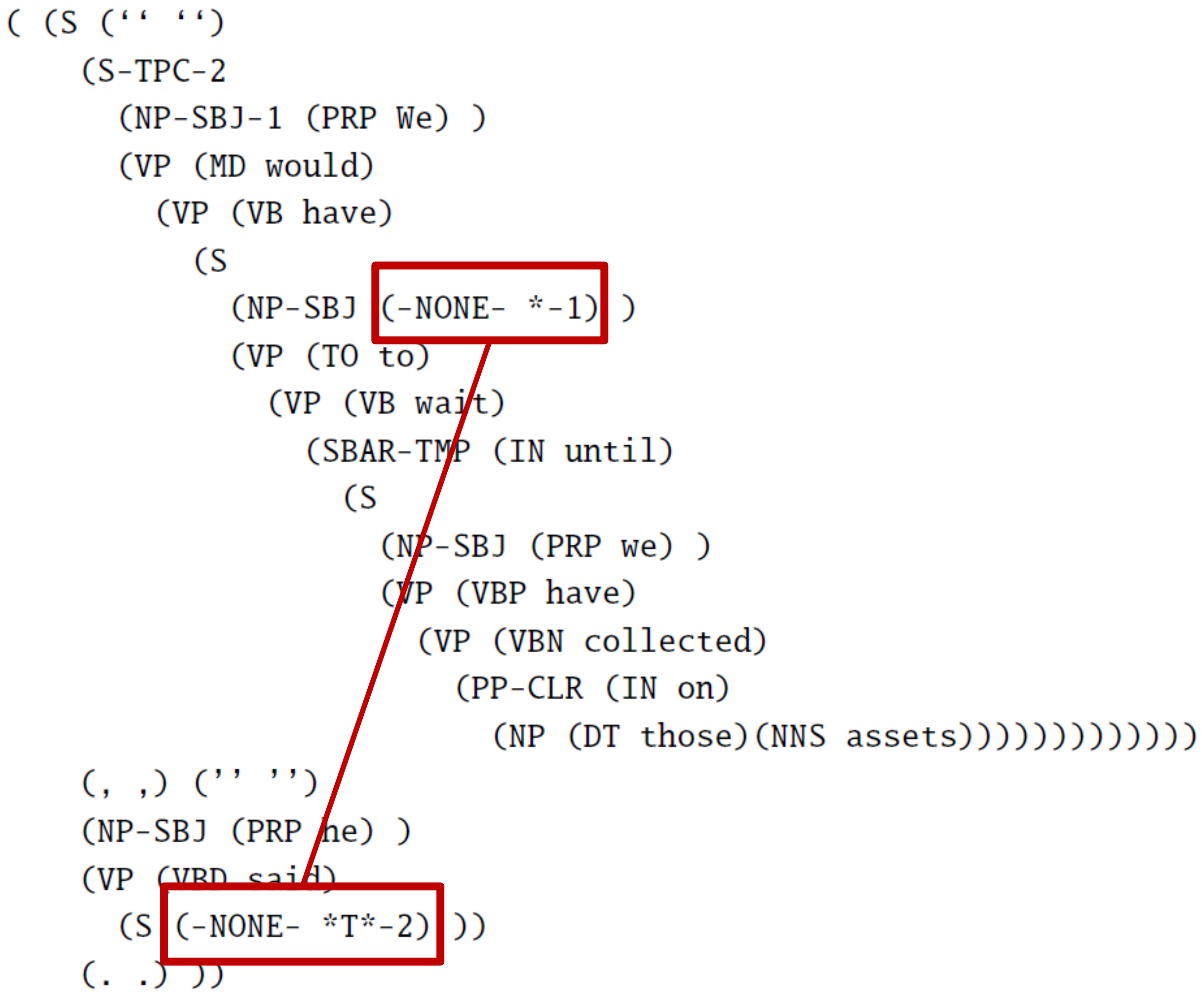
```
((S
   (NP-SBJ (DT That)
     (JJ cold) (, ,)
     (JJ empty) (NN sky) )
   (VP (VBD was)
     (ADJP-PRD (JJ full)
       (PP (IN of)
         (NP (NN fire)
           (CC and)
           (NN light) ))))
   (. .) ))
```



```
((S
   (NP-SBJ The/DT flight/NN )
   (VP should/MD
     (VP arrive/VB
       (PP-TMP at/IN
         (NP eleven/CD a.m/RB ))
       (NP-TMP tomorrow/NN )))))
```

Treebanks and Head Finding

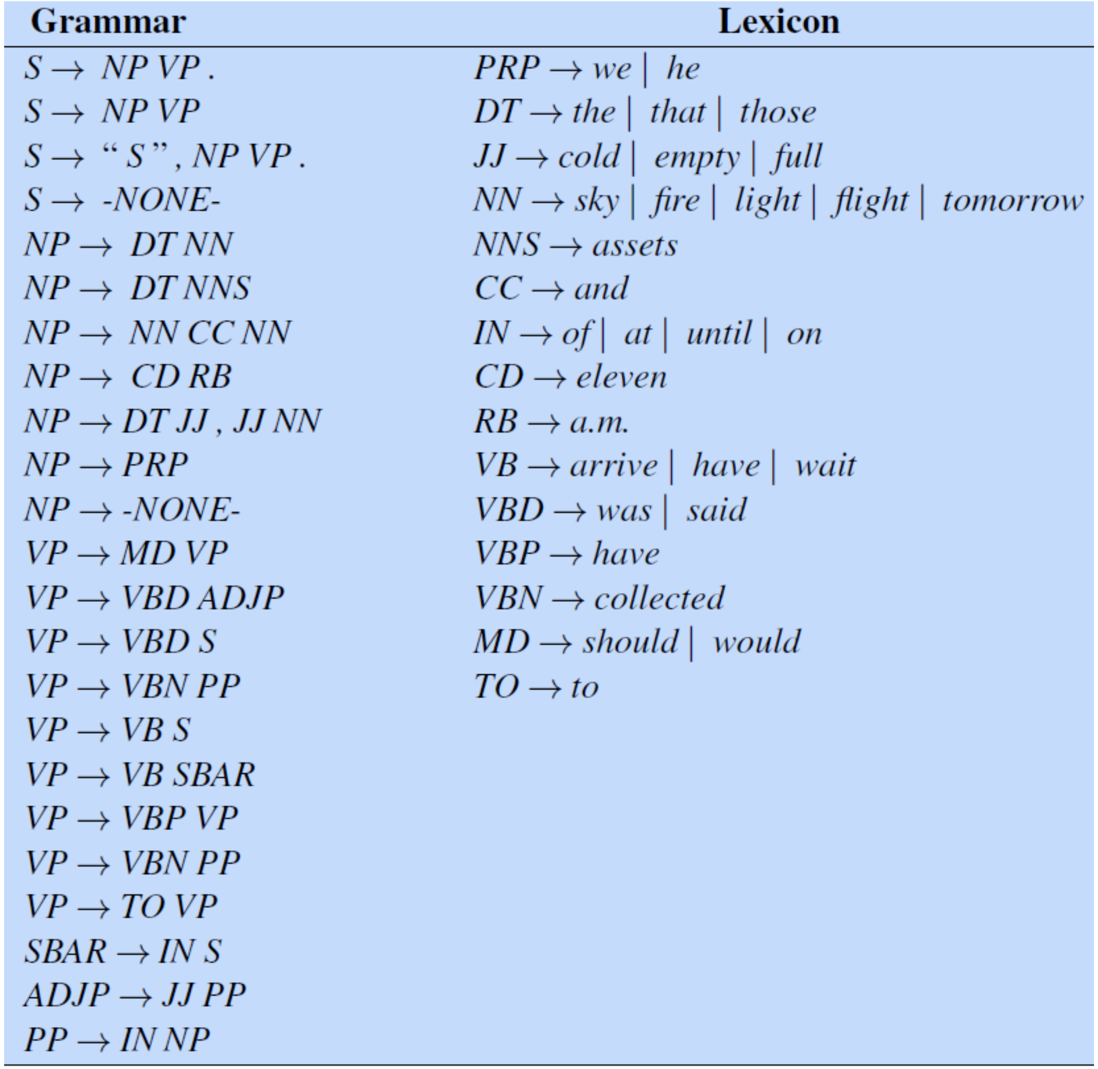
- Long-distant dependencies (syntactic movement) are encoded using -NONE- markers



```
( (S (‘‘ ‘‘)
    (S-TPC-2
      (NP-SBJ-1 (PRP We) )
      (VP (MD would)
        (VP (VB have)
          (S
            (NP-SBJ (-NONE- *-1) )
            (VP (TO to)
              (VP (VB wait)
                (SBAR-TMP (IN until)
                  (S
                    (NP-SBJ (PRP we) )
                    (VP (VBP have)
                      (VP (VBN collected)
                        (PP-CLR (IN on)
                          (NP (DT those)(NNS assets)))))))))))))
    (, ,) (’’ ’’)
    (NP-SBJ (PRP he) )
    (VP (VBD said)
      (S (-NONE- *T*-2) ))
    (. .) ))
```

Treebanks and Head Finding

- Treebanks implicitly encode a grammar
- Treebank 3 has about 17,500 distinct rule types and a million words
- This presents problems for probabilistic parsing algorithms



Grammar	Lexicon
S → NP VP .	PRP → we | he
S → NP VP	DT → the | that | those
S → " S " , NP VP .	JJ → cold | empty | full
S → -NONE-	NN → sky | fire | light | flight | tomorrow
NP → DT NN	NNS → assets
NP → DT NNS	CC → and
NP → NN CC NN	IN → of | at | until | on
NP → CD RB	CD → eleven
NP → DT JJ , JJ NN	RB → a.m.
NP → PRP	VB → arrive | have | wait
NP → -NONE-	VBD → was | said
VP → MD VP	VBP → have
VP → VBD ADJP	VBN → collected
VP → VBD S	MD → should | would
VP → VBN PP	TO → to
VP → VB S	
VP → VB SBAR	
VP → VBP VP	
VP → VBN PP	
VP → TO VP	
SBAR → IN S	
ADJP → JJ PP	
PP → IN NP	

Grammar constructed from previous two examples only

Treebanks and Head Finding

- Lexical head is the word in a phrase which is grammatically most important
- Head words are tricky to define for many phrases
- Many systems use handwritten rules to automatically select headwords from a treebank, guided by statistical analysis of the treebank

- Further reading: Collins, M. (2003). Head-Driven Statistical Models for Natural Language Parsing, Computational Linguistics 2003

Lexicalized Grammars

- Some grammars emphasize lexical features over phrase-structure
- Combinatory Categorial Grammar (CCG)
  - Set of categories
  - Mapping from lexicon words to categories
  - Set of composition rules for categories (forward and backward)
  - CCG allows both left to right AND word-by-word composition, which mirrors human language processing and is quite powerful
- Similar to phrase-structure grammars, CCG approaches are trained from annotated CCG Treebanks
  - CCGBank is the largest CCG Treebank
  - Demo http://groups.inf.ed.ac.uk/ccg/ccgbank.html
  - Dataset https://catalog.ldc.upenn.edu/LDC2005T13

Required Reading

- Constituency Grammars
  - Jurafsky and Martin, Speech and Language Processing, 3rd edition (online) >> chapter 12

Questions

- Panopto Quiz - 1 minute brainstorm for interactive questions

  Please write down in Panopto quiz in **1 minute** two or three questions that you would like to have answered at the next interactive session.

  Do it **right now** while its fresh.

  Take a screen shot of your questions and **bring them with you** at the interactive session so you have something to ask.