

Context-free Grammars and Syntax

CS-585

Natural Language Processing

Derrick Higgins

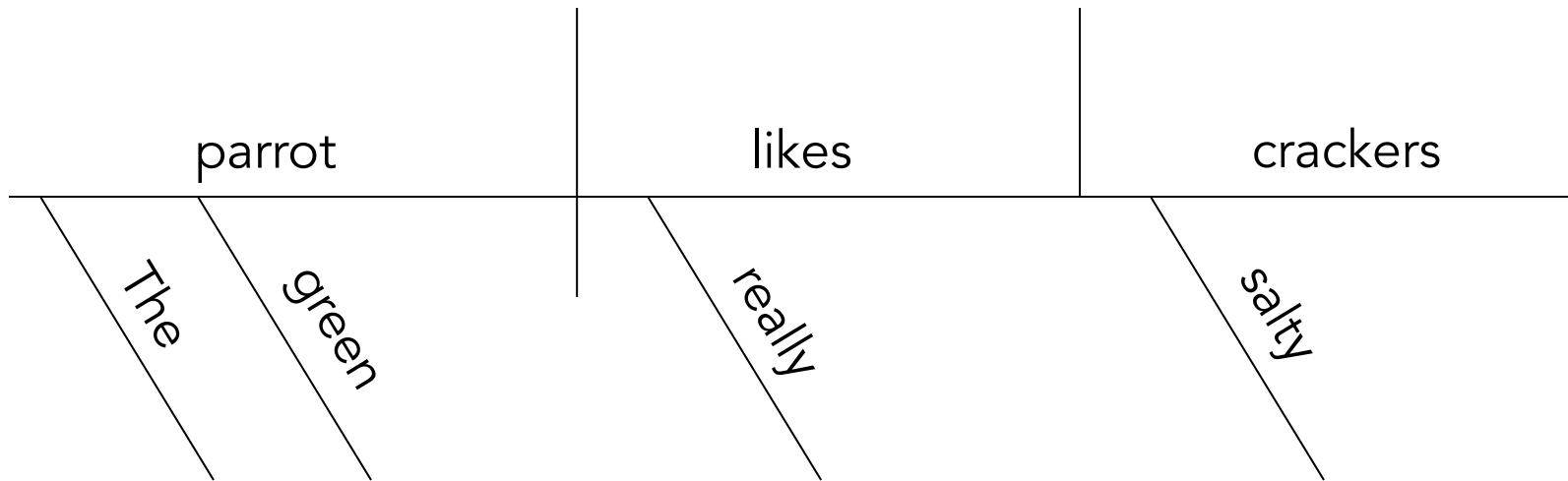
GRAMMATICAL REPRESENTATIONS

Grammatical representations

- Grammatical study has been going on for centuries
 - See the 4th-century grammarian Panini
- But it's only been in the last couple of centuries that people have been representing linguistic structure using trees or other hierarchical representations

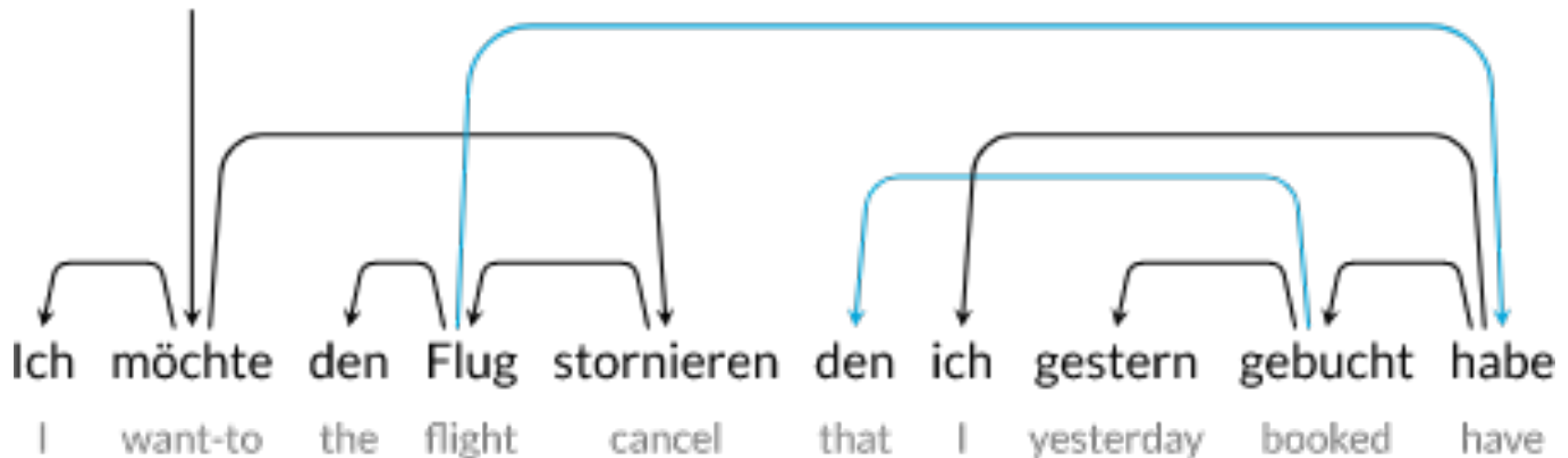
Sentence diagrams

Reed & Kellogg. 1877. Higher Lessons in English.



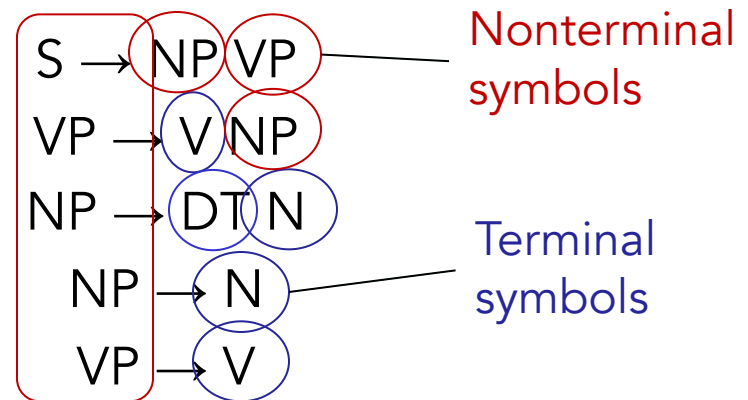
Dependency graphs

Tesnière, Lucien, 1959. Éléments de syntaxe structurale.



Generative Grammar and Phrase Structure

- Chomsky, Noam. 1957.
Syntactic structures
 - Treat human languages such as English and Sankrit like formal languages such as predicate logic
 - A *generative* process that determines admissible sequences of symbols (and by exclusion, inadmissible sequences)
 - Languages defined by *rewrite rules* that indicate options for expressing sentence parts:



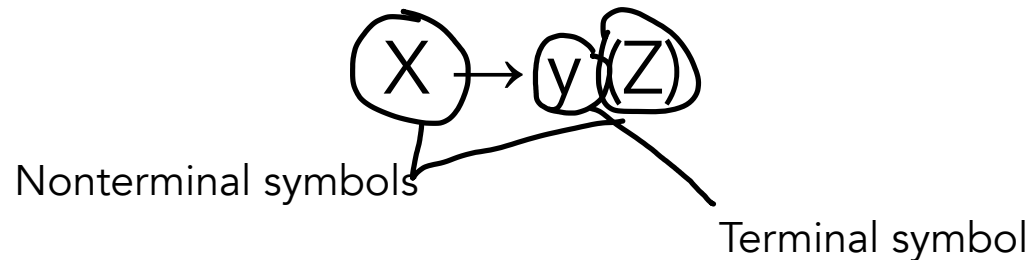
Lexical symbols

V → saw
V → heard
DT → the
DT → a
N → cat
N → dog

GRAMMAR TYPES

Regular (Finite-State) Grammars

- Rewrite rules of form



- "Root" symbol S
- Same class as regular expressions
- Limited expressivity – for example, incapable of capturing a language such as

$$A^n B^n$$

Context-free Grammars

- Rewrite rules of form

$$X \rightarrow y Z$$
$$X \rightarrow y$$
$$X \rightarrow Y Z$$

Terminal symbols $[x, y, z, \dots]$

Nonterminal symbols $[X, Y, Z, \dots]$

- For example:

$$S \rightarrow NP VP$$
$$NP \rightarrow DT N$$
$$DT \rightarrow \text{the}$$
$$N \rightarrow \text{dog}$$

...

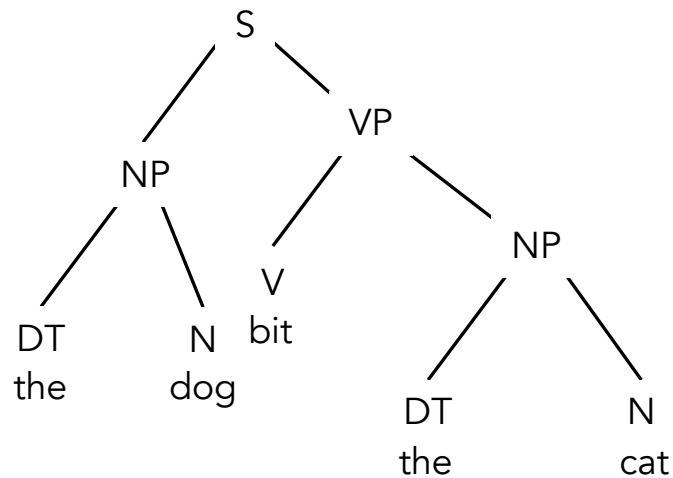
Context-Free Grammars

A sentence is generated by a series of rewrite operations

S
NP VP
DT N VP
DT N V NP
DT N V DT N
the N V DT N
the dog V DT N
the dog bit DT N
the dog bit the N
the dog bit the cat

$S \rightarrow NP VP$
 $NP \rightarrow DT N$
 $VP \rightarrow V NP$
 $NP \rightarrow DT N$
 $DT \rightarrow the$
 $N \rightarrow dog$
 $V \rightarrow bit$
 $DT \rightarrow the$
 $N \rightarrow cat$

Context-Free Grammars



$S \rightarrow NP VP$

$NP \rightarrow DT N$

$VP \rightarrow V NP$

$NP \rightarrow DT N$

$DT \rightarrow the$

$N \rightarrow dog$

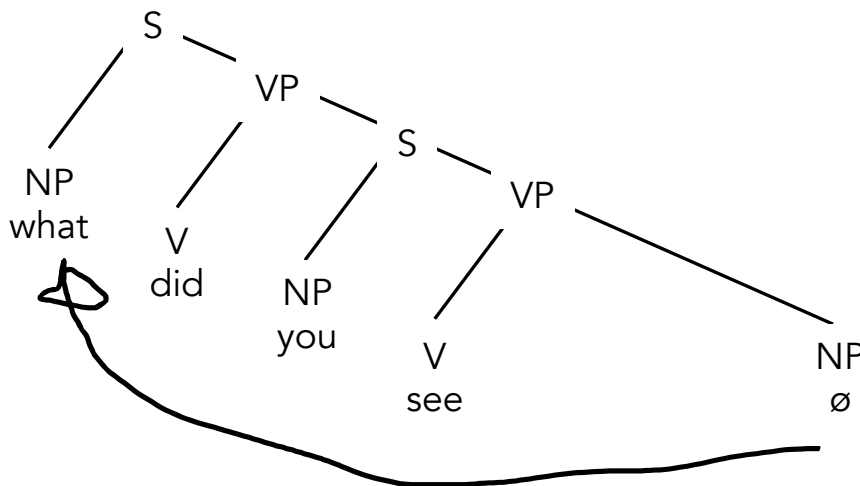
$V \rightarrow bit$

$DT \rightarrow the$

$N \rightarrow cat$

Transformational Grammars

- After rewrite rules apply to create a syntactic tree, transformation operations sensitive to tree structure can further modify it
- Motivated by phenomena such as “wh-extraction”
 - Grammar is simplified if we assume the existence of a “missing” noun phrase



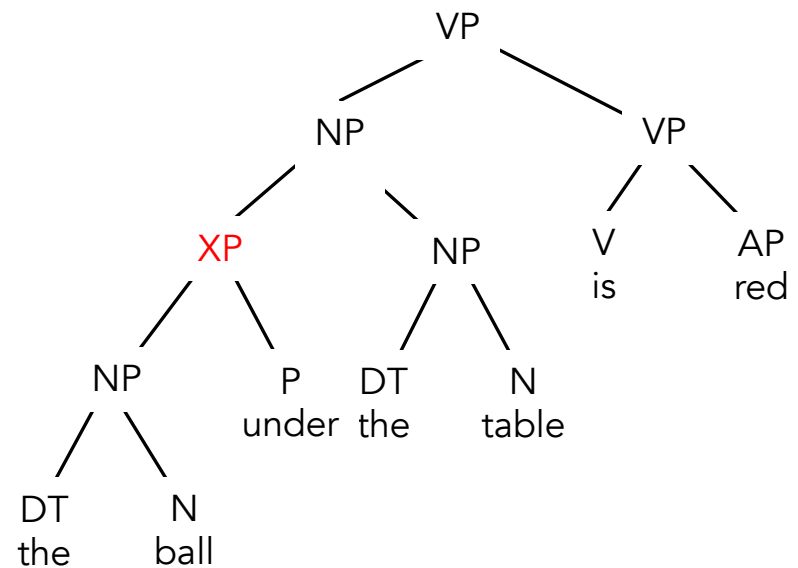
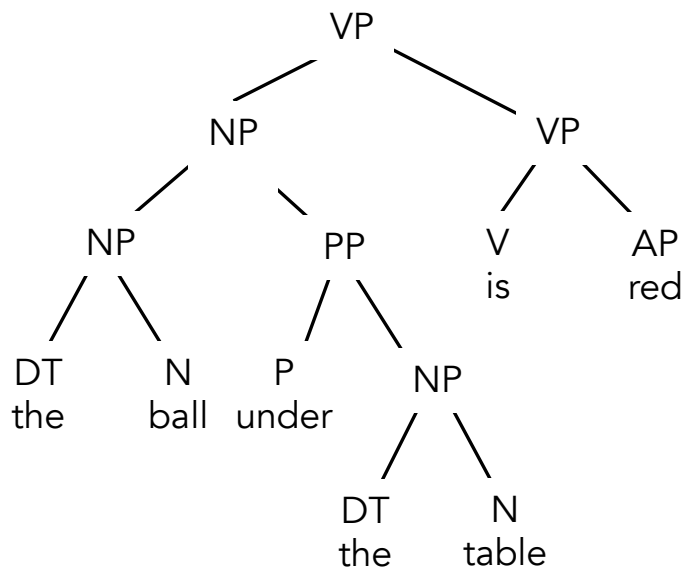
Grammatical Representations for NLP

- Transformations are powerful and expressive, but computationally intractable, and not a helpful static representation of structure
- So almost all NLP work on syntax is done in terms of context-free phrase structure representations
 - Widely used resources such as the Penn Treebank for training and evaluating models
 - Well-established methodologies for CFG parsing

CONSTITUENCY AND PHRASES

Constituents

- There are many ways we could potentially break down a sentence into grammatical units (phrases or *constituents*)



- What is the “correct” way?

Constituency Tests

Conjunction: Only constituents can be conjoined.

The ball [under the table] and [near the door] is red.

** [The ball under] and [the truck on] the table are red.*

Furthermore, conjoined phrases must be of the same category

** The ball [under the table] and [that has my initials on it] is red.*

Constituency Tests

- Clefting: Only constituents can be extracted into a cleft construction.

Where I saw the the ball is [under the table].

** What/where/how I saw the table is [the ball under].*

Constituency Tests

- Fragments: Only constituents can stand alone as incomplete sentence utterances

I saw the ball under the table.

Where?

[Under the table]

I saw the ball under the table.

What/how/where?

** [The ball under]*

Phrase Types

- Noun phrases
 - Headed by a noun
 - May start with a determiner, include modifying adjectives, prepositional phrases and postnominal relative clauses
 - Serves as arguments of prepositions or verbs, or as sentential subject

[_{NP} The fellow I met last week]

[_{NP} The horse with the bit in its teeth]

[_{NP} Four quarters]

[_{NP} We]

Phrase Types

- Verb phrases
 - Headed by a verb
 - Auxiliary verbs head VPs with recursive structure
 - May include adverbial or PP modifiers
 - Includes verb object arguments, but not sentence subject

[_{VP} Finally took a week-long vacation]

[_{VP} Saw the moon with a telescope]

[_{VP} Will [_{VP} have [_{VP} been [_{VP} winning]]]]

[_{VP} Left]

Phrase Types

- Prepositional phrases
 - Headed by a preposition
 - Include noun phrase object
 - Often adjoined to VPs or NPs; may also be verbal complements

[_{PP} On the moon]

[_{PP} Between here and eternity]

[_{PP} One week ago]

Phrase Types

- Adjective phrases
 - Headed by an adjective
 - Various complex adjectival structures

[_{AP} Red and gold]

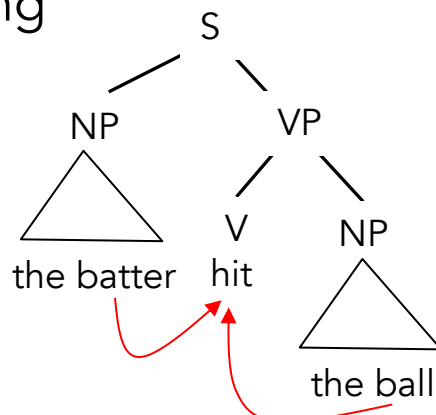
[_{AP} Certain to succeed]

[_{AP} Bigger than a breadbox]

[_{VP} Happy as a clam]

Phrasal Heads

- The *head* of a phrase is the a word that determines its attributes
 - Typically of the same category as the phrase: the head of a noun phrase is a noun, the head of a prepositional phrase is a preposition, etc.
 - Attributes of the head (e.g., tense in the case of verbs, number and case in the case of nouns) are shared by the phrase as a whole
 - Relationships between heads of phrases are strongly predictive for parsing



SYNTACTIC STRUCTURE

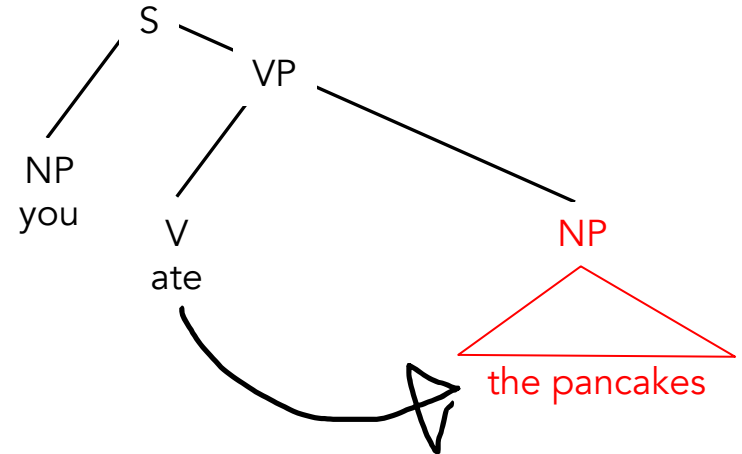
Complements and Adjuncts

- Complement: Syntactic elements that are required in a particular context
 - Category determined by a specific lexical item in the phrase
 - Relationship to other syntactic units may be expressed via case
 - Limited number of slots to fill
- Adjunct: Optional elements added to phrases of a particular category
 - Largely independent of the lexical head of the phrase
 - Possibly more than one

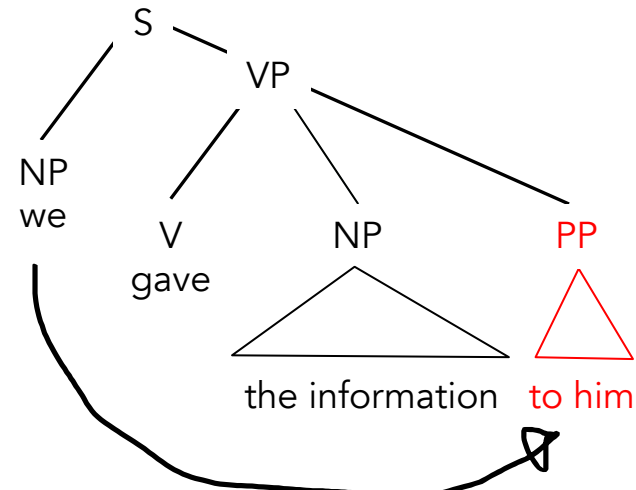
Complements

For example

- Direct objects



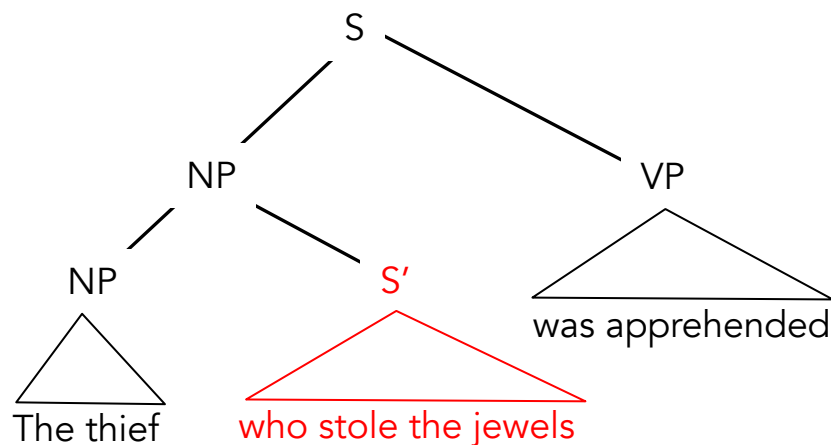
- Prepositional phrase complements of ditransitive verbs



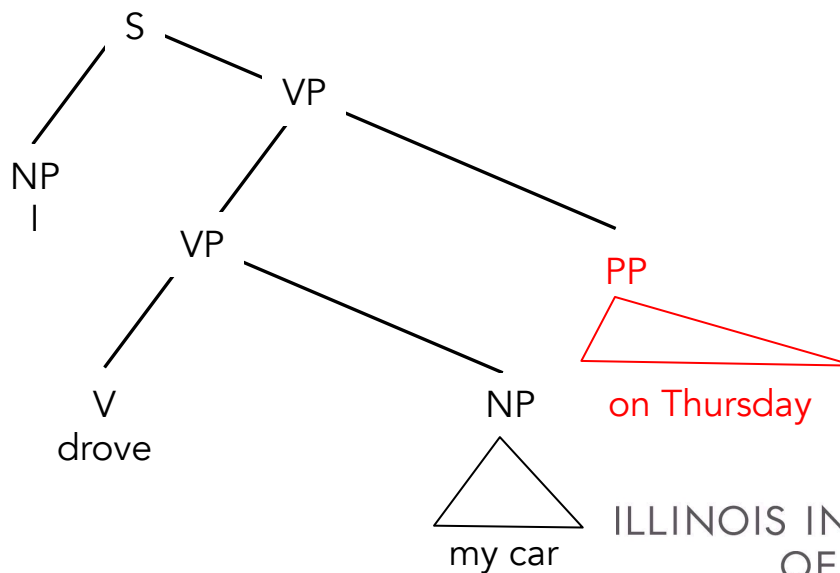
Adjuncts

For example

- Relative clauses

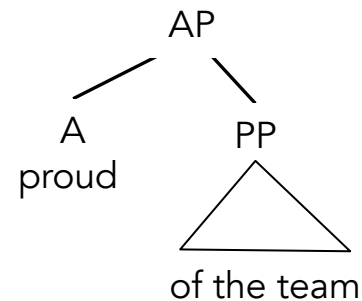
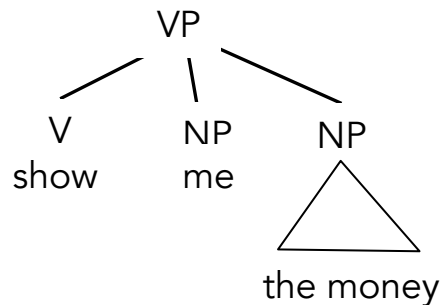
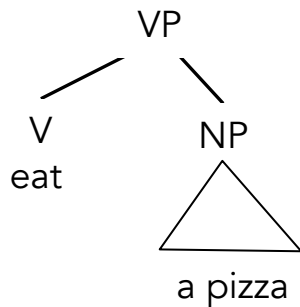


- Adverbial prepositional phrases



Subcategorization

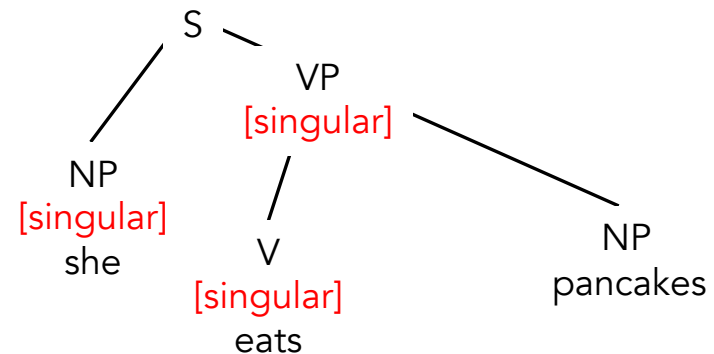
- Subcategorization is the relationship between a syntactic head word and the dependents it requires
 - A transitive verb like “eat” subcategorizes for a single noun phrase
 - A ditransitive verb like “show” subcategorizes for two noun phrases
 - The adjective “proud” subcategorizes for a prepositional phrases headed by “of”



Agreement

- *Agreement* phenomena: words in a phrase that take different forms based on the number, case, person, etc. of a noun
- Motivates feature-based syntactic representations

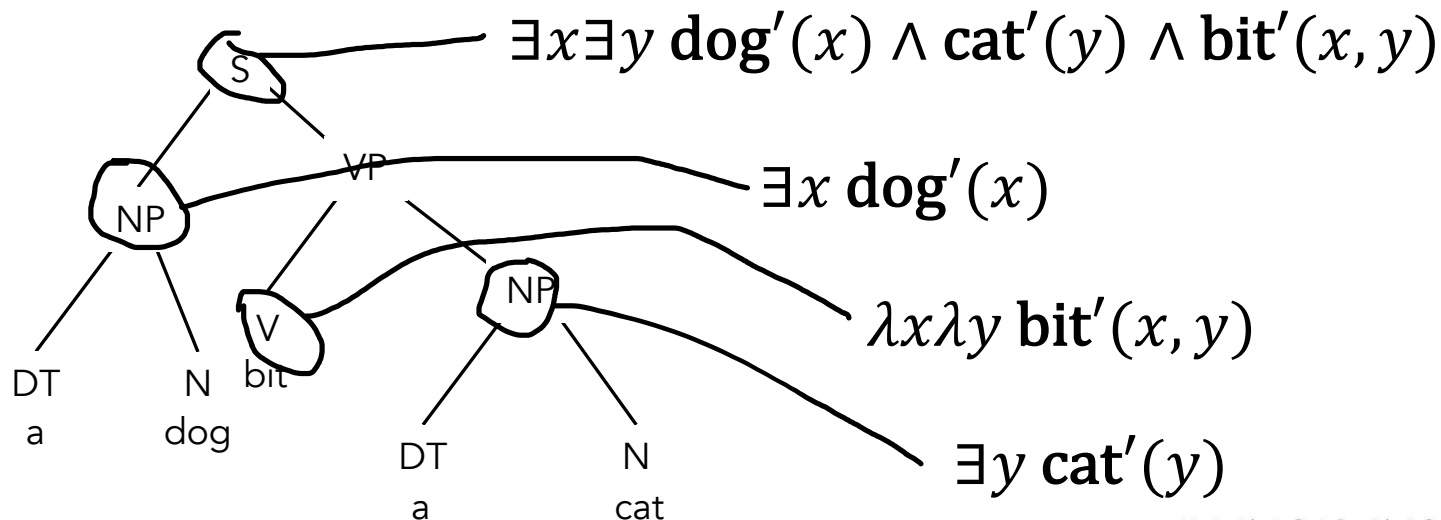
$S \rightarrow NP[\alpha \text{ number}] VP[\alpha \text{ number}]$
 $VP[\alpha \text{ number}] \rightarrow V[\alpha \text{ number}] NP$



SYNTAX AND SEMANTICS

Compositionality

- Compositionality is that semantic structure should mirror syntactic structure
 - Each phrase has a meaning
 - The meaning of larger units is a function of the meaning of smaller units and the way in which they are combined



Remember these phrase-level sentiment predictions?

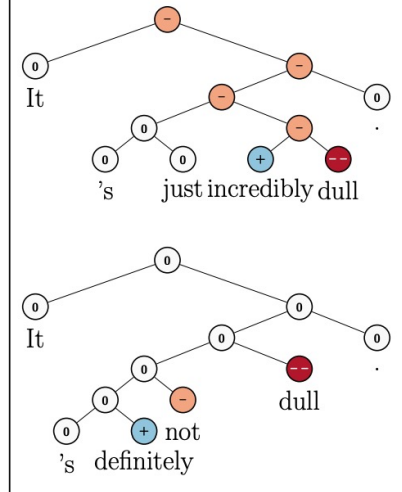
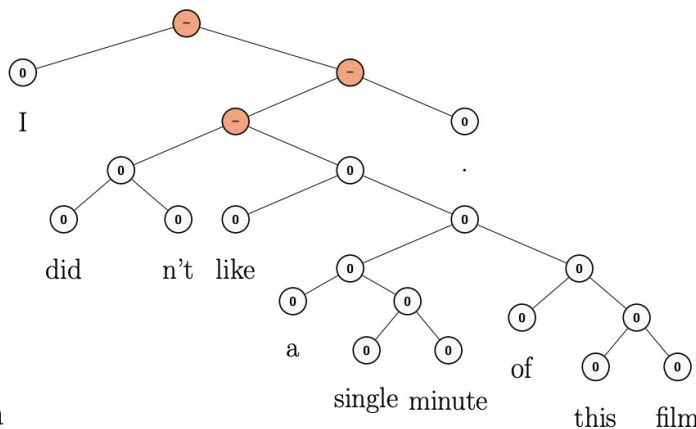
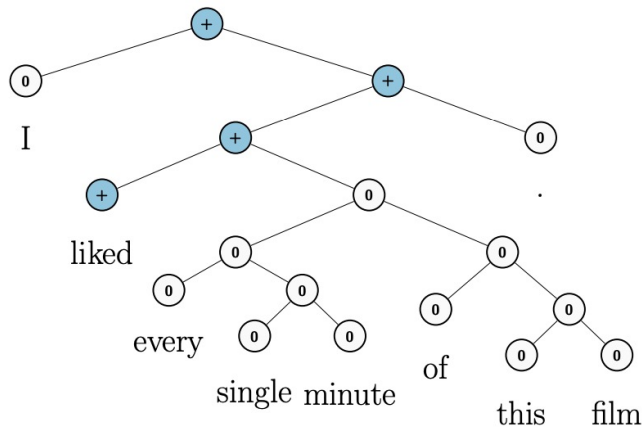
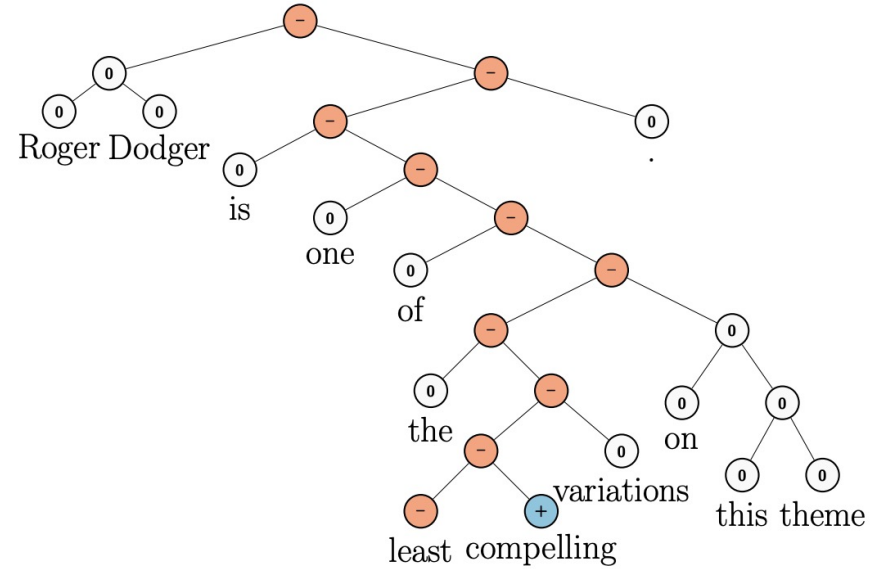
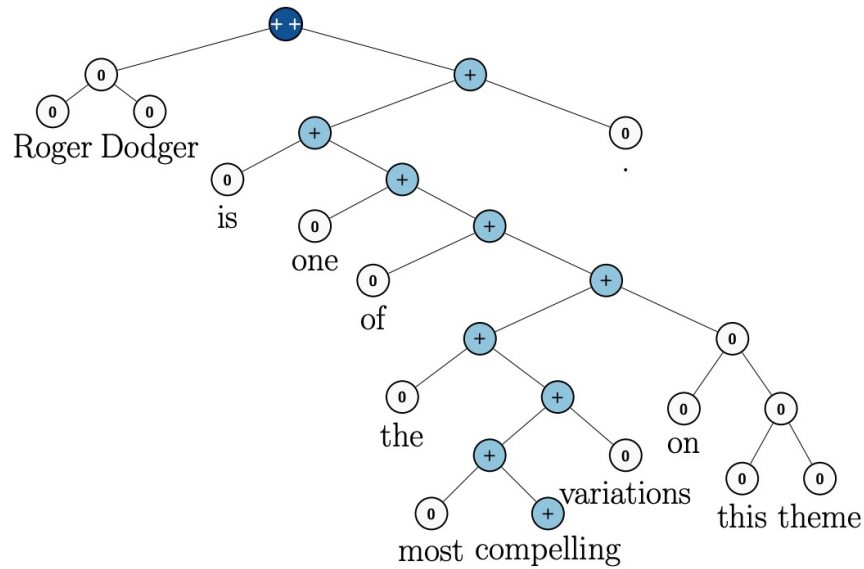
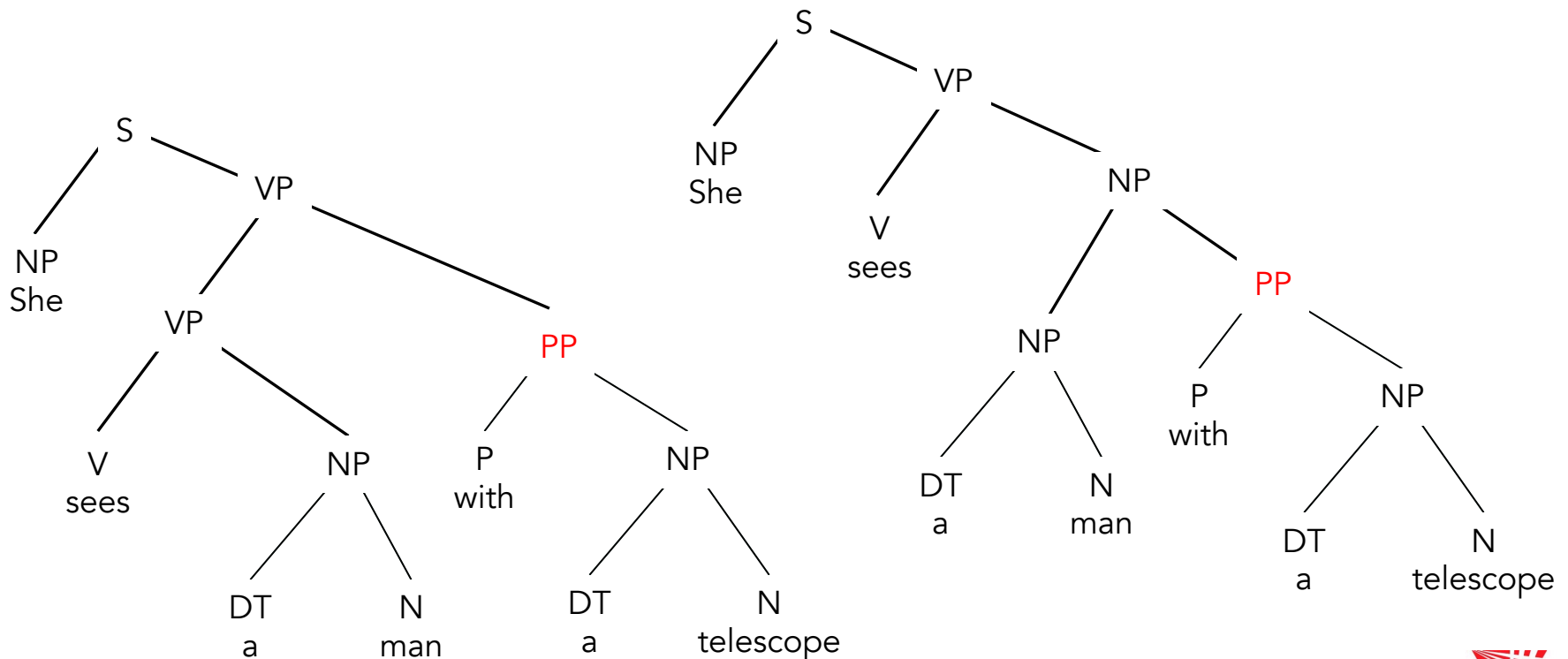


Figure 9: RNTN prediction of positive and negative (bottom right) sentences and their negation.

Structural Ambiguity

- Related to compositionality – sometimes meaning differences can be traced to differences in syntactic structures



TREE REPRESENTATIONS FOR NLP

Treebanks

- A treebank is a set of linguistic tree structures that humans have assigned to texts following a set of annotation guidelines
- These can be used for training and evaluation of syntactic parsing models
- The best-known treebank is the Penn Treebank
 - About 7 million words from newswire text, and assorted fiction and nonfiction genres
- There are others
 - Stuttgart TIGER treebank of German
 - Penn Chinese Treebank
 - Penn Arabic Treebank
 - UAM Treebank of Spanish

Penn Treebank Bracketing

1.2.2 SINV

The SINV label is used for subject-auxiliary inversion in the case of negative inversion, conditional inversion, locative inversion, and some topicalizations. (SINV is not used with questions. See section 1.2.6 and section 1.2.5 for the treatment of subject-auxiliary inversion in the case of yes/no questions and *wh*-questions, respectively.) Inverted auxiliaries are unlabeled.

```
(SINV (ADVP-TMP Never)
      had
      (NP-SBJ I)
      (VP seen
        (NP such a place)))
```

When the inversion results in a conditional clause (i.e., when it is equivalent to (SBAR-ADV if...), the SINV is enclosed in SBAR-ADV).

```
(S (SBAR-ADV (SINV had
              (NP-SBJ Casey)
              (VP thrown
                (NP the ball)
                (ADVP-MNR harder))))
    ,
    (NP-SBJ it)
    (VP would
      (VP have
        (VP reached
          (NP home plate)
          (PP-TMP in
            (NP time)))))))
```

318 pages of instructions!

Treebank Bracketing

- Diagrammatic trees are nice graphical representations of syntactic structure, but we need computer-readable representations
- The typical convention is to
 - Show constituent structure as nested brackets, with spaces inserted between adjacent constituents
 - Label the opening bracket of a constituent with its category

Treebank Bracketing

```
(S (NP she)
  (VP (VP (V sees)
    (NP (DT a)
      (N man)
    )
  )
  (PP (P with)
    (NP (DT a)
      (N telescope)
    )
  )
)
)
```

