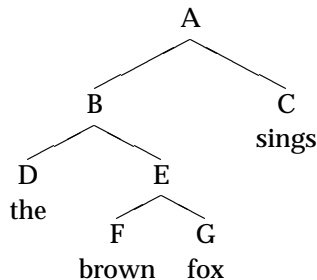


## Phrase Structure Rules Structure within the NP

### 1 Definitions

- (1) a tree for 'the brown fox sings'



- Linguistic trees have **nodes**. The nodes in (1) are A, B, C, D, E, F, and G.
- There are two kinds of nodes: internal nodes and terminal nodes. The internal nodes in (1) are A, B, and E. The terminal nodes are C, D, F, and G. Terminal nodes are so called because they are not expanded into anything further. The tree ends there. Terminal nodes are also called **leaf** nodes. The leaves of (1) are really the words that constitute the sentence 'the brown fox sings' i.e. 'the', 'brown', 'fox', and 'sings'.

- (2) a. A set of nodes form a **constituent** iff they are exhaustively dominated by a common node.
- b. X is a **constituent** of Y iff X is dominated by Y.
- c. X is an **immediate constituent** of Y iff X is immediately dominated by Y.

Notions such as subject, object, prepositional object etc. can be defined structurally. So a subject is the NP immediately dominated by S and an object is an NP immediately dominated by VP etc.

- (3) a. If a node X immediately dominates a node Y, then X is the **mother** of Y, and Y is the **daughter** of X.
- b. A set of nodes are sisters if they are all immediately dominated by the same (mother) node.

We can now define a host of relationships on trees - grandmother, granddaughter, descendant, ancestor etc.

We have seen earlier that c-command is one relationship on trees that is quite useful. It can now be defined in purely structural terms.

- (4) A c-commands B if and only if A does not dominate B and the node that immediately dominates A dominates B.

- **CONDITION C**

(5) A pronoun cannot refer to a proper name it c-commands.

Note that Condition C is a negative condition. It never tells you what a particular pronoun must refer to. It only tells you what it cannot refer to.

In general, if a pronoun cannot refer to a proper name (despite agreeing in gender and number), you can conclude that the pronoun c-commands the proper name.

- **The NO CROSSING BRANCHES CONSTRAINT**

(6) If one node X precedes another node Y, then all descendants of X must also precede Y and all descendants of Y.

## 2 How to grow trees

Where do the trees that we use to analyze linguistic structure come from?

In a way, they are just representations of facts that exist out in the world - the facts that we can discover using constituency test. So one way to make trees is by doing empirical work - taking a sentence, applying various constituency tests to the words in the sentence, and then drawing a tree based on the results of our tests.

This empirical method is ultimately the only correct way to deduce 'tree structure'. However, in most cases, we can simplify things considerably by using **Phrase Structure Rules**.

**Phrase Structure Rules** are rules of the sort

$X \rightarrow YZ$

This rule says 'take the node X and expand it into the nodes Y and Z'. Alternately, going from right to left (or from below), it says 'if you have a Y and a Z next to each other, you can combine them to make an X'.<sup>1</sup>

Phrase structure rules can be categorial i.e. rules that expand categories into other categories, or they can also be lexical i.e. rules that expand category labels by word (lexical items).<sup>2</sup>

- A grammar can then be thought of as a set of phrase structure rules (categorial rules plus lexical rules).

The categorial rules can be thought of as (part of) the syntax and the lexical rules as (part of) the lexicon.

### 2.1 Some Phrase Structure Rules for English

(7) **Categorial Rules**

a.  $S \rightarrow NP \text{ Modal VP}$

b.  $VP \rightarrow V \text{ AP PP}$

c.  $AP \rightarrow \text{ADVP A}$

d.  $\text{ADVP} \rightarrow \text{ADV}$

e.  $PP \rightarrow P \text{ NP}$

---

<sup>1</sup>Such phrase structure rules are called Context Free Grammars (CFG) and were invented by Noam Chomsky in 1956. A closely related model was used by Pāṇini to describe the grammar of Sanskrit in around 500 B.C.

<sup>2</sup>Why must we have at least some lexical rules?

f.  $NP \rightarrow D N$

(8) Lexical Rules

- a.  $N \rightarrow \text{girl}$
- b.  $N \rightarrow \text{boy}$
- c.  $Adv \rightarrow \text{incredibly}$
- d.  $A \rightarrow \text{conceited}$
- e.  $V \rightarrow \text{seem}$
- f.  $Modal \rightarrow \text{must}$
- g.  $P \rightarrow \text{to}$
- h.  $D \rightarrow \text{that}$
- i.  $D \rightarrow \text{this}$

Some sentences these rules will **generate**:

- (9)
- a. This boy must seem incredibly conceited to that girl.
  - b. This boy must seem incredibly conceited to this girl.
  - c. This boy must seem incredibly conceited to that boy.
  - d. This boy must seem incredibly conceited to this boy.
  - e. This girl must seem incredibly conceited to that girl.
  - f. This girl must seem incredibly conceited to this girl.
  - g. This girl must seem incredibly conceited to that boy.
  - h. This girl must seem incredibly conceited to this boy.

How many more sentences will these rules generate?

• Optional constituents

How do we handle cases like:

- (10) This boy must seem incredibly stupid.

## 2.2 Introducing infinity

We know that human languages can contain sentences of arbitrary length. Consider (11) which stands for an infinite number of sentences.

- (11) He believes that he believes that he believes that he believes that . . . he ate pizza.

So if all of human language is to be generated by a set of phrase structure rules, the relevant set of phrase structure rules should generate an infinite number of sentences.

How can that be done?

Let us try to analyze (11), starting with a more manageable (12).

- (12) He believes that he ate pizza.

We start with the following categorial rules:

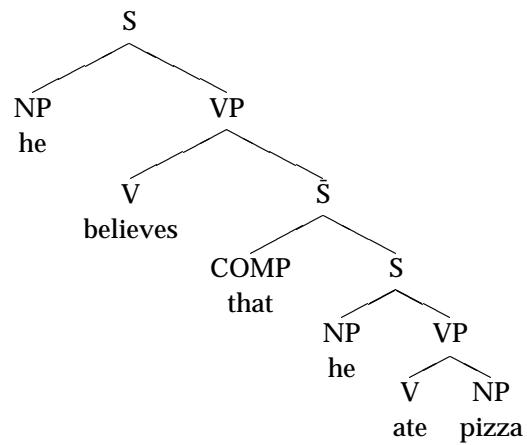
- (13) a.  $S \rightarrow NP VP$   
 b.  $VP \rightarrow V \bar{S}$   
 c.  $\bar{S} \rightarrow COMP S$   
 d.  $VP \rightarrow V NP$

We need the following lexical rules:

- (14) a.  $NP \rightarrow he$   
 b.  $NP \rightarrow pizza$   
 c.  $V \rightarrow ate$   
 d.  $V \rightarrow believes$   
 e.  $COMP \rightarrow that$

Now we can generate (12). This is shown in (15).

(15)



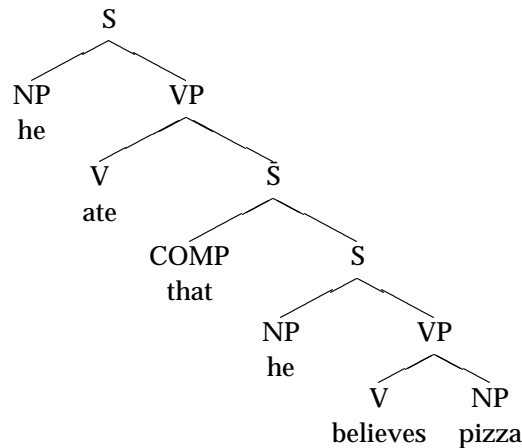
But is (12) all that the rules in (13) and (14) will generate?

How many sentences will (13) and (14) generate?

### 2.2.1 Overgeneration

The rules in (13) and (14) will also generate sentences (see the structure below) like:

- (16) \*He ate that he believes pizza.



How can we constrain phrase structure rules so that such overgeneration does not take place?

### 3 Noun Phrases

So far, we have seen two kinds of categories:

word-level categories such as N, V, A, P etc. (somewhat imprecisely, words) and

phrase-level categories such as NP, VP, AP, PP etc. (somewhat imprecisely, sequences of words which can 'stand on their own').

We will now investigate if these two kinds of categories are all we need a third category which lies in between words and full phrases.

Consider the following NP:

(17) the king of England

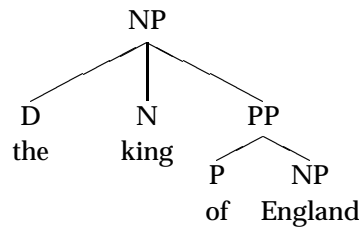
We feel quite confident saying that 'the king of England' is an NP. What else can we say about its structure?

There seems to be a lot of evidence that **of England** is a PP. It can be co-ordinated, shared in **shared constituent** co-ordination. It can also function as a sentence fragment and be preposed.

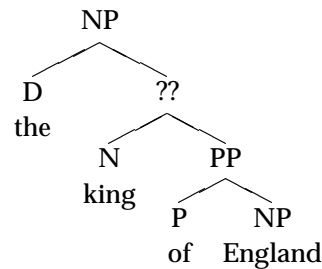
- (18) a. the king [<sub>PP</sub> of England] and [<sub>PP</sub> of the empire]. (coordination)  
 b. He is the king, and she is the queen, [<sub>PP</sub> of England]. (shared constituent coordination)  
 c. A: Was he the king of Livonia?  
 B: No, [<sub>PP</sub> of England]. (sentence fragment)  
 d. [<sub>PP</sub> Of which country] was he the king?

At this point we have two options:

(19)



(20)



There is evidence from constituency tests that the sequence of words ‘king of England’ forms a constituent.

- ‘king of England’ can undergo co-ordination with another similar sequence.

(21) Vivian dared defy the [king of England] and [ruler of the Empire]?

- ‘king of England’ can serve as the **shared constituent** in **shared constituent co-ordination**.

(22) Edward was the last, and some people say the best, [king of England].

- There is a proform that replaces sequences like ‘king of England’.

(23) The present [king of England] is more popular than the last **one**.

So ‘king of England’ forms a constituent that excludes *the*. Thus we have evidence for the tree in (20).

This evidence doesn’t actually rule out the tree in (19). It is not easy to rule out (19) on the basis of the discussion so far. However, an assumption that natural language structures only involve binary branching could be used to block structures like (19).

### 3.1 What kind of constituent is ‘king of England’?

In other words, what is the name of the node labeled ?? in (20)?

Let us assume that it is an NP.

We find that this assumption is problematic in many ways.

- ‘king of England’ does not have the distribution of ‘normal’/‘full’ noun phrases. Normal NPs can occur in subject position, in object position, and as a prepositional object. ‘king of England’ cannot appear in any of these positions.

(24) a. subject:

- i. [The king of England] invaded several countries.
- ii. \* [King of England] invaded several countries.

b. object:

- i. I saw [the king of England] on the T yesterday.
- ii. \* I saw [king of England] on the T yesterday.

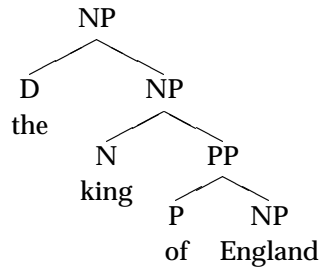
c. prepositional object:

- i. I didn’t give any money to [the king of England].

ii. \* I didn't give any money to [king of England].

- Consider the tree for 'the king of England' under the assumption that 'king of England' is also an NP.

(25)



From this tree, we can read of the phrase structure rules involved in building it. They are shown in (26).

(26) a. Categorical Rules:

- i. NP → D NP
- ii. NP → N PP
- iii. PP → P NP

b. Lexical Rules:

- i. D → the
- ii. N → king
- iii. P → of
- iv. NP → England

Note in particular the categorical rule (26a.i). It has the unusual property that it expands a node label into itself. Such rules are called **recursive** and this phenomena is called **recursion**.<sup>3</sup> So we can go from NP to [D NP] to [D D NP] to [D D D NP] and so on. In principle, using the rules in (26), we can generate NPs like those in (27).

- (27) a. \* the the king of England  
 b. \* the the the king of England  
 c. \* the . . . the the the king of England

Now, it is very clear that none of the NPs in (27) are good noun phrases in English. From this we can conclude that the categorical rule (26a.i), which is the source of the recursion, cannot be correct.

So:

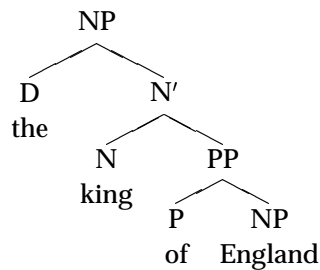
- 'king of England' cannot be an NP
- and yet
- 'king of England' is a constituent of some sort.

Let us call nominal constituents that are bigger than words but still not full phrases  $\bar{N}$  (or **N'** or **N-bar**).

Our tree now becomes:

<sup>3</sup>We have already seen a case of recursion, though there the recursion was in two steps.

(28)



NPs are sometimes called N-double bars or  $N''$ . N are sometimes called  $N^0$ .

## 4 Complements and Adjuncts

Consider the phrase-structure rules responsible for generating (28) <sup>4</sup>:

- (29)
- a.  $NP \rightarrow D N'$
  - b.  $N' \rightarrow N PP$
  - c.  $PP \rightarrow P NP$

We see that D combines with an  $N'$  to its left and forms an NP. Similarly P combines with an NP to its left and forms a PP.

Likewise, (29) says that an N combines with any PP that follows it (i.e. any **postnominal** NP) and forms an  $N'$ .

But is this really the case? Do the PPs in (30a, b) have the same relation to the N?

- (30)
- a. a student [of Physics]
  - b. a student [with long hair]

It seems not. Consider the following pattern:

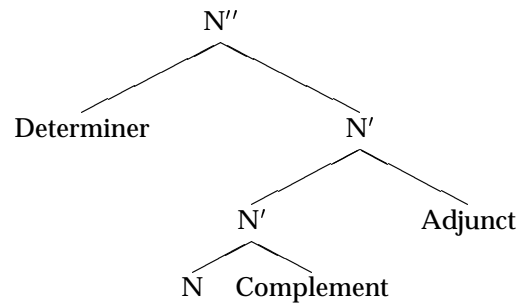
- (31)
- a. i. He is [a student of Physics].  
ii. = He is [studying Physics].
  - b. i. He is [a student with long hair].  
ii.  $\neq$  He is [studying long hair].

Pps like 'of Physics' are called **complements**, while Pps like 'with long hair' are called **adjuncts**. Corresponding to this difference in terminology, a structural difference is also proposed. This is shown in (32).

(32)

---

<sup>4</sup>From now on, we will only consider the categorial rules. The lexical rules are straightforward.



In terms of phrase structure rules this is:

- (33) a.  $N'' \rightarrow D N'$  (Determiner Rule)  
 b.  $N' \rightarrow N' PP$  (Adjunct Rule)  
 c.  $N' \rightarrow N PP$  (Complement Rule)

The rules in (33) make a prediction - if an NP contains both a complement PP and an adjunct PP, the complement PP should precede the adjunct PP. This prediction turns out to be true.

- (34) a. the student [of Physics] [with long hair]  
 b. \* the student [with long hair] [of Physics]

#### 4.1 Optional Constituents of the Noun Phrase

Do all NPs have to contain a determiner, a noun, a complement PP, and an adjunct PP?

Well, they have to contain an N, otherwise they wouldn't be NPs. What about the others?

Consider the rules in (33). If you wanted to make an NP, would it be necessary to apply the Adjunct rule?

You could take an N and a complement PP and make an  $N'$ . Then you **could** combine the  $N'$  with an adjunct PP to make another  $N'$ .

You could, but you don't have to. You can now just combine your adjunct-less  $N'$  with a D on its left to make an NP.

So, NPs don't have to contain adjuncts. In other words, the adjunct rule is an **optional** rule.

Still, the rules in (33) insist that every NP must have a determiner (the Determiner rule) and a complement PP (the Complement rule). This is, however, just false.

- (35) a. the student  
 b. the student with long hair

(35a) is an NP without a complement PP, (35a) show that an NP without a complement PP can still take an adjunct PP. How can we modify our phrase structure rules to handle these case?

For this purpose, we will introduce new terminology: **(A)** means that A is optional. So we can now change our complement rule from (36a) to (36b).

- (36) a.  $N' \rightarrow N PP$  (Old Complement Rule)  
 b.  $N' \rightarrow N (PP)$  (New Complement Rule)

We also find optionality of determiners cf. (37a-e).

- (37) a. cheese from Greece  
 b. students  
 c. students with long hair  
 d. students of physics  
 e. students of physics with long hair

However, this optionality is **lexically determined** i.e. it only works for certain nouns - noncount nouns and plural count nouns but not singular count nouns.<sup>5</sup>

- (38) a. \* Student likes pizza  
 b. \* Student with long hair likes pizza

For such nouns, we can modify the determiner rule in the way we modified the complement rule in (36).

- (39) a.  $NP \rightarrow D N'$  (Old Determiner Rule)  
 b.  $NP \rightarrow (D) N'$  (New Determiner Rule)

However, we have to think of a way to block the NPs in (38) from being generated.

## 4.2 Non-branching Phrases

Consider the new complement rule:

- (40)  $N' \rightarrow N (PP)$  (New Complement Rule)

This rule is really equivalent to the following two rules:

- (41) a.  $N' \rightarrow N PP$   
 b.  $N' \rightarrow N$

(41a) is nothing new. (41b) is definitely new. It tells us something unexpected. According to (41b), **student** in 'the student' is both an  $N'$  as well as an  $N$ , while **student in the student of Physics** is only an  $N$ , not an  $N'$ . Similarly, **student** in 'the student with long hair' should be both an  $N'$  as well as an  $N$ . We can check to see if these predictions are true.

The test we will use is substitution by the  $N'$  pro-form **one**.

- (42) a. The [student] with long hair is dating the **one** with short hair.  
 b. This [student] works harder than that **one**.  
 c. \* The [student] of chemistry was older than the **one** of Physics.

What can co-ordination tell us here?

---

<sup>5</sup>How can we be sure that the word sequences in (37) are NPs?

### 4.3 A bit more on N'

Both (43a, b) are responsible for the creation of N''s.

- (43) a.  $N' \rightarrow N' PP$  (Adjunct Rule)
- b.  $N' \rightarrow N PP$  (Complement Rule)

How can we be sure that the node created by the complement rule isn't N-bar1 and the node created by the adjunct rule N-bar2?

Again by constituency test: we know that only like categories can be co-ordinated and we find that N' created by the two different rules can be co-ordinated.

- (44) the [ [students of Chemistry with long hair] and [professors of Physics]]

In addition, the pro-N' **one** can refer to N's created by either rule.

- (45) a. Which [student of Physics]? The **one** with long hair?
- b. Which [student of Physics with long hair]? That one?

Hence we can conclude that the 'output' of both the rules is indeed one kind of node, which we call N'.