Context-Free Grammars

Carl Pollard
Ohio State University

Linguistics 680
Formal Foundations
Tuesday, November 10, 2009

These slides are available at:

http://www.ling.osu.edu/∼scott/680

(1) **Context-Free Grammars (CFGs)**

A CFG is an ordered quadruple $\langle T, N, D, P \rangle$ where

a. T is a finite set called the **terminals**;

b. N is a finite set called the **nonterminals**

c. D is a finite subset of $N \times T$ called the **lexical entries**;

d. P is a finite subset of $N \times N^+$ called the **phrase structure rules** (PSRs).

(2) **CFG Notation**

a. '$A \rightarrow t$ ' means $\langle A, t \rangle \in D$.

b. '$A \rightarrow A_0 \dots A_{n-1}$' means $\langle A, A_0 \dots A_{n-1} \rangle \in P$.

c. '$A \rightarrow \{s_0, \dots s_{n-1}\}$' abbreviates $A \rightarrow s_i$ $(i < n)$.

(3) **A 'Toy' CFG for English (1/2)**

$T = \{$Fido, Felix, Mary, barked, bit, gave, believed, heard, the, cat, dog, yesterday$\}$

$N = \{\text{S}, \text{NP}, \text{VP}, \text{TV}, \text{DTV}, \text{SV}, \text{Det}, \text{N}, \text{Adv}\}$

D consist of the following lexical entries:

NP $\rightarrow$ {Fido, Felix, Mary}

VP $\rightarrow$ barked

TV $\rightarrow$ bit

DTV $\rightarrow$ gave

SV $\rightarrow$ {believed, heard}

Det $\rightarrow$ the

N $\rightarrow$ {cat, dog}

Adv $\rightarrow$ yesterday

(4) **A 'Toy' CFG for English (2/2)**

P consists of the following PSRs:

$$\mathrm{S} \rightarrow \mathrm{NP\ VP}$$

$$\mathrm{VP} \rightarrow \{\mathrm{TV\ NP},\ \mathrm{DTV\ NP\ NP},\ \mathrm{SV\ S},\ \mathrm{VP\ Adv}\}$$

$$\mathrm{NP} \rightarrow \mathrm{Det\ N}$$

(5) **Context-Free Languages (CFLs)**

a. Given a CFG $\langle T, N, D, P\rangle$, we can define a function C from N to (T-)languages (we write C_A for $C(A)$) as described below.

b. The C_A are called the **syntactic categories** of the CFG (and so a nointerminal can be thought of as a name of a syntactic category).

c. A language is called **context-free** if it is a syntactic category of some CFG.

(6) **Historical Notes**

- Up until the mid 1980's an open research questions was whether NLs (considered as sets of word strings) were context-free languages (CFLs).
- Chomsky maintained they were not, and his invention of transformational grammar (TG) was motivated in large part by the perceived need to go beyond the expressive power of CFGs.
- Gazdar and Pullum (early 1980's) refuted all published arguments that NLs could not be CFLs.
- Together with Klein and Sag, they developed a context-free framework, generalized phrase structure grammar (GPSG), for syntactic theory.
- But in 1985, Shieber published a paper arguing that Swiss German cannot be a CFL.
- Shieber's argument is still generally accepted today.

(7) **Defining the Syntactic Categories of a CFG (1/2)**

a. We will recursively define a function $h : \omega \rightarrow \wp(T^*)^N$.

b. Intuitively, for each nonterminal A, the sets $h(n)(A)$ are successively larger approximations of C_A.

c. Then C_A is defined to be $C_A =_{\text{def}} \bigcup_{n \in \omega} h(n)(A)$.

(8) **Defining the Syntactic Categories of a CFG (2/2)**

d. We define h using RT with X, x, F set as follows:

   i. $X = \wp(T^*)^N$

   ii. x is the function that maps each $A \in N$ to the set of length-one strings t such that $A \to t$.

   iii. F is the function from X to X that maps a function $L : N \to \wp(T^*)$ to the function that maps each nonterminal A to the union of $L(A)$ with the set of all strings that can be obtained by applying a PSR $A \to A_0 \ldots A_{n-1}$ to strings $s_0, \ldots, s_{n-1}$, where, for each $i < n$, s_i belongs to $L(A_i)$. In other words: $F(L)(A) =$
   $F(L) \cup \bigcup\{L(A_0) \bullet \ldots \bullet L(A_{n-1}) \mid A \to A_0 \ldots A_{n-1}\}$.

   iv. Given these values of X, x, and F, the RT guarantees the existence of a unique function h from ω to functions from N to $\wp(T^*)$.

(9) **Proving that a String Belongs to a Category (1/2)**

a. With the C_A *formally* defined as above, the two clauses in the *informal* recursive definition (Chapter 6, section 5):

   i. (Base Clause) If $A \rightarrow t$, then $t \in C_A$.

   ii. (Recursion Clause) If $A \rightarrow A_0\ \ldots A_{n-1}$ and for each $i < n$, $s_i \in C_{A_i}$, then $s_0 \ldots s_{n-1} \in C_A$.

   become true assertions.

b. This in turn provides a simple-minded way to prove that a string belongs to a syntactic category (if in fact it does!).

(10) **Proving that a String Belongs to a Category (2/2)**

c. By way of illustration, consider the string
$s =$ **Mary heard Fido bit Felix yesterday**.

d. We can (and will) prove that $s \in C_{\mathrm{S}}$.

e. But most syntacticians would say that s corresponds to *two different sentences*, one roughly paraphrasable as *Mary heard yesterday that Fido bit Felix* and another roughly paraphrasable as *Mary heard that yesterday, Fido bit Felix*.

f. Of course, these two sentences mean different things; but more relevant for our present purposes is that we can also characterize the difference between the two sentences purely in terms of *two distinct ways of proving* that $s \in C_{\mathrm{S}}$.

(11) **First Proof**

a. From the lexicon and the base clause, we know that **Mary, Fido, Felix** $\in C_{\mathrm{NP}}$, **heard** $\in C_{\mathrm{SV}}$, **bit** $\in C_{\mathrm{TV}}$, and **yesterday** $\in C_{\mathrm{Adv}}$.

b. Then, by repeated applications of the recursion clause, it follows that:

1. since **bit** $\in C_{\mathrm{TV}}$ and **Felix** $\in C_{\mathrm{NP}}$, **bit Felix** $\in C_{\mathrm{VP}}$;
2. since **bit Felix** $\in C_{\mathrm{VP}}$ and **yesterday** $\in C_{\mathrm{Adv}}$, **bit Felix yesterday** $\in C_{\mathrm{VP}}$;
3. since **Fido** $\in C_{\mathrm{NP}}$ and **bit Felix yesterday** $\in C_{\mathrm{VP}}$, **Fido bit Felix yesterday** $\in C_{\mathrm{S}}$;
4. since **heard** $\in C_{\mathrm{SV}}$ and **Fido bit Felix yesterday** $\in C_{\mathrm{S}}$, **heard Fido bit Felix yesterday** $\in CP_{\mathrm{VP}}$; and finally,
5. since **Mary** $\in C_{\mathrm{NP}}$ and **heard Fido bit Felix yesterday** $\in C_{\mathrm{VP}}$, **Mary heard Fido bit Felix yesterday** $\in C_{\mathrm{S}}$.

(12) **Second Proof**

a. Same as for first proof.

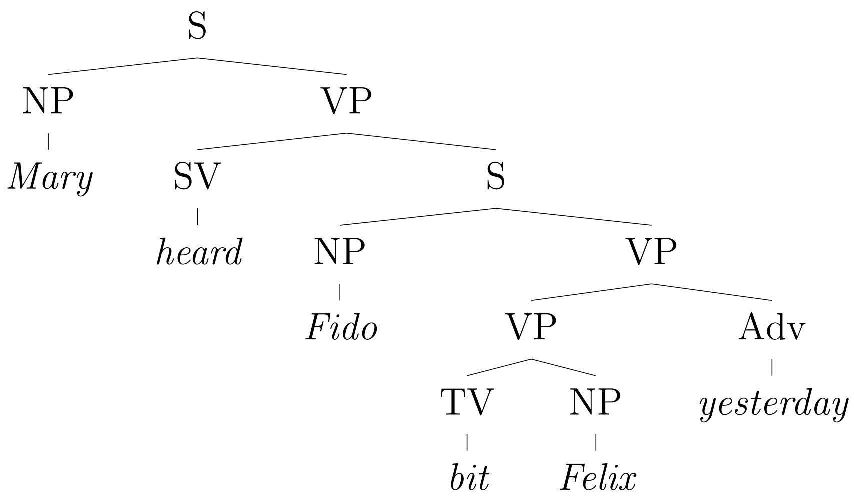
b. Then, by repeated applications of the recursion clause, it follows that:

1. since **Fido** $\in C_{\mathrm{NP}}$ and **bit Felix** $\in C_{\mathrm{VP}}$, **Fido bit Felix** $\in C_{\mathrm{S}}$;
2. since **heard** $\in C_{\mathrm{SV}}$ and **Fido bit Felix** $\in C_{\mathrm{S}}$, **heard Fido bit Felix** $\in C_{\mathrm{VP}}$;
3. since **heard Fido bit Felix** $\in C_{\mathrm{VP}}$ and **yesterday** $\in C_{\mathrm{Adv}}$, **heard Fido bit Felix yesterday** $\in C_{\mathrm{VP}}$; and finally,
4. since **Mary** $\in C_{\mathrm{NP}}$ and **heard Fido bit Felix yesterday** $\in C_{\mathrm{VP}}$, **Mary heard Fido bit Felix yesterday** $\in C_{\mathrm{S}}$.

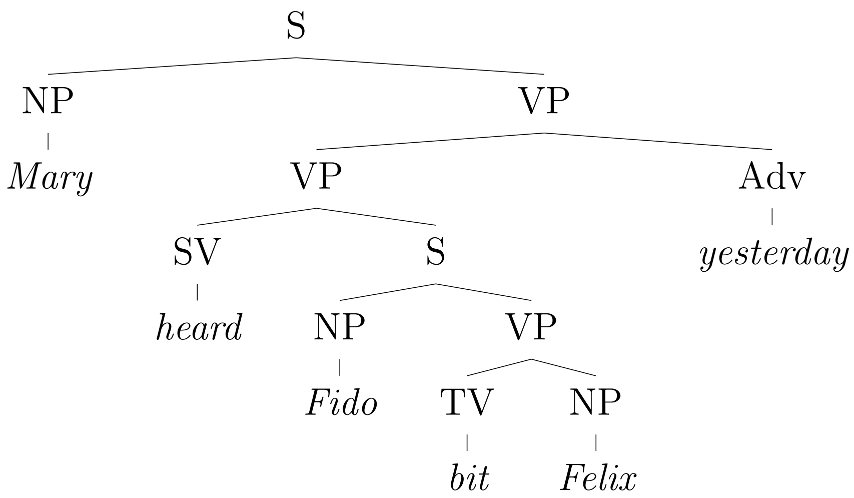
(13) **Proofs vs. Trees**

- The analysis of NL syntax in terms of proofs is characteristic of the family of theoretical approaches collectively known as **categorial grammar**, initiated by Lambek (1958).
- But the most widely practiced approaches (sometimes referred to as **mainstream generative grammar**) analyze NL syntax in terms of *trees*, which will be introduced in a formally precise way in Chapter 7, section 3.
- For now, we just note that the two proofs above would correspond in a more 'mainstream' syntactic approach to the two trees represented informally by the two diagrams:

Tree corresponding to first proof:



Tree corresponding to second proof:



- Intuitively, it seems clear that there is a close relationship between the proof-based approach and the tree-based one, but the nature of the relationship cannot be made precise till we know more about trees and about proofs.