Grammar Formalisms

Yoav Goldberg

(the nice slides are by Julia Hockenmaier)

Recap of last lecture

- **Strong vs weak generative capacity**
- **Why the structure of natural language cannot be modeled by a FSA**

Today’s lecture

- A bit more on context-free grammars
- Some formal language theory;
  the Chomsky Hierarchy
- Bounded and unbounded
  non-local dependencies
- The Penn Treebank

CFGs and center embedding

The mouse ate the corn.
The mouse that the snake ate ate the corn.
The mouse that the snake that the hawk ate ate ate the corn.
....

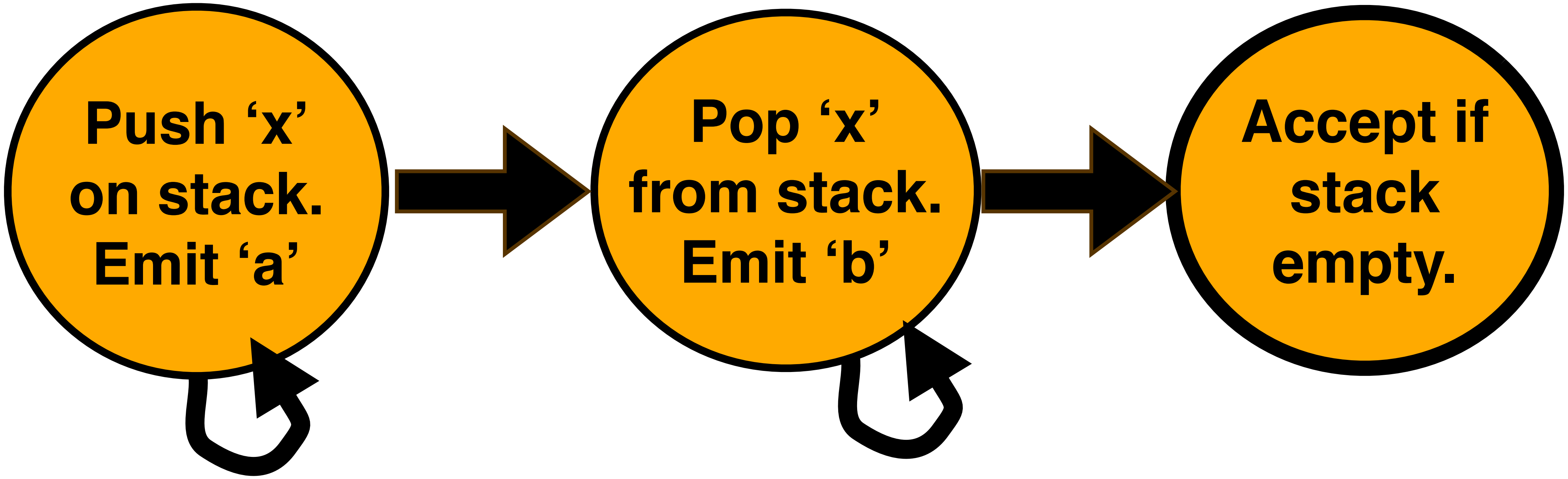
These sentences are all grammatical.
They can be generated by a CFG:

S → NP VP
NP → NP RelClause
RelClause → *that* NP *ate*

Linguists distinguish between a speaker's
- competence (grammatical knowledge) **and**
- performance (processing and memory limitations)

CFGs are equivalent to Pushdown automata (PDAs)

PDAs are FSAs with a stack:
Emit a symbol *and* push/pop a symbol from the stack



This is equivalent to the following CFG:

S → a X b
X → a X b
X → a b

Generating a^nb^n

	Stack:	String:
1. Push x on stack. Emit a.	Stack: x	String: a
2. Push x on stack. Emit a.	Stack: xx	String: aa
3. Push x on stack. Emit a.	Stack: xxx	String: aaa
4. Push x on stack. Emit a.	Stack: xxxx	String: aaaa
5. Pop x off stack. Emit b.	Stack: xxx	String: aaaab
6. Pop x off stack. Emit b.	Stack: xx	String: aaaabb
7. Pop x off stack. Emit b.	Stack: x	String: aaaabbb
8. Pop x off stack. Emit b	Stack:	String: aaaabbbb

Center embedding in German

...daß ich [Hans schwimmen] sah
...that I Hans swim saw
...that I saw [Hans swim]

...daß ich [Maria [Hans schwimmen] helfen] sah
...that I Maria Hans swim help saw
...that I saw [Mary help [Hans swim]]

...daß ich [Anna [Maria [Hans schwimmen] helfen] lassen] sah
...that I Anna Maria Hans swim help let saw
...that I saw [Anna let [Mary help [Hans swim]]]

... and in Dutch...

...dat ik Hans zag zwemmen
...that I Hans saw swim
...that I saw [Hans swim]

...dat ik Maria Hans zag helpen zwemmen
...that I Maria Hans saw help swim
...that I saw [Mary help [Hans swim]]

...dat ik Anna Maria Hans zag laten helpen zwemmen
...that I Anna Maria Hans saw let help swim
...that I saw [Anna let [Mary help [Hans swim]]]

Such cross-serial dependencies require mildly context-sensitive grammars

The Chomsky Hierarchy refined

	Language	**Automata**	**Parsing complexity**	**Dependencies**
Type 3	Regular	Finite-state	linear	adjacent words
Type 2	Context-Free	Pushdown	cubic	nested
	Mildly Context-sensitive	**Extended Pushdown**	**polynomial**	**cross-serial**
Type 1	Context-sensitive	Linear Bounded	exponential	
Type 0	Recursively Enumerable	Turing machine		

(we'll return to this later in the course)

Where do we get the grammar from?

- **Write it by hand:**
  Coverage and software engineering problems
- **Learn/Induce it (from raw text):**
  This doesn’t work so well
- **Read it off a treebank:**
  Gives statistics as well as coverage.
  But: creating treebanks = lots of manual labor.

The Penn Treebank

- **The first large syntactically annotated corpus**
  - Wall Street Journal (50,000 sentences, 1 million words)
  - also Switchboard, Brown corpus,ATIS

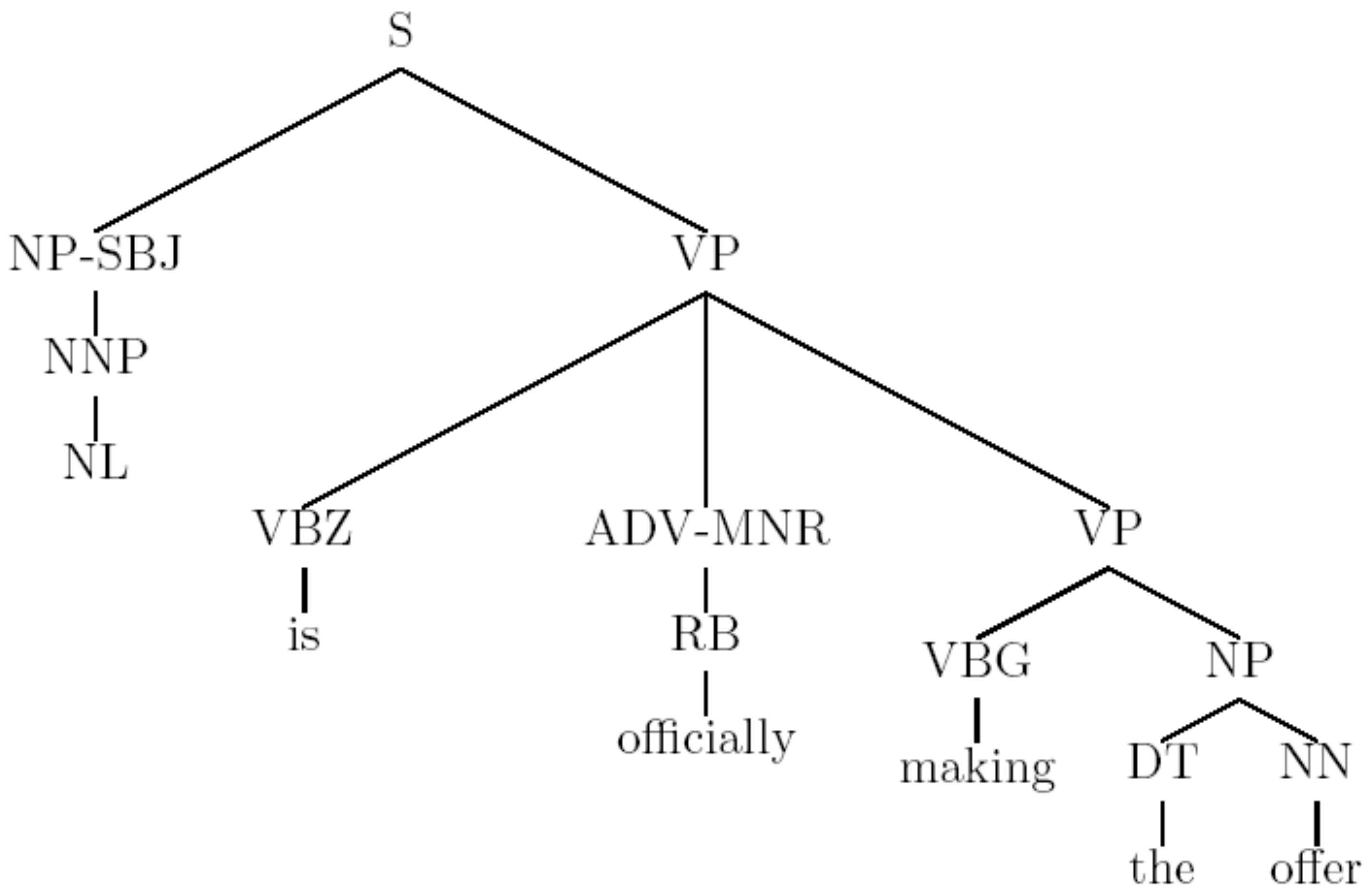
- **The annotation:**
  - POS-tagged (Ratnaparkhi's MXPOST)
  - Manually annotated with phrase-structure trees
  - Relatively detailed analyses (exception: NPs)
  - *Traces* and other *null elements* used to represent non-local dependencies
  - Designed to allow extraction of predicate-argument structure

- **Standard data set for English parsers**

The Treebank label set

- **48 preterminals (tags):**
  - 36 POS tags, 12 other symbols (punctuation etc.)
  - Simplified version of Brown tagset (87 tags)
    (cf. Lancaster-Oslo/Bergen (LOB) tag set: 126 tags)
  - 1M words too little data to allow more fine-grained distinctions?
  - eliminate redundancy that is otherwise recoverable
- **14 nonterminals:**
  - standard inventory (S, NP, VP,...)

A simple example

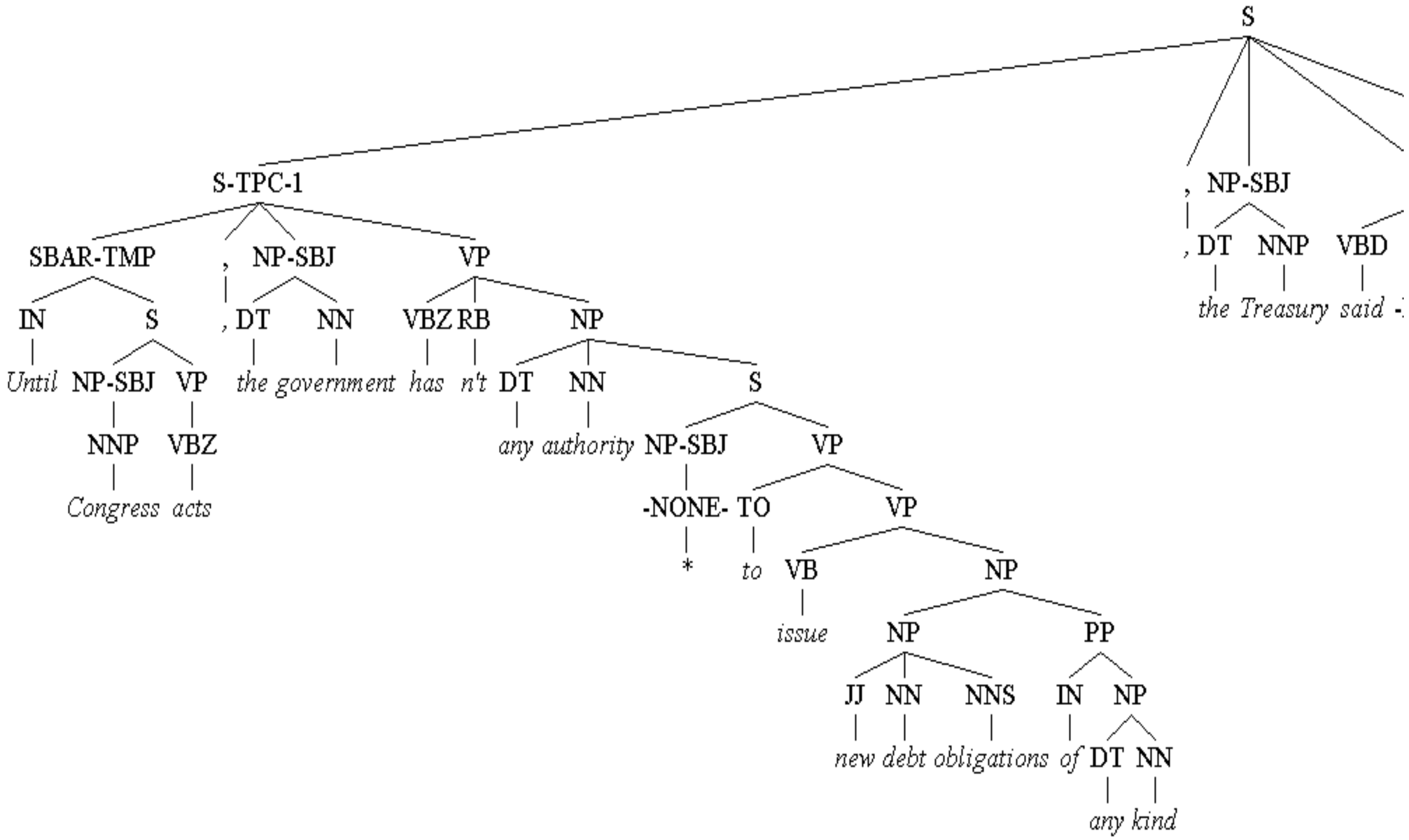


- **Relatively flat structures:**
  - There is no noun level
  - VP arguments and adjuncts appear at the same level

- **Function tags (-SBJ, -MNR)**

A more realistic (partial) example

Until Congress acts, the government hasn't any authority to issue new debt obligations of any kind, the Treasury said .... .



Predicate-argument structure

What is “the meaning” of a (declarative) sentence?

***I am eating sushi*.**

- Truth-conditional semantics:
  We know the meaning of a sentence, if we know under which situations it is true.

- We also want to be able to draw inferences.

- Both require translation into an expression in some formal logic.

Translating language into formal logic....

.... is way beyond the scope of this course!!!

.... and is far from being a solved problem:

- Linguistic issues: quantifiers, tense/aspect, ...
- Coverage!!!

Predicate-argument structure

A simpler task:
Translate a sentence into an expression that describes the relations between the entities described in the sentence.

Who does what to whom?
eat(I, sushi)

NB: typically words stand in for entities.
Grammatical functions (subject, object) replaced with "thematic roles" (agent, patient,....)

Dependency structure

An even simpler task:
Translate a sentence into an expression that describes the relations between the words in the sentence.

Dependency grammars and parsers often ignore some classes of dependencies

Syntactic categories vs. grammatical functions

- **The mapping from syntactic categories to dependency types or grammatical functions is not one-to-one:**
  eat [NP dinner] [NP Monday night]

- **The Penn Treebank solution: function tags**
  *eat [NP dinner] [NP-**TMP** Monday night]*

Function tags in the Penn Treebank

- **Inventory:**
  -TMP, -LOC, -PRD, -SBJ, -CLR, -ADV, -MNR
- **Constituents whose grammatical function differs from the (implicitly assumed) default have function tags.**
- **Useful, but sometimes inconsistent**

The dependencies so far:

- **Arguments:**
  - verbs take arguments: subject, object, complements, ...
  - **Heads subcategorize for their arguments**

- **Adjuncts/Modifiers:**
  - adjectives modify nouns,
  - adverbs modify VPs or adjectives,
  - PPs modify NPs or VPs
  - **Heads do not subcategorize for their modifiers (modifiers subcategorize for the head)**

**These are all “local” dependencies that can typically be expressed in a CFG.
Each word is the dependant of *one other word*.
Hence, dependency *trees.***