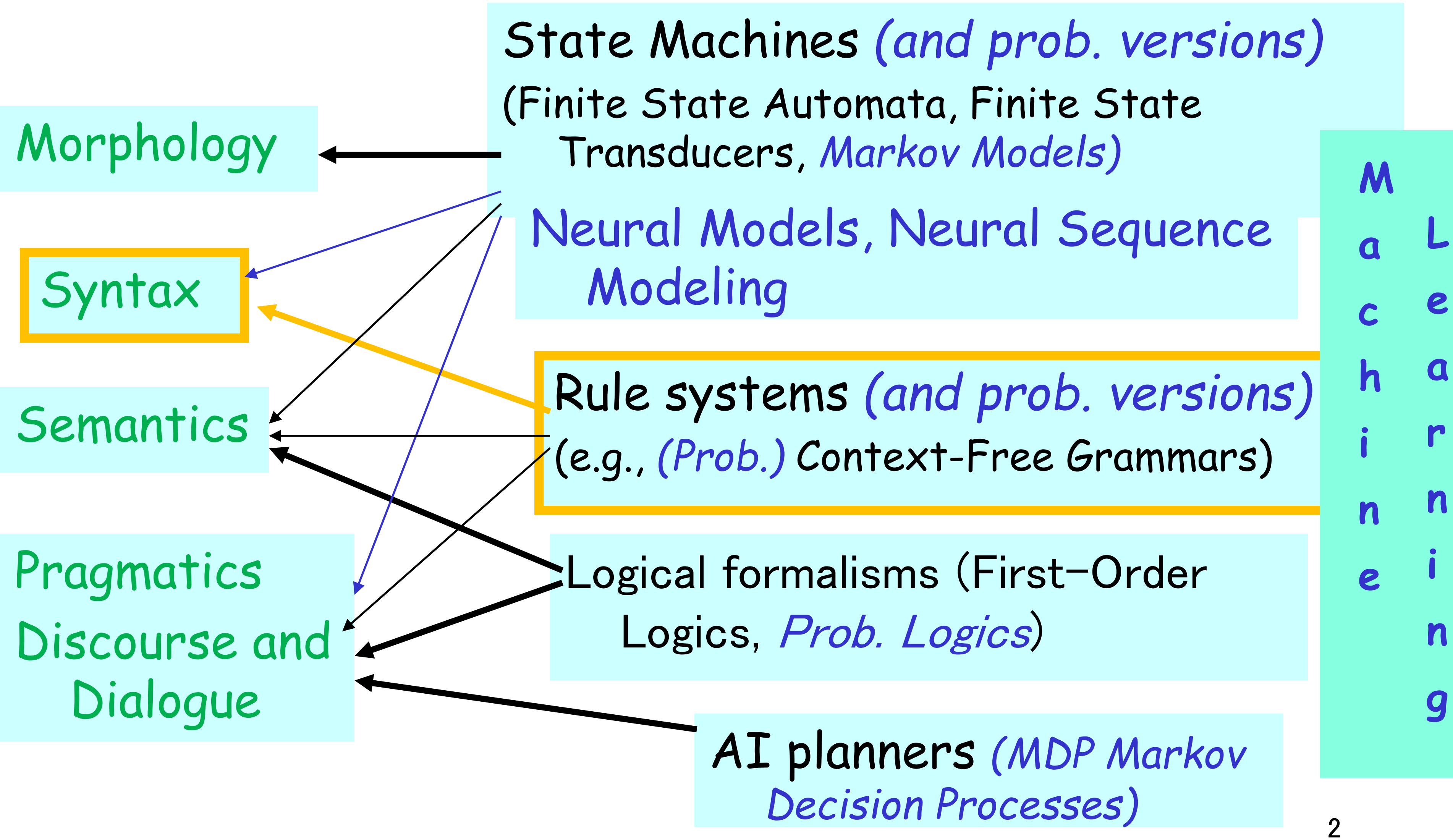
Intelligent Systems (AI-2)

Computer Science cpsc422, Lecture 27

Nov, 10, 2017

NLP: Knowledge–Formalisms Map
(including probabilistic formalisms)



Lecture Overview

- Recap English Syntax and Parsing
- Key Problem with parsing: Ambiguity
- Probabilistic Context Free Grammars (PCFG)
- Treebanks and Grammar Learning

Key Constituents: Examples

Head

NP → N
NP → Det N

(Specifier) X (Complement)

Constituent	(Specifier)	X	(Complement)
• Noun phrases (NP)	• (Det)	N	(PP)
	the	cat	on the table
• Verb phrases (VP)	• (Qual)	V	(NP)
	never	eat	a cat
• Prepositional phrases (PP)	• (Deg)	P	(NP)
	almost	in	the net
• Adjective phrases(AP)	• (Deg)	A	(PP)
	very	happy	about it
• Sentences (S)	• (NP)	(-)	(VP)
	a mouse	--	ate it

Context Free Grammar (CFG)

- **4-tuple** (non-term., term., productions, start)
- (N, Σ, P, S)
- P is a set of rules $A \rightarrow \alpha$; $A \in \mathbf{N}$, $\alpha \in (\Sigma \cup N)^*$

$N = \{X\ Y\}$ $\Sigma = \{a\ b\ c\}$ $P =$

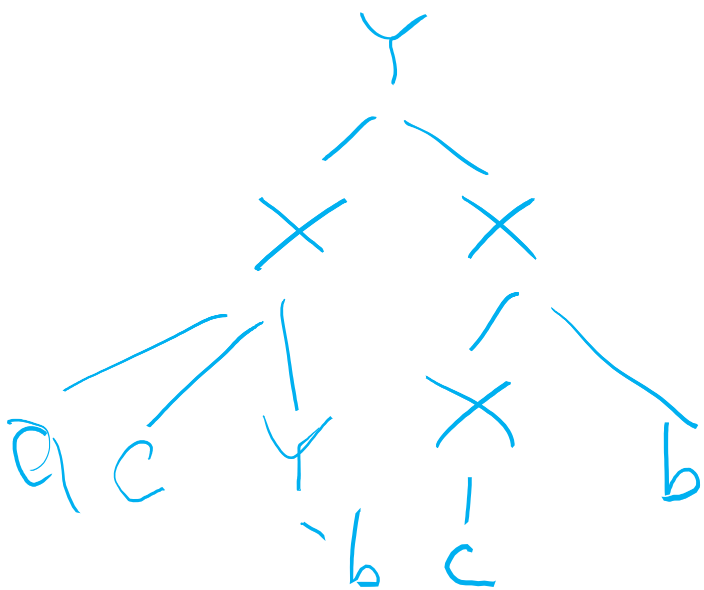
$X \rightarrow Xb$

$Y \rightarrow XX$

$X \rightarrow acY$

$X \rightarrow c$

$Y \rightarrow b$



CFG Example

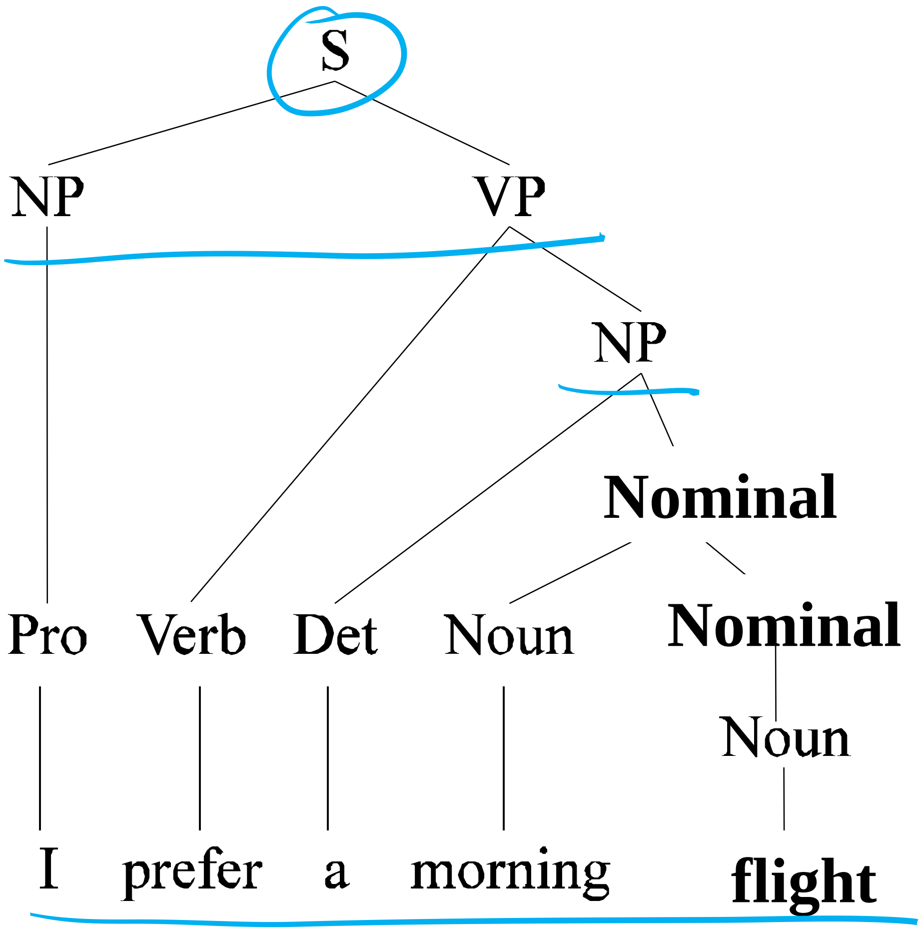
Grammar with example phrases

S	→	*NP VP*	I + want a morning flight
NP	→	*Pronoun*	I
	|	*Proper-Noun*	Los Angeles
	|	*Det Nominal*	a + flight
Nominal	→	*Noun Nominal*	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

Lexicon

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* | ...

Derivations as Trees



$S \rightarrow NP\ VP$

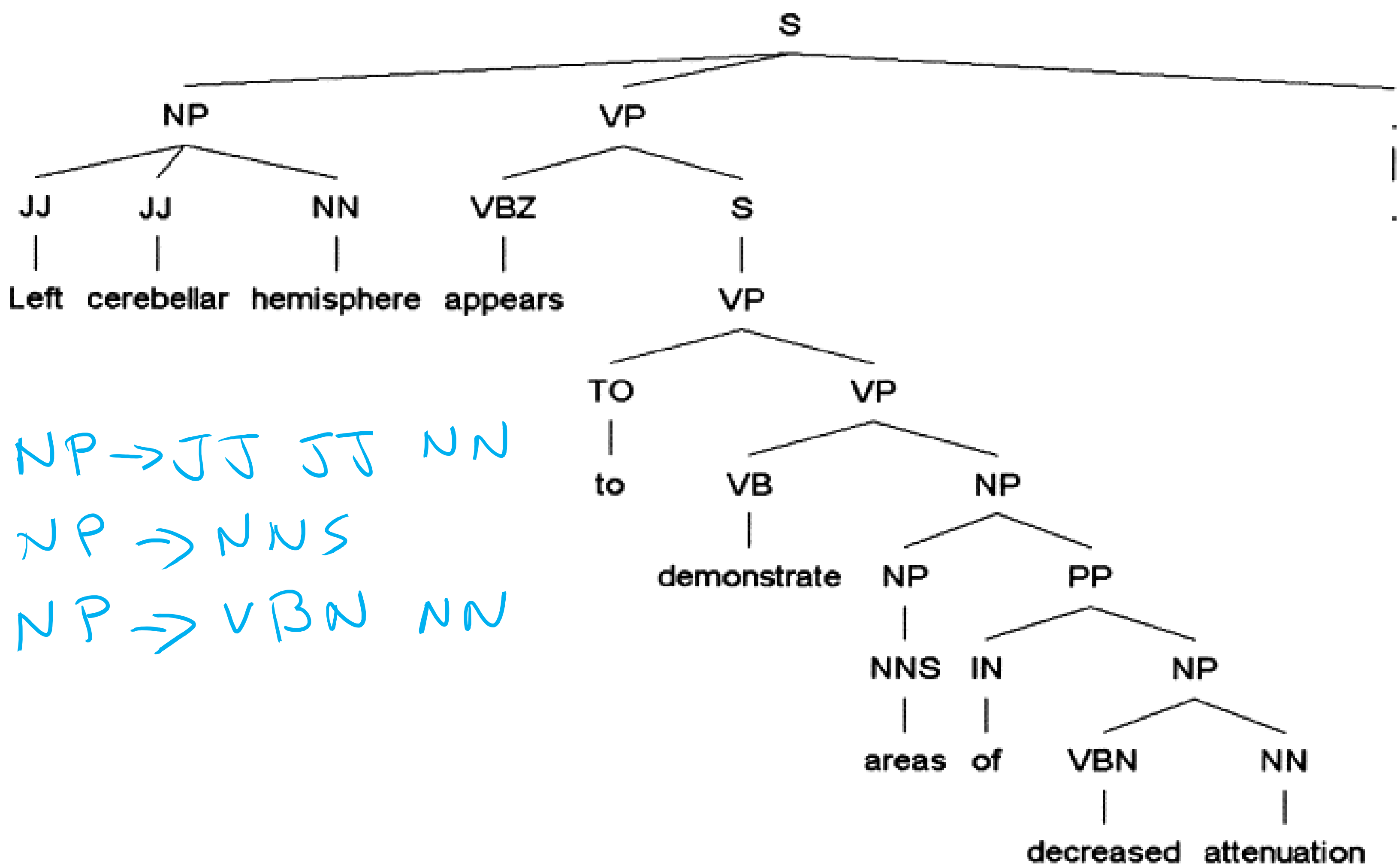
$NP \rightarrow Pronoun$
$\quad |\ Proper\text{-}Noun$
$\quad |\ Det\ Nominal$

$Nominal \rightarrow Noun\ Nominal$
$\quad |\ Noun$

$VP \rightarrow Verb$
$\quad |\ Verb\ NP$
$\quad |\ Verb\ NP\ PP$
$\quad |\ Verb\ PP$

$PP \rightarrow Preposition\ NP$

Example of relatively complex parse tree



NP -> JJ JJ NN
NP -> NNS
NP -> VBN NN

Journal of the American Medical Informatics Association, 2005, Improved Identification of Noun Phrases in Clinical Radiology Reports Using a High-Performance Statistical Natural Language Parser Augmented with the UMLS Specialist Lexicon

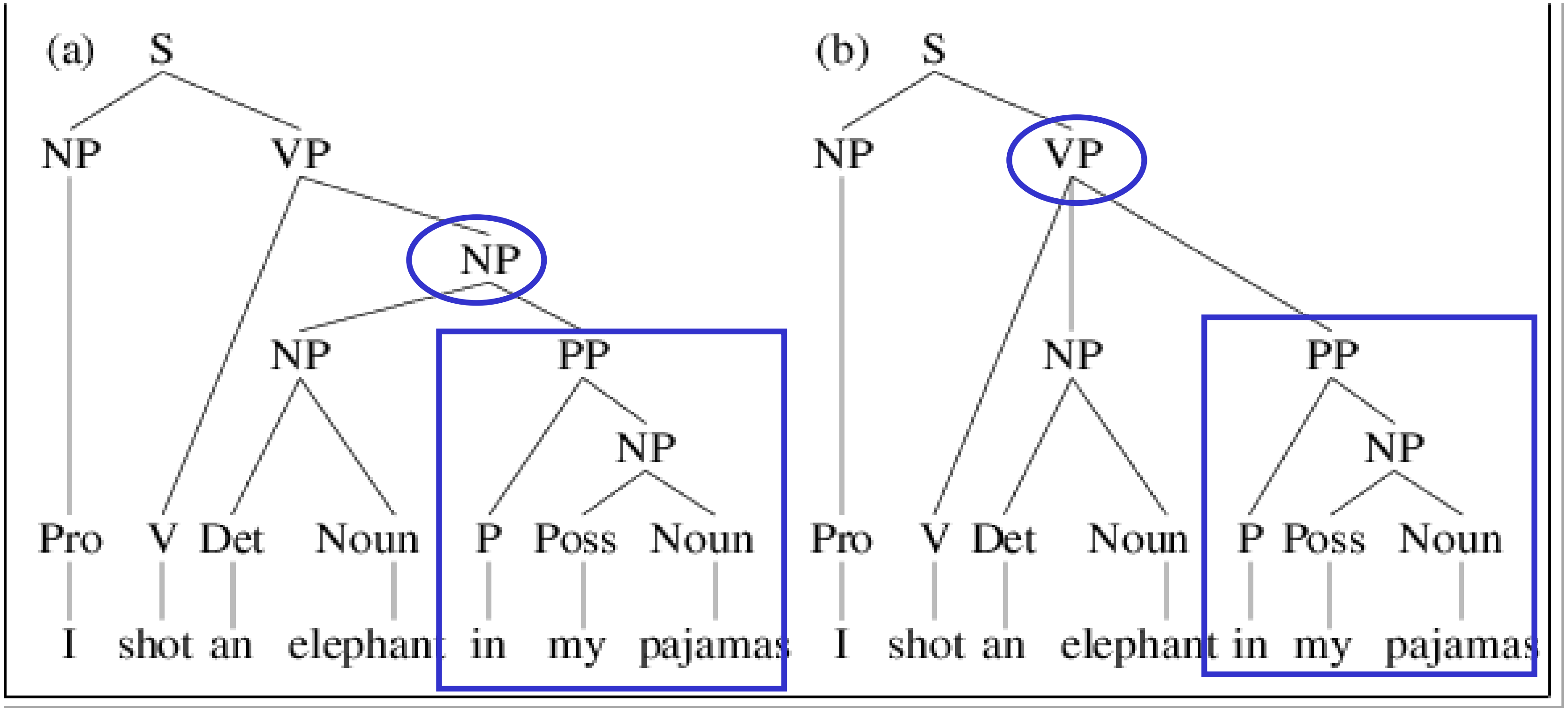
Lecture Overview

- Recap English Syntax and Parsing
- **Key Problem with parsing: Ambiguity**
- Probabilistic Context Free Grammars (PCFG)
- Treebanks and Grammar Learning

Structural Ambiguity (Ex. 1)

VP -> V NP ; NP -> NP PP
VP -> V NP PP

"I shot an elephant in my pajamas"



Structural Ambiguity (Ex.2)

"I saw Mary passing by cs2"

```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBD saw)
      (S
        (NP (NNP Mary))
        (VP (VBG passing)
          (PP (IN by)
            (NP (NNP cs2)))))))
```

"I saw Mary passing by cs2"

```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBD saw)
      (NP (NNP Mary))
      (S
        (VP (VBG passing)
          (PP (IN by)
            (NP (NNP cs2))))))))
```

Structural Ambiguity (Ex. 3)

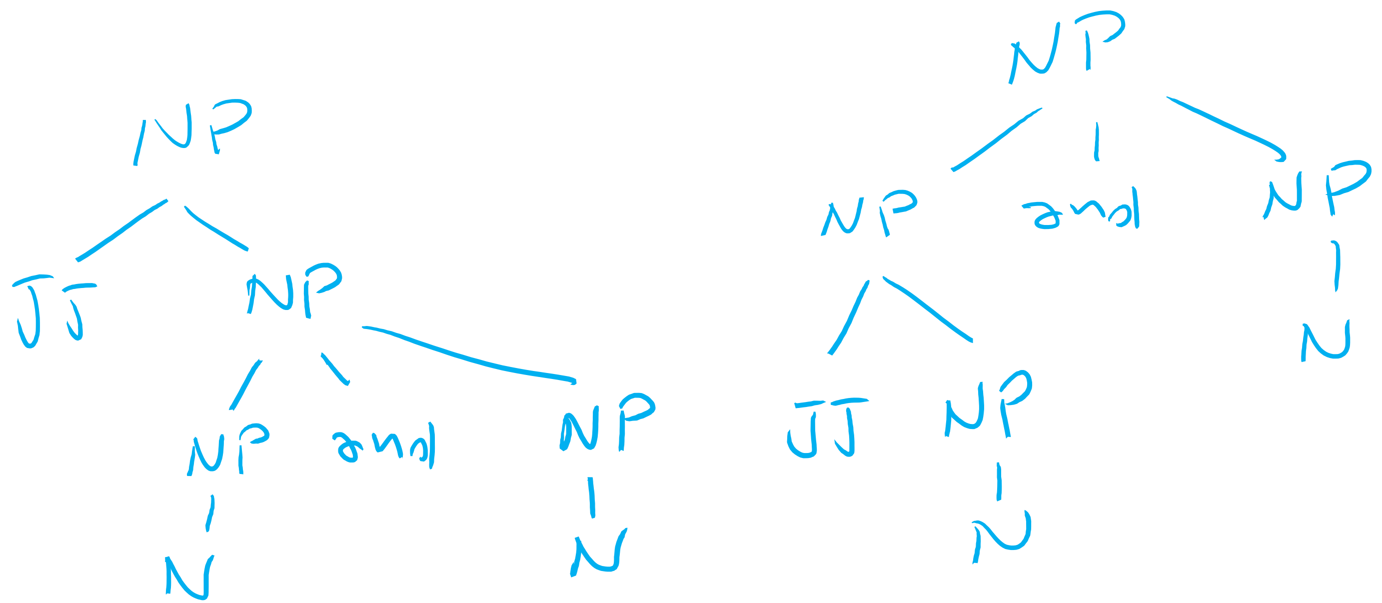
- **Coordination "new student <u>and</u> profs"**

JJ N N

NP -> NP and NP

NP -> JJ NP

NP -> N



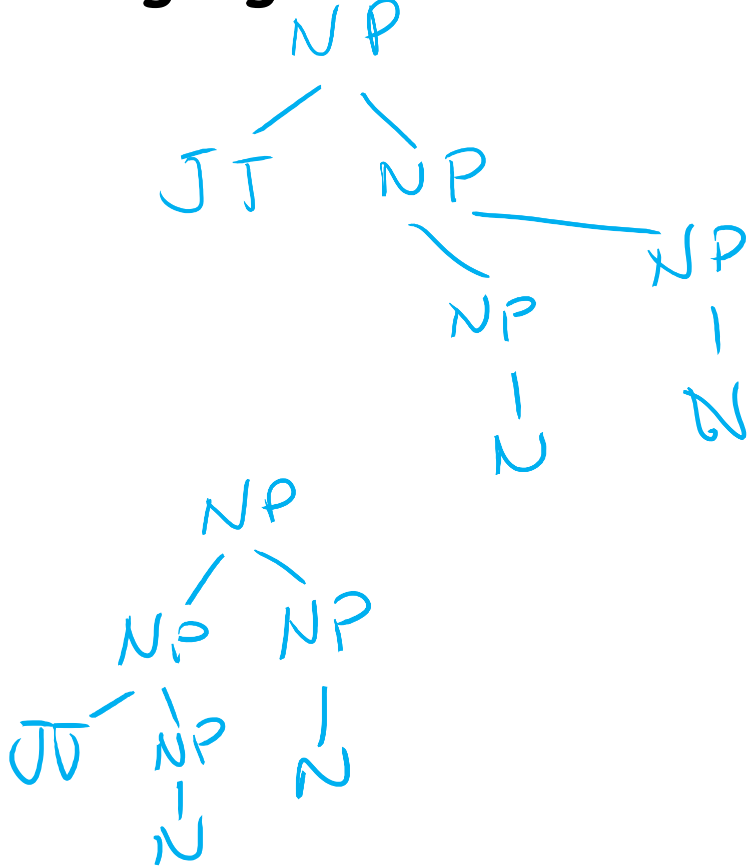
Structural Ambiguity (Ex. 4)

- **NP-bracketing “French language teacher”**

NP -> JJ NP

NP -> N

NP -> NP NP



Lecture Overview

- Recap English Syntax and Parsing
- Key Problem with parsing: Ambiguity
- **Probabilistic Context Free Grammars (PCFG)**
- Treebanks and Grammar Learning (acquiring the probabilities)
- Intro to Parsing PCFG

Probabilistic CFGs (PCFGs)

- **GOAL: assign a probability to parse trees and to sentences**
- **Each grammar rule is augmented with a conditional probability**

- **If these are <u>all the rules for VP</u> and .55 is the** P(VP->Verb | VP)

VP -> Verb	.55
VP -> Verb NP	.40
VP -> Verb NP NP	??

- **What ?? should be ?**

A. 1
B. 0
C. .05
D. None of the above

iclicker.

Probabilistic CFGs (PCFGs)

- **GOAL: assign a probability to parse trees and to sentences**
- **Each grammar rule is augmented with a conditional probability**
  - **The expansions for a given non-terminal sum to 1**

VP -> Verb	.55
VP -> Verb NP	.40
VP -> Verb NP NP	.05

P(VP->Verb | VP)
P(VP->Verb NP | VP)
P(VP->Verb NP NP | VP)

Formal Def: 5-tuple (N, Σ, P, S, D)

Sample PCFG

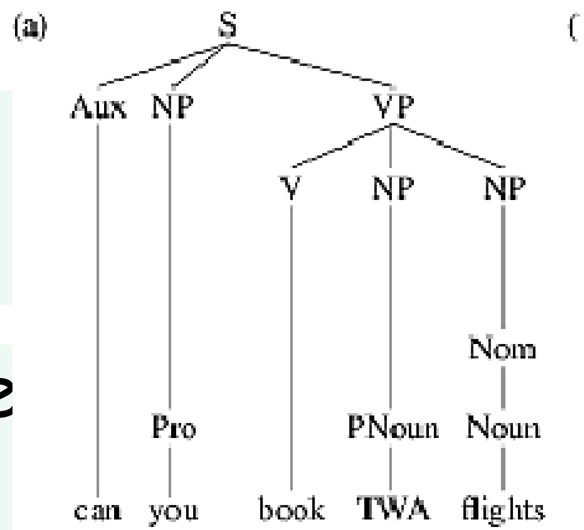
S	→	$NP\ VP$	[.80]	Det	→	$that$ [.05] | the [.80] | a [.15]	
S	→	$Aux\ NP\ VP$	[.15]	$Noun$	→	$book$	[.10]
S	→	VP	[.05]	$Noun$	→	$flights$	[.50]
NP	→	$Det\ Nom$	[.20]	$Noun$	→	$meal$	[.40]
NP	→	$Proper\text{-}Noun$	[.35]	$Verb$	→	$book$	[.30]
NP	→	Nom	[.05]	$Verb$	→	$include$	[.30]
NP	→	$Pronoun$	[.40]	$Verb$	→	$want$	[.40]
Nom	→	$Noun$	[.75]	Aux	→	can	[.40]
Nom	→	$Noun\ Nom$	[.20]	Aux	→	$does$	[.30]
Nom	→	$Proper\text{-}Noun\ Nom$	[.05]	Aux	→	do	[.30]
VP	→	$Verb$	[.55]	$Proper\text{-}Noun$	→	TWA	[.40]
VP	→	$Verb\ NP$	[.40]	$Proper\text{-}Noun$	→	$Denver$	[.40]
VP	→	$Verb\ NP\ NP$	[.05]	$Pronoun$	→	you [.40] | I [.60]	

PCFGs are used to….

- **Estimate Prob. of parse tree**

  A. Sum of the probs of all the rules applied

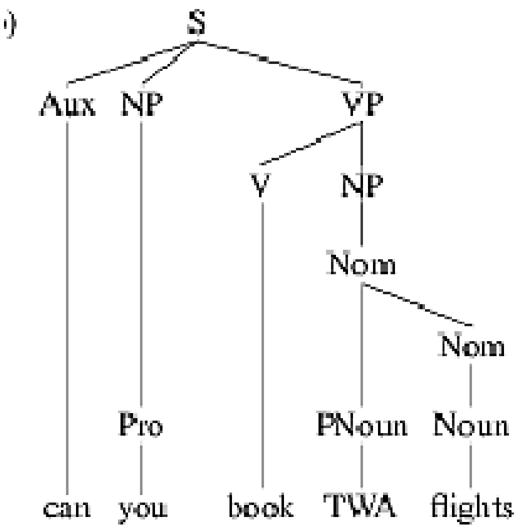
  B. Product of the probs of all the rules applied



- **Estimate Prob. of a sentence**

  A. Sum of the probs of all the parse trees

  B. Product of the probs of all the parse trees



PCFGs are used to....

- **Estimate Prob. of parse tree**

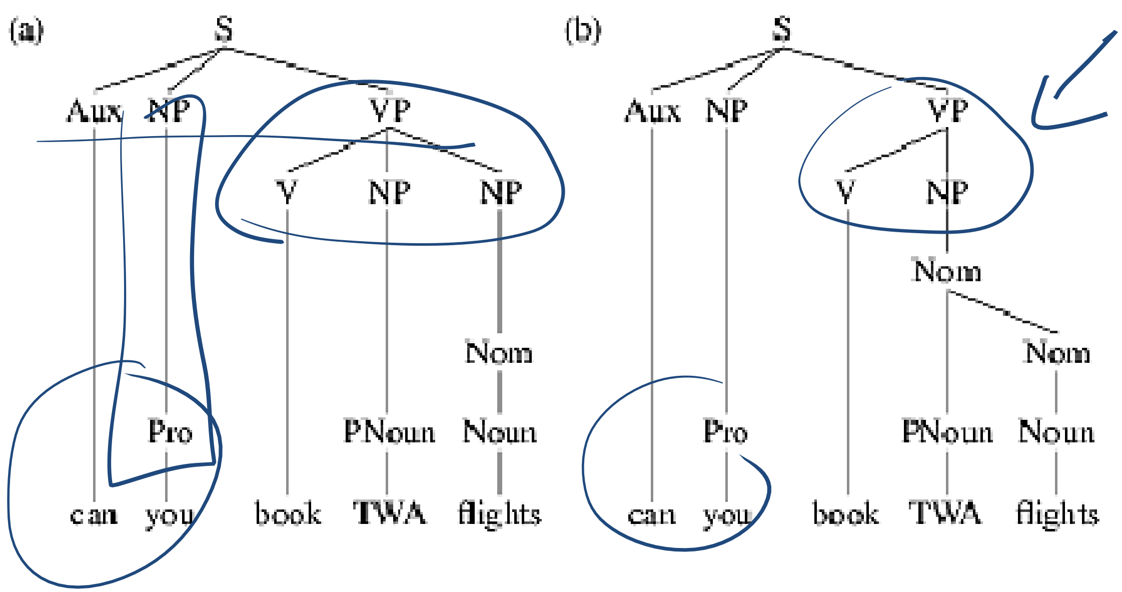
$$P(Tree) = \prod_{node \in Tree} P(\text{expansion for node})$$

- **Estimate Prob. to sentences**

$$P(Sentence) = \sum_{Tree \in \text{Sentence parses}} P(Tree)$$

Example

$P(Tree^a) = .15 * .4 * \ldots = 1.5 \times 10^{-6}$ $\quad$ $P(Tree^b) = .15 * .4 * \ldots = 1.7 \times 10^{-6}$



$P(\text{"Can you...."}) =$

$1.7 \times 10^{-6} + 1.5 \times 10^{-6} =$

3.2×10^{-6}

(a)

Rules			P
S	→	Aux NP VP	.15
NP	→	Pro	.40
VP	→	V NP NP	.05
NP	→	Nom	.05
NP	→	PNoun	.35
Nom	→	Noun	.75
Aux	→	Can	.40
~~NP~~	~~→~~	~~Pro~~	.40
Pro	→	you	.40
Verb	→	book	.30
PNoun	→	TWA	.40
Noun	→	flights	.50

(b)

Rules			P
S	→	Aux NP VP	.15
NP	→	Pro	.40
VP	→	V NP	.40
NP	→	Nom	.05
Nom	→	PNoun Nom	.05
Nom	→	Noun	.75
Aux	→	Can	.40
~~NP~~	~~→~~	~~Pro~~	.40
Pro	→	you	.40
Verb	→	book	.30
Pnoun	→	TWA	.40
Noun	→	flights	.50

Lecture Overview

- Recap English Syntax and Parsing
- Key Problem with parsing: Ambiguity
- Probabilistic Context Free Grammars (PCFG)
- **Treebanks and Grammar Learning (acquiring the probabilities)**

Treebanks

- DEF. corpora in which each sentence has been paired with a parse tree
- These are generally created
  - Parse collection with parser
  - human annotators revise each parse
- Requires detailed annotation guidelines
  - POS tagset
  - Grammar
  - instructions for how to deal with particular grammatical constructions.

Penn Treebank

- **Penn TreeBank is a widely used treebank.**

- Most well known is the Wall Street Journal section of the Penn TreeBank.
  - 1 M words from the 1987-1989 Wall Street Journal.

```
( (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) ))
    (. .) ))
```

S → "S" NP VP .

S → NP VP

Treebank Grammars

- Such grammars tend to contain lots of rules….
- For example, the Penn Treebank has 4500 different rules for VPs! Among them...

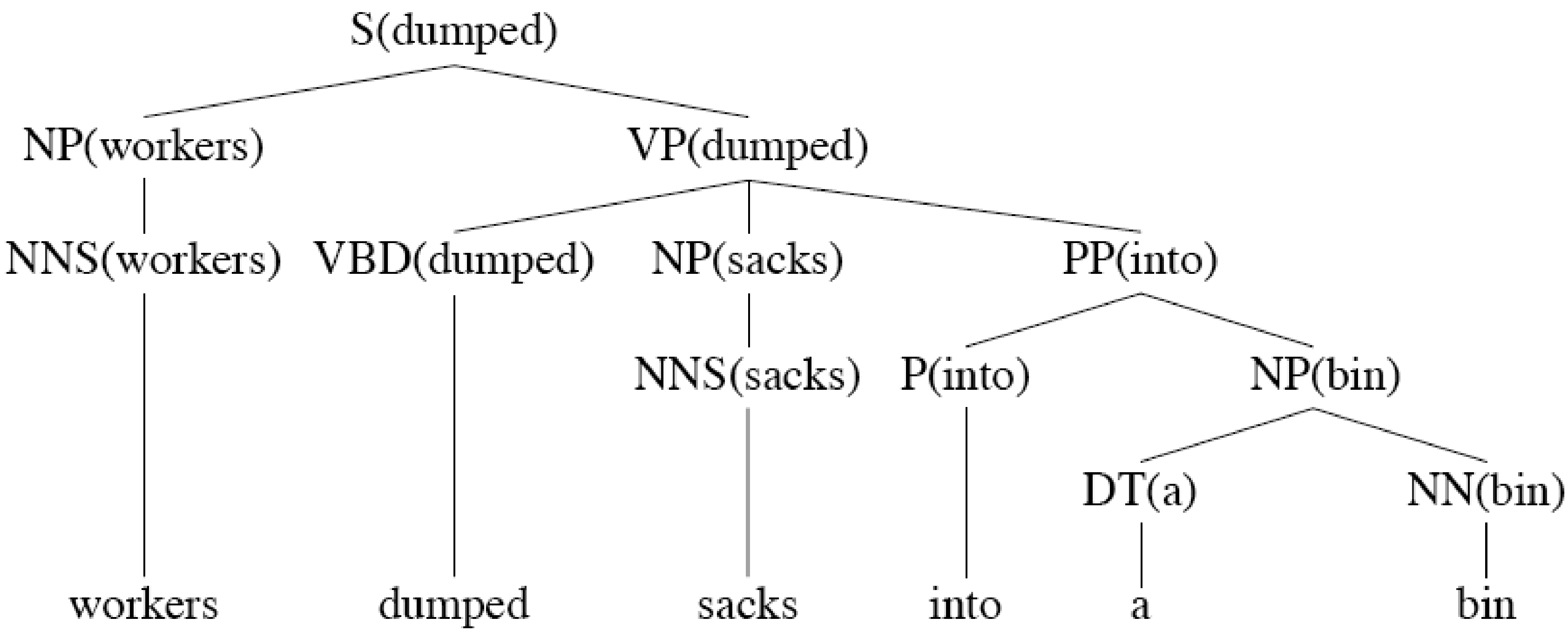
$$VP \rightarrow VBD\ PP$$
$$VP \rightarrow VBD\ PP\ PP$$
$$VP \rightarrow VBD\ PP\ PP\ PP$$
$$VP \rightarrow VBD\ PP\ PP\ PP\ PP$$

Heads in Trees

- Finding heads in treebank trees is a task that arises frequently in many applications.
  - Particularly important in statistical parsing

- We can visualize this task by annotating the nodes of a parse tree with the heads of each corresponding node.

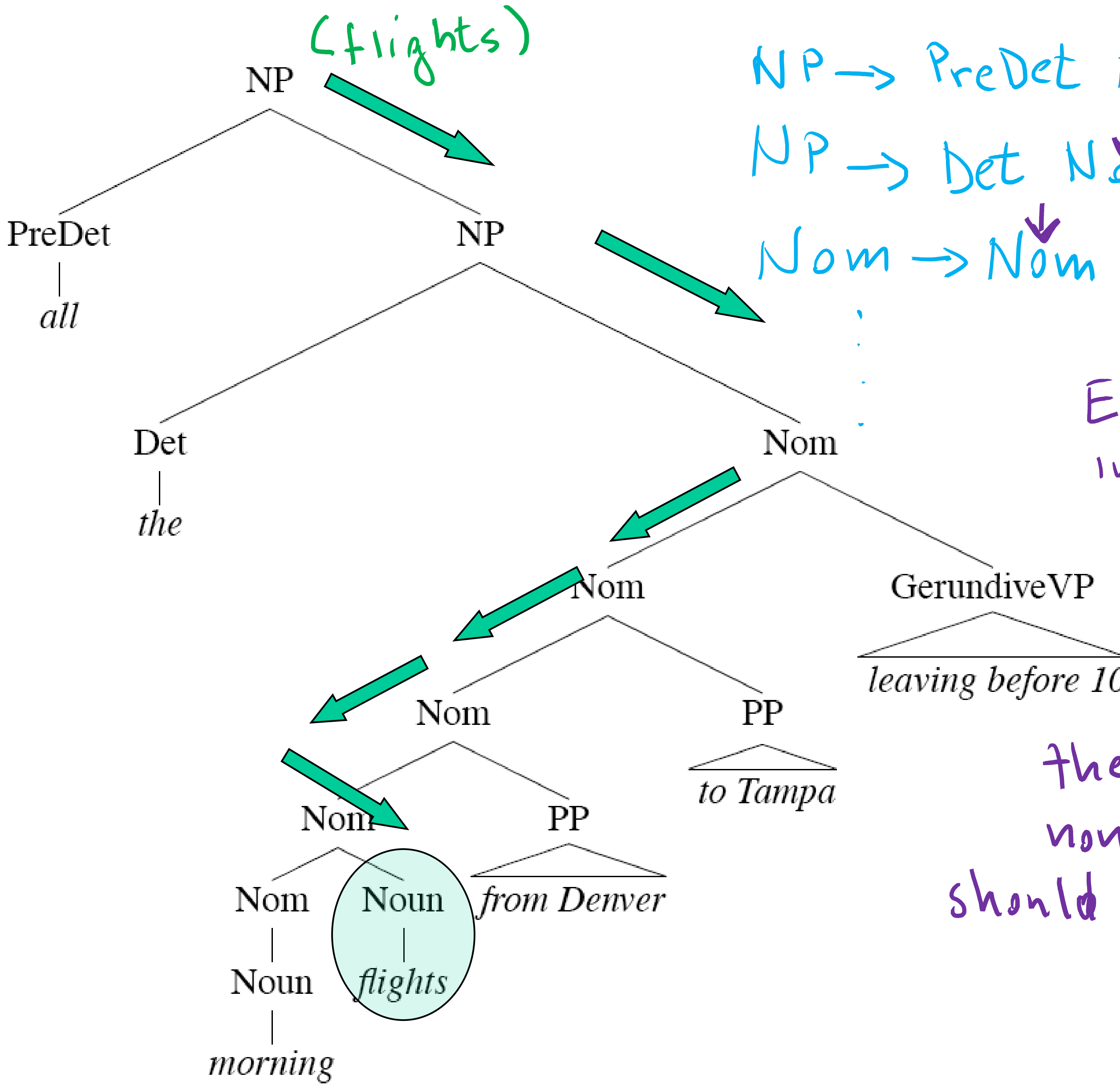
Lexically Decorated Tree



Head Finding

- **The standard way to do head finding is to use a simple set of tree traversal rules specific to each non-terminal in the grammar.**
- **Each rule in the PCFG specifies where the head of the expanded non-terminal should be found**

Noun Phrases



NP → PreDet NP

NP → Det Nom

Nom → Nom GerundiveVP

Each rule in the PCFG specifies where the head of the expanded non-terminal should be found

Acquiring Grammars and Probabilities

Manually parsed text corpora (e.g., PennTreebank)

- **Grammar**: read it off the parse trees

Ex: if an NP contains an ART, ADJ, and NOUN then we create the rule *NP -> ART ADJ NOUN*.

NP -> Pro

- **Probabilities:**

$$P(A \rightarrow \alpha | A) = \frac{count(A \rightarrow \alpha)}{\sum_{\forall \gamma} count(A \rightarrow \gamma)} = \frac{count(A \rightarrow \alpha)}{count(A)}$$

Ex: if the *NP -> ART ADJ NOUN* rule is used 50 times and all NP rules are used 5000 times, then the rule's probability is … .01

Example

If you look at all the parse trees in the bank you find three rules for NP

	How many times
① NP → ART ADJ NOUN	50
② NP → NOUN	4000
③ NP → PRONOUN	950
	5000 total # of NP expansions

$P(① | NP) = 50/5000 = .01$

$P(② | NP) = 4000/5000 = .8$

$P(③ | NP) = 950/5000 = .19$

$.19 = 1 - (.01 + .8)$

Learning Goals for today's class

You can:

- Provide a formal definition of a PCFG
- Apply a PCFG to compute the probability of a parse tree of a sentence as well as the probability of a sentence
- Describe the content of a treebank
- Describe the process to identify a head of a syntactic constituent
- Compute the probability distribution of a PCFG from a treebank

Next class on Wed

- Parsing Probabilistic CFG: CKY parsing
- PCFG in practice: Modeling Structural and Lexical Dependencies

Assignment-3 due on Nov 20
(last year took students 8-18 hours)

Assignment-4 will be out on the same day