

Probabilistic CKY

600.465 - Intro to NLP - J. Elinger

1

Our bane: Ambiguity

- John saw Mary
 - Typhoid Mary
 - Phillips screwdriver Mary
 - note how rare rules interact*
- I see a bird
 - is this 4 nouns – parsed like "city park scavenger bird"?
 - rare parts of speech, plus systematic ambiguity in noun sequences*
- Time flies like an arrow
 - Fruit flies like a banana
 - Time reactions like this one
 - Time reactions like a chemist
 - or is it just an NP?*

600.465 - Intro to NLP - J. Elinger

2

Our bane: Ambiguity

- John saw Mary
 - Typhoid Mary
 - Phillips screwdriver Mary
 - note how rare rules interact*
- I see a bird
 - is this 4 nouns – parsed like "city park scavenger bird"?
 - rare parts of speech, plus systematic ambiguity in noun sequences*
- Time | flies like an arrow NP VP
 - Fruit flies | like a banana NP VP
 - Time | reactions like this one V[stem] NP
 - Time reactions | like a chemist S PP
 - or is it just an NP?*

600.465 - Intro to NLP - J. Elinger

3

How to solve this combinatorial explosion of ambiguity?

- First try parsing without any weird rules, throwing them in only if needed.
- Better: every rule has a weight. A tree's weight is total weight of all its rules. Pick the overall lightest parse of sentence.
- Can we pick the weights automatically? We'll get to this later ...

600.465 - Intro to NLP - J. Elinger

4

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3								
1		NP 4 VP 4							
2				P 2 V 5					
3						Det 1			
4								N 8	

- 1 S → NP VP
- 6 S → Vst NP
- 2 S → S PP
- 1 VP → V NP
- 2 VP → VP PP
- 1 NP → Det N
- 2 NP → NP PP
- 3 NP → NP NP
- 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3								
1		NP 4 VP 4							
2				P 2 V 5					
3						Det 1			
4								N 8	

- 1 S → NP VP
- 6 S → Vst NP
- 2 S → S PP
- 1 VP → V NP
- 2 VP → VP PP
- 1 NP → Det N
- 2 NP → NP PP
- 3 NP → NP NP
- 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						
	Vst 3								
1			NP 4						
			VP 4						
2				P 2					
				V 5					
3						Det 1			
4							N 8		

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						
	Vst 3		S 8						
1			NP 4						
			VP 4						
2				P 2					
				V 5					
3						Det 1			
4							N 8		

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						
	Vst 3		S 8						
			S 13						
1			NP 4						
			VP 4						
2				P 2					
				V 5					
3						Det 1			
4							N 8		

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						
	Vst 3		S 8						
			S 13						
1			NP 4						
			VP 4						
2				P 2					
				V 5					
3						Det 1			
4							N 8		

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						
	Vst 3		S 8						
			S 13						
1			NP 4						
			VP 4						
2				P 2					
				V 5					
3						Det 1	NP 10		
4							N 8		

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						
	Vst 3		S 8						
			S 13						
1			NP 4						
			VP 4						
2				P 2					
				V 5					
3						Det 1	NP 10		
4							N 8		

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4							
2			P 2 V 5					PP 12	
3					Det 1	NP 10			
4						N 8			

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4							
2			P 2 V 5					PP 12	
3					Det 1	NP 10			
4						N 8			

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4							
2			P 2 V 5					PP 12	
3					Det 1	NP 10			
4						N 8			

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4						NP 18	
2			P 2 V 5					PP 12	
3					Det 1	NP 10			
4						N 8			

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4						NP 18 S 21	
2			P 2 V 5					PP 12	
3					Det 1	NP 10			
4						N 8			

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12	
3					Det 1	NP 10			
4						N 8			

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							NP 24
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							NP 24 S 22
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							NP 24 S 22 S 27
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							NP 24 S 22 S 27
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13							NP 24 S 22 S 27 NP 24
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3	NP 10						NP 24	
	Vst 3	S 8						S 22	
		S 13						S 27	
1		NP 4						NP 24	
		VP 4						S 27	
2			P 2					NP 24	
			V 5					S 27	
3					Det 1			NP 24	
								S 27	
4								NP 24	
								S 27	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3	NP 10						NP 24	
	Vst 3	S 8						S 22	
		S 13						S 27	
1		NP 4						NP 24	
		VP 4						S 27	
2			P 2					NP 24	
			V 5					S 27	
3					Det 1			NP 24	
								S 27	
4								NP 24	
								S 27	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3	NP 10						NP 24	
	Vst 3	S 8						S 22	
		S 13						S 27	
1		NP 4						NP 24	
		VP 4						S 27	
2			P 2					NP 24	
			V 5					S 27	
3					Det 1			NP 24	
								S 27	
4								NP 24	
								S 27	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3	NP 10						NP 24	
	Vst 3	S 8						S 22	
		S 13						S 27	
1		NP 4						NP 24	
		VP 4						S 27	
2			P 2					NP 24	
			V 5					S 27	
3					Det 1			NP 24	
								S 27	
4								NP 24	
								S 27	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3	NP 10						NP 24	
	Vst 3	S 8						S 22	
		S 13						S 27	
1		NP 4						NP 24	
		VP 4						S 27	
2			P 2					NP 24	
			V 5					S 27	
3					Det 1			NP 24	
								S 27	
4								NP 24	
								S 27	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3	NP 10						NP 24	
	Vst 3	S 8						S 22	
		S 13						S 27	
1		NP 4						NP 24	
		VP 4						S 27	
2			P 2					NP 24	
			V 5					S 27	
3					Det 1			NP 24	
								S 27	
4								NP 24	
								S 27	

1 S → NP VP
 6 S → Vst NP
 2 S → S PP
 1 VP → V NP
 2 VP → VP PP
 1 NP → Det N
 2 NP → NP PP
 3 NP → NP NP
 0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
1								NP 24	
								S 27	
								S 22	
								S 27	
2								NP 18	
								S 21	
								VP 18	
3								PP 12	
								VP 16	
4								NP 10	
								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
1								NP 18	
								S 21	
								VP 18	
2								PP 12	
								VP 16	
3								NP 10	
								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Which entries do we need?

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
1								NP 18	
								S 21	
								VP 18	
2								PP 12	
								VP 16	
3								NP 10	
								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Which entries do we need?

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
1								NP 18	
								S 21	
								VP 18	
2								PP 12	
								VP 16	
3								NP 10	
								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Not worth keeping ...

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
1								NP 18	
								S 21	
								VP 18	
2								PP 12	
								VP 16	
3								NP 10	
								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

... since it just breeds worse options

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
1								NP 18	
								S 21	
								VP 18	
2								PP 12	
								VP 16	
3								NP 10	
								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Keep only best-in-class!

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8 S 13						NP 24 S 22 S 27 S 27 S 22 S 27	
1		NP 4 VP 4						NP 18 S 21 VP 18	
2			P 2 V 5					PP 12 VP 16	
3					Det 1			NP 10	
4								N 8	

inferior stock

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

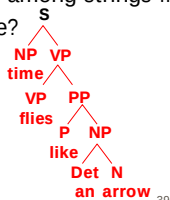
Keep only best-in-class!
(and backpointers so you can recover parse)

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3 Vst 3	NP 10 S 8						NP 24 S 22 NP 18 S 21 VP 18	
1		NP 4 VP 4						PP 12 VP 16	
2			P 2 V 5					NP 10 N 8	
3					Det 1				
4									

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Probabilistic Trees

- Instead of lightest weight tree, take highest probability tree
- Given any tree, your assignment 1 generator would have some probability of producing it!
- Just like using n-grams to choose among strings ...
- What is the probability of this tree?

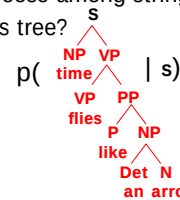


600.465 - Intro to NLP - J. Eisner

39

Probabilistic Trees

- Instead of lightest weight tree, take highest probability tree
- Given any tree, your assignment 1 generator would have some probability of producing it!
- Just like using n-grams to choose among strings ...
- What is the probability of this tree?



- You rolled a lot of independent dice ...

600.465 - Intro to NLP - J. Eisner

40

Chain rule: One word at a time

$$\begin{aligned}
 p(\text{time flies like an arrow}) &= p(\text{time}) \\
 &\quad * p(\text{flies} \mid \text{time}) \\
 &\quad * p(\text{like} \mid \text{time flies}) \\
 &\quad * p(\text{an} \mid \text{time flies like}) \\
 &\quad * p(\text{arrow} \mid \text{time flies like an})
 \end{aligned}$$

600.465 - Intro to NLP - J. Eisner

41

Chain rule + backoff (to get trigram model)

$$\begin{aligned}
 p(\text{time flies like an arrow}) &= p(\text{time}) \\
 &\quad * p(\text{flies} \mid \text{time}) \\
 &\quad * p(\text{like} \mid \text{time flies}) \\
 &\quad * p(\text{an} \mid \text{time flies like}) \\
 &\quad * p(\text{arrow} \mid \text{time flies like an})
 \end{aligned}$$

600.465 - Intro to NLP - J. Eisner

42

Chain rule – written differently

$$\begin{aligned}
 p(\text{time flies like an arrow}) \\
 &= p(\text{time}) \\
 &\quad * p(\text{time flies} \mid \text{time}) \\
 &\quad * p(\text{time flies like} \mid \text{time flies}) \\
 &\quad * p(\text{time flies like an} \mid \text{time flies like}) \\
 &\quad * p(\text{time flies like an arrow} \mid \text{time flies like an})
 \end{aligned}$$

$$\text{Proof: } p(x, y \mid x) = p(x \mid x) * p(y \mid x, x) = 1 * p(y \mid x)$$

600.465 - Intro to NLP - 1. Eliezer

43

Chain rule + backoff

$$\begin{aligned}
 p(\text{time flies like an arrow}) \\
 &= p(\text{time}) \\
 &\quad * p(\text{time flies} \mid \text{time}) \\
 &\quad * p(\text{time flies like} \mid \text{time flies}) \\
 &\quad * p(\text{time flies like an} \mid \text{time flies like}) \\
 &\quad * p(\text{time flies like an arrow} \mid \text{time flies like an})
 \end{aligned}$$

$$\text{Proof: } p(x, y \mid x) = p(x \mid x) * p(y \mid x, x) = 1 * p(y \mid x)$$

600.465 - Intro to NLP - 1. Eliezer

44

Chain rule: One node at a time

$$\begin{aligned}
 p(\text{time flies like an arrow} \mid s) &= p(\text{time} \mid s) * p(\text{flies} \mid \text{time}) * p(\text{like} \mid \text{time flies}) * p(\text{an} \mid \text{time flies like}) * p(\text{arrow} \mid \text{time flies like an}) \\
 &\quad * p(\text{time} \mid s) * p(\text{flies} \mid \text{time}) * p(\text{like} \mid \text{time flies}) * p(\text{an} \mid \text{time flies like}) * p(\text{arrow} \mid \text{time flies like an}) * \dots
 \end{aligned}$$

600.465 - Intro to NLP - 1. Eliezer

45

Chain rule + backoff

model you used
in homework 1!
(called "PCFG")

$$\begin{aligned}
 p(\text{time flies like an arrow} \mid s) &= p(\text{time} \mid s) * p(\text{flies} \mid \text{time}) * p(\text{like} \mid \text{time flies}) * p(\text{an} \mid \text{time flies like}) * p(\text{arrow} \mid \text{time flies like an}) \\
 &\quad * p(\text{time} \mid s) * p(\text{flies} \mid \text{time}) * p(\text{like} \mid \text{time flies}) * p(\text{an} \mid \text{time flies like}) * p(\text{arrow} \mid \text{time flies like an}) * \dots
 \end{aligned}$$

600.465 - Intro to NLP - 1. Eliezer

46

Simplified notation

model you used
in homework 1!
(called "PCFG")

$$\begin{aligned}
 p(\text{time flies like an arrow} \mid s) &= p(s \rightarrow \text{NP VP} \mid s) * p(\text{NP} \rightarrow \text{flies} \mid \text{NP}) \\
 &\quad * p(\text{VP} \rightarrow \text{VP NP} \mid \text{VP}) \\
 &\quad * p(\text{VP} \rightarrow \text{flies} \mid \text{VP}) * \dots
 \end{aligned}$$

600.465 - Intro to NLP - 1. Eliezer

47

Already have a CKY alg for weights ...

$$\begin{aligned}
 w(\text{time flies like an arrow} \mid s) &= w(s \rightarrow \text{NP VP}) + w(\text{NP} \rightarrow \text{flies} \mid \text{NP}) \\
 &\quad + w(\text{VP} \rightarrow \text{VP NP}) \\
 &\quad + w(\text{VP} \rightarrow \text{flies} \mid \text{VP}) + \dots
 \end{aligned}$$

Just let $w(x \rightarrow yz) = -\log p(x \rightarrow yz \mid x)$
Then lightest tree has highest prob

48

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
								NP 24	
								S 27	
								S 27	
1			NP 4					NP 18	
			VP 4					S 21	
								VP 18	
2					P 2			PP 12	
					V 5			VP 16	
3						Det 1		NP 10	
4								N 8	

multiply to get 2^{-22}

2^{-8}

2^{-12}

2^{-2}

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
								NP 24	
								S 27	
								S 27	
1			NP 4					NP 18	
			VP 4					S 21	
								VP 18	
2					P 2			PP 12	
					V 5			VP 16	
3						Det 1		NP 10	
4								N 8	

multiply to get 2^{-22}

2^{-8}

2^{-13}

2^{-2}

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Why probabilities not weights?

- We just saw probabilities are really just a special case of weights ...
- ... *but* we can estimate them from training data by counting and smoothing! Yay!
- Warning: What kind of training corpus do we need?

A slightly different task

- Been asking: What is probability of generating a given *tree* with your homework 1 generator?
 - To pick tree with highest prob: useful in parsing.
- But could also ask: What is probability of generating a given *string* with the generator?
 - (i.e., with the *-t* option turned off)
 - To pick string with highest prob: useful in speech recognition, as substitute for an n-gram model.
 - ("Put the file in the folder" vs. "Put the file and the folder")
 - To get prob of generating string, must add up probabilities of all trees for the string ...

Could just add up the parse probabilities

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 8					S 22	
			S 13					S 27	
								NP 24	
								S 27	
								S 27	
								S 27	
1			NP 4					NP 18	
			VP 4					S 21	
								VP 18	
2					P 2			PP 12	
					V 5			VP 16	
3						Det 1		NP 10	
4								N 8	

oops, back to finding exponentially many parses

2^{-22}

2^{-27}

2^{-27}

2^{-22}

2^{-27}

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Any more efficient way?

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S 2 ⁻⁸					S 22	
			S 2 ⁻¹³					S 27	
								NP 24	
								S 27	
								S 27	
1			NP 4					NP 18	
			VP 4					S 21	
								VP 18	
2					P 2			PP 12	
					V 5			VP 16	
3						Det 1		NP 10	
4								N 8	

1 S → NP VP
6 S → Vst NP
2 S → S PP
1 VP → V NP
2 VP → VP PP
1 NP → Det N
2 NP → NP PP
3 NP → NP NP
0 PP → P NP

Add as we go ... (the "inside algorithm")

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10					NP 24	
	Vst 3		S $2^{-8}+2^{-13}$					S 22	
								S 27	
								NP 24	
1								S 27	
								S 2^{-22}	
								$+2^{-27}$	
								NP 18	
2			NP 4					S 21	
			VP 4					VP 18	
								PP 2^{-12}	
								VP 16	
3				P 2				NP 10	
				V 5				N 8	
4						Det 1			

1 $S \rightarrow NP VP$

6 $S \rightarrow Vst NP$

$2^{-2} S \rightarrow S PP$

1 $VP \rightarrow V NP$

2 $VP \rightarrow VP PP$

1 $NP \rightarrow Det N$

2 $NP \rightarrow NP PP$

3 $NP \rightarrow NP NP$

0 $PP \rightarrow P NP$

- 1 S → NP VP
- 6 S → Vst NP
- 2⁻² S → S PP
- 1 VP → V NP
- 2 VP → VP PP
- 1 NP → Det N
- 2 NP → NP PP
- 3 NP → NP NP
- 0 PP → P NP

Add as we go ... (the "inside algorithm")

time	1	flies	2	like	3	an	4	arrow	5
0	NP 3		NP 10						NP 2 ⁻²²
	Vst 3		S 2 ^{-8+2⁻¹³}						+2 ⁻²⁷
									S 2 ⁻²²
									+2 ⁻²⁷
1									+2 ⁻²⁷
									+2 ⁻²²
									+2 ⁻²⁷
									NP 18
2									S 21
									VP 18
									PP 2 ⁻¹²
									VP 16
3									NP 10
									N 8
4									

1 S → NP VP

6 S → Vst NP

2⁻² S → S PP

1 VP → V NP

2 VP → VP PP

1 NP → Det N

2 NP → NP PP

3 NP → NP NP

0 PP → P NP

- 1 S → NP VP
- 6 S → Vst NP
- 2⁻² S → S PP
- 1 VP → V NP
- 2 VP → VP PP
- 1 NP → Det N
- 2 NP → NP PP
- 3 NP → NP NP
- 0 PP → P NP