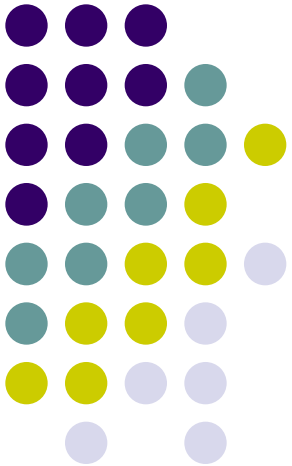
Probabilistic Earley Parsing

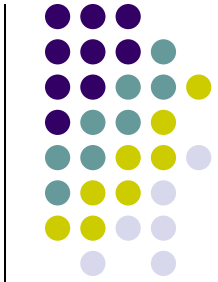
Charlie Kehoe, Spring 2004



Based on the 1995 paper by Andreas Stolcke:

An Efficient Probabilistic Context-Free Parsing Algorithm that Computes Prefix Probabilities

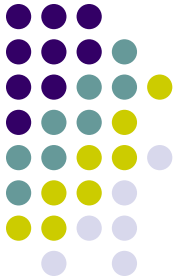
Overview

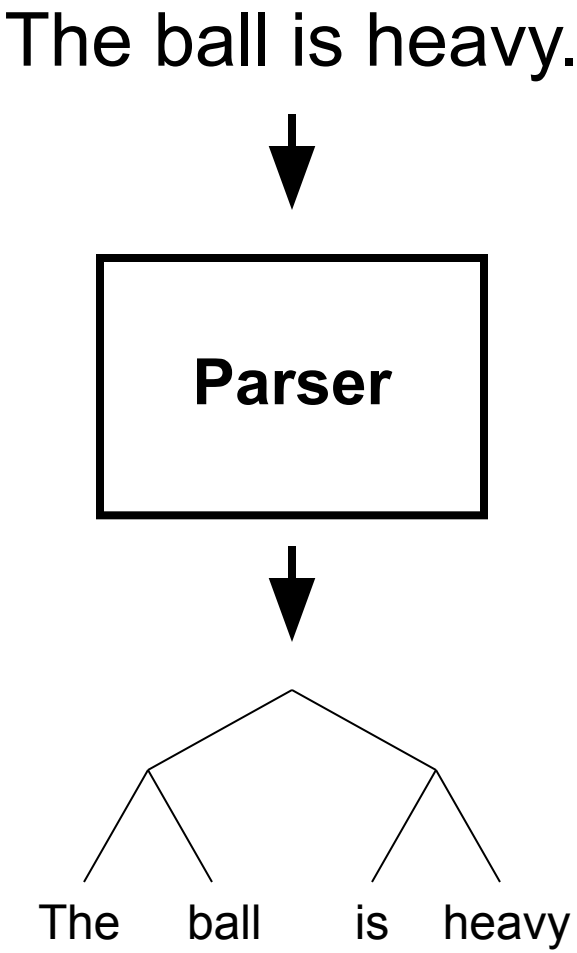


- What is this paper all about?

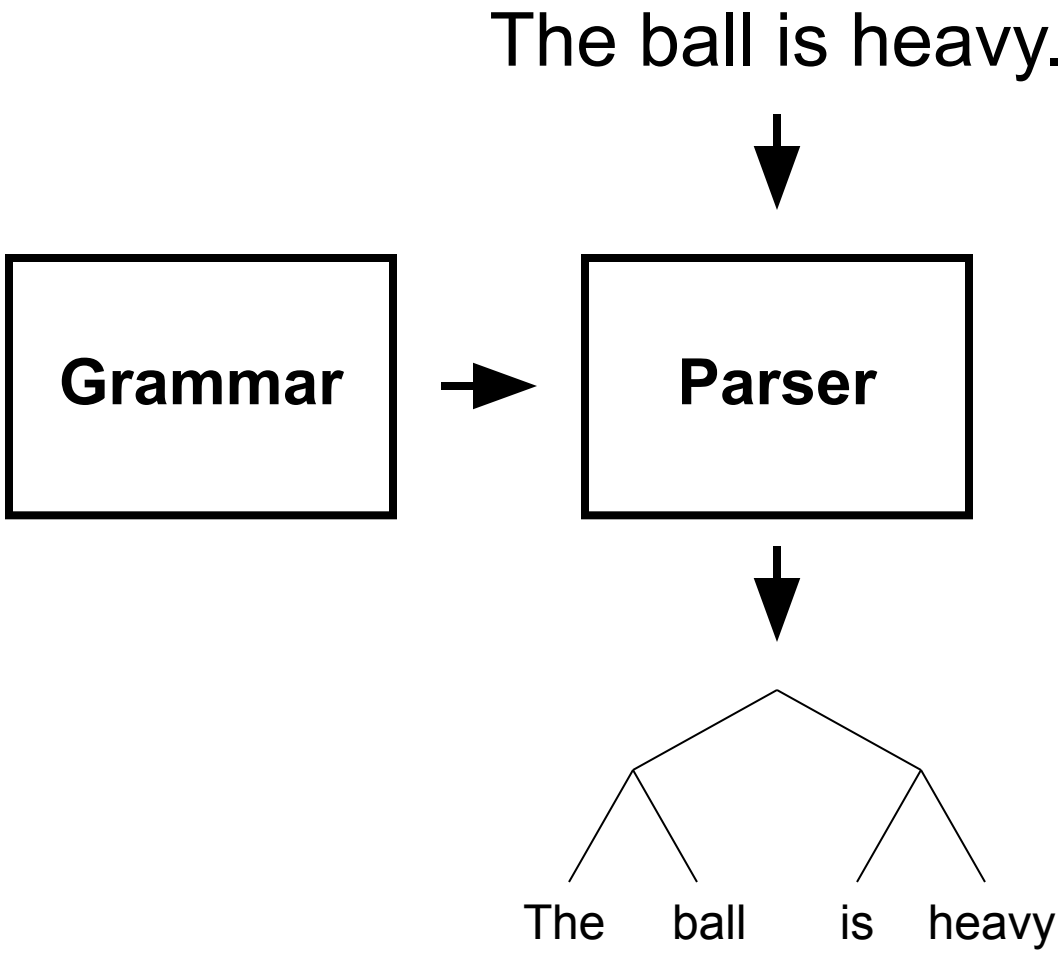
- Key ideas from the title:
  - Context-Free Parsing
  - Probabilistic
  - Computes Prefix Probabilities
  - Efficient

Context-Free Parsing

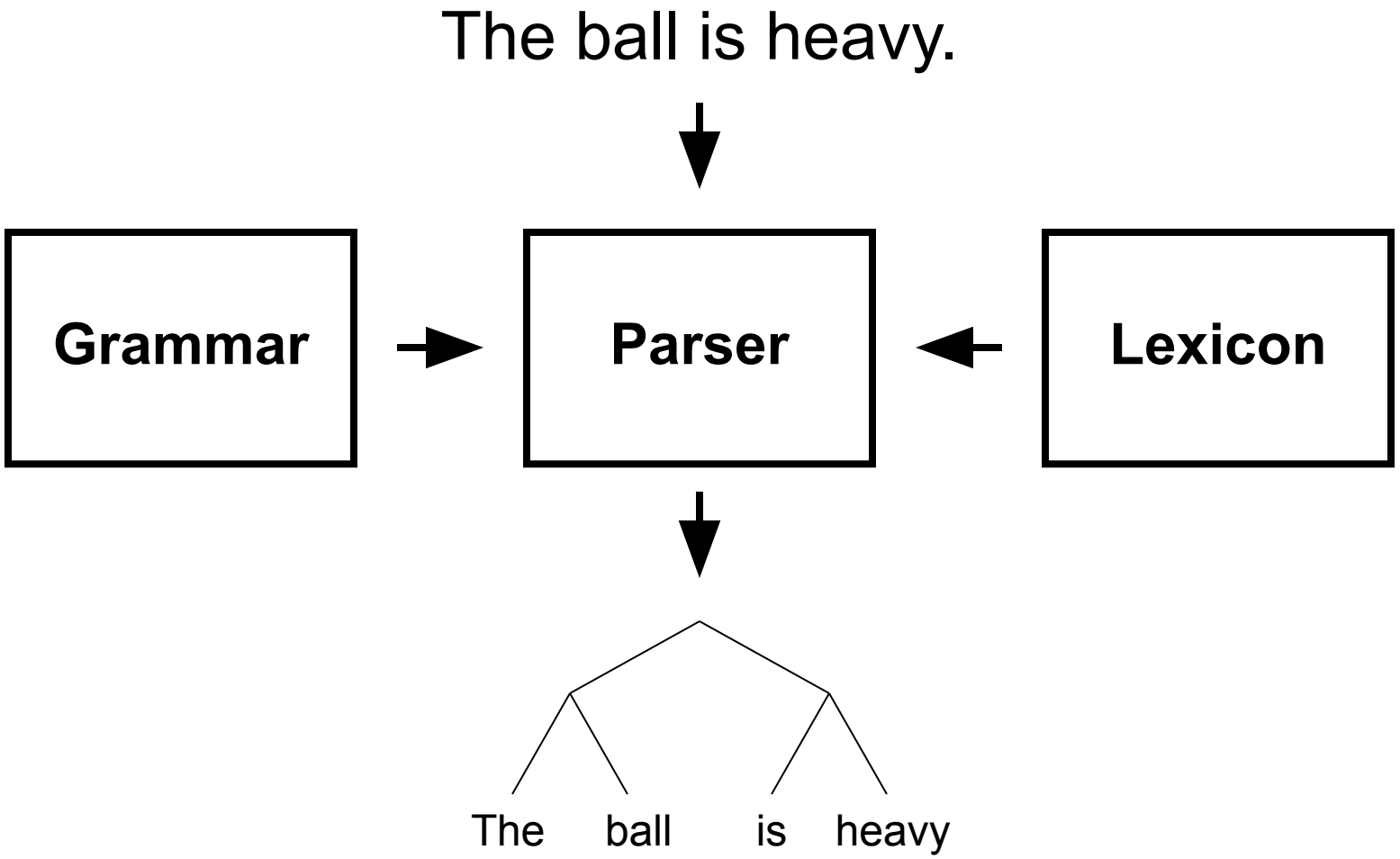




Context-Free Parsing

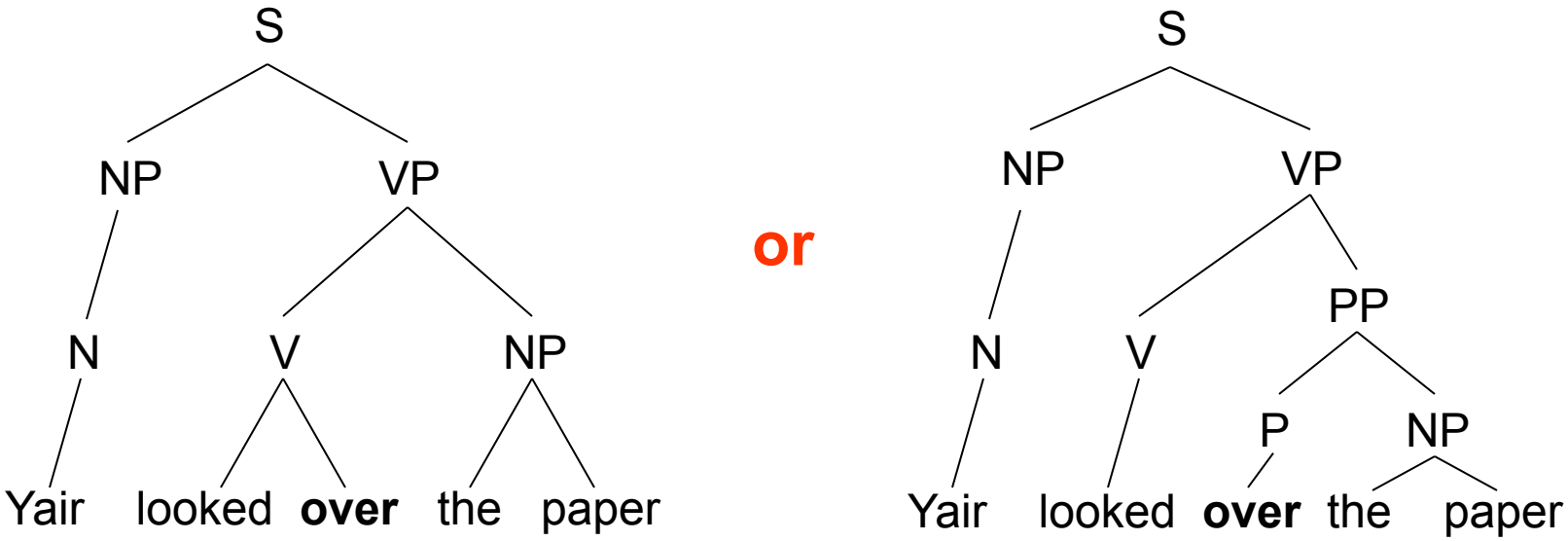


Context-Free Parsing



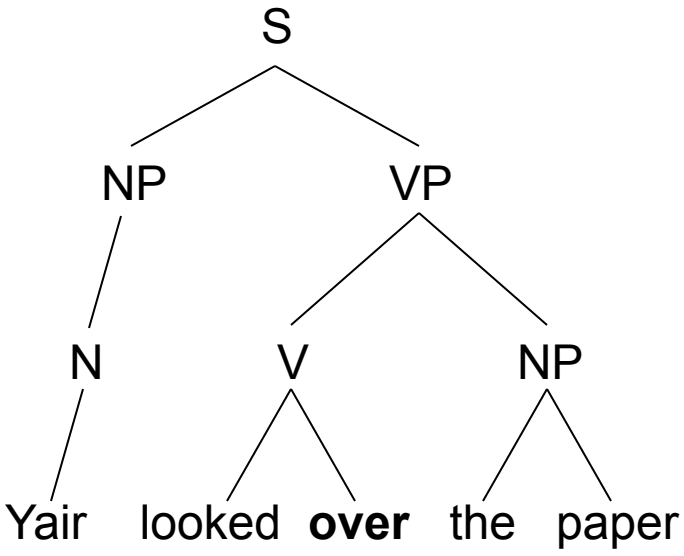
Context-Free Parsing

- What if there are multiple legal parses?
- Example: “Yair looked over the paper.”
- How does the word “over” function?



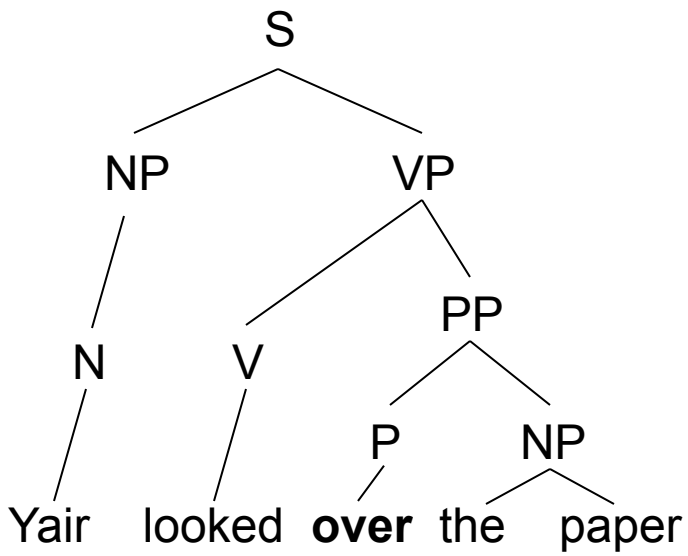
Probabilistic Parsing

- Use probabilities to find the most likely parse
- Store typical probabilities for words and rules
- In this case:



P = 0.99

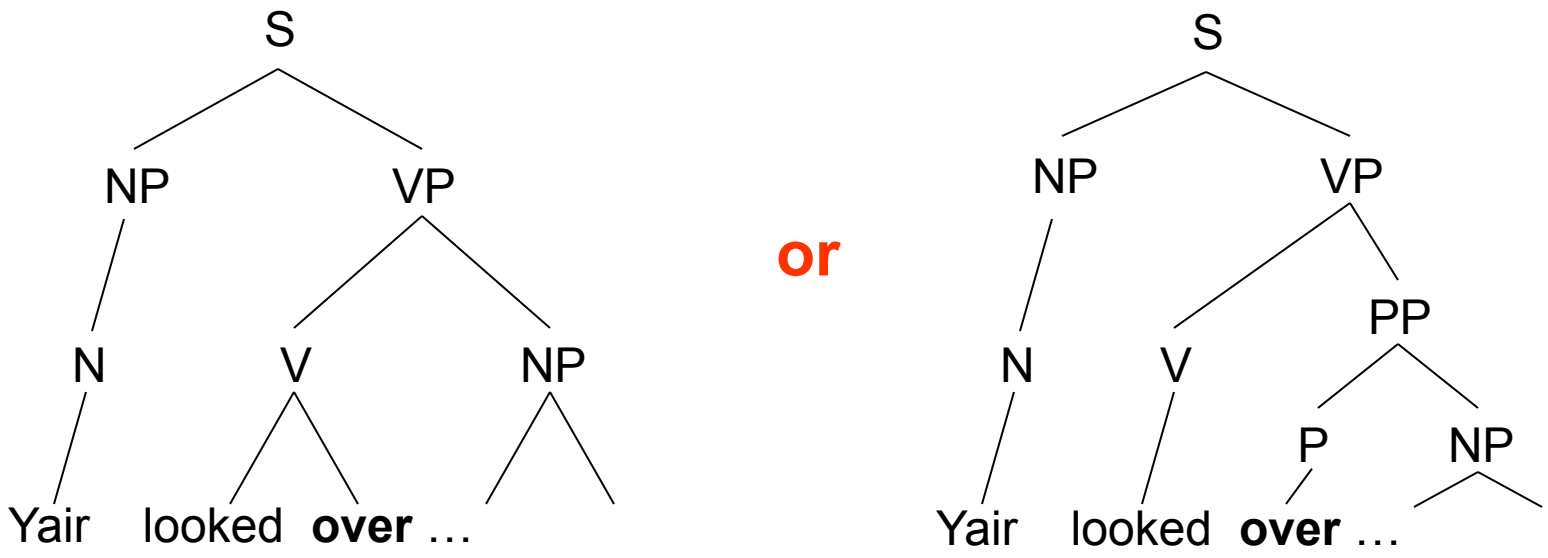
or



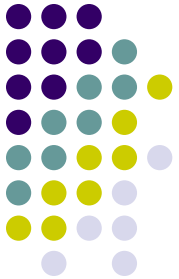
P = 0.01

Prefix Probabilities

- How likely is a partial parse?

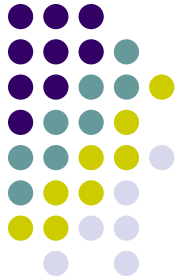


Efficiency



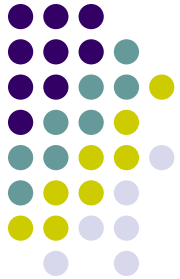
- The Earley algorithm (upon which Stolcke builds) is one of the most efficient known parsing algorithms
- Probabilities allow intelligent pruning of the developing parse tree(s)

Parsing Algorithms



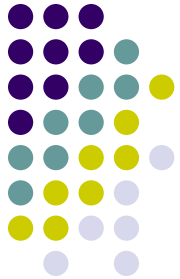
- How do we construct a parse tree?
  - Work from grammar to sentence (top-down)
  - Work from sentence to grammar (bottom-up)
  - Work from both ends at once (Earley)
- Predictably, Earley works best

Earley Parsing Overview



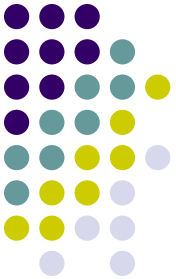
- Start with a root constituent, e.g. sentence
- Until the sentence is complete, repeatedly
  - Predict: expand constituents as specified in the grammar
  - Scan: match constituents with words in the input
  - Complete: propagate constituent completions up to their parents
- Prediction is top-down, while scanning and completion are bottom-up

Earley Parsing Overview



- Earley parsing uses a *chart* rather than a tree to develop the parse
- Constituents are stored independently, indexed by word positions in the sentence
- Why do this?
  - Eliminate recalculation when tree branches are abandoned and later rebuilt
  - Concisely represent multiple parses

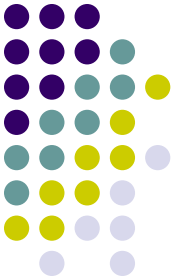
Earley Parsing Example



	the	ball	is	heavy
S	Begin			

$S \rightarrow NP\ VP \quad NP \rightarrow ART\ N \quad VP \rightarrow V\ ADJ$

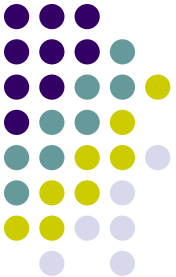
Earley Parsing Example



	the	ball	is	heavy
S	Begin			
NP	Begin			

S → NP VP NP → ART N VP → V ADJ

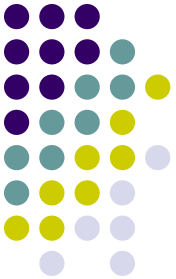
Earley Parsing Example



	the	ball	is	heavy
S	Begin			
NP	Pending			
ART	Scan			

S → NP VP NP → ART N VP → V ADJ

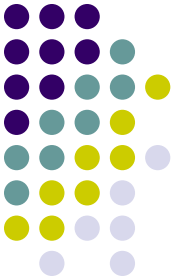
Earley Parsing Example



	the	ball	is	heavy
S	Begin			
NP		Complete		
ART	Scan			
N		Scan		

S → NP VP NP → ART N VP → V ADJ

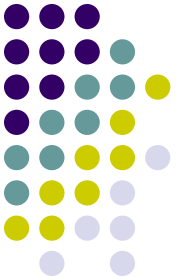
Earley Parsing Example



	the	ball	is	heavy
S		Pending		
NP		Complete		
ART	Scan			
N		Scan		

S → NP VP NP → ART N VP → V ADJ

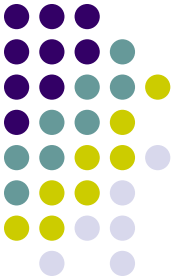
Earley Parsing Example



	the	ball	is	heavy
S		Pending		
NP		Complete		
ART	Scan			
N		Scan		
VP			Begin	

S → NP VP NP → ART N VP → V ADJ

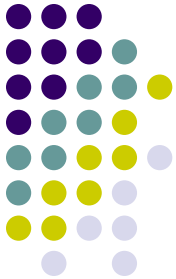
Earley Parsing Example



	the	ball	is	heavy
S		Pending		
NP		Complete		
ART	Scan			
N		Scan		
VP			Pending	
V			Scan	

S → NP VP NP → ART N VP → V ADJ

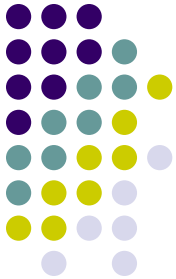
Earley Parsing Example



	the	ball	is	heavy
S		Pending		
NP		Complete		
ART	Scan			
N		Scan		
VP				Complete
V			Scan	
ADJ				Scan

S → NP VP NP → ART N VP → V ADJ

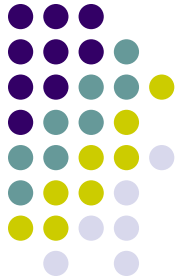
Earley Parsing Example



	the	ball	is	heavy
S				Complete
NP		Complete		
ART	Scan			
N		Scan		
VP				Complete
V			Scan	
ADJ				Scan

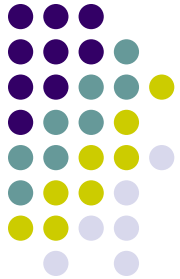
S → NP VP NP → ART N VP → V ADJ

Probabilistic Parsing



- How do we parse probabilistically?
  - Assign probabilities to grammar rules and words in lexicon
  - Grammar and lexicon “randomly” generate all possible sentences in the language
  - P(parse tree) = P(sentence generation)

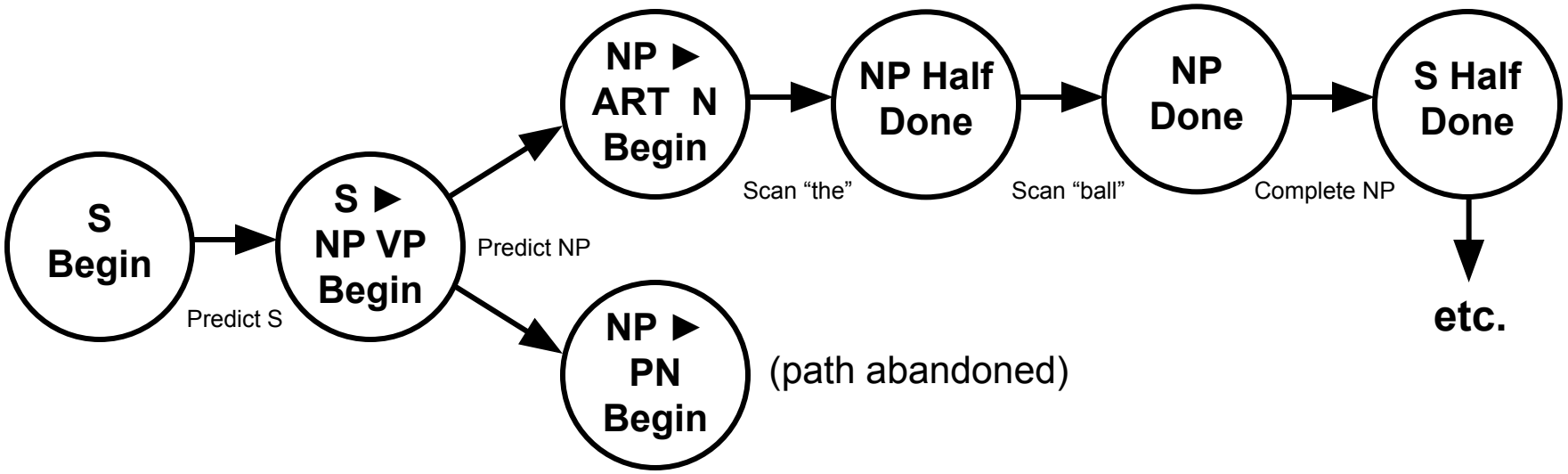
Probabilistic Parsing



- Terminology
  - Earley state: each step of the processing that a constituent undergoes. Examples:
    - Starting sentence
    - Half-finished sentence
    - Complete sentence
    - Half-finished noun phrase
    - etc.
  - Earley path: a sequence of linked states
    - Example: the complete parse just described

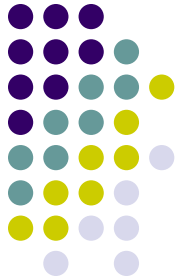
Probabilistic Parsing

- Can represent the parse as a Markov chain:



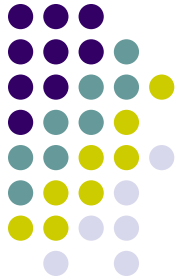
- Markov assumption (state probability is independent of path) applies, due to CFG

Probabilistic Parsing



- Every Earley path corresponds to a parse tree
- P(tree) = P(path)
- Assign a probability to each state transition
  - Prediction: probability of grammar rule
  - Scanning: probability of word in lexicon
  - Completion: accumulated (“inner”) probability of the finished constituent
- P(path) = product of P(transition)s

Probabilistic Parse Example



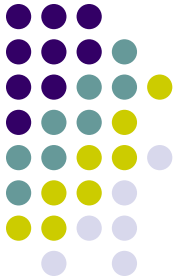
Grammar

Rule	P
S → NP VP	1.0
NP → ART N	0.7
NP → PN	0.3
VP → V NP	0.4
VP → V ADJ	0.6

Lexicon

word	PS	P
the	ART	1.0
is	V	1.0
ball	N	0.8
apple	N	0.2
heavy	ADJ	0.4
blue	ADJ	0.6

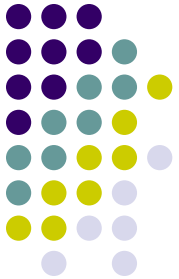
Probabilistic Parse Example



	the	ball	is	heavy	P
S	Begin				1.0

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

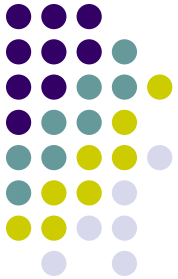
Probabilistic Parse Example



	the	ball	is	heavy	P
S	Begin				1.0
NP	Begin				0.7
NP	Begin				0.3

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

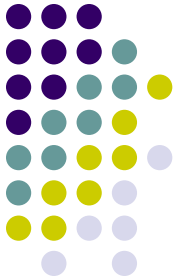
Probabilistic Parse Example



	the	ball	is	heavy	P
S	Begin				1.0
NP	Pending				0.7
NP	Failed				0.3
ART	Scan				1.0

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

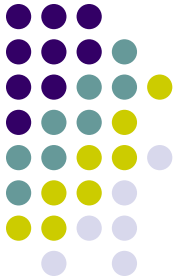
Probabilistic Parse Example



	the	ball	is	heavy	P
S	Begin				1.0
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

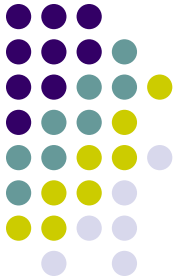
Probabilistic Parse Example



	the	ball	is	heavy	P
S		Pending			0.56
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

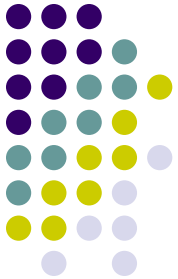
Probabilistic Parse Example



	the	ball	is	heavy	P
S		Pending			0.56
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8
VP			Begin		0.4
VP			Begin		0.6

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

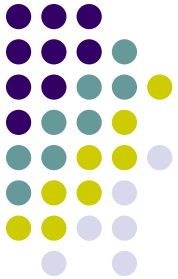
Probabilistic Parse Example



	the	ball	is	heavy	P
S		Pending			0.56
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8
VP			Pending		0.4
VP			Pending		0.6
V			Scan		1.0

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

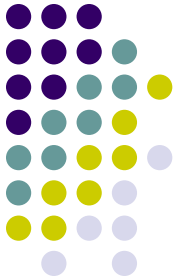
Probabilistic Parse Example



	the	ball	is	heavy	P
S		Pending			0.56
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8
VP			Pending		0.4
VP			Pending		0.6
V			Scan		1.0
NP				Begin	0.7
NP				Begin	0.3

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

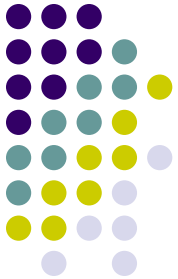
Probabilistic Parse Example



	the	ball	is	heavy	P
S		Pending			0.56
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8
VP			Pending		0.4
VP			Pending		0.6
V			Scan		1.0
NP				Failed	0.7
NP				Failed	0.3

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

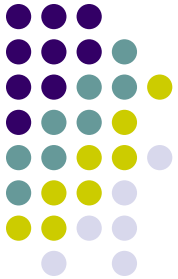
Probabilistic Parse Example



	the	ball	is	heavy	P
S		Pending			0.56
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8
VP			Failed		0.4
VP				Complete	0.24
V			Scan		1.0
ADJ				Scan	0.4

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

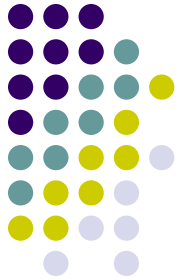
Probabilistic Parse Example



	the	ball	is	heavy	P
S				Complete	0.1344
NP		Complete			0.56
ART	Scan				1.0
N		Scan			0.8
VP				Complete	0.24
V			Scan		1.0
ADJ				Scan	0.4

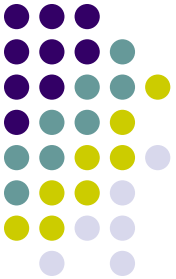
Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

Prefix Probabilities



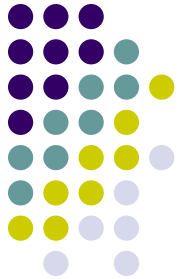
- Current algorithm reports parse tree probability when the sentence is completed
- What if we don't have a full sentence?
- Probability is tracked by constituent ("inner"), rather than by path ("forward")

Prefix Probabilities



- Solution: add a separate path probability
- Same as before, but propagate down on prediction step
- This is the missing link to chain the path probabilities together

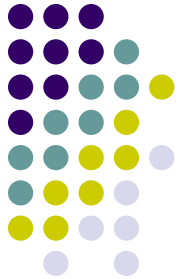
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S	Begin				1.0	1.0

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

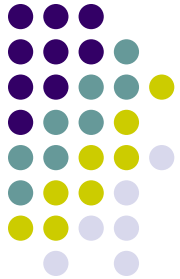
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S	Begin				1.0	1.0
NP	Begin				0.7	0.7

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

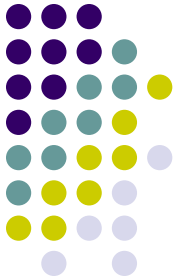
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S	Begin				1.0	1.0
NP	Pending				0.7	0.7
ART	Scan				1.0	(N/A)

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

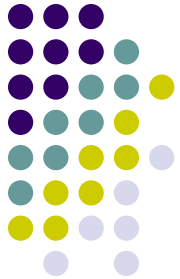
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S	Begin				1.0	1.0
NP		Complete			0.56	0.56
ART	Scan				1.0	(N/A)
N		Scan			0.8	(N/A)

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

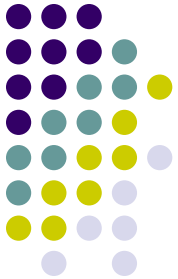
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S		Pending			0.56	0.56
NP		Complete			0.56	0.56
ART	Scan				1.0	(N/A)
N		Scan			0.8	(N/A)

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

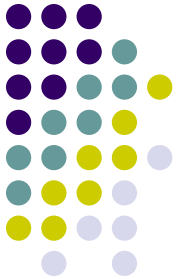
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S		Pending			0.56	0.56
NP		Complete			0.56	0.56
ART	Scan				1.0	(N/A)
N		Scan			0.8	(N/A)
VP			Begin		0.6	0.336

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

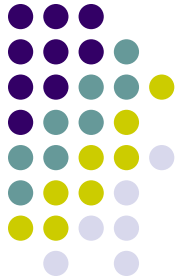
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S		Pending			0.56	0.56
NP		Complete			0.56	0.56
ART	Scan				1.0	(N/A)
N		Scan			0.8	(N/A)
VP			Pending		0.6	0.336
V			Scan		1.0	(N/A)

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

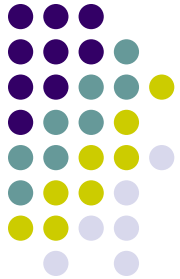
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S		Pending			0.56	0.56
NP		Complete			0.56	0.56
ART	Scan				1.0	(N/A)
N		Scan			0.8	(N/A)
VP				Complete	0.24	0.1344
V			Scan		1.0	(N/A)
ADJ				Scan	0.4	(N/A)

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

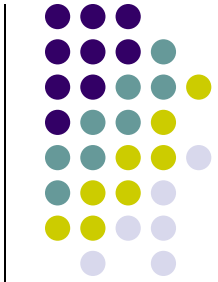
Prefix Probability Example



	the	ball	is	heavy	P_{inner}	$P_{forward}$
S				Complete	0.1344	0.1344
NP		Complete			0.56	0.56
ART	Scan				1.0	(N/A)
N		Scan			0.8	(N/A)
VP				Complete	0.24	0.1344
V			Scan		1.0	(N/A)
ADJ				Scan	0.4	(N/A)

Rule	S → NP VP	NP → ART N	NP → PN	VP → V NP	VP → V ADJ
P	1.0	0.7	0.3	0.4	0.6

Summary



- Use Earley chart parser for efficient parsing, even with ambiguous or complex sentences
- Use probabilities to choose among multiple possible parse trees
- Track constituent probability for complete sentences
- Also track path probability for incomplete sentences