

# Earley Parsing

Natalie Parde

UIC CS 421

# Earley Parsing

- Top-down dynamic parsing approach
- Table is length  $n+1$ , where  $n$  is equivalent to the number of words
- Table entries contain three types of information:
  - A subtree corresponding to a single grammar rule
  - Information about the progress made in completing the subtree
  - The position of the subtree with respect to the input

In Earley  
parsing,  
table entries  
are known  
as states.

- States include structures called **dotted rules**
- A • within the righthand side of a state's grammar rule indicates the progress made towards recognizing it
- A state's **position with respect to the input is represented by two numbers**, indicating (1) where the state begins, and (2) where its dot lies

# Example States

- Input: Book that flight.
- $S \rightarrow \bullet VP, [0,0]$ 
  - Top-down prediction for this particular kind of S
  - First 0: Constituent predicted by this state should begin at the start of the input
  - Second 0: Dot lies at the start of the input as well
- $NP \rightarrow Det \bullet Nominal, [1,2]$ 
  - NP begins at position 1
  - Det has been successfully parsed
  - Nominal is expected next
- $VP \rightarrow V NP \bullet, [0,3]$ 
  - Successful discovery of a tree corresponding to a VP that spans the entire input

# Earley Algorithm

- An Earley parser moves through the  $n+1$  sets of states in a chart in order
- At each step, one of three operators is applied to each state depending on its status
  - Predictor
  - Scanner
  - Completer
- States can be added to the chart, but are never removed
- The algorithm never backtracks
- The presence of  $S \rightarrow \alpha \bullet$ ,  $[0, n]$  indicates a successful parse

# Earley Operators: Predictor

## Predictor

- Creates new states
- Applied to any state that has a non-terminal immediately to the right of its dot (as long as the non-terminal is not a POS category)
- New states are placed into the same chart entry as the generating state
- They begin and end at the same point in the input where the generating state ends

$S \rightarrow \cdot VP, [0,0]$

- $VP \rightarrow \cdot Verb, [0,0]$
- $VP \rightarrow \cdot Verb NP, [0,0]$
- $VP \rightarrow \cdot Verb NP PP, [0,0]$
- $VP \rightarrow \cdot Verb PP, [0,0]$
- $VP \rightarrow \cdot VP PP, [0,0]$

# Earley Operators: Scanner

- Used when a state has a POS category to the right of the dot
- Examines input and incorporates a state corresponding to the prediction of a word with a particular POS into the chart
- $VP \rightarrow \bullet \text{ Verb NP}, [0,0]$ 
  - Since category following the dot is a part of speech (Verb)....
  - $\text{Verb} \rightarrow \text{book} \bullet, [0,1]$

# Earley Operators: Completer

- Applied to a state when its dot has reached the right end of the rule
- Indicates that the parser has successfully discovered a particular grammatical category over some span of input
- Finds all previously created states that were searching for this grammatical category, and creates new states that are copies with their dots advanced past the grammatical category
- $NP \rightarrow \text{Det Nominal} \bullet, [1,3]$ 
  - What incomplete states end at position 1 and expect an NP?
  - $VP \rightarrow \text{Verb} \bullet NP, [0,1]$
  - $VP \rightarrow \text{Verb} \bullet NP PP, [0,1]$
  - So, add  $VP \rightarrow \text{Verb NP} \bullet, [0,3]$  and the new incomplete  $VP \rightarrow \text{Verb NP} \bullet PP, [0,3]$  to the chart



# Earley Algorithm: Example

Chart	State	Rule	Start, End	Added By
0	S0	$\gamma \rightarrow \bullet S$	0, 0	Start State
0	S1	$S \rightarrow \bullet NP VP$	0, 0	Predictor
0	S2	$S \rightarrow \bullet VP$	0, 0	Predictor
0	S3	$NP \rightarrow \bullet Det Nominal$	0, 0	Predictor
0	S4	$VP \rightarrow \bullet Verb$	0, 0	Predictor
0	S5	$VP \rightarrow \bullet Verb NP$	0, 0	Predictor

• Book that flight.

Det  $\rightarrow$  that | this | a | the  
Noun  $\rightarrow$  book | flight | meal | money  
Verb  $\rightarrow$  book | include | prefer

S  $\rightarrow$  NP VP  
S  $\rightarrow$  VP  
NP  $\rightarrow$  Det Nominal  
Nominal  $\rightarrow$  Noun  
VP  $\rightarrow$  Verb  
VP  $\rightarrow$  Verb NP

# Earley Algorithm: Example

Book • that flight.

Det → that | this | a | the  
Noun → book | flight | meal | money  
Verb → book | include | prefer

S → NP VP  
S → VP  
NP → Det Nominal  
Nominal → Noun  
VP → Verb  
VP → Verb NP

Chart	State	Rule	Start, End	Added By
0	S0	$\gamma \rightarrow \bullet S$	0, 0	Start State
0	S1	$S \rightarrow \bullet NP VP$	0, 0	Predictor
0	S2	$S \rightarrow \bullet VP$	0, 0	Predictor
0	S3	$NP \rightarrow \bullet Det Nominal$	0, 0	Predictor
0	S4	$VP \rightarrow \bullet Verb$	0, 0	Predictor
0	S5	$VP \rightarrow \bullet Verb NP$	0, 0	Predictor
1	S6	$Verb \rightarrow book \bullet$	0, 1	Scanner
1	S7	$VP \rightarrow Verb \bullet$	0, 1	Completer
1	S8	$VP \rightarrow Verb \bullet NP$	0, 1	Completer
1	S9	$S \rightarrow VP \bullet$	0, 1	Completer
1	S10	$NP \rightarrow \bullet Det Nominal$	1, 1	Predictor

# Earley Algorithm: Example

Book that • flight.

Det → that | this | a | the  
Noun → book | flight | meal | money  
Verb → book | include | prefer

S → NP VP  
S → VP  
NP → Det Nominal  
Nominal → Noun  
VP → Verb  
VP → Verb NP

Chart	State	Rule	Start, End	Added By
0	S0	$\gamma \rightarrow \bullet S$	0, 0	Start State
0	S1	$S \rightarrow \bullet NP VP$	0, 0	Predictor
0	S2	$S \rightarrow \bullet VP$	0, 0	Predictor
0	S3	$NP \rightarrow \bullet Det Nominal$	0, 0	Predictor
0	S4	$VP \rightarrow \bullet Verb$	0, 0	Predictor
0	S5	$VP \rightarrow \bullet Verb NP$	0, 0	Predictor
1	S6	$Verb \rightarrow book \bullet$	0, 1	Scanner
1	S7	$VP \rightarrow Verb \bullet$	0, 1	Completer
1	S8	$VP \rightarrow Verb \bullet NP$	0, 1	Completer
1	S9	$S \rightarrow VP \bullet$	0, 1	Completer
1	S10	$NP \rightarrow \bullet Det Nominal$	1, 1	Predictor
2	S11	$Det \rightarrow that \bullet$	1, 2	Scanner
2	S12	$NP \rightarrow Det \bullet Nominal$	1, 2	Completer
2	S13	$Nominal \rightarrow \bullet Noun$	2, 2	Predictor

# Earley Algorithm: Example

Book that flight. •

Det → that | this | a | the  
Noun → book | flight | meal | money  
Verb → book | include | prefer

S → NP VP  
S → VP  
NP → Det Nominal  
Nominal → Noun  
VP → Verb  
VP → Verb NP

Chart	State	Rule	Start, End	Added By
0	S0	$\gamma \rightarrow \bullet S$	0, 0	Start State
0	S1	$S \rightarrow \bullet NP VP$	0, 0	Predictor
0	S2	$S \rightarrow \bullet VP$	0, 0	Predictor
0	S3	$NP \rightarrow \bullet Det Nominal$	0, 0	Predictor
0	S4	$VP \rightarrow \bullet Verb$	0, 0	Predictor
0	S5	$VP \rightarrow \bullet Verb NP$	0, 0	Predictor
1	S6	$Verb \rightarrow book \bullet$	0, 1	Scanner
1	S7	$VP \rightarrow Verb \bullet$	0, 1	Completer
1	S8	$VP \rightarrow Verb \bullet NP$	0, 1	Completer
1	S9	$S \rightarrow VP \bullet$	0, 1	Completer
1	S10	$NP \rightarrow \bullet Det Nominal$	1, 1	Predictor
2	S11	$Det \rightarrow that \bullet$	1, 2	Scanner
2	S12	$NP \rightarrow Det \bullet Nominal$	1, 2	Completer
2	S13	$Nominal \rightarrow \bullet Noun$	2, 2	Predictor
3	S14	$Noun \rightarrow flight \bullet$	2, 3	Scanner
3	S15	$Nominal \rightarrow Noun \bullet$	2, 3	Completer
3	S16	$NP \rightarrow Det Nominal \bullet$	1, 3	Completer
3	S17	$VP \rightarrow Verb NP \bullet$	0, 3	Completer
3	S18	$S \rightarrow VP \bullet$	0, 3	Completer

# Which states participate in the final parse?

Chart	State	Rule	Start, End	Added By
0	S0	$\gamma \rightarrow \bullet S$	0, 0	Start State
0	S1	$S \rightarrow \bullet NP VP$	0, 0	Predictor
0	S2	$S \rightarrow \bullet VP$	0, 0	Predictor
0	S3	$NP \rightarrow \bullet Det Nominal$	0, 0	Predictor
0	S4	$VP \rightarrow \bullet Verb$	0, 0	Predictor
0	S5	$VP \rightarrow \bullet Verb NP$	0, 0	Predictor
1	S6	$Verb \rightarrow book \bullet$	0, 1	Scanner
1	S7	$VP \rightarrow Verb \bullet$	0, 1	Completer
1	S8	$VP \rightarrow Verb \bullet NP$	0, 1	Completer
1	S9	$S \rightarrow VP \bullet$	0, 1	Completer
1	S10	$NP \rightarrow \bullet Det Nominal$	1, 1	Predictor
2	S11	$Det \rightarrow that \bullet$	1, 2	Scanner
2	S12	$NP \rightarrow Det \bullet Nominal$	1, 2	Completer
2	S13	$Nominal \rightarrow \bullet Noun$	2, 2	Predictor
3	S14	$Noun \rightarrow flight \bullet$	2, 3	Scanner
3	S15	$Nominal \rightarrow Noun \bullet$	2, 3	Completer
3	S16	$NP \rightarrow Det Nominal \bullet$	1, 3	Completer
3	S17	$VP \rightarrow Verb NP \bullet$	0, 3	Completer
3	S18	$S \rightarrow VP \bullet$	0, 3	Completer

**As with  
CKY, the  
example  
algorithm  
acted as a  
recognizer.**

- We can retrieve parse trees by adding a field to store information about the completed states that generated constituents
- How to do this?
  - Have the Completer operator add a pointer to the previous state onto a list of constituent states for the new state
  - When an S is found in the final chart, just follow pointers backward

# Which states participate in the final parse?

Chart	State	Rule	Start, End	Added By (Backward Pointer)
0	S0	$\gamma \rightarrow \bullet S$	0, 0	Start State
0	S1	$S \rightarrow \bullet NP VP$	0, 0	Predictor
0	S2	$S \rightarrow \bullet VP$	0, 0	Predictor
0	S3	$NP \rightarrow \bullet Det Nominal$	0, 0	Predictor
0	S4	$VP \rightarrow \bullet Verb$	0, 0	Predictor
0	S5	$VP \rightarrow \bullet Verb NP$	0, 0	Predictor
1	S6	<b><math>Verb \rightarrow book \bullet</math></b>	<b>0, 1</b>	<b>Scanner</b>
1	S7	$VP \rightarrow Verb \bullet$	0, 1	Completer
1	S8	$VP \rightarrow Verb \bullet NP$	0, 1	Completer
1	S9	$S \rightarrow VP \bullet$	0, 1	Completer
1	S10	$NP \rightarrow \bullet Det Nominal$	1, 1	Predictor
2	S11	<b><math>Det \rightarrow that \bullet</math></b>	<b>1, 2</b>	<b>Scanner</b>
2	S12	$NP \rightarrow Det \bullet Nominal$	1, 2	Completer
2	S13	$Nominal \rightarrow \bullet Noun$	2, 2	Predictor
3	S14	<b><math>Noun \rightarrow flight \bullet</math></b>	<b>2, 3</b>	<b>Scanner</b>
3	S15	$Nominal \rightarrow Noun \bullet$	2, 3	Completer (S14)
3	S16	$NP \rightarrow Det Nominal \bullet$	1, 3	Completer (S11, S15)
3	S17	$VP \rightarrow Verb NP \bullet$	0, 3	Completer (S6, S16)
3	S18	$S \rightarrow VP \bullet$	0, 3	Completer (S17)

# Successful Earley Parse

