

Context Free Parsing and Shared Forest Construction

CS674
Spring 2007

```
CKY( $N, \Sigma, P, R, L, w$ )
1   $n \leftarrow \text{length}(w)$ 
2  Initialize  $\text{chart}[0..n-1, 1..n] \leftarrow \emptyset$ 
3  for  $i$  from 0 to  $n-1$ 
4      do for  $\langle a, A \rangle$  in  $L$ 
5          do if  $a = w[i]$ 
6              then  $\text{chart}[i, i+1] \leftarrow \text{chart}[i, i+1] \cup \{A\}$ 
7  for  $\text{length}$  from 2 to  $n$ 
8      do for  $\text{left}$  from 0 to  $n - \text{length}$ 
9          do  $\text{right} \leftarrow \text{left} + \text{length}$ 
10             for  $\text{split}$  from  $\text{left} + 1$  to  $\text{right} - 1$ 
11                 do for  $A \rightarrow BC$  in  $P_e$ 
12                     do if  $B \in \text{chart}[\text{left}, \text{split}]$  and  $C \in \text{chart}[\text{split}, \text{right}]$ 
13                         then  $\text{chart}[\text{left}, \text{right}] \leftarrow \text{chart}[\text{left}, \text{right}] \cup \{A\}$ 
14  return  $\text{chart}$ 
```

Inside Tree Semantics for Chart

x is a grammar symbol ($x \in N$)

$x \in \text{chart}[\text{start}, \text{end}]$ if and only if the grammar licenses as a phrase a tree with root symbol x , and terminal string $w[\text{start}] \dots w[\text{end}]$

```
CKY-FILTER( $N, \Sigma, P, R, L, \text{chart}[0 \dots n - 1, 1 \dots n]$ )
1 Initialize  $\text{nchart}[0 \dots n - 1, 1 \dots n] \leftarrow \emptyset$ 
2  $\text{nchart}[0, n] \leftarrow \text{chart}[0, n] \cap R$ 
3 for  $\text{length}$  from  $n$  to 2
4   do for  $\text{left}$  from 0 to  $n - \text{length}$ 
5     do  $\text{right} \leftarrow \text{left} + \text{length}$ 
6       for  $\text{split}$  from  $\text{left} + 1$  to  $\text{right} - 1$ 
7         do for  $A \rightarrow BC$  in  $P$ 
8           do if  $B \in \text{chart}[\text{left}, \text{split}]$  and  $C \in \text{chart}[\text{split}, \text{right}]$ 
9             and  $A \in \text{nchart}[\text{left}, \text{right}]$ 
10              then  $\text{nchart}[\text{left}, \text{split}] \leftarrow \text{nchart}[\text{left}, \text{split}] \cup \{B\}$ 
11                  $\text{nchart}[\text{split}, \text{right}] \leftarrow \text{nchart}[\text{split}, \text{right}] \cup \{C\}$ 
12 return  $\text{nchart}$ 
```

Complete Tree Semantics for Chart

$x \in \text{chart}[\text{start}, \text{end}]$ iff for some tree t

- the grammar licenses t as a phrase
alternatively sentence
- the terminal string of t is w
- for some address a in $\text{dom}(t)$
 $t(a) = x$
the span of a in t is $[\text{start}, \text{end}]$

Span symbols and span rules

$\langle \text{NP}, \text{start}, \text{end} \rangle$ NP spanning start, end

$\text{DP} \rightarrow \text{DT NP}$

$\langle \text{DP}, \text{start}, \text{end} \rangle \rightarrow$

$\langle \text{DT}, \text{start}, \text{split} \rangle \langle \text{NP}, \text{split}, \text{end} \rangle$

$_0$ almost $_1$ every $_2$ fat $_3$ cat $_4$

$\langle \text{DT}, 0, 2 \rangle \quad \langle \text{NP}, 2, 4 \rangle$

$\therefore \langle \text{DP}, 0, 4 \rangle$

```

CKY-SPAN-PF( $N, \Sigma, P, R, L, \text{chart}[0\dots n - 1, 1\dots n], w$ )
1   $N', P', R', L' \leftarrow \emptyset$ 
2  for  $A$  in  $\text{chart}[0, n]$ 
3      do if  $A \in R$ 
4          then  $\text{nchart}[0, n] \leftarrow \text{nchart}[0, n] \cup \{A\}$ 
5               $N' \leftarrow N' \cup \{\langle A, 0, n \rangle\}$ 
6               $R' \leftarrow R' \cup \{\langle A, 0, n \rangle\}$ 
7  for  $\text{length}$  from  $n$  to  $2$ 
8      do for  $\text{left}$  from  $0$  to  $n - \text{length}$ 
9          do  $\text{right} \leftarrow \text{left} + \text{length}$ 
10         for  $\text{split}$  from  $\text{left} + 1$  to  $\text{right} - 1$ 
11             do for  $A \rightarrow BC$  in  $P$ 
12                 do if  $B \in \text{chart}[\text{left}, \text{split}]$  and  $C \in \text{chart}[\text{split}, \text{right}]$ 
13                     and  $A \in \text{nchart}[\text{left}, \text{right}]$ 
14                         then  $\text{nchart}[\text{left}, \text{split}] \leftarrow \text{nchart}[\text{left}, \text{split}] \cup \{B\}$ 
15                              $\text{nchart}[\text{split}, \text{right}] \leftarrow \text{nchart}[\text{split}, \text{right}] \cup \{C\}$ 
16                              $N' \leftarrow N' \cup \{\langle B, \text{left}, \text{split} \rangle, \langle C, \text{split}, \text{right} \rangle\}$ 
17                              $P' \leftarrow P' \cup \{\langle A, \text{left}, \text{right} \rangle \rightarrow \langle B, \text{left}, \text{split} \rangle \langle C, \text{split}, \text{right} \rangle\}$ 
18  for  $i$  from  $0$  to  $n - 1$ 
19      do for  $A$  in  $\text{nchart}[i, i + 1]$ 
20          do  $L' \leftarrow L' \cup \{\langle w[i], \langle A, i, i + 1 \rangle \rangle\}$ 
21  return  $\langle N', \Sigma, P', R', L' \rangle$ 

```

Mapped Grammar

Projection function $f: \langle x, \text{start}, \text{end} \rangle \mapsto x$

Use grammar $\langle \Sigma, N', R', L', \text{Start}' \rangle$ and $f: N' \rightarrow N$
to represent a set of trees with labels in N .

$\langle \Sigma, N', R', L', \text{Start}' \rangle$ represents t iff for some
tree t' ,

$\langle \Sigma, N', R', L', \text{Start}' \rangle$ licenses t' as a sentence
 $t = f \circ t'$

Relabeling

Given t' licensed by mapped grammar

$$\text{dom}(t) =_{\text{def}} \text{dom}(t')$$

$$t'(a) =_{\text{def}} f(t(a))$$

$$t'(a) = \langle \text{DP}, 0, 4 \rangle$$

$$t(a) = f(t'(a)) = \text{DP}$$

For span grammars, f projection to the grammar symbol.

Input sentence

[mr249@green Bank 4]% cat s3.in

the

guy

told

her

about

it

Grammar

```
egrep ROOT bank.gram | sort -nr | head -3  
25007  ROOT S'  
1308   ROOT SINV'  
734    ROOT NP'
```

Lexicon

```
[mr249@green Bank 11]% egrep '^ab' bank.lex  
aback  RB 1  
abacuses  NNS 1  
abandon VB 10  
abandoned  VBD 9 VBN 6  
abandoning  VBG 2  
abandons  VBZ 1  
abashed  JJ 1
```

Parsing – finding trees

[mr249@green Bank 17]% cat s3.in | lopar -in bank -trees -quiet | head -3

(ROOT (SQ (NP-SBJ (NNP the))(VP (NN guy)(PP-CLR (PP (VBN told)(NP (PRP her)))(PP (IN about)(NP (PRP it))))))

(ROOT (SQ (NP-SBJ (NN the))(VP (NN guy)(PP-CLR (PP (VBN told)(NP (PRP her)))(PP (IN about)(NP (PRP it))))))

(ROOT (SQ (NP-SBJ (JJ the))(VP (NN guy)(PP-CLR (PP (VBN told)(NP (PRP her)))(PP (IN about)(NP (PRP it))))))

...

Parsing – shared forest construction

[mr249@green Bank 20]% cat s3.in | lopar -in bank -quiet | gawk -f '{printf("%s %s",NR,\$0)}

0 ROOT 0 6 1384 1 % 1380 78 % 1378 79 %
1377 80 % 1376 81 % 1381 82 % 1373 100 %
1372 166 % 1375 174 % 1383 192 %%

1 SQ 0 6 1740 2 9 % 1740 20 27 %%

2 NP-SBJ 0 1 966 3 % 950 4 % 939 5 % 924 6
% 991 7 % 882 8 %%

9 VP 1 6 1779 10 11 %%