Parsing
Final term exam, 07.02.2017

Laura Kallmeyer

Winter 2016/2017, Heinrich-Heine-Universität Düsseldorf

Klausurdauer: 90 Minuten.

Hilfmittel: Sämtliche Unterrichtsmaterialien und Notizen in nicht-elektronischer Form.

Questions can be answered in English or in German.

Question 1 (Left Corner Parsing, 7 pts)

Consider the CFG G with non-terminals $N = \{S\}$, terminals $T = \{a, b\}$, start symbol S and productions $S \to aS \mid Sb \mid a \mid b$.

Given an input word abb, give the Left Corner Recognition trace, i.e, the set of stack triples, for this input. Indicate for each triple the operation that has lead to it together with the number of the antecedent triple and, in case of reduce, the predicted production.

We assume the following two conditions:

1. *The number of terminals on the top-down prediction stack must not increase the number of terminals on the stack of completed elements.*
2. *A reduce can only be performed if the top-down stack is not empty.*

The table starts as follows:

	Γ_{compl}	Γ_{td}	Γ_{lhs}	*operation*
1.	abb	S	–	
2.	bb	$S\$S$	S	*reduce from 1.,* $S \to aS$
3.	bb	$\$S$	S	*reduce from 1.,* $S \to a$
4.	b	$\$S\$S$	SS	*reduce from 2.,* $S \to b$
5.	Sbb	S	–	*move from 3.*
6.	Sb	$S\$S$	S	*move from 4.*

(Note that for some of the triples, there are several ways to obtain them. In this case, there is no need to repeat them, just add the additional operation into the corresponding row.)

Solution:

	Γ_{compl}	Γ_{td}	Γ_{lhs}	operation
1.	abb	S	–	
2.	bb	S$S	S	reduce from 1., $S \rightarrow aS$
3.	bb	$S	S	reduce from 1., $S \rightarrow a$
4.	b	SS	SS	reduce from 2., $S \rightarrow b$
5.	Sbb	S	–	move from 3.
6.	Sb	S$S	S	move from 4.
7.	bb	b$S	S	reduce from 5., $S \rightarrow Sb$
8.	bb	–	–	remove from 5.
9.	b	bSS	SS	reduce from 6., $S \rightarrow Sb$
10.	b	$S	S	remove from 6. or 7.
11.	b	bS	SS	reduce from 7., $S \rightarrow b$
12.	–	SS	SS	remove from 9.
13.	Sb	S	–	move from 10.
14.	Sb	b$S	S	move from 11.
15.	S	S$S	S	move from 12.
16.	b	b$S	S	reduce from 13., $S \rightarrow Sb$
17.	b	–	–	remove from 13.
18.	–	$S	S	remove from 15. or 16.
19.	S	S	–	move from 18.
20.	–	–	–	remove from 19.

Question 2 (Earley Parsing, 9 pts)

Consider again the same grammar as in the preceding exercise.

Give the chart resulting from an Earley-recognition of aab with prediction lookahead and completion lookahead:

Predict with lookahead: $\frac{[A \rightarrow \alpha \bullet B\beta, i, j]}{[B \rightarrow \bullet\gamma, j, j]} \quad B \rightarrow \gamma \in P, w_{i+1} \in First(\gamma) \text{ or } \epsilon \in First(\gamma)$

Complete with lookahead: $\frac{[A \rightarrow \alpha \bullet B\beta, i, j], [B \rightarrow \gamma\bullet, j, k]}{[A \rightarrow \alpha B \bullet \beta, i, k]} \quad w_{k+1} \in First(\beta) \text{ or } \epsilon \in First(\beta)$

Solution:

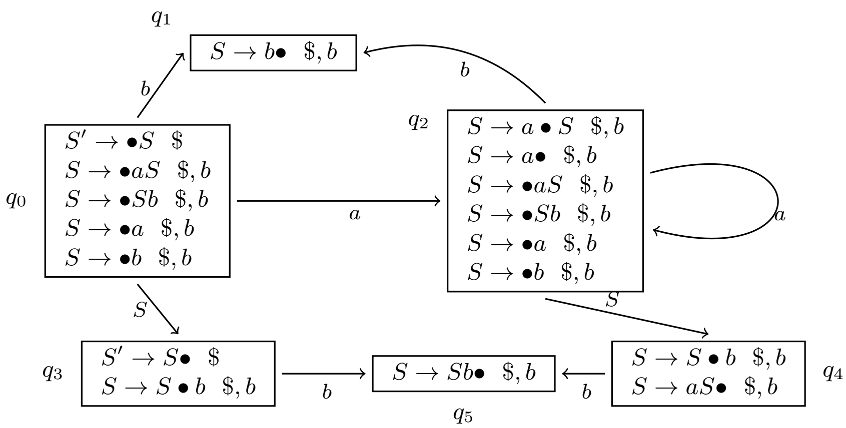
3	$S \rightarrow S \bullet b$ $S \rightarrow aS\bullet$ $S \rightarrow Sb\bullet$	$S \rightarrow aS\bullet$ $S \rightarrow S \bullet b$ $S \rightarrow Sb\bullet$	$S \rightarrow b\bullet$		
2	$S \rightarrow S \bullet b$ $S \rightarrow aS\bullet$	$S \rightarrow S \bullet b$ $S \rightarrow a \bullet S$ $S \rightarrow a\bullet$	$S \rightarrow \bullet Sb$ $S \rightarrow \bullet b$		
1	$S \rightarrow a \bullet S$ $S \rightarrow a\bullet$	$S \rightarrow \bullet a$ $S \rightarrow \bullet Sb$ $S \rightarrow \bullet aS$			
0	$S \rightarrow \bullet a$ $S \rightarrow \bullet Sb$ $S \rightarrow \bullet aS$				
	0	1	2	3	4

Question 3 (LR, 8 pts)

Consider again the grammar from the preceding questions.

Construct the LR(1)-automaton for this grammar with canonical lookahead computation.

Solution:



Question 4 (Tomita, 9 pts) *Now consider the following CFG G with non-terminals $N = \{S, T\}$, terminals $T = \{a, b\}$, start symbol S and productions*

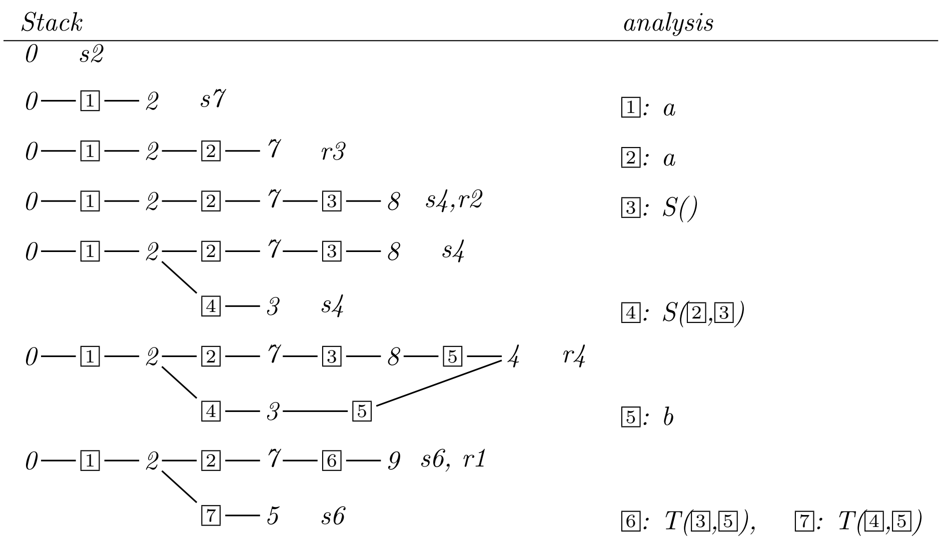
1. $S \to aT$, 2. $S \to aS$, 3. $S \to \varepsilon$, 4. $T \to Sb$, 5. $T \to Tb$.

The LR(1) parse table for this CFG is as follows:

	a	b	$	S	T
0	s2		r3	1	
1			acc		
2	s7	r3	r3	3	5
3		s4	r2		
4		r4	r4		
5		s6	r1		
6		r5	r5		
7	s7	r3	r3	8	9
8		s4, r2	r2		
9		s6, r1	r1		

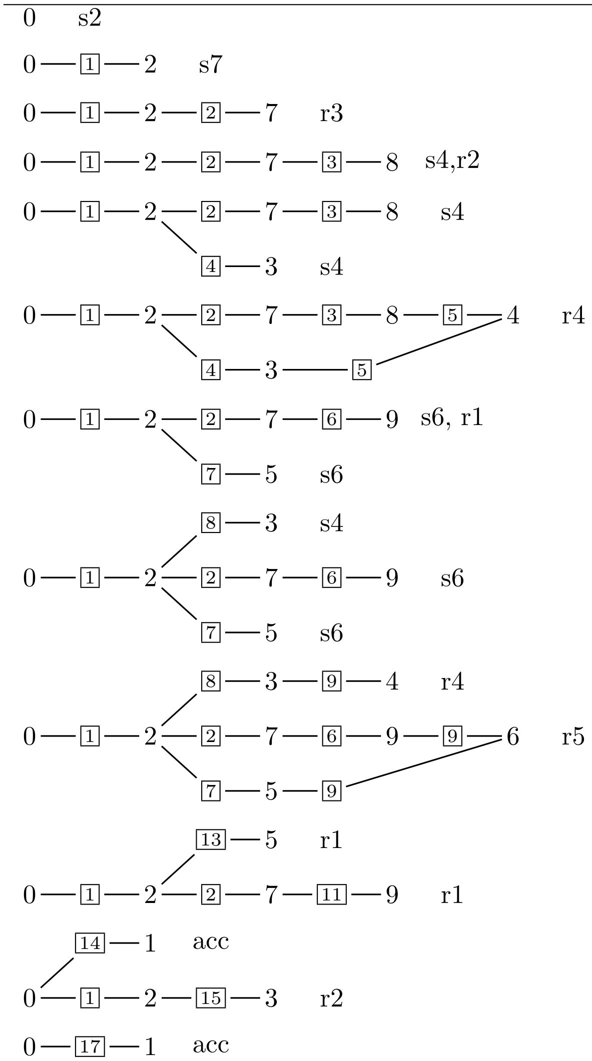
Complete the Tomita parse trace for aabb, i.e., the sequence of graph-structured stacks together with the compact parse forest.

The first steps are as follows:



Solution:

Stack	analysis



[1]: a

[2]: a

[3]: S()

[4]: S([2],[3])

[5]: b

[6]: T([3],[5]), [7]: T([4],[5])

[8]: S([2],[6])

[9]: b

[10]: T([8],[9]), [11]:T([6],[9]), [12]: T([7],[9]), [13]:[[10],[12]]

[14]: S([1],[13]), [15]:S([2],[11])

[16]: S([1],[15]), [17]:[[14],[16]]

Question 5 (PCFG, Viterbi Parsing, 5 pts) *Now consider the following CFG G with non-terminals $N = \{S, T\}$, terminals $T = \{a, b\}$, start symbol S and productions*

0.1 (-1)	$S \rightarrow AT$	*0.7 (-0.15)*	$S \rightarrow AS$
0.2 (-0.7)	$S \rightarrow a$	*0.6 (-0.22)*	$T \rightarrow SB$
0.4 (-0.4)	$T \rightarrow TB$	*1 (0)*	$A \rightarrow a$
1 (0)	$B \rightarrow b$		

Preceding each production, its probability is given together with the log_{10} value of the probability in parentheses.

Give the viterbi chart one obtains when using this PCFG in a CYK-parsing of the input aabb. Calculate with the log values instead of the probabilities. This means that instead of multiplying probabilities, you just add their log values.

Note that the higher a probability, the higher the log value and the better the item. (We are using $log_{10}(p)$, not $|log_{10}(p)|$.)

Explain in particular your computation for S and T in the field with $i = 1$ and $l = 4$.

l					
4	-2.32:S, -1.47:T				
3	-1.07:T, -1.92:S	-1.32:T			
2	-0.85:S	-0.92:T			
1	-0.7:S, 0:A	-0.7:S, 0:A	0:B	0:B	
	1	2	3	4	i

Computation for [S,1,4]: production $S \to AT$: 0-1.32-1=-2.32; production $S \to AS$: not possible.

Computation for [T,1,4]: $T \to SB$:-1.92+0-0.22=-2.14; $T \to TB$: -1.07+0-0.4=-1.47. The latter is better since $-2.14 < -1.47$.

Question 6 (SX outside estimates, 6 pts) *Consider the PCFG from slide 14–15 with its inside estimates:*

PCFG $G = \langle \{N, A\}, \{camping, car, nice, red, ugly, green, house, bike\}, P, N \rangle$ *with productions:*

$0.1(1) : N \to NN$	$0.2(0.7) : N \to AN$	$0.1(1) : N \to red$	$0.1(1) : N \to green$
$0.1(1) : N \to car$	$0.1(1) : N \to bike$	$0.2(0.7) : N \to camping$	$0.1(1) : N \to house$
$0.3(0.5) : A \to nice$	$0.25(0.6) : A \to ugly$	$0.2(0.7) : A \to red$	$0.25(0.6) : A \to green$

(The numbers preceding the productions give their probabilities p followed by the $|log_{10}(p)|$ *in parentheses.)*

The inside estimates for lengths l up to 4 are as follows:

A	0.5	∞	∞	∞
N	0.7	1.9	3.1	4.3
	1	*2*	*3*	*4* l

Compute the outside SX estimates $out(X, n_l, l, n_r)$ *for this grammar for sentence length* $n = n_l + l + n_r = 3$*, only for the non-terminal* $X = N$*. Detail your computation by giving each time the set of sums you minimize over.*

Solution:

- $l = 3$: $out(N, 0, 3, 0) = 0$
- $l = 2$:

  $out(N, 0, 2, 1) = min\{1 + 0.7 + 0\} = 1.7$

  $out(N, 1, 2, 0) = min\{1 + 0.7 + 0, 0.7 + 0.5 + 0\} = 1.2$
- $l = 1$:

  $out(N, 0, 1, 2) = min\{1 + 0.7 + 1.7, 1 + 1.9 + 0\} = 2.9$

  $out(N, 1, 1, 1) = min\{1 + 0.7 + 1.2, 1 + 0.7 + 1.4, 0.7 + 0.5 + 1.4\} = 2.6$

  $out(N, 2, 1, 0) = min\{1 + 0.7 + 1.2, 1 + 1.9 + 0, 0.7 + 0.5 + 1.2\} = 2.4$

Question 7 (A* parsing, 6 pts) *Take again the grammar from the preceding question. The outside SX estimates for sentence length 4 that we have computed on the course slides are as follows:*

- $l = 4$*:*

  $out(A, 0, 4, 0) = \infty$, $out(N, 0, 4, 0) = 0$
- $l = 3$*:*

  $out(A, 0, 3, 1) = 1.4, \quad out(A, 1, 3, 0) = \infty$

  $out(N, 0, 3, 1) = 1.7, \quad out(N, 1, 3, 0) = 1.2$
- $l = 2$*:*

  $out(A, 2, 2, 0) = \infty, \quad out(A, 1, 2, 1) = 2.6, \quad out(A, 0, 2, 2) = 2.6$

  $out(N, 0, 2, 2) = 2.9, \quad out(N, 1, 2, 1) = 2.9, \quad out(N, 2, 2, 0) = 2.9$

- $l = 1$:

  $out(A,3,1,0) = \infty$, $out(A,2,1,1) = 4.3$, $out(A,1,1,2) = 3.8$, $out(A,0,1,3) = 3.8$
  $out(N,3,1,0) = 4.1$, $out(N,2,1,1) = 4.1$, $out(N,1,1,2) = 3.8$, $out(N,0,1,3) = 3.8$

As input consider "nice green bike house".

Show the weighted deductive CYK-Parsing with chart and agenda using this grammar and input with weights as described on slide 18 (incorporating the viterbi inside score and the SX outside estimate).

Write each weight as a pair (w_i, w_o) where w_i is the inside viterbi score and w_o the outside estimate (using $|log_{10}(p)|$ instead of p).

Concerning the chart column, it is enough to list only new items in each row. (This is dfferent from the agenda where items are not only added but also removed and reordering depending on weights takes place.)

The trace starts as follows:

Chart	*Agenda*
	$(0.5, 3.8) : [A, 0, 1], (0.6, 3.8) : [A, 1, 2], (1, 3.8) : [N, 1, 2], (1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4]$
$(0.5, 3.8) : [A, 0, 1]$	$(0.6, 3.8) : [A, 1, 2], (1, 3.8) : [N, 1, 2], (1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4]$
$(0.6, 3.8) : [A, 1, 2]$	$(1, 3.8) : [N, 1, 2], (1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4]$
$(1, 3.8) : [N, 1, 2]$	$(1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4], (0.5 + 1 + 0.7, 2.9) : [N, 0, 2]$

Explain for every new item that you add to the agenda how you computed its score. More precisely, give the sum for the new score or, if there are several possibilities, the set of sums that you minimize over.

Solution:

Chart	*Agenda*
	$(0.5, 3.8) : [A, 0, 1], (0.6, 3.8) : [A, 1, 2], (1, 3.8) : [N, 1, 2], (1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4]$
$(0.5, 3.8) : [A, 0, 1]$	$(0.6, 3.8) : [A, 1, 2], (1, 3.8) : [N, 1, 2], (1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4]$
$(0.6, 3.8) : [A, 1, 2]$	$(1, 3.8) : [N, 1, 2], (1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4]$
$(1, 3.8) : [N, 1, 2]$	$(1, 4.1) : [N, 2, 3], (1, 4.1) : [N, 3, 4], (0.5 + 1 + 0.7, 2.9) : [N, 0, 2]$
$(1, 4.1) : [N, 2, 3]$	$(1, 4.1) : [N, 3, 4], (2.2, 2.9) : [N, 0, 2], (min\{1 + 1 + 1, 0.7 + 0.6 + 1\}, 2.9) : [N, 1, 3]$
$(1, 4.1) : [N, 3, 4]$	$(2.2, 2.9) : [N, 0, 2], (2.3, 2.9) : [N, 1, 3], (1 + 1 + 1, 2.9) : [N, 2, 4]$
$(2.2, 2.9) : [N, 0, 2]$	$(2.3, 2.9) : [N, 1, 3], (3, 2.9) : [N, 2, 4], (2.2 + 1 + 1, 1.7) : [N, 0, 3]$
$(2.3, 2.9) : [N, 1, 3]$	$(min\{4.2, 0.5 + 2.3 + 0.7\}, 1.7) : [N, 0, 3], (2.3 + 1 + 1, 1.2) : [N, 1, 4], (3, 2.9) : [N, 2, 4]$
$(3.5, 1.7) : [N, 0, 3]$	$(4.3, 1.2) : [N, 1, 4], (3.5 + 1 + 1, 0) : [N, 0, 4], (3, 2.9) : [N, 2, 4]$
$(4.3, 1.2) : [N, 1, 4]$	$(min\{5.5, 0.5 + 4.3 + 0.7\}, 0) : [N, 0, 4], (3, 2.9) : [N, 2, 4]$

Goal item reached, parsing stops. (There are several possible best parse trees with the same scores.)