Einführung in die Computerlinguistik
Vorbereitung der Abschlussklausur

Laura Kallmeyer

SS 2012, Heinrich-Heine-Universität Düsseldorf

Erlaubte Hilfsmittel: Eine Din-A4 Seite mit Notizen. Kein Taschenrechner.

Aufgabe 1 *Betrachten Sie die folgende CFG G : $N = \{S, A, B, C, D\}$, $T = \{a, b, c, d\}$, Startsymbol S.*

Produktionen:

$S \to abS \quad S \to dA \quad A \to B$
$B \to aBb \quad B \to \varepsilon \quad B \to C \quad D \to a$

1. *Geben Sie eine zu G äquivalente CFG G' ohne nutzlose Symbole an.*
2. *Berechnen Sie die Menge N_ε der Nichtterminalen, aus denen sich in G' ε ableiten lässt und geben Sie eine zu G' äquivalente ε-freie CFG G'' an.*
3. *Geben Sie eine zu G'' äquivalente CFG G''' ohne unäre Produktionen und ohne nutzlose Symbole.*
4. *Transformieren Sie G''' in Chomsky Normal Form.*

Lösung:

1. Produktionen in G':
   $S \to abS \quad S \to dA \quad A \to B$
   $B \to aBb \quad B \to \varepsilon$
2. $N_\varepsilon = \{B, A\}$

   Produktionen in G'':
   $S \to abS \quad S \to dA \quad S \to d \quad A \to B$
   $B \to aBb \quad B \to ab$
3. Produktionen in G'':
   $S \to abS \quad S \to dA \quad S \to d$
   $A \to aBb \quad A \to ab \quad B \to aBb \quad B \to ab$
4. Neue Produktionen:
   $S \to T_aX_1 \quad S \to T_dA \quad S \to d$
   $A \to T_aX_2 \quad A \to T_aT_b \quad B \to T_aX_2 \quad B \to T_aT_b$
   $X_1 \to T_bS \quad X_2 \to BT_b$
   $T_a \to a \quad T_b \to b \quad T_d \to d$

Aufgabe 2 *Betrachten Sie die folgenden Merkmalsstrukturen formuliert als Attribut-Wert Matrizen:*

$$S_1 = \begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{nominal} \\ \text{CASE} & \boxed{1} \\ \text{AGR} & \boxed{4} \end{bmatrix} \\ \text{PRED} & \begin{bmatrix} \textit{verbal} \\ \text{SUBJCASE} & \boxed{1} \\ \text{ASSIGNCASE} & \boxed{3} \\ \text{AGR} & \boxed{4} \\ \text{SCOMP} & \begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{nominal} \\ \text{CASE} & \boxed{3} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$S_2 = \begin{bmatrix} \textit{sentence} \\ \text{PRED} & \begin{bmatrix} \textit{ecm} \\ \text{SUBJCASE} & \text{nom} \\ \text{ASSIGNCASE} & \text{acc} \\ \text{PHON} & \text{expects} \\ \text{AGR} & \begin{bmatrix} \textit{agreement} \\ \text{NUM} & \text{sg} \\ \text{PER} & 3 \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

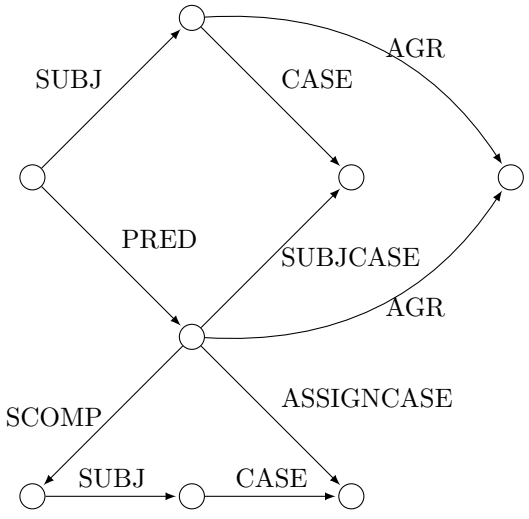
$$S_3 = \begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{pronoun} \\ \text{CASE} & \text{acc} \\ \text{PHON} & \text{him} \\ \text{AGR} & \begin{bmatrix} \textit{agreement} \\ \text{NUM} & \text{sg} \\ \text{PER} & 3 \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Typen: ecm *ist ein Untertyp von* verbal *und* pronoun *ein Untertyp von* nominal.

1. *Geben Sie für S_1 den entsprechenden Graphen an.*
2. *Berechnen Sie $S_1 \sqcup S_2$, $S_1 \sqcup S_3$ und $S_1 \sqcup S_2 \sqcup S_3$. In den Fällen, in denen keine Unifikation möglich ist (Ergebnis $\perp$) begründen Sie, warum die Unifikation scheitert.*

Lösung:

1.



2. $S_1 \sqcup S_2 =$

$$\begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{nominal} \\ \text{CASE} & \text{nom} \\ \text{AGR} & \begin{bmatrix} \textit{agreement} \\ \text{NUM} & \text{sg} \\ \text{PER} & 3 \end{bmatrix} \end{bmatrix} \\ \text{PRED} & \begin{bmatrix} \textit{ecm} \\ \text{SUBJCASE} & \text{nom} \\ \text{ASSIGNCASE} & \text{acc} \\ \text{PHON} & \text{expects} \\ \text{AGR} & \begin{bmatrix} \textit{agreement} \\ \text{NUM} & \text{sg} \\ \text{PER} & 3 \end{bmatrix} \\ \text{SCOMP} & \begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{nominal} \\ \text{CASE} & \text{acc} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$S_1 \sqcup S_3 =$

$$\begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{pronoun} \\ \text{CASE} & \text{acc} \\ \text{PHON} & \text{him} \\ \text{AGR} & \begin{bmatrix} \textit{agreement} \\ \text{NUM} & \text{sg} \\ \text{PER} & 3 \end{bmatrix} \end{bmatrix} \\ \text{PRED} & \begin{bmatrix} \textit{verbal} \\ \text{SUBJCASE} & \text{acc} \\ \text{ASSIGNCASE} & \boxed{3} \\ \text{AGR} & \begin{bmatrix} \textit{agreement} \\ \text{NUM} & \text{sg} \\ \text{PER} & 3 \end{bmatrix} \\ \text{SCOMP} & \begin{bmatrix} \textit{sentence} \\ \text{SUBJ} & \begin{bmatrix} \textit{nominal} \\ \text{CASE} & \boxed{3} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$S_1 \sqcup S_2 \sqcup S_3 = \bot$ wegen unterschiedlicher CASE Werte unter SUBJ.

Aufgabe 3 *Betrachten Sie die folgende CFG G : $N = \{S, A, B\}$, $T = \{a, b\}$, Startsymbol S.*
Produktionen:

$S \to A \quad S \to B \quad A \to bB \quad A \to aAa \quad B \to b \quad B \to bB$

Geben Sie die Trace an, die sich bei einem Top-Down-Parsing von $w = abba$ ergibt, d.h., alle Paare aus Stack und verbleibender Eingabe. Vermerken Sie jeweils, durch welche Operation und aus welchem anderen Paar ein neues Paar entstanden ist.

Lösung:

	Resteingabe	Stack	
1.	abba	S	
2.	abba	A	predict(1)
3.	abba	B	predict(1)
4.	abba	bB	predict(2)
5.	abba	aAa	predict(2)
6.	abba	b	predict(3)
7.	abba	bB	predict(3)
8.	bba	Aa	scan(5)
9.	bba	bBa	predict(8)
10.	bba	aAaa	predict(8)
11.	ba	Ba	scan(9)
12.	ba	ba	predict(11)
13.	ba	bBa	predict(11)
14.	a	a	scan(12)
15.	a	Ba	scan(13)
16.	a	ba	predict(15)
17.	a	bBa	predict(15)
18.	ε	ε	scan(14)

Aufgabe 4 *Betrachten Sie folgende kontextfreie Grammatik:* $S \rightarrow aSd|aAd$, $A \rightarrow b|bc$

1. *Geben sie ein direktionales bottem-up parsing (shift-reduce parsing) für die Eingabe* $w = aabcdd$ *an.*

   Genauer: Geben Sie den Parstrace in Form einer Tabelle an, wobei jede Zeile eine Kombination von verbleibender Eingabe und Stack ist. Zusätzlich soll in jeder Zeile vermerkt werden, durch welche Operation sich (shift oder reduce) diese Konfiguration (d.h. diese Zeile) ergeben hat, und aus welcher anderen Konfiguration sie entstanden ist.

Index	*Resteingabe*	*Stack*	*Operation*	*entstanden aus*
1.	*aabcdd*	ϵ	–	–
2.	*abcdd*	*a*	*shift*	*1*

   Gehen Sie beim Parsen davon aus, dass bei shift-reduce Konflikten zunächst die reduce Möglichkeit verfolgt wird, und nur wenn diese nicht zum Erfolg führt, die shift Alternative ausprobiert wird.

2. *Woran erkennt der Parser, dass das Wort zu der von der Grammatik generierten Sprache gehört?*

Lösung:

1.

Index	Resteingabe	Stack	Operation	entstanden aus	
1.	aabcdd	ϵ	–	–	
2.	abcdd	a	shift	1	
3.	bcdd	aa	shift	2	
4.	cdd	aab	shift	3	
5.	cdd	aaA	reduce	4	→ hier geht es nicht weiter!
6.	dd	aabc	shift	4	
7.	dd	aaA	reduce	6	
8.	d	aaAd	shift	7	
9.	d	aS	reduce	8	
10.	ϵ	aSd	reduce	9	
11.	ϵ	S	reduce	10	

2. Wenn der Stack genau ein S entählt und die Resteingabe leer ist.

Aufgabe 5

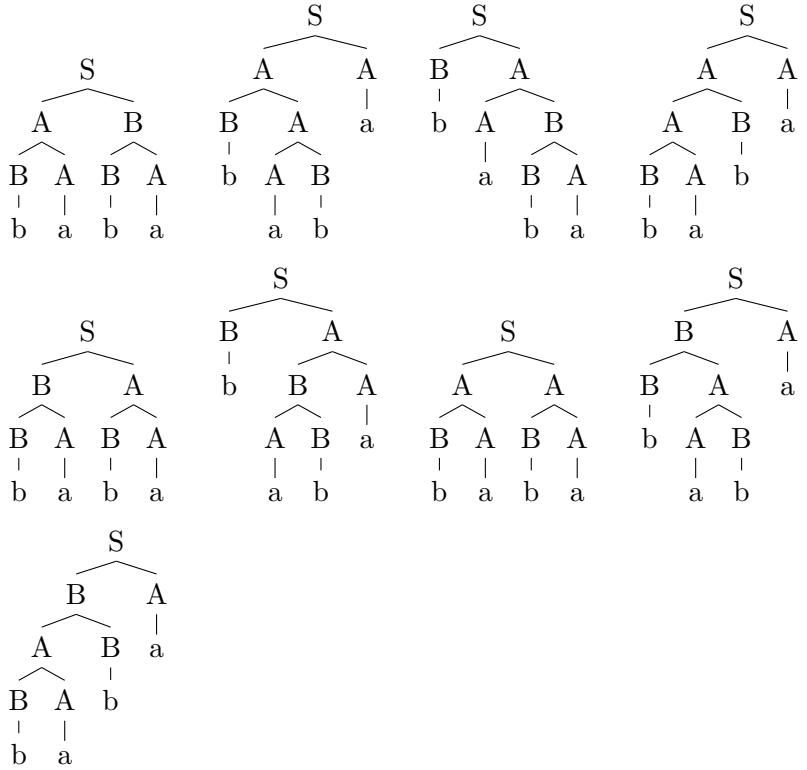
$2 \cdot 10^{-1}$: $S \to AB$ $3 \cdot 10^{-1}$: $S \to AA$ $5 \cdot 10^{-1}$: $S \to BA$
$4 \cdot 10^{-1}$: $A \to BA$ $3 \cdot 10^{-1}$: $A \to AB$
$6 \cdot 10^{-1}$: $B \to BA$ $3 \cdot 10^{-1}$: $B \to AB$
$1 \cdot 10^{-1}$: $B \to b$ $3 \cdot 10^{-1}$: $A \to a$

Betrachten Sie die Eingabe $w = baba$.

1. *Wieviel Lesarten hat baba? Geben Sie die entsprechenden Parsbäume an und berechnen Sie die Wahrscheinlichkeiten der Parsbäume.*
2. *Wie hoch ist die Insidewahrscheinlichkeit von A mit Positionen* 2, 4, *bezogen auf diese Eingabe?*
3. *Wie hoch ist die Outsidewahrscheinlichkeit von A mit Position* 2, 3, *bezogen auf diese Eingabe?*

Lösung:

1.



$p(t_1) = 2 \cdot 10^{-1} \cdot 4 \cdot 10^{-1} \cdot 1 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} \cdot 6 \cdot 10^{-1} \cdot 1 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} = 432 \cdot 10^{-7}$
$p(t_2) = 3 \cdot 10^{-1} \cdot 4 \cdot 10^{-1} \cdot 1 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} \cdot 1 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} = 324 \cdot 10^{-7}$
$p(t_3) = 5 \cdot 10^{-1} \cdot 1 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} \cdot 6 \cdot 10^{-1} \cdot 1 \cdot 10^{-1} \cdot 3 \cdot 10^{-1} = 810 \cdot 10^{-7}$
$p(t_4) = 3 \cdot 3 \cdot 3 \cdot 4 \cdot 1 \cdot 3 \cdot 1 \cdot 10^{-7} = 324 \cdot 10^{-7}$
$p(t_5) = 4 \cdot 6 \cdot 3 \cdot 1 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-7} = 648 \cdot 10^{-7}$
$p(t_6) = 4 \cdot 4 \cdot 3 \cdot 1 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-7} = 432 \cdot 10^{-7}$
$p(t_7) = 3 \cdot 4 \cdot 3 \cdot 1 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-7} = \ldots$
$p(t_8) = 5 \cdot 6 \cdot 3 \cdot 1 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-7} = \ldots$
$p(t_9) = 5 \cdot 3 \cdot 3 \cdot 1 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-7} = \ldots$

$t(w) = p(t_1) + p(t_2) + p(t_3) + p(t_4) + p(t_5) + p(t_6) + p(t_7) + p(t_8) + p(t_9)$

2. Man kann sich überlegen, dass es zwei Ableitungsteilbäume gibt, die aus A die Kette von 2 bis 4 herleiten, siehe t_3 und t_6. Die Wahrscheinlichkeiten dieser beiden Bäume müssen summiert werden, es ergibt sich also:

$$3 \cdot 6 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-5} + 4 \cdot 3 \cdot 3 \cdot 1 \cdot 3 \cdot 10^{-5} = 270 \cdot 10^{-5}$$

   Alternativ kann man auch den Inside Algorithmus soweit nötig durchrechnen.

3. Es gibt zwei Möglichkeiten, aus S die Kette bAa abzuleiten, siehe t_2 und t_8. Daher ergibt sich

$$(3 \cdot 4 + 5 \cdot 6) \cdot 1 \cdot 3 \cdot 10^{-4} = 126 \cdot 10^{-4}$$

Aufgabe 6 *Betrachten Sie nun die PCFG mit folgenden Produktionen:*

$2 \cdot 10^{-1}$: $S \to AB$ $3 \cdot 10^{-1}$: $S \to AA$ $5 \cdot 10^{-1}$: $S \to BA$
$4 \cdot 10^{-1}$: $A \to BA$ $3 \cdot 10^{-1}$: $A \to AB$
$7 \cdot 10^{-1}$: $B \to BA$ $2 \cdot 10^{-1}$: $B \to AB$
$1 \cdot 10^{-1}$: $B \to b$ $3 \cdot 10^{-1}$: $A \to a$

Geben Sie die Chart an, die sich bei einem probabilistischen CYK Parsing von $w = aba$ mit dieser Grammatik ergibt.

Lösung:

l			
3	$126 \cdot 10^{-5}$: $S \to AB$, 1 $189 \cdot 10^{-5}$: $A \to AB$, 1 $126 \cdot 10^{-5}$: $B \to BA$, 2		
2	$6 \cdot 10^{-3}$: $S \to AB$, 1 $9 \cdot 10^{-3}$: $A \to AB$, 1 $6 \cdot 10^{-3}$: $B \to AB$, 1	$15 \cdot 10^{-3}$: $S \to BA$, 1 $12 \cdot 10^{-3}$: $A \to BA$, 1 $21 \cdot 10^{-3}$: $B \to BA$, 1	
1	$3 \cdot 10^{-1}$: $A \to a$	$1 \cdot 10^{-1}$: $B \to b$	$3 \cdot 10^{-1}$: $A \to a$
	1	2	3
	a	b	a i