
Vorlesung “Grammatikformalismen”

3. Übung (24.05.2019)

Sommersemester 2019 – Prof. Dr. Alexander Koller

Verwenden Sie in dieser Übung Alto, um mit RTG- und RDG-Grammatiken zu arbeiten. Details finden Sie im Alto-Wiki:

<https://github.com/coli-saar/alto/wiki/TreeAutomata>

<https://github.com/coli-saar/alto/wiki/RDG>

1 Reguläre Baumsprachen

Betrachten Sie die kontextfreie Grammatik G_1 mit den folgenden fünf Produktionsregeln und dem Startsymbol A :

$$A \rightarrow A A \mid A B \mid B A \mid B B \quad B \rightarrow c$$

- Geben Sie eine RTG G_2 über der Signatur $\Sigma = \{a|_2, f|_2, b|_1, c|_0\}$ an, deren Sprache genau die Parsebäume von G_1 enthält. $a|_2 \in \Sigma$ bedeutet, dass das Symbol a in Σ die Arität 2 hat. (Ich schreibe der Klarheit halber die Knotenlabels im Parsebaum mit Kleinbuchstaben. Diese können Sie als Terminalsymbole in Ihrer RTG verwenden, so dass Sie für die Nichtterminalsymbole weiterhin Großbuchstaben verwenden können.)
- Wir nennen einen Baum *lokal rechtsgeklammert*, wenn jeder Knoten im Baum, der eine rechte Schwester hat, höchstens ein Kind hat. Geben Sie jeweils ein Beispiel für einen lokal rechtsgeklammerten und einen nicht lokal rechtsgeklammerten Baum aus $L(G_2)$ an.
- Geben Sie eine reguläre Baumgrammatik G_3 an, deren Sprache genau die lokal rechtsgeklammerten Bäume über Σ enthält. Sie können mit zwei Nichtterminalsymbolen auskommen.
- Geben Sie eine reguläre Baumgrammatik G_4 an mit $L(G_4) = L(G_2) \cap L(G_3)$. *Hinweis:* Analog zur Produktkonstruktion beim Schnitt von endlichen Automaten können Sie für die Nichtterminale von G_4 Paare von Nichtterminalen aus G_2 und Nichtterminalen aus G_3 nehmen und dann die Regeln von G_4 aus Paaren von Regeln in G_2 und G_3 bauen.

2 CCG

- a) Betrachten Sie die CCG-Grammatik G_5 , bestehend aus den Regeln $>$, $<$, $>\mathbf{B}$, $<\mathbf{B}$ und $<\mathbf{B}\times$ und dem folgenden Lexikon:

$$a : A/A \quad b : B\backslash B \quad c : B\backslash A \quad d : A \quad e : S\backslash B$$

Das Startsymbol der Grammatik ist S . Geben Sie CCG-Ableitungen für die Strings $acde$, $adce$ und $acabdbe$ an.

- b) Die folgende RDG-Grammatik G_6 (mit Startsymbol S) ist ein Versuch, eine zu G_5 schwach äquivalente RDG-Grammatik anzugeben:

$$\begin{aligned} A &\rightarrow \langle a, 01 \rangle(A) \\ B &\rightarrow \langle b, 10 \rangle(B) \\ B &\rightarrow \langle c, 10 \rangle(A) \\ A &\rightarrow \langle d, 0 \rangle \\ S &\rightarrow \langle e, 10 \rangle(B) \end{aligned}$$

Die Idee ist, dass die Valenz- und Wortstellungsinformationen der einzelnen lexikalischen Kategorien in den RDG-Regeln nachgebaut worden sind: Der RDG-Lexikoneintrag für e verlangt z.B. ein Argument B auf der linken Seite und gibt ein S zurück, genau wie es die Kategorie $S\backslash B$ in der CCG-Grammatik angibt.

Überprüfen Sie, ob G_5 und G_6 wirklich die gleiche Sprache beschreiben. Wenn nein, geben Sie einen String an, der in der einen, aber nicht der anderen Sprache liegt.

- c) Erklären Sie, woran es liegt, dass G_5 mehr Strings beschreibt als G_6 . Wie bringen Sie dieses Ergebnis mit der Tatsache in Einklang, dass CCG und TAG schwach äquivalent sind und jede TAG als RDG dargestellt werden kann?

3 TAG

Geben Sie eine RDG-Grammatik an, die zu Ihrer schweizerdeutschen TAG-Grammatik aus der 1. Übung äquivalent bezüglich der Übersetzung von TAG-Ableitungsbäumen in Dependenzbäume ist. Die Features, die Sie in der 1. Übung verwendet haben, können Sie für diese Aufgabe ignorieren.