Formale Methoden 1

Gerhard Jäger
Gerhard.Jaeger@uni-bielefeld.de

Uni Bielefeld, WS 2007/2008

12. Dezember 2007

Baumdiagramme

Ein Baumdiagramm eines Satzes stellt drei Arten von Information dar:

- die Konstituenten-Struktur des Satzes,
- die grammatische Kategorie jeder Konstituente, sowie
- die lineare Anordnung der Konstituenten.

Konventionen

- Ein Baum besteht aus **Knoten**, die durch
- **Kanten** verbunden werden.
- Kanten sind implizit von oben nach unten **gerichtet** (ähnlich zu Hasse-Diagrammen, wo die implizite Richtung aber von unten nach oben ist.)
- Jeder Knoten ist mit einem **Etikett** (engl. **label**) versehen.

Dominanz

- Ein Knoten x **dominiert** Knoten y wenn es eine zusammenhängende Seq von gerichteten Ästen gibt, die mit x beginnt und mit y endet.
- Für einen Baum T bildet

$$D_T = \{\langle x, y \rangle | x \text{ dominiert } y \text{ in } T\}$$

  die zugehörige **Dominanz-Relation**
- D_T ist eine schwache Ordnung, also reflexiv, transitiv und anti-symmetrisch.

Konventionen

- Wenn x nach D_T der unmittelbare Vorgänger von y ist, dann **dominiert** x y **unmittelbar**.
- Der unmittelbare Vorgänger von x bzgl. D_T heißt der **Mutterknoten** von x.
- Die unmittelbaren Nachfolger von x heißen **Tochterknoten** von x.
- Wenn zwei Knoten nicht identisch sind, aber den selben Mutterknoten haben, heißen sie **Schwesterknoten**.
- Jeder Baum hat endlich viele Knoten.
- Jeder Baum hat ein Infimum. Das Infimum heißt **Wurzel** oder **Wurzelknoten** des Baums.
- Die maximalen Elemente eines Baumes heißen **Blätter**.

Präzedenz

- Baum-Diagramme beinhalten (anders als Hasse-Diagramm) Informationen über die lineare Abfolge der Knoten.
- Knoten x **geht** Knoten y **voran** (engl. x *precedes* y) gdw. x links von y steht und keiner der beiden Knoten den anderen dominiert.
- Für einen Baum T bildet

$$\{\langle x, y\rangle | x \text{ geht } y \text{ voran}\}$$

  die zugehörige **Präzedenz-Relation**.
- P_T ist eine starke Ordnung, also irreflexiv, transitiv und asymmetrisch.

Exklusivität

In einem Baum T stehen die Knoten x und y in der Präzedenz-Relation (also $P_t(x,y)$ oder $P_t(y,x)$) gdw. sie nicht in der Dominanz-Relation stehen (also weder $D_T(x,y)$ noch $D_T(y,x)$).

Nicht-Überkreuzung

Wenn in einem Baum der Knoten x dem Knoten y vorangeht, dann geht jeder Knoten x', der von x dominiert wird, jedem Knoten y' voran, der von y dominiert wird.

Diese Bedingung schließt aus, dass

- ein Knoten mehrere Mutterknoten hat, oder dass
- sich Äste überkreuzen.

Etikettierung

Für jeden Baum T gibt es eine Etikettierungs-Funktion L_T, die jedem Knoten ein Etikett zuweist.

- L_T muss nicht injektiv sein (mehrere Knoten können das selbe Etikett tragen).
- Bei Ableitungsbäumen werden Blätter (auch **Terminal-Knoten** genannt) auf Terminalsymbole abgebildet und alle anderen Knoten auf Nichtterminal-Symbole.

Mit Hilfe dieser Eigenschaften von Bäumen können *Theoreme* bewiesen werden, also Sachverhalte, die für alle Bäume gelten. Zum Beispiel

Theorem

Wenn x und y Schwesterknoten sind, dann gilt entweder $P(x, y)$ oder $P(y, x)$.

Theorem

Die Menge der Blätter eines Baumes sind durch P total geordnet.

Grammatiken und Bäume

- Bäume repräsentieren die relevanten Aspekte einer Ableitung
- Zusammenhang zwischen Ableitung und Baum am einfachsten, wenn alle Regeln der Grammatik die Form

$$A \to \alpha$$

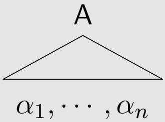
haben (mit $A \in V_N$ und $\alpha \in (V_T \cup V_N)^*$)

Grammatiken und Bäume

Definition

Eine Grammatik $G = \langle V_T, V_N, S, R \rangle$, bei der alle Regeln als linke Seite genau ein Nichtterminal-Symbol haben, **generiert** einen Baum T genau dann wenn

- die Wurzel von T mit S etikettiert ist,
- die Blätter entweder mit Terminalsymbolen oder it ϵ etikettiert sind, sowie
- es für jeden Teilbaum



in T eine Regel $A \to \alpha_1, \cdots, \alpha_n$ in R gibt.

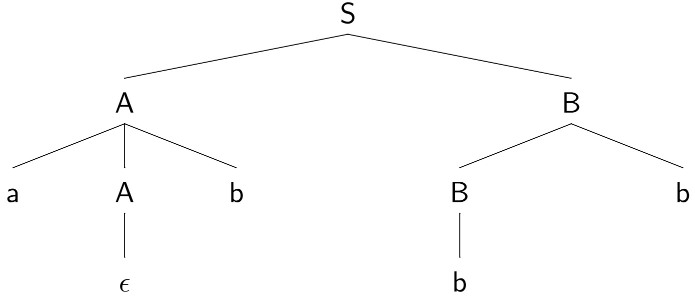
Grammatiken und Bäume

Beispiel-Grammatik

$$
\begin{array}{rcl}
G & = & \langle \{a,b\}, \{S,A,B\}, S, R \rangle \\
R & = & \left\{ \begin{array}{ll} S \rightarrow AB & B \rightarrow Bb \\ A \rightarrow aAb & B \rightarrow b \\ A \rightarrow \epsilon & \end{array} \right\}
\end{array}
$$

Grammatiken und Bäume

Diese Grammatik generiert z.B. folgende Baum:



Frage: Welche Sprache wird durch diese Grammatik generiert?

Kontext-sensitive Regeln

Manchmal möchte man die Anwendung bestimmter Regeln auf bestimmte Kontexte einschränken. Zum Beispiel:

- $D \rightarrow$ des nur wenn das nachfolgende Nomen maskulin oder neutrum singular Genitiv ist
- $/d/ \rightarrow [d]$ nur wenn das Segment nicht am Wortende steht
- [Präteritum] $\rightarrow -t-$ nur wenn direkt davor der Stamm eines schwachen Verbs steht
- ...

Aufgabe: Finde mehr Beispiele für kontext-abhängige grammatische Regeln!

Kontext-sensitive Regeln

- übliches Format für kontext-sensitive Regeln:

$$A \rightarrow \gamma/\alpha\_\beta$$

- A: Nichtterminal-Symbol
- α, β, γ: Ketten von Terminal- und Nichtterminal-Symbolen
- $\gamma \neq \epsilon$
- α_β gibt Kontext an, in dem die Regel $A \rightarrow \gamma$ angewendet werden kann
- „offizielle“ Schreibweise:

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

Die Chomsky-Hierarchie

Unterschiedliche Einschränkung für das Format von Grammatik-Regeln führt zu folgender Hierarchie von Grammatik-Typen:

Chomsky-Hierarchie

Typ 0	keine Einschränkung	
Typ 1	Regeln der Form $\alpha A \beta \rightarrow \alpha \gamma \beta$ $A \in V_N, \alpha \neq \epsilon$	*kontext-sensitive Grammatik*
Typ 2	Regeln der Form $A \rightarrow \gamma$ $A \in V_N$	*kontext-freie Grammatik*
Typ 3	Regeln der Form $A \rightarrow xB$ oder $A \rightarrow x$ $A, B \in V_N, x \in V_T^*$	*reguläre Grammatik*

Die Chomsky-Hierarchie

- keine strikte Hierachie, weil bei kontext-freien Grammatiken ϵ als rechte Seite auftreten kann, bei kontext-sensitiven Grammatiken jedoch nicht

$$\textbf{Typ 3} \subset \textbf{Typ 2} \not\subset \textbf{Typ 1} \subset \textbf{Typ 0}$$

Die Chomsky-Hierarchie

Grammatik-Hierarchie entspricht Hierarchie formaler Sprachen:

- *Typ-0-Sprachen* („rekursiv aufzählbare Sprachen“): Sprachen, die von Typ-0-Grammatik generiert werden
- *Typ-1-Sprachen* („kontext-sensitive Sprachen“): Sprachen, die von Typ-1-Grammatik generiert werden
- *Typ-2-Sprachen* („kontext-freie Sprachen“): Sprachen, die von Typ-2-Grammatik generiert werden
- *Typ-3-Sprachen* („reguläre Sprachen“): Sprachen, die von Typ-3-Grammatik generiert werden

Theorem

Wenn L eine kontext-freie Sprache ist und $\epsilon \notin L$, dann ist L auch eine kontext-sensitive Sprache.

Die Chomsky-Hierarchie

- Alle kontext-freien und kontext-sensitiven Sprachen sind **entscheidbar** — es gibt für jede dieser Sprachen ein Computerprogramm, das in endlicher Zeit entscheidet, ob eine gegebene Kette zu der Sprache gehört oder nicht.
- Rekursiv aufzählbare Sprachen sind nicht immer entscheidbar. Zum Beispiel bildet die Menge aller beweisbaren mathematischen Aussagen eine rekursiv aufzählbare Sprache, die nicht entscheidbar ist.
- Kontext-freie Sprachen können vom Computer effizient verarbeitet werden (maximal kubische Zeitkomplexität).
- Reguläre Sprachen können vom Computer sehr effizient verarbeitet werden (lineare Zeitkomplexität).
- Kontext-sensitive Sprachen sind im allgemeinen Fall nicht effizient verarbeitbar.

Die Chomsky-Hierarchie

- 1957 (Chomsky): Nachweis, dass Englisch keine reguläre Sprache ist.
- 1957 (Chomsky): Vermutung, dass natürliche Sprachen generell nicht kontext-frei, sondern kontext-sensitiv sind.
- 1982 (Pullum & Gazdar): „English as a context-free language" — Argumente, dass Englisch (und alle anderen nat. Sprachen) kontext-frei sind.
- 1984 (Huybregts), 1985 (Shieber): Beweis, dass Schweizerdeutsch nicht kontext-frei ist
- 1985 (Culy): Beweis, dass Bambara (westafrikanische Sprache) nicht kontext-frei ist
- Die meisten phonologischen und morphologischen Prozesse in natürlichen Sprachen können durch reguläre Grammatiken erfasst werden.

Die Chomsky-Hierarchie