Automaten und Formale Sprachen Tutorium

Grammatiken

Christopher Blöcker (inf8871)

FH Wedel, SS 2010

Beispiel : Bezeichner in Pascal (BNF)

<BEZEICHNER>	::=	<BUCHSTABE> (<BUCHSTABE> | <ZIFFER>)*
<BUCHSTABE>	::=	**'A' | 'B' | 'C' | 'D' | 'E' | 'F' | 'G' | 'H' | 'I' | 'J' | 'K' | 'L' | 'M' | 'N' | 'O' | 'P' | 'Q' | 'R' | 'S' | 'T' | 'U' | 'V' | 'W' | 'X' | 'Y' | 'Z**
<ZIFFER>	::=	**'0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9'**

Definition : Grammatik

Eine Grammatik ist ein 4-Tupel.

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S})$$

mit

$\mathcal{N}$	Endliche, nicht-leere Menge der Nicht-Terminale
$\mathcal{T}$	Endliche, nicht-leere Menge der Terminale $\mathcal{N} \cap \mathcal{T} = \emptyset$
$\mathcal{R}$	Menge der Produktionen mit $\mathcal{R} \subseteq (\mathcal{N} \cup \mathcal{T})^+ \times (\mathcal{N} \cup \mathcal{T})^*$
$\mathcal{S}$	Das Startsymbol mit $\mathcal{S} \in \mathcal{N}$

$\mathcal{L} = \{w \in \Sigma^* \mid w = a^n b^n, n \in \mathbb{N}\}$

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S})$$

mit

$$\begin{array}{lcl} \mathcal{N} & = & \{S\} \\ \mathcal{T} & = & \{a, b\} \\ \mathcal{R} & : & S \to \epsilon \\ & & S \to aSb \\ \mathcal{S} & = & S \end{array}$$

Anmerkungen

- Die Regeln $S \to \epsilon$ und $S \to aSb$ können zusammengefasst werden zu $S \to \epsilon \mid aSb$.
- Die Zeile $\mathcal{S} = S$ ist eigentlich überflüssig.

Sprachklassen

Die Menge aller Sprachen wurde durch Chomsky eingeteilt in die Chomsky-Hierarchie mit den 4 Sprachklassen

- Typ-3 : Reguläre Sprachen $\mathcal{L}_3$
- Typ-2 : Kontext-Freie Sprachen $\mathcal{L}_2$
- Typ-1 : Kontext-Sensitive Sprachen $\mathcal{L}_1$
- Typ-0 : Rekursiv Aufzählbare Sprachen $\mathcal{L}_0$

Chomsky Hierarchie

$$\mathcal{L}_3 \subsetneq \mathcal{L}_2 \subsetneq \mathcal{L}_1 \subsetneq \mathcal{L}_0$$

Grammatiken

Jede dieser Sprachklassen lässt sich durch Grammatiken des jeweiligen Types erzeugen.
Dabei unterliegen die Produktionen gewissen Restriktionen.

Typ-3 Grammatik (Reguläre Sprachen)

Sei

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S}).$$

$\mathcal{G}$ ist vom Typ-3, wenn die Produktionen folgende Form haben:

$$\begin{aligned} A &\rightarrow \epsilon \\ A &\rightarrow a \\ A &\rightarrow aB \end{aligned}$$

mit

- $A, B \in \mathcal{N}$
- $a \in \mathcal{T}$

Die hier gezeigte Grammatik ist rechtslinear.
(Der Ableitungsbaum wächst nach rechts.)

Typ-3 Grammatik (Reguläre Sprachen)

Zu jeder rechtslinearen Typ-3-Grammatik gibt es eine linkslineare Grammatik, die dieselbe Sprache erzeugt.
Rechts- und linkslineare Regeln dürfen nicht vermischt werden, sonst können auch Sprachen erzeugt werden, die nicht vom Typ-3 sind.

$\mathcal{L} = \{w \in \Sigma^* \mid w = a^n b^n, n \in \mathbb{N}\}$

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S})$$

mit

$$\begin{array}{lcl} \mathcal{N} & = & \{S\} \\ \mathcal{T} & = & \{a, b\} \\ \mathcal{R} & : & S \rightarrow aA \mid \epsilon \\ & & A \rightarrow Sb \\ \mathcal{S} & = & S \end{array}$$

Typ-2 Grammatik (Kontext-freie Sprachen)

Sei

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S}).$$

$\mathcal{G}$ ist vom Typ-2, wenn die Produktionen folgende Form haben:

$$A \quad \rightarrow \quad w$$

mit

- $A \in \mathcal{N}$
- $w \in (\mathcal{N} \cup \mathcal{T})^*$

Typ-1 Grammatik (Kontext-sensitive Sprachen)

Sei

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S}).$$

$\mathcal{G}$ ist vom Typ-1, wenn die Produktionen folgende Form haben:

$$uAv \quad \rightarrow \quad u\phi v$$

mit

- $A \in \mathcal{N}$
- $u, v \in (\mathcal{N} \cup \mathcal{T})^*$
- $\phi \in (\mathcal{N} \cup \mathcal{T})^+$

Typ-0 Grammatik (Rekursiv aufzählbare Sprachen)

Sei

$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S}).$$

$\mathcal{G}$ ist vom Typ-0, wenn die Produktionen folgende Form haben:

$$u \quad \to \quad v$$

mit

- $u \in (\mathcal{N} \cup \mathcal{T})^+$
- $v \in (\mathcal{N} \cup \mathcal{T})^*$

Erzeugte Sprache

Sei
$$\mathcal{G} = (\mathcal{N}, \mathcal{T}, \mathcal{R}, \mathcal{S}).$$

Die von $\mathcal{G}$ erzeugte Sprache ist

$$\mathcal{L}(\mathcal{G}) = \{w \in \Sigma^* \mid \mathcal{S} \xrightarrow{*} w\}.$$

"$\xrightarrow{*}$" bezeichnet eine Folge von Ableitungsschritten.

Definition : Ableitungsschritt

Ein Ableitungsschritt ist das Anwenden einer Produktion auf eine Satzform, z.B.:

- $r_1 : A \rightarrow aBc$
- $w = xyz\underline{A}zyx$

$w \xrightarrow{r_1} w'$ mit $w' = xyz\underline{aBc}zyx$.

Definition : Mehrdeutigkeit

Eine Grammatik ist mehrdeutig, wenn es ein Wort gibt, für das unterschiedliche Ableitungsbäume existieren.
Ob eine Grammatik mehrdeutig ist, ist im Allgemeinen nicht entscheidbar. (D.h. es kann für eine konkrete Grammatik entscheidbar sein, ob sie mehrdeutig ist, es existiert aber kein Algorithmus, der diese Frage für jede Grammatik entscheidet.)

Nicht-Determinismus in Grammatiken

Grammatiken sind nicht-deterministisch, da die angewendeten Produktionsregeln zufällig ausgewählt werden.
Soll eine Grammatik deterministisch sein, so darf es immer nur eine Produktionsregel zur Auswahl geben.

Exkurs

Feinere Unterteilung der Sprachklassen

Vereinbarung

Im Folgenden bezeichne

- $\mathcal{L}_{i,D}$ die Menge der Deterministischen Typ-i-Sprachen
- $\mathcal{L}_{i,N}$ die Menge der Nicht-Deterministischen Typ-i-Sprachen

Reguläre Sprachen (Typ-3)

$$\mathcal{L}_{3,D} = \mathcal{L}_{3,N}$$

Dies ist leicht einzusehen, da zu jedem **NEA** ein äquivalenter **DEA** angegeben werden kann.

Kontext-freie Sprachen (Typ-2)

$$\mathcal{L}_{2,D} \subsetneq \mathcal{L}_{2,N}$$

NKA sind mächtiger als **DKA**.

Kontext-Sensitive Sprachen (Typ-1)

$$\mathcal{L}_{1,D} \overset{?}{\subseteq} \mathcal{L}_{1,N}$$

Es ist bisher nicht bekannt, ob die Inclusion echt oder unecht ist.

Rekursiv Aufzählbare Sprachen (Typ-0)

$$\mathcal{L}_{0,D} = \mathcal{L}_{0,N}$$

NDTM können durch **DTM** simuliert werden, was im Allgemeinen exponentielle Zeit benötigt.

Rekursiv Aufzählbare Sprachen $\mathcal{L}_{re}$

Sei $\mathcal{L} \subseteq \Sigma^*, x \in \mathcal{L}, \mathcal{M}$ eine **TM** mit $\mathcal{L}_\mathcal{M} = \mathcal{L}$.
$\mathcal{L}$ ist Rekursiv Aufzählbar, wenn gilt

$$\mathcal{M}(x) = \begin{cases} 1 & \text{wenn } x \in \mathcal{L} \\ \mathit{undef} & \text{sonst} \end{cases}$$

$\mathcal{M}$ **erkennt** $\mathcal{L}$.

Rekursive Sprachen $\mathcal{L}_r$

Sei $\mathcal{L} \subseteq \Sigma^*, x \in \mathcal{L}, \mathcal{M}$ eine **TM** mit $\mathcal{L}_\mathcal{M} = \mathcal{L}$.
$\mathcal{L}$ ist Rekursiv wenn gilt

$$\mathcal{M}(x) = \begin{cases} 1 & \text{wenn } x \in \mathcal{L} \\ 0 & \text{wenn } x \notin \mathcal{L} \end{cases}$$

$\mathcal{M}$ **entscheidet** $\mathcal{L}$.

Zusammenhang

Es gilt

$$\mathcal{L}_3 \subsetneq \mathcal{L}_2 \subsetneq \mathcal{L}_1 \subsetneq \mathcal{L}_0.$$

Außerdem gilt

- $\mathcal{L}_{3,D} = \mathcal{L}_{3,N} = \mathcal{L}_3$
- $\mathcal{L}_{2,D} \subsetneq \mathcal{L}_{2,N} = \mathcal{L}_2$
- $\mathcal{L}_{1,D} \overset{?}{\subseteq} \mathcal{L}_{1,N} = \mathcal{L}_1$
- $\mathcal{L}_{0,D} = \mathcal{L}_{0,N} = \mathcal{L}_0$
- $\mathcal{L}_r \subsetneq \mathcal{L}_{re} = \mathcal{L}_0$

$\Rightarrow$

$$\mathcal{L}_{3,D} = \mathcal{L}_{3,N} = \mathcal{L}_3 \subsetneq \mathcal{L}_{2,D} \subsetneq \mathcal{L}_{2,N} = \mathcal{L}_2 \subsetneq \mathcal{L}_{1,D} \overset{?}{\subseteq} \mathcal{L}_{1,N} = \mathcal{L}_1$$

$$\mathcal{L}_1 \subsetneq \mathcal{L}_r \subsetneq \mathcal{L}_{0,D} = \mathcal{L}_{0,N} = \mathcal{L}_{re} = \mathcal{L}_0$$