6. Formale Grammatiken

Idee: Konstruktion aller Wörter in einer Sprache

Beispiel: $L = \{0^n\}_{n \in \mathbb{N}_0}$

S Startsymbol

$S \to \lambda$, $S \to 0S$ Regel (S, λ), $(S, 0S)$

$S \to 0S \to 00S \to 000S \to 000$

Definition 6.1 (Grammatiken)

Eine Grammatik ist ein Tupel $G = (N, T, P, S)$. Dabei ist

- N das Alphabet der Nichtterminalsymbole/Variablen,
- T das Alphabet der Terminalsymbole mit $N \cap T = \emptyset$,
- $P \subseteq ((N \cup T)^* \setminus T^*) \times (N \cup T)^*$ eine endliche Menge von Regeln/Produktionen, wobei wir für ein Paar $(u, v) \in P$ auch $u \to v$ schreiben
- $S \in N$ das Startsymbol.

Eine Satzform von G ist ein Wort $s \in (N \cup T)^*$ und ein Terminalwort von G ist ein Wort $t \in T^*$.

Definition 6.2 (Ableitung)

Sei $G=(N,T,P,S)$ eine Grammatik. Eine Satzform w' von G ist in einem Schritt aus einer Satzform w von G ableitbar, wenn es Satzformen u, v, x, y von G gibt, so dass $w=xuy$, $u \to v \in P$ und $w'=xvy$ gelten. Es bezeichne $\to_G$ die Relation auf der Menge der Satzformen von G, so dass $w \to_G w'$ genau dann für Satzformen von G gilt, wenn w' aus w in einem Schritt ableitbar ist.

Für Satzformen u, v von G ist eine Ableitung von v aus u eine Folge $u=w_1, \ldots, w_n=v$ mit $w_i \to_G w_{i+1}$ $\forall i \in [n-1]$ und eine Ableitung von v in G ist eine Ableitung von v aus S in G. Für $n \in \mathbb{N}$ schreiben wir $u \to_G^n v$ wenn es eine Ableitung von v aus u der Länge n gibt und wir schreiben $u \to_G^* v$ wenn eine Ableitung von v aus u in G existiert.

Definition 6.3 (erzeugte Sprache)

Sei $G=(N,T,P,S)$ eine Grammatik. Die von G erzeugte Sprache $L(G)$ ist die Menge aller Wörter $w \in T^*$ für die es eine Ableitung von w in G gibt.

Lemma 6.4

Sei $G=(N,T,P,S)$ eine Grammatik und seien u, v, w, x, y Satzformen von G und seien $n, m \in \mathbb{N}$ mit $u \xrightarrow[G]{m} v$ und $w \xrightarrow[G]{n} xuy$.

Dann gilt $w \xrightarrow[G]{m+n-1} xvy$.

Beweis: Sei $\alpha_1, \ldots, \alpha_n = xuy$ eine Ableitung von xuy aus w und $u = \beta_1, \ldots, \beta_m = v$ eine Ableitung von v aus u. Dann ist

$$w = \alpha_1, \ldots, \alpha_{n-1}, x\beta_1 y, \ldots, x\beta_m y = xvy.$$

eine Ableitung von xvy aus w in G der Länge $n+m-1$. $\square$

Im folgenden beschäftigen wir uns mit dem Thema welche Sprache Grammatiken verschiedener Komplexitätsstufen erzeugen können.

Satz GS

Eine Sprache ist genau dann rek. aufzählbar, wenn sie von einer Grammatik erzeugt wird.

Beweisidee:

Wird eine Sprache L von einer Grammatik G erzeugt, so ist L die erkannte Sprache einer TM, die in geeigneter Weise Ableitungen von G erzeugt, prüft ob diese Ableitung dem Wort der Eingabe entspricht und gegebenenfalls akzept. wenn eine Ableitung der Eingabe gefunden ist.

Gegeben eine rek. aufzählbare Sprache L und eine TM M, die L erkennt. So konstruieren wir ähnlich dem Postschen Korrespondenzproblems Regeln und

Symbole, sodass wir die Arbeitsweise der TM modellieren können und entsprechend mit einem Terminalwort enden wenn dies von der TM erkannt wird. $\square$

Definition 6.6 (rechtslinear)

Eine Grammatik $G = (N, T, P, S)$ ist rechtslinear, wenn alle Regeln von der Form

$$X \to uY \quad \text{oder} \quad X \to u$$

mit $X, Y \in N$ und $u \in T^*$ sind.

Hier ist es sinnvoll endliche Automaten zu betrachten bei denen es nicht $\forall$ Zustände q und Eingabesymbole a ein Tripel (q, a, q') in der Übergangsrelation geben muss. Solche Automaten sind zwangsläufig nicht det.

Satz 6.7

Eine Sprache ist genau dann regulär, wenn sie von einer rechtslin. Grammatik erzeugt wird.

Beweisidee:

Zunächst überzeugt man sich davon, dass eine Sprache L genau dann von einer rechtslin. Grammatik erzeugt wird wenn sie von einer Grammatik $G = (N, T, P, S)$ erzeugt wird bei der alle Regeln von der Form

$$X \to aY \quad \text{oder} \quad X \to \lambda$$

mit $X, Y \in N$ und $a \in T$ sind. Eine solche Grammatik wird als Grammatik in Simulationsform bezeichnet.

$X \to abY$ x streichen

$X \to aX_1$ hinzufügen

$X_1 \to bY$

Die Sprache L, die von einer rechtslin Grammatik (in Simulationsform) gebildet wird, wird von dem EA

$$A = (N, T, \Delta, S, \{X \in N : X \to \lambda \in P\})$$

mit

$$\Delta = \{(X, a, Y) \in N \times T \times N : X \to aY \in P\}$$

erkannt.

Umgekehrt ist es einfach zu sehen, dass jede reg. Sprache von einer rechtslin. Grammatik erzeugt wird. $\square$

Beispiel 6.8

Die Sprache $\{0\}^*$ wird von der rechtslin. Grammatik

$$G = (\{S\}, \{0,1\}^*, \{S \to 0S, S \to \lambda\}, S)$$

erzeugt und vom EA

$$A = (\{S\}, \{0,1\}^*, \{(S,0,S)\}, S, \{S\})$$

erkannt.

Sowohl auf der Seite der Maschinenmodelle als auch auf der Seite der Grammatiken gibt es weitere wichtige Sprachklassen, die sich ergeben wenn die Maschinenarbeitsweise oder die Menge der zulässigen Regeln weniger stark eingeschränkt wird als bei EA und rechtslin. Grammatiken.

Definition 6.9 (kontextfrei)

Eine Grammatik $G = (N, T, P, S)$ ist kontextfrei, wenn alle Regeln von G von der Form

$$X \to w$$

mit $X \in N$ und $w \in (N \cup T)^*$ sind. Eine Sprache ist kontextfrei, wenn sie von einer kontextfreien Grammatik erzeugt wird. Die Menge aller kontextfreien Sprachen bezeichnen wir mit **CF**.

Definition 6.10 (Lexikographische Ordnung)

Sei Σ ein Alphabet. Die lexikographische Ordnung auf Σ^* ist die lineare Ordnung $\leq$ auf Σ^* für die $u \leq v$ für $u, v \in \Sigma^*$ genau dann gilt wenn eine der folgenden Bed. erfüllt ist.

(L1) $u \sqsubseteq v$.

(L2) Es gibt ein $i \in [\min\{|u|, |v|\}]$ mit $u(j) = v(j)$ $\forall j \in [i-1]$ und $u(i) \neq v(i)$ und $u(i) \leq v(i)$ für eine geg. lin. Ordnung $\leq$ auf Σ.