

Lernzettel

Grammatiken: Typen, Generierung und Ableitungsregeln

Universität: Technische Universität Berlin
Kurs/Modul: Theoretische Grundlagen der Informatik
Erstellungsdatum: September 19, 2025



Zielorientierte Lerninhalte, kostenlos!
Entdecke zugeschnittene Materialien für deine Kurse:

<https://study.AllWeCanLearn.com>

Theoretische Grundlagen der Informatik

Lernzettel: Grammatiken: Typen, Generierung und Ableitungsregeln

(1) Grundbegriffe. Eine Grammatik G besteht aus vier Elementen: N (Nichtterminale), Σ (Terminalsymbole), P (Produktionsregeln) und Startsymbol S . Die von G erzeugte Sprache ist

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

Die Relation $\Rightarrow^*$ bedeutet eine Folge von Ableitungen unter den Regeln aus P . Eine Produktion hat die Form

$$A \rightarrow \alpha, \quad A \in N, \alpha \in (N \cup \Sigma)^*.$$

(2) Typen der Grammatiken (Chomsky-Hierarchie). - Typ-3 (reguläre Grammatiken): Regeln der Form

$$A \rightarrow a \quad \text{oder} \quad A \rightarrow aB, \quad A, B \in N, a \in \Sigma.$$

Diese Sprachen sind regulär und können durch DFA/NFA erkannt werden.

- Typ-2 (kontextfreie Grammatiken): Regeln der Form

$$A \rightarrow \gamma, \quad A \in N, \gamma \in (N \cup \Sigma)^*.$$

Diese Sprachen sind kontextfrei und werden durch Kellerautomaten erkannt.

- Typ-1 (kontext-sensitive Grammatiken): Regeln der Form

$$uAv \rightarrow u\gamma v, \quad A \in N, \gamma \in (N \cup \Sigma)^+, \text{ mit } |uAv| \leq |u\gamma v|.$$

(-Regeln nur für Startsymbol möglich, ansonsten nicht.)

- Typ-0 (unrestriktete Grammatiken): Regeln der Form

$$\alpha \rightarrow \beta, \quad \alpha \neq \varepsilon, \alpha, \beta \in (N \cup \Sigma)^*.$$

Diese Sprachen sind rekursiv aufzählbar.

(3) Generierung und Ableitungsregeln. Eine Ableitung ist eine Folge von Ersetzungen

$$S \Rightarrow \alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow w,$$

wobei jeder Schritt durch eine Produktion aus P erfolgt. - Linke Ableitung: immer das linksstehende Nichtterminal wird durch eine rechte Seite ersetzt. - Rechte Ableitung: immer das rechtsstehende Nichtterminal wird ersetzt. - Parse-Tree: grafische Darstellung der Ableitungen; der Wurzelknoten ist S , Blätter bilden das Wort w .

(4) Beispiel-Grammatik und Ableitung. Gegeben sei die kontextfreie Grammatik G_0 mit

$$S \rightarrow a S b \mid \varepsilon.$$

Sie erzeugt die Sprache

$$L(G_0) = \{ a^n b^n \mid n \geq 0 \}.$$

Ableitungsschritte zur Wortfolge $a^2 b^2$:

$$S \Rightarrow a S b \Rightarrow a a S b b \Rightarrow a a \varepsilon b b \Rightarrow a a b b.$$

(5) Normalformen. - Chomsky-Normalform (CNF):

$$A \rightarrow BC \quad \text{oder} \quad A \rightarrow a,$$

mit $A, B, C \in N$ und $a \in \Sigma$. Falls $\varepsilon \in L(G)$, gilt zusätzlich $S \rightarrow \varepsilon$.

- Greibach-Normalform (GNF):

$$A \rightarrow a\alpha, \quad a \in \Sigma, \quad \alpha \in (N \cup \Sigma)^*.$$

(6) Zusammenhang mit Automaten. - Sprachen der Typ-3-Grammatiken sind regulär; sie werden von endlichen Automaten erkannt. - Sprachen der Typ-2-Grammatiken sind kontextfrei; sie werden von Kellerautomaten erkannt. - Sprachen der Typ-1-Grammatiken sind kontextsensitiv; sie werden von linear-beschränkten Automaten erkannt. - Sprachen der Typ-0-Grammatiken sind rekursiv aufzählbar; sie werden von Turingmaschinen erkannt.

(7) Übungsbeispiel – CNF/ GNForm-Überlegung. Betrachte die Grammatik

$$S \rightarrow AB \mid a, \quad A \rightarrow a, \quad B \rightarrow b.$$

Sie ist bereits CNF (außer, dass ε hier nicht vorkommt). Zur CNF-Konversion würden ggf. lange rechte Seiten weiter in Binärapporthe zerlegt oder Hilfs-Nichtterminale eingeführt.

(8) Hinweise zur Ableitungspraxis. - Um eine gegebene Grammatik in CNF zu bringen, zerlegt man lange rechte Seiten in Binärregeln und ersetzt Terminals in langen Regeln durch neue Nichtterminale. - Für Sprachen, die durch Typ-3-Grammatiken erzeugt werden, existieren äquivalente DFAs/NFAs. - Für Sprachen, die durch Typ-2-Grammatiken erzeugt werden, existieren äquivalente Kellerautomaten.

Hinweis zum Kursziel. Die behandelten Grammatiken und Ableitungsregeln dienen dazu, formale Sprachen zu erkennen, zu klassifizieren und deren Komplexität in der Chomsky-Hierarchie abzuschätzen.