Grammatik und Sprachen

Complete	☐
Importancy	
Notes	☐
Readings	☐

Grammatiken sind nützliche Modelle zum Entwurf von Software, die Daten rekursiv verarbeiten (Parser). Somit dienen Grammatiken als Regelwerk zur Bildung von Sätzen einer Sprache, dabei wird der Fokus auf die Analyse rekursiver Strukturen wie Programmiersprachen.

Was ist nun eine Grammatik G?

- die Grammatik G ist ein Quadrupel $G = (V, \Sigma, R, S)$
  - V ist eine endliche Menge von *Variablen* (oder Nichtterminale)
  - Σ ein Alphabet, die *Terminale* , $V \cap \Sigma = \varnothing$
  - R eine endliche Menge von *Regeln*
    - eine Regel ist ein Paar $P \to Q$ von Wörtern, sodass $P, Q \in (V \cup \Sigma)^*$ und in P mindest eine *Variable* vorkommt
    - P wir dauch *Prämisse* genannt
    - Q wird auch *Konklusion* der Regel genannt
  - $S \in V$ das *Startsymbol* von G

💡 Variablen werden als Großbuchstaben geschrieben und Terminale als Kleinbuchstaben

> Bei **mehreren** Regeln einer Prämisse werden die Konklusionen auf der rechten Seite zusammengefasst: $P \to Q_1|Q_2|Q_3\dots$

Definitionen von der Ableitbarkeit

Sei $G = (V, \Sigma, R, S)$ eine Grammatik und $x, y \in (V \cup \Sigma)^*$.

Dann heißt y aus x in G *direkt ableitbar,* wenn gilt:

$$\exists\, u, v \in (V \cup \Sigma)^*, \exists\, (P \to Q) \in R \text{ sodass } (x = uPv \;\; und \;\; y = uQv)$$

Kurz geschrieben:

$x \underset{G}{\Rightarrow} y$, sollte die Grammatik G aus dem Kontext folgen, dann schreiben wir $x \Rightarrow y$

Dann ist y aus x in G *ableitbar,* wenn es eine natürliche Zahl $k \in \mathbb{N}$ und Wörter $w_0, w_1, \dots, w_k \in (V \cup \Sigma)^*$ gibt, sodass:

$$x = w_0 \underset{G}{\Rightarrow} w_1 \underset{G}{\Rightarrow} \dots \underset{G}{\Rightarrow} w_k = y$$

das heißt $x = y$ für $k = 0$. Symbolisch schreiben wir $x \underset{G}{\Rightarrow} y$, beziehungsweise $x \overset{*}{\Rightarrow}$y.

💡 $\overset{*}{\Rightarrow}$ heißt soviel wie beliebig oft
$\overset{*}{\underset{G}{\Longrightarrow}}$ heißt soviel beliebig oft ableitbar in einer Grammatik

Definitionen einer Sprache

- Die vom Startsymbol ableitbaren Wörter heißen *Satzformen*
- Elemente von Σ^* werden *Terminalwörter* genannt
- Satzformen die Terminalwörter sind, heißen *Sätze*
- Sätze können mehrere Ableitungen haben und es kann Satzformen geben, die nicht weiter abgeleitet werden können

Die Menge aller Sätze

$$L(G) := \{x \in \Sigma^* \mid S \overset{*}{\underset{G}{\Longrightarrow}} x\}$$

wird die von der Grammatik G *erzeugte Sprache* genannt.
Zwei Grammatiken G_1 und G_2 heißen *äquivalent,* wenn $L(G_1) = L(G_2)$

Klassen von Grammatiken

Sei $G = (V, \Sigma, R, S)$ eine Grammatik, dann heißt G

- *rechtslinear*

  wenn für alle Regeln

  > $P \rightarrow Q$ in R
  > $P \in V$ und $Q \in (V \cup \Sigma)^*$ gilt

- *kontextfrei*

  wenn für alle Regeln

  > $P \rightarrow Q$
  > $P \in V$ und $Q \in (V \cup \Sigma)^*$ gilt

- *kontextsensitiv*

  wenn für alle Regeln $P \rightarrow Q$ gilt:

  1. entweder es existieren $u, v, w \in (V \cup \Sigma)^*$ und $A \in V$, sodass
     $P = uAv$ und $Q = uwv$ wobei $|w| \geq 1$

     > D.h. wenn der Kontext um die variable gleich bleibt, auch nachdem sie ersetzt worden ist

  2. oder $P = S$ und $Q = \epsilon$

     > Wenn $S \rightarrow \epsilon \in R$, dann kommt S nicht in der Konklusion vor

- *beschränkt*

  wenn für alle Regeln $P \rightarrow Q$ entweder gilt:

  1. $|P| \leq |Q|$ oder
  2. $P = Q$ und $Q = \epsilon$

  wenn $S \rightarrow \epsilon \in G$, dann kommt S nicht in einer Konklusion vor

> 💡 sollte links vom Pfeil mehr als eine Variable stehen, dann kann sie nicht **rechtslinear und kontextfrei** sein

Klassen von Sprachen

Die Klassen von Sprachen sind aufbauend auf die eingeführten Grammatiken definiert

- *regulär (Typ 3)*

  wenn eine rechtslineare Grammatik G existiert

  > $L = L(G)$

- *kontextfrei (Typ 2)*

  wenn eine kontextfreie Grammatik G existiert

  > $L = L(G)$

- *kontextsensitiv (Typ 1)*

  wenn eine kontextsensitive Grammatik G existiert

  > $L = L(G)$

- *beschränkt*

  wenn eine beschränkte Grammatik G existiert

  > $L = L(G)$

- *rekursiv aufzählbar (Typ 0)*

  wenn eine beschränkte Grammatik G existiert

  > $L = L(G)$