Theoretische Informatik 1

WiSe 23/24

Prof. Dr. Sebastian Siebertz
AG Theoretische Informatik
MZH, Raum 3160
siebertz@uni-bremen.de

Universität Bremen

Grammatiken

- *Grammatiken* sind neben Automaten und regulären Ausdrücken ein weiteres Mittel, um formale Sprachen zu definieren:
  - Wir beginnen mit einem Startsymbol.
  - Regeln erlauben es, wiederholt (Teil-)Wörter durch andere Wörter zu ersetzen.

Beispiel

- Startsymbol: S
- Regeln:
  1) $S \longrightarrow aSb$
  2) $S \longrightarrow \varepsilon$
- Eine mögliche Ableitung eines Wortes ist:

$$S \xrightarrow{1} aSb \xrightarrow{1} aaSbb \xrightarrow{1} aaaSbbb \xrightarrow{2} aaabbb$$

- Symbol S: *nichtterminales Symbol*
- Wir sind nur an erzeugten Wörtern interessiert, die keine nichtterminalen Symbole enthalten: *Terminalwörter*.
- Die erzeugte Sprache: $\{a^n b^n : n \geq 0\}$

Grammatiken

Definition Grammatik

Eine *Grammatik* ist von der Form $G = (N, \Sigma, P, S)$, wobei

- N und Σ endliche, disjunkte Alphabete von *Nichtterminalsymbolen* bzw. *Terminalsymbolen* sind,
- $S \in N$ das *Startsymbol* ist,
- $P \subseteq (N \cup \Sigma)^+ \times (N \cup \Sigma)^*$ eine endliche Menge von Ersetzungsregeln (*Produktionen*) ist.

- Wir schreiben Produktionen $(u, v) \in P$ gewöhnlich als $u \longrightarrow v$.
- Wir schreiben Elemente von N mit Großbuchstaben und Elemente von Σ mit Kleinbuchstaben.

Beispiel

- Folgendes Tupel ist eine Grammatik: $G = (N, \Sigma, P, S)$ mit

$$
\begin{array}{lrcl}
- & N & = & \{S, B\} \\
- & \Sigma & = & \{a, b, c\} \\
- & P & = & \{S \longrightarrow aSBc, \\
 & & & \quad S \longrightarrow abc, \\
 & & & \quad cB \longrightarrow Bc, \\
 & & & \quad bB \longrightarrow bb\}
\end{array}
$$

Ableitbarkeit, erzeugte Sprache

Definition Ableitbarkeit, erzeugte Sprache

Sei $G = (N, \Sigma, P, S)$ eine Grammatik und x, y Wörter aus $(N \cup \Sigma)^*$.

1) *y aus x direkt ableitbar:*
   $x \vdash_G y \Leftrightarrow x = x_1 u x_2$ und $y = x_1 v x_2$ mit $u \longrightarrow v \in P$ und $x_1, x_2 \in (N \cup \Sigma)^*$
2) *y aus x in n Schritten ableitbar:*
   $x \vdash_G^n y \Leftrightarrow x \vdash_G x_1 \vdash_G \cdots \vdash_G x_{n-1} \vdash_G y$ für $x_1, \ldots, x_{n-1} \in (N \cup \Sigma)^*$
3) *y aus x ableitbar:*
   $x \vdash_G^* y \Leftrightarrow x \vdash_G^n y$ für ein $n \geq 0$
4) Die durch G *erzeugte Sprache* ist
   $L(G) := \{w \in \Sigma^* \mid S \vdash_G^* w\}$.

Beispiel

- Ableitungen in der Grammatik $G = (N, \Sigma, P, S)$ mit
  - $N = \{S, B\}$
  - $\Sigma = \{a, b, c\}$
  - $$\begin{aligned} P = \{S &\longrightarrow aSBc, \\ S &\longrightarrow abc, \\ cB &\longrightarrow Bc, \\ bB &\longrightarrow bb\}: \end{aligned}$$
    - $S \vdash_G abc$
    - $S \vdash_G aSBc \vdash_G aaSBcBc \vdash_G aaabcBcBc \vdash_G aaabBccBc \vdash_G^2 aaabBBccc \vdash_G^2 aaabbbccc$
- $L(G) = \{a^n b^n c^n \mid n \geq 1\}$.
- Beweis: Nächste Woche

Die Chomsky-Hierarchy

Definition Chomsky-Hierarchy

Es sei $G = (N, \Sigma, P, S)$ eine Grammatik.

- Jede Grammatik G ist Grammatik vom *Typ 0.*
- G ist Grammatik vom *Typ 1 (monoton),* falls alle Regeln *nicht verkürzend* sind, also die Form $w \longrightarrow u$ haben wobei $w, u \in (\Sigma \cup N)^+$ und $|u| \geq |w|$.

  Ausnahme: Die Regel $S \longrightarrow \varepsilon$ ist erlaubt, wenn S in keiner Produktion auf der rechten Seite vorkommt.
- G ist Grammatik vom *Typ 2 (kontextfrei),* falls alle Regeln die Form $A \longrightarrow w$ haben mit $A \in N, w \in (\Sigma \cup N)^*$.
- G ist Grammatik vom *Typ 3 (rechtslinear),* falls alle Regeln die Form $A \longrightarrow uB$ oder $A \longrightarrow u$ haben mit $A, B \in N$, $u \in \Sigma^*$.

Kontextfrei versus kontextsensitiv

- Typ 2 heißt *kontextfrei*, weil in kontextfreien Regeln

$$A \longrightarrow w$$

  die linke Seite nur aus Nichterminalsymbol A besteht.

- Ersetzung daher *unabhängig vom Kontext* im Wort.
- Grammatiken der *Typen 0 und 1* erlauben Regeln

$$u_1 A u_2 \longrightarrow u_1 w u_2.$$

- Ersetzung von A durch w *abhängig vom Kontext* u_1 links und u_2 rechts.

Beispiele

$$S \longrightarrow aSb$$
$$S \longrightarrow \varepsilon$$

Typ 2 (kontextfrei)

$$S \longrightarrow aS$$
$$S \longrightarrow bS$$
$$S \longrightarrow abB$$
$$B \longrightarrow aB$$
$$B \longrightarrow bB$$
$$B \longrightarrow \varepsilon$$

Typ 3 (rechtslinear)

$$S \longrightarrow aSBc$$
$$S \longrightarrow abc$$
$$cB \longrightarrow Bc$$
$$bB \rightarrow bb$$

Typ 1 (monoton)

Sprachklassen der Chomsky-Hierarchie

Definition Sprachklassen

Für $i = 0, 1, 2, 3$ ist die *Klasse der Typ-i-Sprachen definiert* als

$$\mathcal{L}_i := \{L(G) \mid G \text{ ist Grammatik vom Typ } i\}.$$

- Jede Typ 3 Grammatik ist auch eine Typ 2 Grammatik.
- Jede Typ 1 Grammatik ist auch eine Typ 0 Grammatik.
- Typ 2 und Typ 3 Grammatiken erlauben das Verkürzen des abgeleiteten Wortes, sie sind also nicht immer Typ 1 Grammatiken!
- Später: Jede Typ-2-Grammatik kann in äquivalente Typ-1 Grammatik umgewandelt werden.

Sprachklassen der Chomsky-Hierarchie

- Also bilden die Sprachtypen eine *Hierarchie*.
- Diese ist sogar *strikt*.

Satz

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

- Nach Definition der Grammatiktypen gilt $\mathcal{L}_3 \subseteq \mathcal{L}_2$ und $\mathcal{L}_1 \subseteq \mathcal{L}_0$.
- Die Inklusion $\mathcal{L}_2 \subseteq \mathcal{L}_1$ werden wir später zeigen.
- Auch die Striktheit der Inklusionen werden wir erst später beweisen.

Rechtslinear = regulär

Satz: Typ 3 = regulär

Die Typ-3-Sprachen sind genau die regulären Sprachen, d. h.:

$$\mathcal{L}_3 = \{L \mid L \text{ ist regulär}\}$$

Beweis.

- Zeige: Jede Typ-3-Sprache ist regulär.
- Zeige: Jede reguläre Sprache ist eine Typ-3-Sprache.

Jede Typ-3-Sprache ist regulär.

- Sei $L \in \mathcal{L}_3$, d. h. $L = L(G)$ für eine Typ-3-Grammatik $G = (N, \Sigma, P, S)$.
- Erinnerung:
  - Jede Regel hat die Form $A \longrightarrow uB$ oder $A \longrightarrow u$ für $A, B \in N, u \in \Sigma^*$.
- Es gilt $w_1 \cdots w_n \in L(G)$ (für $w_i \in \Sigma^*$) g.d.w. es gibt eine Ableitung

$$\begin{aligned} S = B_0 \vdash_G w_1 B_1 \vdash_G w_1 w_2 B_2 \vdash_G \dots \\ \vdash_G w_1 \dots w_{n-1} B_{n-1} \vdash_G w_1 \dots w_{n-1} w_n. \end{aligned} \tag{$*$}$$

- Nichtterminale verhalten sich wie Zustände eines Automaten (zu jedem Zeitpunkt genau eines).
- Produktionen verhalten sich wie Wortübergänge.

Jede Typ-3-Sprache ist regulär.

- Wir konstruieren *NEA mit Wortübergängen* $\mathcal{A} = (N \cup \{\Omega\}, \Sigma, S, \Delta, \{\Omega\})$, wobei
  - $\Omega \notin N$ akzeptierender Zustand ist und
  - $\Delta = \{(A, w, B) \mid A \longrightarrow wB \in P\} \cup \{(A, w, \Omega) \mid A \longrightarrow w \in P\}$.
- Nun gilt $w_1 \dots w_n \in L(G) \Leftrightarrow$ es gibt Ableitung

$$\begin{aligned} S = B_0 \vdash_G w_1 B_1 \vdash_G w_1 w_2 B_2 \vdash_G \dots \\ \vdash_G w_1 \dots w_{n-1} B_{n-1} \vdash_G w_1 \dots w_{n-1} w_n. \end{aligned}$$

- Dies ist genau dann der Fall, wenn im Automaten ein Lauf existiert

$$S \xrightarrow{w_1}_{\mathcal{A}} B_1 \xrightarrow{w_2}_{\mathcal{A}} \cdots \xrightarrow{w_n}_{\mathcal{A}} \Omega.$$

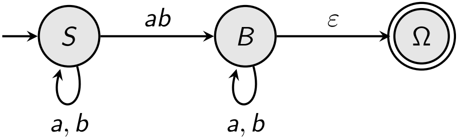
- Also $L(\mathcal{A}) = L(G)$.

Beispiel

Die Grammatik

$$\begin{array}{rcl} P = \{S & \longrightarrow & aS, \\ S & \longrightarrow & bS, \\ S & \longrightarrow & abB, \\ B & \longrightarrow & aB, \\ B & \longrightarrow & bB, \\ B & \longrightarrow & \varepsilon\} \end{array}$$

liefert den folgenden NEA mit Wortübergängen:



Jede reguläre Sprache ist Typ-3-Sprache

- Sei $L = L(\mathcal{A})$ für einen NEA $\mathcal{A} = (Q, \Sigma, q_s, \Delta, F)$.
- Definiere Typ-3-Grammatik $G = (N, \Sigma, P, S)$:

$$\begin{aligned} N &:= Q \qquad S := q_s \\ P &:= \{p \longrightarrow aq \mid (p,a,q) \in \Delta\} \cup \{p \longrightarrow \varepsilon \mid p \in F\} \end{aligned}$$

- Ein Lauf in $\mathcal{A}$ der Form

$$q_s \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \xrightarrow{a_3} \cdots \xrightarrow{a_n} q_n$$

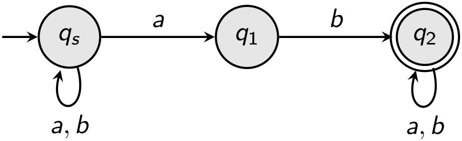
mit $q_n \in F$ entspricht nun genau einer Ableitung

$$q_s \vdash_G a_1 q_1 \vdash_G a_1 a_2 q_2 \vdash_G \ldots \vdash_G a_1 \ldots a_n q_n \vdash_G a_1 \ldots a_n.$$

- Also $L(\mathcal{A}) = L(G)$.

Beispiel

Der NEA



liefert die Grammatik mit den rechtslinearen Produktionen

$$
\begin{aligned}
P = \{q_s &\longrightarrow aq_s, \\
q_s &\longrightarrow bq_s, \\
q_s &\longrightarrow aq_1, \\
q_1 &\longrightarrow bq_2, \\
q_2 &\longrightarrow aq_2, \\
q_2 &\longrightarrow bq_2, \\
q_2 &\longrightarrow \varepsilon\}
\end{aligned}
$$

Zusammenfassung und Ausblick

Heute:

- Definition: Grammatiken
- Chomsky-Hierarchy für Grammatiken.
- Typ-3 Grammatiken beschreiben reguläre Sprachen.

Nächste Woche:

- Erste Beweise für Striktheit der Chomsky-Hierarchie