Grundlagen der Theoretischen Informatik

Sommersemester 2021

28.04.2021

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de

Bis jetzt

1. Motivation
2. Terminologie
3. Endliche Automaten und reguläre Sprachen
4. Kellerautomaten und kontextfreie Sprachen
5. Turingmaschinen und rekursiv aufzählbare Sprachen
6. Berechenbarkeit, (Un-)Entscheidbarkeit
7. Komplexitätsklassen P und NP

Bis jetzt

- **Alphabete, Wörter**
  - Operationen auf Wörtern

    Konkatenation, i-te Potenz, Reverse
- **Sprache**
  - Operationen auf Sprachen

    Konkatenation, i-te Potenz, Reverse, Kleene-Hülle
- **Reguläre Ausdrücke**
- **Grammatiken**
  - Ableitung
  - Die von einer Grammatik erzeugte Sprache.
- **Die Chomsky-Hierarchie**

Grammatik

Definition. Eine Grammatik G über einem Alphabet Σ ist ein Tupel $G = (V, T, R, S)$. Dabei ist

- V eine endliche Menge von **Variablen**
- $T \subseteq \Sigma$ eine endliche Menge von **Terminalen** mit $V \cap T = \emptyset$
- R eine endliche Menge von **Regeln**
- $S \in V$ das **Startsymbol**

Konvention (meistens): **Variablen** als Großbuchstaben; **Terminale** als Kleinbuchstaben.

Eine **Regel** ist ein Element $(P, Q) \in \big((V \cup T)^* \; V \; (V \cup T)^*\big) \times (V \cup T)^*$.

- P und Q sind Wörter über $(V \cup T)$
- P muss mindestens eine Variable enthalten
- Q ist beliebig

Schreibweise: $P \rightarrow_G Q$, $P \rightarrow Q$ (P: Prämisse, Q: Conclusio)

Erzeugte Sprache, Äquivalenz

Ableitung, Rechnung.

- $w \Longrightarrow_G w'$ („w geht über in w'“) falls

  $$\exists u, v \in (V \cup T)^* \; \exists P \to Q \in R \; \big(w = uPv \text{ und } w' = uQv\big)$$

- $w \Longrightarrow_G^* w'$ falls es Wörter $w_0, \ldots, w_n \in (V \cup T)^*$ $(n \geq 0)$ gibt mit
  - $w = w_0$
  - $w_n = w'$
  - $w_i \Longrightarrow_G w_{i+1}$ für $0 \leq i < n$

Definition (Erzeugte Sprache) Gegeben: Eine Grammatik G
Die von G erzeugte Sprache $L(G)$ ist die Menge aller **terminalen** Wörter, die durch G vom Startsymbol S aus erzeugt werden können:

$$L(G) \quad := \quad \{w \in T^* \mid S \Longrightarrow_G^* w\}$$

Definition Zwei Grammatiken G_1, G_2 heißen **äquivalent** gdw $L(G_1) = L(G_2)$.

Die Chomsky-Hierarchie

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **rechtslinear** gdw

$$\forall (P \to Q) \in R \;\; \big(P \in V \text{ und } Q \in T^* \cup T^+ V\big)$$

Das heißt, bei jeder Regelanwendung:

- Links eine **einzelne Variable**; Rechts **höchstens eine Variable**
- Wenn rechts eine Variable steht, steht sie **ganz rechts im Wort** und das Wort enthält **mindestens ein Terminalsymbol**

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **kontextfrei** gdw

$$\forall (P \to Q) \in R \;\; \big(P \in V \text{ und } Q \in (V \cup T)^*\big)$$

Das heißt, bei jeder Regelanwendung:

- Links eine **einzelne Variable**; Rechts steht etwas beliebiges
- Die Prämisse macht keine Aussage, was der Kontext dieser Variablen ist („kontextfrei“)

Die Chomsky-Hierarchie

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **kontextsensitiv** gdw $\forall (P \to Q) \in R$:

1. $\exists u, v, \alpha \in (V \cup T)^* \; \exists A \in V \; \big(P = uAv$ und $Q = u\alpha v$ mit $|\alpha| \geq 1\big)$, **oder** die Regel hat die Form $S \to \varepsilon$
2. S nicht in Q

Das heißt, bei jeder Regelanwendung:

- Eine Variable A wird in einen String α mit $|\alpha| \geq 1$ überführt; die Ersetzung von A durch α findet nur statt, wenn der in der Regel geforderte **Kontext (u und v) vorhanden ist**
- **Das Wort wird nicht kürzer, außer bei $\varepsilon \in L$**

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **beschränkt** gdw $\forall (P \to Q) \in R$:

1. $|P| \leq |Q|$, **oder** die Regel hat die Form $S \to \varepsilon$
2. S nicht in Q

Das heißt, bei jeder Regelanwendung:

- Die Conclusio ist mindestens so lang wie die Prämisse, außer bei $\varepsilon \in L$.
- Das Wort wird nicht kürzer, außer bei $\varepsilon \in L$

Die Chomsky-Hierarchie

Aufbauend auf den Grammatikarten kann man Sprachklassen definieren.

Definition (Sprachklassen)

Klasse	**definiert als**	**Sprache heißt**
$\mathcal{L}_3$, REG	$\{L(G) \mid G \text{ ist rechtslinear}\}$	Typ 3, **regulär**
$\mathcal{L}_2$, CFL	$\{L(G) \mid G \text{ ist kontextfrei}\}$	Typ 2, **kontextfrei**
$\mathcal{L}_1$, CSL	$\{L(G) \mid G \text{ ist kontextsensitiv}\}$	Typ 1, **kontextsensitiv**
$\mathcal{L}_1$, CSL	$\{L(G) \mid G \text{ ist beschränkt}\}$	Typ 1, **beschränkt**
$\mathcal{L}_0$, r.e.	$\{L(G) \mid G \text{ beliebig}\}$	Typ 0, **aufzählbar**
$\mathcal{L}$	$\{L \mid L \subseteq \Sigma^* \}$	**beliebige** Sprache

Abzählbarkeit

Definition (Abzählbarkeit)
Eine Menge M heißt **abzählbar**, wenn

- es eine surjektive Funktion $\quad f : \mathbb{N} \to M \quad$ gibt,
- oder M leer ist.

Lemma. Eine Menge M ist abzählbar, wenn es eine injektive Funktion

$$f : M \to \mathbb{N}$$

gibt.

Beispiele

Abzählbar sind:

- $\mathbb{N}$
- $\mathbb{Q}$
- alle endlichen Mengen
- die Vereinigung zweier abzählbarer Mengen

  Falls Σ_1, Σ_2 abzählbar, so ist $\Sigma_1 \cup \Sigma_2$ abzählbar

- die Vereiningung abzählbar vieler abzählbarer Mengen

  Falls $\Sigma_1, \Sigma_2, \ldots$ abzählbar, so ist $\bigcup_{i \in \mathbb{N}} \Sigma_i$ abzählbar

  Falls Σ abzählbar, so ist Σ^i abzählbar für alle $i \in \mathbb{N}$.
  Beweis: Induktion (IB Klar, IS: $\Sigma^{i+1} = \bigcup_{u \in \Sigma} \{wu \mid w \in \Sigma^i\}$.

Diagonalisierungsargument für Überabzählbarkeit

Theorem. Die Menge $\mathbb{R}$ der rellen Zahlen ist überabzählbar.

Beweis. Wir zeigen, dass schon das Intervall $[0, 1]$ überabzählbar ist.

Annahme: Es gibt eine Aufzählung, also eine surjektive Funktion

$$f : \mathbb{N} \to [0, 1]$$

Dann sei

$$f(i) = 0, d_0^i d_1^i d_2^i \ldots$$

die Dezimaldarstellung der i-ten reellen Zahl.

Diagonalisierungsargument für Überabzählbarkeit

Beweis. Fortsetzung:

Wir definieren eine neue Zahl $d = 0,\overline{d}_0\overline{d}_1\overline{d}_2\dots$ durch

$$\overline{d}_n = \begin{cases} d_n^n + 1 & \text{falls } d_n^n < 9 \\ 0 & \text{sonst} \end{cases}$$

d unterscheidet sich in der n-ten Stelle von d_n.

Also $d \neq d_n$ für alle $n \in \mathbb{N}$

Also kommt d in der Aufzählung nicht vor. Widerspruch!

Wieviele gibt es?

Wieviele

- Grammatiken
- Sprachen
- Algorithmen

gibt es überhaupt?

Wieviele gibt es?

Wieviele

– Grammatiken
– Sprachen
– Algorithmen

gibt es Überhaupt?

Mögliche Antworten:

- Endlich viele
- Unendlich viele
- Abzählbar viele
- Überabzählbar viele
- Nicht klar für Algorithmen, da dieser Begriff nicht genau definiert wurde

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Alphabet Σ, endlich oder abzählbar unendlich

Dann ist Σ^* abzählbar unendlich.

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Alphabet Σ, endlich oder abzählbar unendlich

Dann ist Σ^* abzählbar unendlich.

Beweis.

Σ ist abzählbar, also ist Σ^i abzählbar für alle $i \in \mathbb{N}$.

Σ^* ist die Vereinigung der abzählbar vielen abzählbaren Mengen Σ^i.

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Alphabet Σ, endlich oder abzählbar unendlich

Dann ist die Menge aller Grammatiken über Σ abzählbar unendlich

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Alphabet Σ, endlich oder abzählbar unendlich

Dann ist die Menge aller Grammatiken über Σ abzählbar unendlich

Beweis.

Grammatiken sind endlich und somit als Wörter über einem geeigneten erweiterten Alphabet

$$\Sigma \cup V \cup \{\rightarrow, \ldots\}$$

darstellbar.

Die Menge der Wörter über diesen erweiterten Alphabet ist abzählbar.

Wieviele Algorithmen gibt es?

Lemma. Es gibt (nur) abzählbar viele Algorithmen.

Wieviele Algorithmen gibt es?

Lemma. Es gibt (nur) abzählbar viele Algorithmen.

Beweis.

Algorithmen müssen per Definition eine endliche Beschreibung haben.

Sie sind also als Wörter über einem Alphabet Σ darstellbar
(für jedes abzählbare Σ).

Also sind sie abzählbar.

Wieviele Funktionen $f : \mathbb{N} \to \mathbb{N}$ gibt es?

Lemma. Es gibt überabzählbar viele Funktionen $f : \mathbb{N} \to \mathbb{N}$.

Wieviele Funktionen $f : \mathbb{N} \to \mathbb{N}$ gibt es?

Lemma. Es gibt überabzählbar viele Funktionen $f : \mathbb{N} \to \mathbb{N}$.

Beweis.

Angenommen, es existiere eine Abzählung

$$f_0, f_1, f_2, \ldots f_n, \ldots$$

Dann sei

$$C : \mathbb{N} \to \mathbb{N} \quad \text{mit} \quad C(n) = \begin{cases} 1 & \text{falls } f_n(n) = 0; \\ 0 & \text{sonst} \end{cases}$$

$C(i) \neq f_i(i)$

Also: C ist von allen f_i verschieden. Widerspruch!

Wieviele Funktionen $f : \mathbb{N} \to \{0, 1\}$ gibt es?

Lemma. Es gibt überabzählbar viele Funktionen $f : \mathbb{N} \to \{0, 1\}$.

Beweis. Analog.

Wieviele Sprachen gibt es?

Lemma. Gegeben: Alphabet Σ (endlich oder unendlich).

Die Menge der Sprachen über Σ ist überabzählbar.

Wieviele Sprachen gibt es?

Lemma. Gegeben: Alphabet Σ (endlich oder unendlich).

Die Menge der Sprachen über Σ ist überabzählbar.

Beweis.

Sei eine beliebige Abzählung aller Wörter über Σ gegeben:

$$w_1, w_2, \ldots$$

Dann kann man die Sprachen L über Σ mit den Funktionen $f : \mathbb{N} \to \{0, 1\}$ identifizieren vermittels

$$f(i) = 1 \quad \text{gdw} \quad w_i \in L$$

Von diesen gibt es überabzählbar viele.

Korollar

Korollar.
Nicht jede Sprache kann durch eine Grammatik dargestellt werden.

Zusammenfassung

Gegeben eine Signatur Σ

Abzählbar

- $\mathbb{N}$
- Menge aller Wörter
- Menge aller Grammatiken
- Menge aller Algorithmen

Zusammenfassung

Gegeben eine Signatur Σ

Abzählbar

- $\mathbb{N}$
- Menge aller Wörter
- Menge aller Grammatiken
- Menge aller Algorithmen

Überabzählbar

- Die Menge aller Teilmengen von $\mathbb{N}$
- Die Menge aller reellen Zahlen
- Die Menge aller Funktionen $\quad f : \mathbb{N} \to \mathbb{N} \quad$ bzw. $\quad f : \mathbb{N} \to \{0, 1\}$
- Die Menge aller Sprachen