Kapitel 9
Allgemeine Grammatiken

Abschnitt 9.1
Kontextsensitive Grammatiken

Idee

Wir lassen es zu, dass eine Grammatikregel $A \to \alpha$ nicht immer, sondern nur in gewissen Situationen („Kontexten“) anwendbar ist.

Wir führen deswegen Regeln der Form $\beta_1 A \beta_2 \to \beta_1 \alpha \beta_2$ ein, die besagen: „A darf nur dann durch α ersetzt werden, wenn es zwischen β_1 und β_2 steht.“

Beispiel 9.1

$\mathcal{G}_{abc} = (N, \Sigma, P, S)$ sei die „kontextsensitive“ Grammatik mit Terminalalphabet $\Sigma = \{a, b, c\}$, Nichtterminalalphabet $N = \{S, A, B, C, Y, Z\}$ und folgenden Regeln in P:

$$\begin{array}{lll} S \to abc \mid aABc, & A \to \ aBC \mid aABC, & CB \to YB, \\ YB \to YZ, & YZ \to BZ, & BZ \to BC, \\ aB \to ab, & bB \to bb, & Cc \to cc. \end{array}$$

Definition 9.2

1. Eine kontextsensitive Grammatik ist ein Tupel (N, Σ, P, S), wobei das Nichtterminalalphabet N, das Terminalalphabet Σ und das Startsymbol S wie bei einer kontextfreien Grammatik definiert sind und P eine endliche Menge von Regeln der Form
$$\beta_1 A \beta_2 \to \beta_1 \alpha \beta_2$$
für $A \in N$ und $\alpha, \beta_1, \beta_2 \in (N \cup \Sigma)^*$ mit $\alpha \neq \varepsilon$ ist
2. Seien $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextsensitive Grammatik und $\gamma, \delta \in (N \cup \Sigma)^*$ Satzformen. Dann ist δ direkt herleitbar aus γ (kurz: $\gamma \to_{\mathcal{G}} \delta$), wenn es eine Regel $\beta_1 A \beta_2 \to \alpha$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass
$$\gamma = \gamma_1 \beta_1 A \beta_2 \gamma_2 \quad \text{und} \quad \delta = \gamma_1 \beta_1 \alpha \beta_2 \gamma_2.$$
Intuitiv entsteht also γ aus δ, indem man ein zwischen β_1 und β_2 stehendes A durch α ersetzt.
3. Damit definieren wir Ableitungen und Ableitbarkeit in einer kontextsensitiven Grammatik $\mathcal{G}$ sowie die von $\mathcal{G}$ erzeugte Sprache $L(\mathcal{G})$ wie bei kontextfreien Grammatiken.

Notation

Wir verwenden für kontextsensitive Grammatiken die gleiche Notation wie für kontextfreie Grammatiken, etwa $\xrightarrow{*}_{\mathcal{G}}$, $\xrightarrow{n}_{\mathcal{G}}$, und lassen den Index $\mathcal{G}$ weg, wenn die Grammatik aus dem Kontext hervorgeht.

Bemerkung 9.3

Es ist wichtig bei der Definition kontextsensitiver Grammatiken, in Regeln $\beta_1 A \beta_2 \to \beta_1 \alpha \beta_2$ zu verlangen, dass $\alpha \neq \varepsilon$.
Wir lassen also keine „ε-Regeln“ zu.

Kontextfreie Grammatiken können deswegen keine Sprachen erzeugen, die das leere Wort enthalten. Wir könnten das beheben, indem wir die Regel $S \to \varepsilon$ für das Startsymbol S zulassen (aber keinen anderen ε-Regeln).

Beispiel 9.1 (Forts.) I

$\mathcal{G}_{abc}$ sei wieder die kontextsensitive Grammatik mit den Regeln

$$
\begin{array}{llll}
S \to abc \mid aABc, & A \to aBC \mid aABC, & & \\
CB \to YB, & YB \to YZ, & YZ \to BZ, & BZ \to BC, \\
aB \to ab, & bB \to bb, & Cc \to cc. &
\end{array}
$$

Der einzige Sinn der vier Regeln in der zweiten Zeile ist es, CB in BC zu überführen:

$$CB \to YB \to YZ \to BZ \to BC.$$

Die Nichtterminale Y und Z können auch nur in solch einer Teilableitung vorkommen.
Im folgenden schreiben wir einfach $CB \xrightarrow{4} BC$ und führen diese Teilableitung nicht mehr aus.

Beispielableitungen

Beispiel 9.1 (Forts.) II

- $S \to abc$,
- $S \to a\underline{A}Bc \to aaB\underline{CB}c \xrightarrow{4} a\underline{aB}BCc \to aa\underline{bB}Cc \to aabb\underline{Cc} \to aabbcc$,
- $S \to a\underline{A}Bc \to aa\underline{A}BCBc \to aaaB\underline{CB}CBc \xrightarrow{4} aaaBBC\underline{CB}c \xrightarrow{4}$
  $aaaBB\underline{CB}Cc \xrightarrow{4} aa\underline{aB}BBCCc \to aaa\underline{bB}BCCc \to aaab\underline{bB}CCc \to$
  $aaabbbC\underline{Cc} \to aaabbb\underline{Cc}c \to aaabbbccc$

Behauptung

$L(\mathcal{G}_{abc}) = \{a^n b^n c^n \mid n \geq 1\}$.

Beweisskizze. „$\supseteq$“ Wir zeigen, dass $S \xrightarrow{*} a^n b^n c^n$ für alle $n \geq 1$.

Klar für $n = 1$ mit Regel $S \to abc$

Für $n \geq 2$ leiten wir zunächst mit Hilfe der Regeln in der ersten Zeile $a^n(BC)^{n-1}Bc$ ab.

Beispiel 9.1 (Forts.) III

Mit Hilfe der Regeln aus der zweiten Zeile sortieren wir die Bs und Cs, leiten also $a^n B^n C^{n-1} c$ ab.

Daraus können wir mit Hilfe der Regeln aus der dritten Zeile $a^n b^n c^n$ ableiten.

„$\subseteq$“ Man kann zeigen, dass bis auf die Reihenfolge der Regelanwendungen Ableitungen vom gerade beschriebenen Typ die einzigen sind, mit denen sich Terminalwörter ableiten lassen. $\square$

Definition 9.4

Eine Sprache L ist kontextsensitiv, wenn es eine kontextsensitive Grammatik gibt, die $L \setminus \{\varepsilon\}$ erzeugt.

Satz 9.5

1. *Jede kontextfreie Sprache ist kontextsensitiv.*
2. *Es gibt kontextsensitive Sprachen, die nicht kontextfrei sind.*

Beweis.

1. Jede kontextfreie Grammatik ohne ε-Regeln ist kontextsensitiv.
2. Die Sprache $\{a^n b^n c^n \mid n \geq 1\}$ ist ein Beispiel.

$\square$

Satz 9.6

Es gibt einen Algorithmus zur Lösung des Wortproblems für kontextsensitive Grammatiken:

> *"Gegeben eine kontextsensitive Grammatik $\mathcal{G}$ und Terminalwort w, gehört w zu $L(\mathcal{G})$?"*

Beweisskizze.

Der Schlüssel zum Beweis ist folgende Beobachtung.

Beobachtung 9.7

Für eine kontextsensitive Grammatik $\mathcal{G}$ und Satzformen γ, δ gelte $\gamma \to \delta$. Dann ist $|\gamma| \leq |\delta|$.

Sei nun $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextsensitive Grammatik und $w \in \Sigma^*$.
Ausgehend von w konstruieren wir „rückwärts“ alle $\gamma \in (N \cup \Sigma)^*$ mit $\gamma \xrightarrow{*} w$. Wir können diese als Baum anordnen, dabei ist γ ein Kind von γ', wenn $\gamma \to \gamma'$.

Beobachtung 9.7 garantiert, dass für alle γ in diesem Baum $|\gamma| \leq |w|$ gilt. Deswegen terminiert die Suche. $\square$

Satz 9.6 zeigt, dass das Wortproblem für kontextsensitive Grammatiken entscheidbar ist.

Aber:

Satz 9.8

Das Leerheitsproblem für kontextsensitive Grammatiken ist unentscheidbar.

Automaten für kontextsensitive Sprachen

Gibt es ein Automatenmodell für die Klasse der kontextsensitiven Sprachen?

Ein linear beschränkter Automat (LBA) ist eine Erweiterung des NFA um die Fähigkeit

- auf dem Eingabewort nach links und rechts zu laufen (Zwei-Wege-Automat)
- Zeichen zu drucken

Satz 9.9

Eine Sprache ist genau dann kontextsensitiv, wenn sie LBA-erkennbar ist.

Abschnitt 9.2

Die Chomsky-Hierarchie

Allgemeine Grammatiken

Definition 9.10

1. Eine allgemeine Grammatik ist ein Tupel (N, Σ, P, S), wobei das Nichtterminalalphabet N, das Terminalalphabet Σ und das Startsymbol S wie bei einer kontextfreien Grammatik definiert sind und P eine endliche Menge von Regeln der Form
$$\alpha \to \beta$$
für $\alpha \in (N \cup \Sigma)^* \setminus \Sigma^*$ und $\beta \in (N \cup \Sigma)^*$.
2. Seien $\mathcal{G} = (N, \Sigma, P, S)$ eine allgemeine Grammatik und $\gamma, \delta \in (N \cup \Sigma)^*$ Satzformen. Dann ist δ direkt herleitbar aus γ (kurz: $\gamma \to_{\mathcal{G}} \delta$), wenn es eine Regel $\alpha \to \beta$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass
$$\gamma = \gamma_1 \alpha \gamma_2 \quad \text{und} \quad \delta = \gamma_1 \beta \gamma_2.$$
3. Damit definieren wir Ableitungen und Ableitbarkeit in einer allgemeinen Grammatik $\mathcal{G}$ sowie die von $\mathcal{G}$ erzeugte Sprache $L(\mathcal{G})$ wie bei kontextfreien Grammatiken.

Mächtigkeit allgemeiner Grammatiken

- Es gibt Sprachen, die nicht kontextsensitiv sind, die aber von allgemeinen Grammatiken erzeugt werden.
- Mit allgemeinen Grammatiken lassen sich beliebige Berechnungen simulieren.
- Ein Automatenmodell für allgemeine Grammatiken ist deshalb die Turingmaschine, also ein endlicher Automat mit unbeschränkten Speicher (Vorlesung Berechenbarkeit und Komplexität). Turingmaschinen sind ein universelles Berechnungsmodell.
- Das Wortproblem für allgemeine Grammatiken ist unentscheidbar.

Monotone Grammatiken

Definition 9.11

Eine allgemeine Grammatik $\mathcal{G} = (N, \Sigma, P, S)$ ist monoton, wenn für alle Regeln $\alpha \to \beta \in P$ gilt: $|\alpha| \leq |\beta|$.

Beobachtung 9.7 (umformuliert)

Jede kontextsensitive Grammatik ist monoton.

Satz 9.12

Eine Sprache ist genau dann kontextsensitiv, wenn sie von einer monotonen Grammatik erzeugt wird.

(Ohne Beweis.)

Beispiel 9.14

Folgende monotone Grammatik erzeugt die Sprache
$L_{abc} = \{a^n b^n c^n \mid n \geq 1\}$:

$$S \to aBc \mid aSBc, \qquad cB \to Bc,$$
$$aB \to ab, \qquad bB \to bb.$$

Diese Grammatik ist eine vereinfachte Vesrion der kontextsensitiven Grammatik aus Beispiel 9.1.

Beispiel 9.14 I

Eine andere monotone Grammatik für die Sprache L_{abc} ist folgende Grammatik $\mathcal{G}'_{abc}$:

$$S \to abc \mid aAbc, \qquad Ab \to bA, \qquad Ac \to Bbcc,$$
$$bB \to Bb, \qquad aB \to aaA \mid aa$$

Diese Grammatik hat den Vorteil, dass man relativ leicht beweisen kann, dass sie die Sprache L_{abc} erzeugt.

Beispielableitung:

$$\underline{S} \to a\underline{Ab}c \to ab\underline{Ac} \to a\underline{bB}bcc \to \underline{aB}bbcc \to aabbcc$$

Behauptung

$L(\mathcal{G}'_{abc}) = \{a^n b^n c^n \mid n \geq 1\}$.

Beweis. „$\supseteq$“ Wir zeigen, dass $S \xrightarrow{*} a^n b^n c^n$ für alle $n \geq 1$.

Beispiel 9.14 II

Klar für $n = 1$ mit Regel $S \to abc$
Für $n \geq 1$ zeigen wir:

$$S \xrightarrow{*} a^n B b^{n+1} c^{n+1} \qquad (\star)$$

Aus $(\star)$ folgt $S \xrightarrow{*} a^{n+1} b^{n+1} c^{n+1}$ mittels der Regel $aB \to aa$.

Beweis von $(\star)$: Induktion über n.

Induktionsanfang $n = 1$: $S \xrightarrow{4} aBbbcc$ (siehe Beispielableitung)

Induktionsschritt $n \to n+1$:
Nach Induktionsannahme gilt

$$S \xrightarrow{*} a^n B b^{n+1} c^{n+1}.$$

Auf $a^n B b^{n+1} c^{n+1}$

- wenden wir $aB \to aaA$ an und erhalten $a^{n+1} A b^{n+1} c^{n+1}$,
- wenden dann $(n+1)$ mal $Ab \to bA$ an und erhalten $a^{n+1} b^{n+1} A c^{n+1}$,

Beispiel 9.14 III

- wenden dann $Ac \to Bbcc$ an und erhalten $a^{n+1} b^{n+1} B b c^{n+2}$,
- und wenden schließlich $(n+1)$-mal $bB \to Bb$ an und erhalten $a^{n+1} B b^{n+2} c^{n+2}$.

„$\subseteq$“ Wenn $S \xrightarrow{1} w$, dann $w = abc$, und die Behauptung ist gezeigt.
Andernfalls ist der Ableitungsbeginn

$$S \to aAbc \to abAc \to abBbcc \to aBbbcc.$$

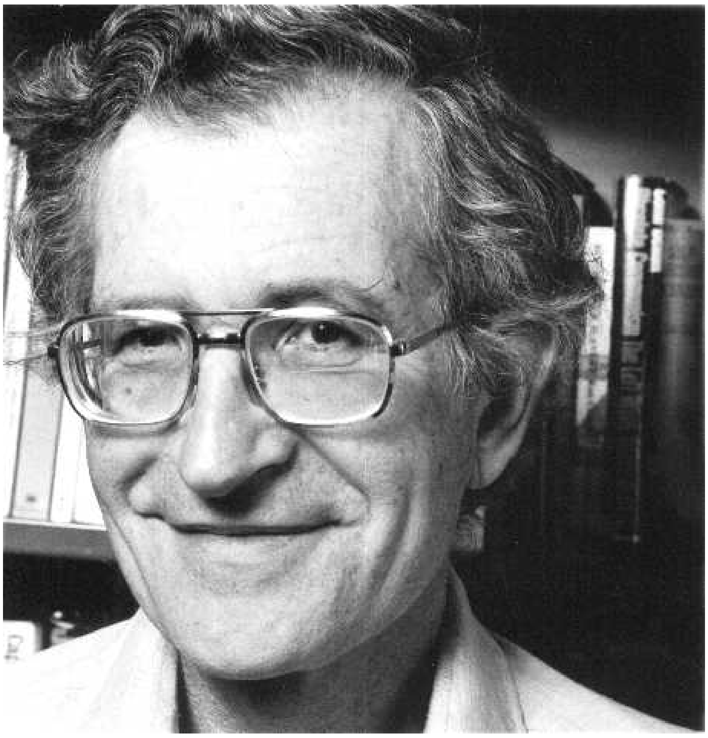
Von $a^n B b^{n+1} c^{n+1}$ sind Übergang nach $a^{n+1} b^{n+1} c^{n+1}$ oder $a^{n+1} A b^{n+1} c^{n+1}$ möglich.
Von $a^{n+1} A b^{n+1} c^{n+1}$ ist die Ableitung wieder eindeutig bis $a^{n+1} B b^{n+2} c^{n+2}$.
Nur aus Satzformen $a^n B b^{n+1} c^{n+1}$ sind also Terminalwörter direkt ableitbar, und zwar solche der Form $a^{n+1} b^{n+1} c^{n+1}$. $\square$

Grammatik-Klassifikation nach Chomsky

- Allgemeine Grammatiken sind vom Typ 0.
- Kontextsensitive Grammatiken sind vom Typ 1.
- Kontextfreie Grammatiken sind vom Typ 2.
- Rechtslineare Grammatiken sind vom Typ 3.

Eine Sprache ist vom Typ i, wenn sie von einer Grammatik vom Typ i erzeugt wird.

Die vier Klassen von Sprachen der Typen $3, 2, 1, 0$ bilden die Chomsky-Hierarchie.



Noam Chomsky
($*$ 7.12.1928, Philadelphia)
Noam Chomsky, *Three Models for the Description of Language*, IRE Transactions on Information Theory 2 (1956), pp. 113 – 124.

Algorithmische Eigenschaften

	Typ 0 (allgemein)	Typ 1 (kontextsensitiv)	Typ 2 (kontextfrei)	Typ 3 (regulär)
Wortproblem	**U**	**E**	**E**	**E**
Leerheit	**U**	**U**	**E**	**E**
Äquivalenz	**U**	**U**	**U**	**E**

E = entscheidbar, **U** = unentscheidbar