Informatik III

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

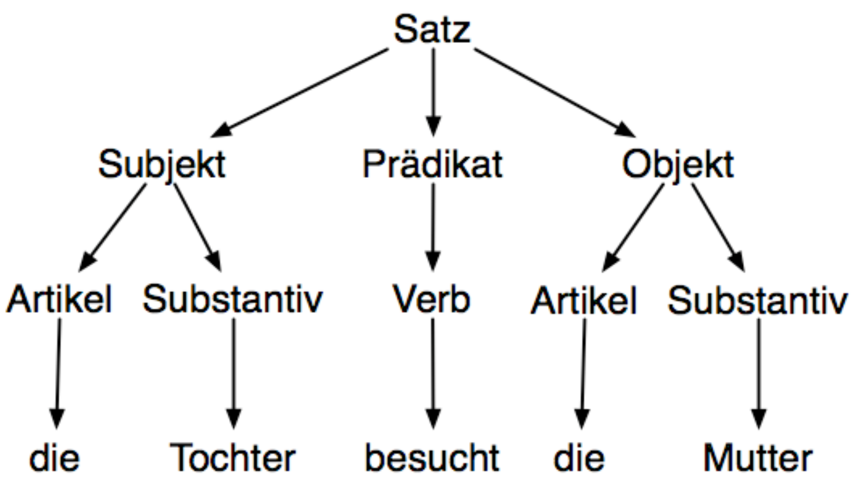
Christian Schindelhauer
Wintersemester 2006/07
6. Vorlesung
10.11.2006
schindel@informatik.uni-freiburg.de

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Kontextfreie Grammatik

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Grammatik - Was ist das?



<SATZ> →
<SUBJEKT> <PRÄDIKAT> <OBJEKT>

<SUBJEKT> → **<ARTIKEL> <SUBSTANTIV>**
<OBJEKT> → **<ARTIKEL> <SUBSTANTIV>**
<PRÄDIKAT> → **<VERB>**

<ARTIKEL> → `die`

<SUBSTANTIV> → `Mutter`
<SUBSTANTIV> → `Tochter`
<SUBSTANTIV> → `Professorin`
<SUBSTANTIV> → `Studentin`

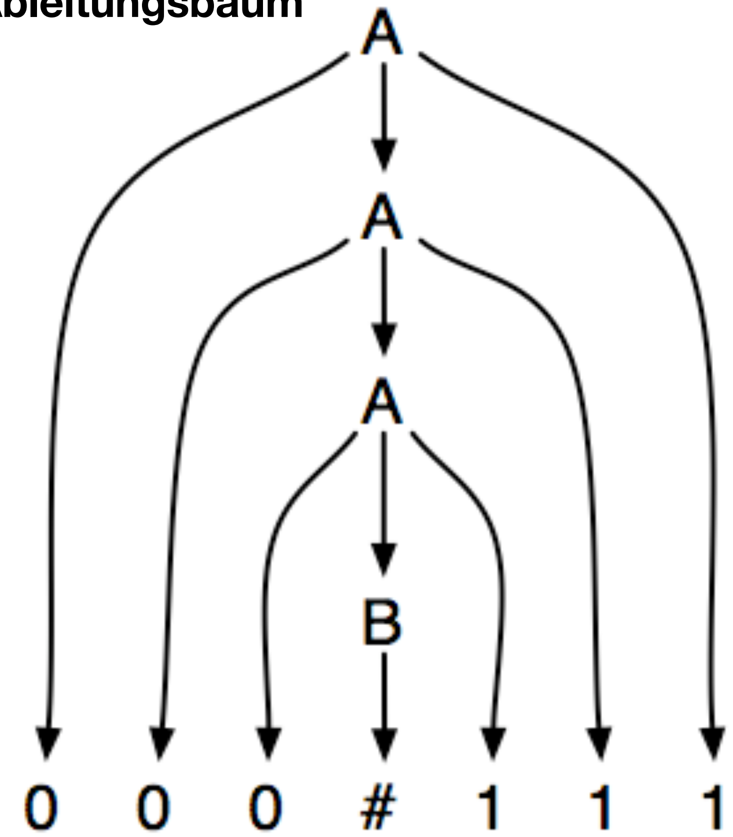
<VERB> → `langweilt`
<VERB> → `prüft`
<VERB> → `belehrt`
<VERB> → `besucht`
<VERB> → `beleidigt`
<VERB> → `tötet`

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Beispiele kontextfreier Grammatiken

- **Kontextfreie Grammatiken sind "Wortersetzer"**
  - $A \rightarrow 0A1$
  - $A \rightarrow B$
  - $B \rightarrow \#$
- **Ersetzungsregeln bestehen aus Produktionen**
- **Variablen, Terminalsymbole (Terminale)**
- **Startvariable: A**
  - A
  - 0A1
  - 00A11
  - 000A111
  - 000B111
  - 000#111 .... und Schluss
- **Das Wort 000#111 ist ein Wort der Grammatik**

Ableitungsbaum



Beispiel einer kontextfreien Grammatik

- **Definition**
  - Eine kontextfreie Grammatik ist ein Vierer-Tupel $G=(V,\Sigma,R,S)$
    - V: Variablen
    - Σ: Terminale
      - V und Σ sind disjunkt
    - R : Ersetzungsregeln
      - $A \to w$ mit $A \in V$, $w \in (V \cup \Sigma)^*$
    - $S \in V$: Startvariable
- **Ableitung**
  - Falls $A \to w$ in R, dann ist $uAv \Rightarrow uwv$
  - $u \Rightarrow^* v$, wenn es ein $k \geq 0$ gibt mit
    - $u \Rightarrow u_1 \Rightarrow u_2 \Rightarrow \ldots \Rightarrow u_k \Rightarrow v$
- **Sprache der Grammatik G**
  - $L(G) = \{ w \in \Sigma^* \mid S \Rightarrow^* v\}$

- **G = ({A,B}, {0,1,#}, R, A)**
- **R = {A → 0A1, A → B, B → #}**
- **Alternative Schreibweise:**
- **R = {A → 0A1 | B, B → #}**

- $$\begin{aligned} \mathtt{A} &\Rightarrow \mathtt{0A1} \\ &\Rightarrow \mathtt{00A11} \\ &\Rightarrow \mathtt{000A111} \\ &\Rightarrow \mathtt{000B111} \\ &\Rightarrow \mathtt{000\#111} \end{aligned}$$

Also:

$\mathtt{A} \Rightarrow^* \mathtt{000\#111}$

Damit ist das Wort 000#111 in der Sprache L(G)

L(G) = {#, 0#1, 00#11, 000#111, 0000#1111, ...}

Probleme mit kontextfreien Grammatiken

- **Mehrdeutigkeit:**
  - Beispiel: Grammatik ({S}, {+,x,3}, R, A}
  - R = {S → S + S,
    S → S x S,
    S → 3}
  - Betrachte das Wort: w = 3+3x3
    - S ⇒ S+S ⇒ S + SxS ⇒* 3 + 3x3
    - S ⇒ SxS ⇒ S+S x S ⇒* 3+3 x 3
  - Die Ableitung ist jeweils verschieden
    - damit ist auch die Interpretation verschieden

- **Epsilon-Regeln**
  - Beispiel: Grammatik ({S,A,B,C}, {a,b,c}, R, S }
  - R = {S → ABC | ε,
    A → BaB | ε,
    B → ε | c,
    C → BAbACB | ε }
  - S ⇒ A*BC*
    ⇒ A**BBAbACB** C
    ⇒ ABBAbA **BAbACB** B
    ⇒ ABBAbA *B*AbA **BAbACB** B B
    ⇒ ABBAbA **c**AbA BAbACB B B
    ⇒* εεεεεbεcεbεεεbεεεεεε
    = bcbb
  - Zwischenergebnisse können gigantisch lang werden
    - und dann wieder verschwinden

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Ist ein Wort in der Sprache?

- **Die Entscheidung, ob ein Wort in einer kontextfreien Grammatik abgeleitet werden kann ist nicht trivial:**
  - Mehrdeutige Ableitung
  - Riesige Worte, die aufgrund der Epsilon-Regeln wieder verschwinden
- **Wenig aussichtsreiche Ansätze:**
  1. Versuche Terminalsymbole von links nach rechts zu erzeugen
     - Führt für viele Sprachen in Sackgassen
  2. Probiere alle Ableitungsmöglichkeiten aus
     - Grammatik mit Epsilon-Regeln: Unendliche Laufzeit
     - Grammatik ohne Epsilon-Regeln: Unglaublich schlechte (= exponentielle) Laufzeit
- **Die Lösungsstrategie:**
  - Chomsky-Normalform
    - Jede Grammatik lässt sich (praktisch) ohne Epsilon-Regeln in einer einfachen Form darstellen
  - Cocke-Younger-Kasami-Algorithms
    - Durch dynamische Programmierung lässt sich die Laufzeit reduzieren

Chomsky-Normalform

- **Definition**
  - Eine kontextfreie Grammatik ist in Chomsky-Normalform, falls jede Regel die Form
    - $A \to BC$ oder
    - $A \to a$ oder
    - $S \to \varepsilon$ hat
  - wobei
    - A,B,C,S sind Variablen
    - a ist ein Terminal
    - B,C sind nicht das Startsymbol,
    - S ist das Startsymbol.
- **Beispiel**
    - $G = (\{A,B,C,N,E\}, \{0,1,\#\}, R, S\}$
    - $R = \{S \to NC, N \to 0, S \to \#,$
      $A \to NC, C \to AE, E \to 1, A \to \#\}$

Chomsky-Normalform

- **Definition**
  - Eine kontextfreie Grammatik ist in Chomsky-Normalform, falls jede Regel die Form
    - $A \rightarrow BC$ oder
    - $A \rightarrow a$ oder
    - $S \rightarrow \varepsilon$ hat
  - wobei
    - A,B,C,S sind Variablen
    - a ist ein Terminal
    - B,C sind nicht das Startsymbol,
    - S ist das Startsymbol.

- **Theorem**
  - Jede kontextfreie Sprache kann in Chomsky-Normalform dargestellt werden.

- **Beispiel**
  - Kontextfreie Grammatik
    - $G = (\{A,B\}, \{0,1,\#\}, R, A)$
    - $R = \{A \rightarrow 0A1, A \rightarrow B, B \rightarrow \#\}$
  - Grammatik mit gleicher Sprache in Chomski-Normalform
    - $G' = (\{A,B,C,N,E\}, \{0,1,\#\}, R, S)$
    - $R = \{S \rightarrow NC, N \rightarrow 0, S \rightarrow \#,$
      $A \rightarrow NC, C \rightarrow AE, E \rightarrow 1, A \rightarrow \#\}$

Chomsky-Normalform

- **Definition**
  - Eine kontextfreie Grammatik ist in Chomsky-Normalform, falls jede Regel die Form
    - $A \rightarrow BC$ oder
    - $A \rightarrow a$ oder
    - $S \rightarrow \varepsilon$ hat
  - wobei
    - A,B,C,S sind Variablen
    - a ist ein Terminal
    - B,C sind nicht das Startsymbol,
    - S ist das Startsymbol.
- **Theorem**
  - Jede kontextfreie Sprache kann in Chomsky-Normalform dargestellt werden.

- **Beweisidee:**
  - Forme alle Produktionsregeln, welche gegen die Normalform verstoßen, um
  - Führe ggf. neue Variablen ein
- **Vier Probleme**
  1. **Startsymbol auf der rechten Seite**
     - Lösung: Neues Startsymbol, Übernahme der direkten Ableitung des alten Startsymbols
  2. **Epsilon-Regeln: $A \rightarrow \varepsilon$**
     - Lösung: Falls A in einer Regel rechts vorkommt, füge neue Regeln ohne A rechts ein
  3. **Eins-zu-eins-Regeln $A \rightarrow B$**
     - Lösung: In Ziele einsetzen
  4. **Lange und gemischte Regeln: $A \rightarrow aBcAbA$**
     - Lösung: neue Variablen/neue Regeln für schrittweise Ableitung

Konstruktion der Chomsky-Normalform

- **Theorem**
  - Jede kontextfreie Sprache kann in Chomsky-Normalform dargestellt werden.
- **Beweis:**
  - Gegeben eine kontextfreie Grammatik $G=(V,\Sigma,R,S)$
  - Konstruiere äquivalente Grammatiken G_1,G_2,G_3,G_4
- **G_4 ist dann eine kontextfreie Grammatik in Chomsky-Normalform**
  - D.h. jede Regel hat die Form
    - $A \rightarrow BC$ oder
    - $A \rightarrow a$ oder
    - $S \rightarrow \varepsilon$ hat
  - wobei
    - A,B,C,S sind Variablen
    - a ist ein Terminal
    - B,C sind nicht das Startsymbol,
    - S ist das Startsymbol.

1. **Konstruiere äquiv. kontextfreie Grammatik G_1, in der das Startsymbol S nicht auf der rechten Seite vorkommt**

- **$G_1 = (V,\Sigma,R',S')$**
- **R' besteht aus allen Regeln von R**
  - und der neuen Regel $S' \rightarrow S$
- **G_1 ist äquivalent zu G und erfüllt Bedingung 1.**

Beweis Fortsetzung

- **Gegeben eine kontextfreie Grammatik $G_1 = (V,\Sigma,R,S)$, in der das Startsymbol S nicht auf der rechten Seite vorkommt**
- **Konstruiere eine äquiv. Grammatik $G_2 = (V,\Sigma,R',S)$ ohne Regeln der Form $A \rightarrow \varepsilon$,**
  - für $A \in V \setminus \{S\}$
- **R' entsteht durch folgende Modifikation von R**
- **Für jede Regel $A \rightarrow \varepsilon$ aus R mit $A \in V \setminus \{S\}$**
  - Lösche Regel $A \rightarrow \varepsilon$ aus R
  - Falls A auf der rechten Seite einer Regel einfach vorkommt,
    - d.h. $B \rightarrow uAv$
  - füge Regel $B \rightarrow uv$ zu R hinzu
    - Falls $uv = \varepsilon$ (also $B \rightarrow A$)
    - füge $B \rightarrow \varepsilon$ nur hinzu, falls $B \rightarrow \varepsilon$ noch nicht behandelt wurde
  - Falls A auf der rechten Seite einer Regel mehrfach vorkommt,
    - z.B. $B \rightarrow uAvAw$
  - füge jede Kombination der Ersetzungen von A durch ε zu R' hinzu
    - Also: $B \rightarrow uAvAw$, $B \rightarrow uvAw$, $B \rightarrow uAvw$, $B \rightarrow uvw$
    - Falls $uvw= \varepsilon$ füge $B \rightarrow \varepsilon$ nur hinzu, falls $B \rightarrow \varepsilon$ noch nicht behandelt wurde

Beweis 2. Fortsetzung

- **Gegeben eine kontextfreie Grammatik $G_2 = (V,\Sigma,R,S)$,**
  - in der das Startsymbol S nicht auf der rechten Seite vorkommt
  - ohne Regeln der Form $A \rightarrow \varepsilon$, für $A \in V \setminus \{S\}$
- **Konstruiere eine äquiv. Grammatik $G_3 = (V,\Sigma,R',S)$ ohne Regeln der Form $A \rightarrow B$**
- **R' entsteht durch folgende Modifikation von R**
- **Für jede Regel $A \rightarrow B$ aus R**
  - Lösche Regel $A \rightarrow B$ aus R
  - Für jede Regel $B \rightarrow u$
    - füge Regel $A \rightarrow u$ hinzu
    - außer
      - $A \rightarrow u$ hat die Form $A \rightarrow C$ für $A \in V$
      - und diese Regel wurde schon behandelt
- **Korrektheitsbeweis folgt durch Betrachtung der Ableitung eines Wortes in G_2 und G_3**

Beweis 3. Fortsetzung

- **Gegeben eine kontextfreie Grammatik $G_3 = (V,\Sigma,R,S)$,**
  - in der das Startsymbol S nicht auf der rechten Seite vorkommt
  - ohne Regeln der Form $A \to \varepsilon$, für $A \in V \setminus \{S\}$
  - ohne Regeln der Form $A \to B$ für $A,B \in V$
- **Konstruiere kontextfreie Grammatik $G_4 = (V',\Sigma,R',S)$ in Chomsky-Normalform, d.h.**
  1. ohne Regeln $A \to u_1u_2...u_m$ für $m \geq 2$, wobei ein Symbol u_i ein Terminal ist
  2. ohne Regeln $A \to u_1u_2...u_m$ für $m \geq 3$

1. **Führe für jedes Terminal a eine Variable V[a] ein**
   - Füge Regeln $V[a] \to a$ für jedes Terminal a hinzu
   - Ersetze in jeder Regel $A \to u_1u_2...u_n$ für $n \geq 2$ die Vorkommen der Terminale u_i durch die Hilfsvariablen $V[u_i]$
2. **Für jede Regel $A \to u_1u_2...u_m$ mit $m \geq 3$**
   - **Lösche diese Regel**
   - Füge neue Variablen $V[u_2...u_m]$, $V[u_3...u_m]$, $V[u_{m-1}u_m]$ hinzu, sowie die Regeln

$$
\begin{aligned}
&A \to u_1V[u_2...u_m],\\
&V[u_2...u_m] \to u_2V[u_3...u_m],\\
&V[u_3...u_m] \to u_3V[u_4...u_m],\\
&\qquad ...\\
&V[u_{m-1}u_m] \to u_{m-1}u_m
\end{aligned}
$$

- **Enstandene Grammatik G_4 ist äquivalent und in Chomsky-Normalform.**

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Beispiel: 1. Schritt

- **Betrachte**
  - kontextfreie Grammatik (V,Σ,R,S) mit den Regeln mit
  - $V = \{A,B,S\}$
  - $\Sigma = \{a,b\}$
  - $R = \{ \quad S \to ASA,$
    $S \to aB,$
    $A \to B,$
    $A \to S,$
    $B \to b,$
    $B \to \varepsilon \}$

- **Äquivalente kontextfreie Grammatik $G' = (V,\Sigma,R,S')$ mit den Regeln**
  - ohne Vorkommen des Startsymbols auf der rechten Seite
  - S' neues Startsymbol

- $R = \{ \quad \mathbf{S' \to S,}$
  $S \to ASA,$
  $S \to aB,$
  $A \to B,$
  $A \to S,$
  $B \to b,$
  $B \to \varepsilon \}$

Beispiel: 2. Schritt

- $R = \{ \; S' \to S,$
  $S \to ASA,$
  $S \to aB,$
  $A \to B,$
  $A \to S,$
  $B \to b,$
  $\mathbf{B \to \varepsilon} \; \}$

- **Entfernen von $B \to \varepsilon$, neue Regeln:**
- $R' = \{ \; S' \to S,$
  $S \to ASA,$
  $S \to aB,$ $\mathbf{S \to a,}$
  $A \to B,$
  $\mathbf{A \to \varepsilon,}$
  $A \to S,$
  $B \to b \; \}$

- **Entfernen von $A \to \varepsilon$, neue Regeln:**
- $R'' = \{ \; \mathbf{S' \to S,}$
  $S \to ASA,$ $\mathbf{S \to SA, S \to AS, S \to S}$,
  $S \to aB,$
  $S \to a,$
  $A \to B,$
  $A \to S,$
  $B \to b \; \}$

Beispiel: 3a. Schritt

- R = { S' → S,
  S → ASA,
  S → SA,
  S → AS,
  S → S,
  S → aB,
  S → a,
  A → B,
  A → S,
  B → b }
- **Entferne S → S**
- R' = { **S' → S,**
  S → ASA,
  S → SA,
  S → AS,
  S → aB,
  S → a,
  A → B,
  A → S,
  B → b }
- **Entferne S' → S, neue Regeln:**
- R'' = { **S' → ASA,** **S' → SA,**
  S' → AS, **S' → aB,**
  S' → a
  S → ASA,
  S → SA,
  S → AS,
  S → aB,
  S → a,
  A → B,
  A → S,
  B → b }
- **Entferne A → B, neue Regeln:**
- R''' = { S' → ASA, S' → SA
  S' → AS, S' → aB,
  S' → a,
  S → ASA,
  S → SA,
  S → AS,
  S → aB,
  S → a,
  A → b,
  A → S,
  B → b }

Beispiel: 3b. Schritt

- R' = { S' → ASA, S' → SA
  S' → AS, S' → aB,
  S' → a,
  S → ASA,
  S → SA,
  S → AS,
  S → aB,
  S → a,
  A → b,
  A → S,
  B → b }

- Entferne **A → S, neue Regeln**
- R'' = { S' → ASA, S' → SA
  S' → AS, S' →aB,
  S' → a,
  S → ASA,
  S → SA,
  S → AS,
  S → aB,
  S → a,
  A → b,
  A → ASA,
  A → SA,
  A → AS,
  A → aB,
  A → a,
  B → b }

Beispiel: 4. Schritt

- R = { **S' → ASA,**
  S' → SA
  S' → AS,
  S' → aB,
  S' → a,
  S → ASA,
  S → SA,
  S → AS,
  S → aB,
  S → a,
  A → b,
  A → ASA,
  A → SA,
  A → AS,
  A → aB,
  A → a,
  B → b }

- Füge Variablen V[SA], V[a] hinzu
- Neue Regeln
- **R' = {** **S' → A V[SA],**
  V[SA] → SA
  S' → SA
  S' → AS,
  S' → V[a]B,
  S' → a,
  S → A V[SA],
  S → SA,
  S → AS,
  S → V[a]B,
  V[a] → a,
  S → a,
  A → b,
  A → A V[SA],
  A → SA,
  A → AS,
  A → V[a] B,
  A → a,
  B → b }
- **Grammatik R' ist in Chomsky-Normalfom**

Das Wort-Problem der kontextfreien Sprachen

- **Gegeben:**
  - Eine kontextfreie Grammatik $G=(V,\Sigma,R,S)$ in Chomsky-Normalform und
  - ein Wort $w \in \Sigma^*$
- **Entscheide:**
  - Ist $w \in L(G)$
- **Oder:**
  - Kann w aus dem Startsymbol S abgeleitet werden?

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

CYK-Algorithmus

Bei Eingabe $w = w_1 \cdots w_n$:

1. Falls $w = \epsilon$ und $S \to \epsilon$ eine Regel ist, akzeptiere.
2. Für $i = 1$ bis n
3.   Für jede Variable v
4.   Teste, ob $v \to b$ eine Regel ist, wobei $b = w_i$
5.   Falls ja, füge v zu $T(i,i)$ hinzu.
6. Für $\ell = 2$ bis n
7.   Für $i = 1$ bis $n - \ell + 1$
8.   Setze $j = i + \ell - 1$
9.   Für $k = i$ bis $j - 1$
10.   Für jede Regel $v \to uz$
11.   Falls $u \in T(i,k)$ und $z \in T(k+1,j)$,
                füge v zu $T(i,j)$ hinzu.
12. Falls S in $T(1,n)$ enthalten ist, akzeptiere. Sonst lehne ab.

l = Länge der Teilfolge

i = Startindex der Teilfolge

j = Schlussindex der Teilfolge

Ende der 6. Vorlesung

CoNe Freiburg

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Informatik III
Christian Schindelhauer
schindel@informatik.uni-freiburg.de