

Theoretische Grundlagen der Informatik

Tutorium 11

Institut für Theoretische Informatik



Der Cocke-Younger-Kasami-Algorithmus (CYK) löst das Wortproblem für kontextfreie Sprachen (CH-2) in $\mathcal{O}(n^3)$. Um CYK anzuwenden, muss die gegebene Grammatik erst in Chomsky-Normalform gebracht werden. Das ist für jede CH-2-Grammatik möglich.

Chomsky-Normalform

Eine CH-2-Grammatik $G = (\Sigma, \mathcal{V}, S, \mathcal{R})$ ist in Chomsky-Normalform, wenn jede Produktion aus \mathcal{R} eine der folgenden Formen hat:

■ $A \rightarrow BC$

■ $A \rightarrow a$

Wobei gilt $A, B, C \in \mathcal{V}$ und $a \in \Sigma$. Um das leere Wort in der Sprache zu erlauben, lässt sich die Grammatik leicht mit neuem Startsymbol S' ergänzen mit der Regel

$$S' \rightarrow S \mid \varepsilon$$

1. Für alle $a \in \Sigma$ und für alle Produktionen, auf deren rechter Seite a vorkommt (außer für $V \rightarrow a$, mit $V \in \mathcal{V}$), wird jedes Vorkommen von a durch ein *neues* Nichtterminalsymbol A ersetzt und die Produktion $A \rightarrow a$ wird hinzugefügt.

Umwandlungsbeispiel (Schritt 1 von 4)

$$S \rightarrow XY$$

$$X \rightarrow aXb \mid Z \mid \varepsilon$$

$$Y \rightarrow ccY \mid \varepsilon$$

$$Z \rightarrow X$$

 \Rightarrow

$$S \rightarrow XY$$

$$X \rightarrow AXB \mid Z \mid \varepsilon$$

$$Y \rightarrow CCY \mid \varepsilon$$

$$Z \rightarrow X$$

$$A \rightarrow a$$

$$B \rightarrow b$$

$$C \rightarrow c$$

2. Für Produktionen mit mehr als zwei Variablen rechts werden *neue Nichtterminale* eingeführt und dazu *passende Produktionen* hinzugefügt.

Umwandlungsbeispiel (Schritt 2 von 4)

$$S \rightarrow XY$$

$$X \rightarrow \textcolor{red}{AXB} \mid Z \mid \varepsilon$$

$$Y \rightarrow \textcolor{red}{CCY} \mid \varepsilon$$

$$Z \rightarrow X$$

$$A \rightarrow a$$

$$B \rightarrow b$$

$$C \rightarrow c$$

 \Rightarrow

$$S \rightarrow XY$$

$$X \rightarrow \textcolor{blue}{FB} \mid Z \mid \varepsilon$$

$$Y \rightarrow \textcolor{blue}{GY} \mid \varepsilon$$

$$Z \rightarrow X$$

$$\textcolor{blue}{F} \rightarrow \textcolor{blue}{AX}$$

$$\textcolor{blue}{G} \rightarrow \textcolor{blue}{CC}$$

$$A \rightarrow a$$

$$B \rightarrow b$$

$$C \rightarrow c$$

3. Entfernen von Produktionen der Form $V \rightarrow \epsilon$ für $V \in \mathcal{V}, v \neq S$
 \Rightarrow „Vorwegnahme“ dieser Produktionen: Für jede Produktion mit einem der obigen V auf der rechten Seite wird eine **neue Produktion** ohne dieses V hinzugefügt.

Umwandlungsbeispiel (Schritt 3 von 4)

$$S \rightarrow XY$$

$$X \rightarrow FB \mid Z \mid \epsilon$$

$$Y \rightarrow GY \mid \epsilon$$

$$Z \rightarrow X$$

$$F \rightarrow AX$$

$$G \rightarrow CC$$

$$A \rightarrow a, B \rightarrow b, C \rightarrow c$$

 \Rightarrow

$$S \rightarrow XY \mid X \mid Y \mid \epsilon$$

$$X \rightarrow FB \mid Z \mid \epsilon$$

$$Y \rightarrow GY \mid G$$

$$Z \rightarrow X \mid \epsilon$$

$$F \rightarrow AX \mid A$$

$$G \rightarrow CC$$

$$A \rightarrow a, B \rightarrow b, C \rightarrow c$$

4. Für Produktionen mit einer Variablen rechts werden Zyklen gesucht, für gefundene Zyklen werden alle Vorkommnisse aller Variablen des Zyklus durch einen Repräsentanten ausgetauscht. Danach werden triviale Produktionen entfernt.

Umwandlungsbeispiel (Schritt 4a von 4)

$$S \rightarrow XY \mid X \mid Y \mid \varepsilon$$

$$X \rightarrow FB \mid Z$$

$$Y \rightarrow GY \mid G$$

$$Z \rightarrow X$$

$$F \rightarrow AX \mid A$$

$$G \rightarrow CC$$

$$A \rightarrow a, B \rightarrow b, C \rightarrow c$$

 \Rightarrow

$$S \rightarrow XY \mid X \mid Y \mid \varepsilon$$

$$X \rightarrow FB \mid X$$

$$Y \rightarrow GY \mid G$$

$$X \rightarrow X$$

$$F \rightarrow AX$$

$$G \rightarrow CC$$

$$A \rightarrow a, B \rightarrow b, C \rightarrow c$$

4. Alle Regeln, die rechts eine einzelne Variable haben, werden durch „Vorziehen“ der Regeln eliminiert.

Außerdem wird ein neues Startsymbol eingeführt, falls eine Regel $S \rightarrow \varepsilon$ existiert.

Umwandlungsbeispiel (Schritt 4b von 4)

$$S \rightarrow XY \mid \textcolor{red}{X} \mid \textcolor{red}{Y} \mid \varepsilon$$

$$X \rightarrow FB$$

$$Y \rightarrow GY \mid \textcolor{red}{G}$$

$$F \rightarrow AX$$

$$G \rightarrow CC$$

$$A \rightarrow a$$

$$B \rightarrow b$$

$$C \rightarrow c$$

$$\Rightarrow$$

$$S \rightarrow S' \mid \varepsilon$$

$$S' \rightarrow XY \mid \textcolor{blue}{FB} \mid \textcolor{blue}{GY} \mid \textcolor{blue}{CC}$$

$$X \rightarrow FB$$

$$Y \rightarrow GY \mid \textcolor{blue}{CC}$$

$$F \rightarrow AX$$

$$G \rightarrow CC$$

$$A \rightarrow a$$

$$B \rightarrow b$$

$$C \rightarrow c$$

Sei $G = (\Sigma, V, S, R)$ die CH-2-Grammatik mit $\Sigma = \{a, b\}$,
 $V = \{A, B, C, D, E, S\}$ und der folgenden Regelmengemenge R :

$$S \rightarrow A \mid aAa \mid bBb \mid \varepsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow C \mid b$$

$$C \rightarrow CDE \mid \varepsilon$$

$$D \rightarrow A \mid B \mid ab$$

$$E \rightarrow S$$

Bestimme eine Grammatik G' für $L(G)$ in Chomsky-Normalform.

CYK ist ein Algorithmus, um das Wortproblem in CH-2 zu lösen. Um den Algorithmus anzuwenden, muss eine Grammatik in Chomsky-Normalform vorliegen.

Grundidee zur Überprüfung eines Wortes der Länge n :

- Wir betrachten $V_{i,j}$ = Menge der Nichtterminale, aus denen das Teilwort der Position i bis j abgeleitet werden kann
- Die Frage, ob $V_{i,j}$ ableitbar ist, lässt sich entscheiden durch Betrachten aller möglichen $V_{i,k}$ und $V_{k+1,j}$
- $V_{i,i}$ sind trivial
- Bottom-up lässt sich dadurch $V_{1,n}$ berechnen
- Ist $S \in V_{1,n}$, so lässt sich das Wort ableiten

Gegeben sei die Grammatik $G = (\mathcal{T}, \mathcal{V}, S, \mathcal{P})$ mit den folgenden Produktionen aus \mathcal{P} :

$$S \rightarrow AX \mid AB$$

$$X \rightarrow SB \mid AB$$

$$A \rightarrow a$$

$$B \rightarrow b$$

1. Lässt sich der CYK-Algorithmus auf G anwenden?
2. Ist das Wort $aaabbb$ in der Sprache $\mathcal{L}(G)$?

Lemma

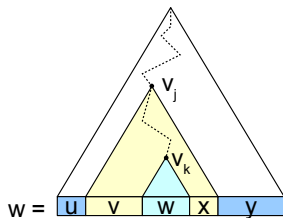
Für jede kontextfreie Sprache L gibt es eine Konstante $n \in \mathbb{N}$, so dass sich jedes Wort $z \in L$ mit $|z| \geq n$ so als

$$z = uvwxy$$

schreiben lässt, dass

- $|vx| \geq 1$,
- $|vwx| \leq n$ und
- für alle $i \geq 0$ das Wort $uv^iwx^iy \in L$ ist.

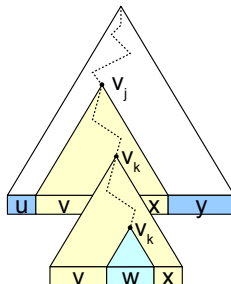
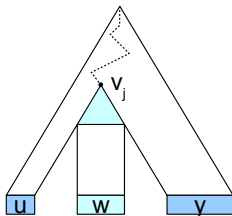
- Jeder Knoten im Ableitungsbaum (wie wir ihn in CYK sehen) steht für ein Nichtterminalsymbol
- Ab einer gewissen Höhe des Baumes (bzw. Länge des Wortes) muss ein Nichtterminal im Baum mehrmals in einer Reihe vorkommen
- Man kann also aus einem Nichtterminalsymbol dasselbe Symbol wieder ableiten
- Da das Wort durch jede Ableitung (außer zu Terminalsymbolen) länger wird, gibt es eine „Schleife“ beim Ableiten
- Diese Schleife kann man also „pumpen“, also beliebig oft (oder auch gar nicht) durchlaufen



} h

Gegeben: Wort $z \in L$
mit $|x| \geq n$

Ableitungsbaum T für z
mit Höhe $h \geq N$



Lemma

Für jede kontextfreie Sprache L gibt es eine Konstante $n \in \mathbb{N}$, so dass für jedes Wort $z \in L$ mit $|z| \geq n$ gilt:

Wenn wir in z mindestens n Buchstaben markieren, so lässt sich z so als $z = uvwxy$ schreiben, dass

- von den mindestens n markierten Buchstaben
 - mindestens einer zu vx gehört und
 - höchstens n zu vw gehören und
- für alle $i \geq 0$ das Wort $uv^iwx^iy \in L$ ist.

- Anwendung genau wie „altes“ Pumping Lemma
- Wird verwendet, um zu zeigen dass eine Sprache nicht kontextfrei ist
- Dazu: Kontraposition bilden und folgern, dass Sprache nicht kontextfrei

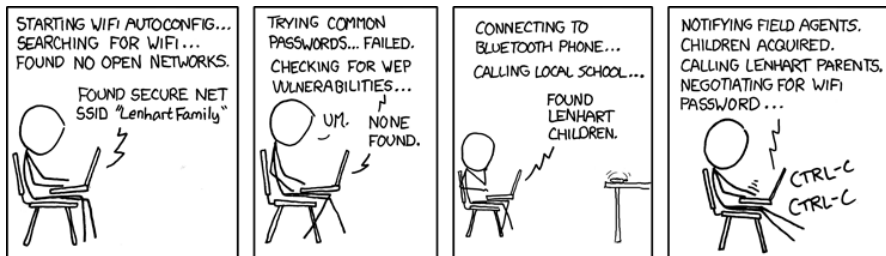
Zeige, dass folgende Sprachen nicht kontextfrei sind:

1. $L_1 = \{w \in \{a, b, c\}^* : |w|_a = |w|_b = |w|_c\}$, wobei $|w|_x$ für die Häufigkeit des Buchstabens x im Wort w stehe.
2. $L_2 = \{a^i b^j c^j \mid j \leq i\}$
3. $L_3 = \{a^i b^j c^j d^j \mid i, j \geq 1\}$

Bis zum nächsten Mal!



<http://twitter.com/herpderpedia>



I hear this is an option in the latest Ubuntu release.



Dieses Werk ist unter einem "Creative Commons Namensnennung-Weitergabe unter gleichen Bedingungen 3.0 Deutschland"-Lizenzvertrag lizenziert. Um eine Kopie der Lizenz zu erhalten, gehen Sie bitte zu <http://creativecommons.org/licenses/by-sa/3.0/de/> oder schreiben Sie an Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

Davon ausgenommen sind das Titelbild, welches aus der März-April 2002 Ausgabe von American Scientist erschienen ist und ohne Erlaubnis verwendet wird, sowie das KIT Beamer Theme. Hierfür gelten die Bestimmungen der jeweiligen Urheber.