Formální jazyky a automaty

Jednoduchá pravidla a epsilon-pravidla, Chomského normální forma, věta o vkládání pro bezkontextové jazyky

Jan Křetínský

Fakulta informatiky, MU Brno

Jaro 2024

Kanonické tvary bezkontextových gramatik

- redukované bezkontextové gramatiky
- gramatiky bez ε-pravidel
- gramatiky bez jednoduchých pravidel
- vlastní gramatiky
- Chomského normální forma
- gramatiky bez levé rekurze
- Greibachové normální forma

ε-pravidla

Definice 3.13.

Řekneme, že CFG $\mathcal{G} = (N, \Sigma, P, S)$ je bez ε-pravidel právě když buď

1. P neobsahuje žádné ε-pravidlo (tj. pravidlo tvaru $A \to \varepsilon$) nebo
2. v P existuje právě jedno ε-pravidlo $S \to \varepsilon$ a S se nevyskytuje na pravé straně žádného pravidla z P.

Příklad

$\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde P obsahuje pravidla

$$
\begin{array}{lcl}
S & \to & aAbBc \\
A & \to & BB \mid a \mid \varepsilon \\
B & \to & AcA \mid b
\end{array}
$$

Algoritmus pro odstranění ε-pravidel

Vstup: CFG $\mathcal{G} = (N, \Sigma, P, S)$
Výstup: CFG $\mathcal{G}' = (N', \Sigma, P', S')$ bez ε-pravidel splňující $L(\mathcal{G}) = L(\mathcal{G}')$

Zkonstruuj $N_\varepsilon = \{A \in N \mid A \Rightarrow^* \varepsilon\}$
Množinu pravidel P' zkonstruuj takto:
for all $A \to X_1 \dots X_n \in P, n > 0$ **do**
    přidej do P' všechna pravidla tvaru $A \to \alpha_1 \dots \alpha_n$ splňující
        (a) pokud $X_i \notin N_\varepsilon$ pak $\alpha_i = X_i$
        (b) pokud $X_i \in N_\varepsilon$ pak α_i je buď X_i nebo ε
        (c) ne všechna α_i jsou ε
if $S \in N_\varepsilon$ **then**
    přidej do P' pravidla $S' \to S \mid \varepsilon$
    $N' := N \cup \{S'\}$
else
    $N' := N; \; S' := S$

Algoritmus pro odstranění ε-pravidel

Vstup: CFG $\mathcal{G} = (N, \Sigma, P, S)$
Výstup: CFG $\mathcal{G}' = (N', \Sigma, P', S')$ bez ε-pravidel splňující $L(\mathcal{G}) = L(\mathcal{G}')$

Zkonstruuj $N_\varepsilon = \{A \in N \mid A \Rightarrow^* \varepsilon\}$
Množinu pravidel P' zkonstruuj takto:
for all $A \to X_1 \dots X_n \in P, n > 0$ **do**
  přidej do P' všechna pravidla tvaru $A \to \alpha_1 \dots \alpha_n$ splňující
    (a) pokud $X_i \notin N_\varepsilon$ pak $\alpha_i = X_i$
    (b) pokud $X_i \in N_\varepsilon$ pak α_i je buď X_i nebo ε
    (c) ne všechna α_i jsou ε
if $S \in N_\varepsilon$ **then**
  přidej do P' pravidla $S' \to S \mid \varepsilon$
  $N' := N \cup \{S'\}$
else
  $N' := N;\ S' := S$

Korektnost algoritmu:
Konečnost
Výsledná gramatika je bez ε-pravidel
Ekvivalence gramatik

Příklad

$\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde P obsahuje pravidla

$$
\begin{array}{lcl}
S & \rightarrow & aAbBc \mid AB \\
A & \rightarrow & BB \mid a \mid \varepsilon \\
B & \rightarrow & AA \mid b
\end{array}
$$

Jednoduchá pravidla

Jednoduchým pravidlem nazýváme každé pravidlo tvaru $A \to B$, kde $A, B \in N$.

$$
\begin{array}{lcl}
S & \to & aAbBc \\
A & \to & aA \mid B \mid a \\
B & \to & bB \mid b
\end{array}
$$

Algoritmus pro odstranění jednoduchých pravidel

Vstup: CFG $\mathcal{G} = (N, \Sigma, P, S)$ bez ε-pravidel
Výstup: CFG $\mathcal{G}' = (N, \Sigma, P', S)$ bez jednoduchých a ε-pravidel, kde $L(\mathcal{G}) = L(\mathcal{G}')$

for all $A \in N$ **do**
  $i := 0;\ M_0 := \{A\}$
  repeat
    $i := i + 1$
    $M_i := M_{i-1} \cup \{C \mid C \to B \in P,\ B \in M_{i-1}\}$
  until $M_i = M_{i-1}$
  $M_A := M_i$

$P' := \emptyset$
for all $B \to \alpha \in P$, které není jednoduché **do**
  do P' přidej $A \to \alpha$ pro každé $A \in M_B$, tj. $A \Rightarrow^*_{\mathcal{G}} B$

Příklad

$\mathcal{G} = (\{S, A, B, C\}, \{a, b, c\}, P, S)$, kde P obsahuje pravidla

$$
\begin{array}{lcl}
S & \rightarrow & ABC \\
A & \rightarrow & aA \mid B \mid a \\
B & \rightarrow & bB \mid A \\
C & \rightarrow & cC \mid A
\end{array}
$$

Korektnost algoritmu

Konečnost.

Výsledná gramatika neobsahuje jednoduchá pravidla ani ε-pravidla.

Ekvivalence gramatik:

$L(\mathcal{G}') \subseteq L(\mathcal{G})$ Nechť $w \in L(\mathcal{G}')$, pak existuje derivace
$S = \alpha_0 \Rightarrow_{\mathcal{G}'} \alpha_1 \Rightarrow_{\mathcal{G}'} \ldots \Rightarrow_{\mathcal{G}'} \alpha_n = w$.
Pokud bylo při kroku $\alpha_i \Rightarrow_{\mathcal{G}'} \alpha_{i+1}$ použito pravidlo $A \to \beta$, pak $A \in M_B$ pro nějaké $B \in N$ takové, že v $\mathcal{G}$ platí $A \Rightarrow^* B$ a $B \to \beta$.
Tedy v $\mathcal{G}$ platí $A \Rightarrow^* \beta$ a $\alpha_i \Rightarrow^* \alpha_{i+1}$.

$L(\mathcal{G}) \subseteq L(\mathcal{G}')$ Nechť $w \in L(\mathcal{G})$, pak existuje levá derivace
$S = \alpha_0 \Rightarrow_{\mathcal{G}} \alpha_1 \Rightarrow_{\mathcal{G}} \ldots \Rightarrow_{\mathcal{G}} \alpha_n = w$.
Tu lze rozdělit na úseky tak, že v každém úseku se použila pouze jednoduchá pravidla anebo žádné jednoduché pravidlo.
Úsek s jednoduchými pravidly lze vypustit (a ponechat pouze jeho počáteční větnou formu). $\square$

Vlastní bezkontextová gramatika

Definice 3.17.

CFG $\mathcal{G} = (N, \Sigma, P, S)$ se nazývá necyklická, právě když neexistuje $A \in N$ takový, že $A \Rightarrow^+ A$.
$\mathcal{G}$ se nazývá vlastní, právě když je bez nepoužitelných symbolů, bez ε-pravidel a necyklická.

Věta 3.18.

Ke každému neprázdnému bezkontextovému jazyku existuje **vlastní** bezkontextová gramatika, která jej generuje.

Důkaz. Z bezkontextové gramatiky pro neprázdný jazyk odstraníme ε-pravidla a jednoduchá pravidla. Odstraněním nepoužitelných symbolů pak získáme vlastní gramatiku. $\square$

Chomského normální forma

Definice 3.19.

Bezkontextová gramatika $\mathcal{G} = (N, \Sigma, P, S)$ je v Chomského normální formě (CNF), právě když $\mathcal{G}$ je bez ε-pravidel a každé pravidlo z P má jeden z těchto tvarů:

1. $A \to BC$, kde $B, C \in N$
2. $A \to a$, kde $a \in \Sigma$
3. $S \to \varepsilon$

Věta 3.21.

Každý bezkontextový jazyk lze generovat bezkontextovou gramatikou v Chomského normální formě.

Příklad

$\mathcal{G} = (\{S, A, B\}, \{a, b\}, P, S)$, kde P obsahuje pravidla

$$
\begin{aligned}
S &\rightarrow AS \mid a \\
A &\rightarrow AB \mid AA \mid a \\
B &\rightarrow b
\end{aligned}
$$

Algoritmus transformace do CNF

Algoritmus 3.6

Princip:

1. $L = \emptyset$
2. $L \neq \emptyset$
   Gramatiku pro L převedeme na vlastní a bez jednoduchých pravidel.

   $$\begin{aligned} X &\rightarrow \varepsilon \\ X &\rightarrow a \\ X &\rightarrow A \\ X &\rightarrow ab \\ X &\rightarrow aB \\ X &\rightarrow Ab \\ X &\rightarrow AB \\ &\vdots \\ X &\rightarrow aBcD \end{aligned}$$

Lemma o vkládání pro bezkontextové jazyky

Lemma o vkládání pro bezkontextové jazyky

Věta 3.24.

Nechť L je CFL. Pak existují $p, q \in \mathbb{N}$ (závisející na L) taková, že každé slovo $z \in L$ delší než p lze psát ve tvaru $z = uvwxy$, kde

- alespoň jedno ze slov v, x je neprázdné (tj. $vx \neq \varepsilon$),
- $|vwx| \leq q$ a
- $uv^iwx^iy \in L$ pro každé $i \in \mathbb{N}_0$.

Poznámka 3.25. Tvrzení zůstává v platnosti i když namísto konstant p, q budeme všude psát jen (jedinou) konstantu n.

Použití Lemmatu o vkládání pro bezkontextové jazyky

Lemma o vkládání je implikace $P \implies Q$, kde P je výrok, že L je CFL a Q jsou uvedené vlastnosti.
Obměnu Lemmatu o vkládání $\neg Q \implies \neg P$ lze použít k důkazu, že nějaký jazyk L **není CFL**: stačí, když ukážeme platnost $\neg Q$.

$\neg Q$:

1. Pro libovolnou konstantu $n \in \mathbb{N}$
2. existuje slovo $z \in L$ delší než n takové, že
3. pro všechny slova u, v, w, x, y splňující
   $z = uvwxy$, $vx \neq \varepsilon$ a $|vwx| \leq n$
4. existuje $i \in \mathbb{N}_0$ takové, že $uv^iwx^iy \notin L$.

Příklad použití Lemmatu o vkládání

$$L = \{a^i b^i c^i \mid i \geq 1\}$$

1. Pro libovolnou konstantu $n \in \mathbb{N}$
2. existuje slovo $z \in L$ delší než n takové, že

3. pro všechny slova u, v, w, x, y splňující
   $z = uvwxy$, $vx \neq \varepsilon$ a $|vwx| \leq n$

4. existuje $i \in \mathbb{N}_0$ takové, že $uv^i wx^i y \notin L$.

$\Longrightarrow$ L není CFL.

Důkaz Lemmatu o vkládání pro bezkontextové jazyky

Nechť L je generován gramatikou v CNF.

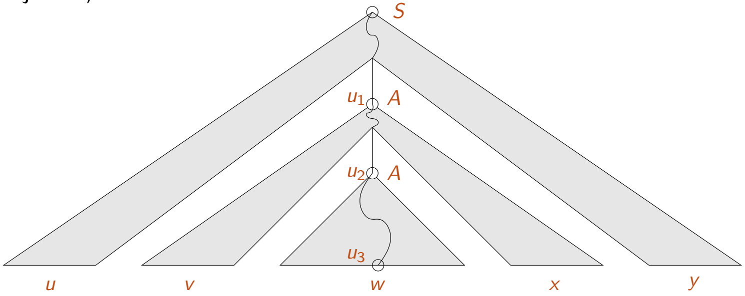
hloubka stromu $\stackrel{def}{=}$ max. počet hran na cestě = max. počet neterminálů na cestě

- Derivační strom hloubky k má max. 2^{k-1} listů $\Longrightarrow$ slovo délky nejvýše 2^{k-1}.
- Derivační strom pro slovo delší než 2^{k-1} má hloubku alespoň $k+1$. Nejdelší cesta obsahuje alespoň $k+1$ neterminálů.

Důkaz Lemmatu o vkládání pro bezkontextové jazyky

- Derivační strom pro slovo delší než 2^{k-1} má cestu délky alespoň $k+1$. Tato cesta obsahuje alespoň $k+1$ neterminálů.

Označme $k = |N|$ a položme $p = 2^{k-1}, q = 2^k$.
Nechť $z \in L$ je slovo delší než p. Pak v libovolném derivačním stromu slova z existuje cesta s alespoň $k+1$ neterminály (vezměme tu nejdelší).



u_1, u_2 poslední opakování neterminálu na této cestě.
Tedy hloubka z u_1 je nejvýše $k+1$, a proto $|vwx| \leq 2^k$. $\square$