

Moderné regulárne výrazy

Tatiana Tóthová

školiteľ: RNDr. Michal Forišek PhD.

Fakulta matematiky, fyziky a informatiky
Univerzita Komenského

28.mája 2015

Motivácia:

- teoretický model regulárnych výrazov bol v praxi rozšírený o nové konštrukcie

Cieľ :

- preskúmať z hľadiska formálnych jazykov – Chomského hierarchia, vlastnosti triedy jazykov, zložitosť

Regex tvoria **znaky** a **metaznaky**

Základné operácie:

- $(e_1)(e_2)$ zreťazenie (bez znaku)
- $(e_1)|(e_2)$ alternácia
- $(e_1)^*$ Kleeneho uzáver

Navyše: ľubovoľný znak $.$, začiatok slova $^$, koniec slova $\$$

Ďalšie operácie:

- množiny znakov $[abc], [a - z]$
- komplementy množín znakov $[\wedge abc], [\wedge a - z]$
- opakovania $?, \{n\}, \{n, m\}, +$

Regex má konečnú dĺžku!

Zložitejšie konštrukcie (1)

Číslovanie zátvoriek zľava doprava, podľa otváracej zátvorky.
Nečíslujú sa konštrukcie ($? \dots$)

Spätné referencie $\backslash k$ – odkazujú sa na k -te zátvorky

$$\alpha \underset{k}{(} \underset{k}{\beta)} \gamma \backslash k \delta$$

$$w = \underbrace{x_1 \dots x_{i-1}}_{\alpha} \underbrace{\overbrace{x_i \dots x_{j-1}}^{w_k}}_{\underset{k}{(} \underset{k}{\beta)}} \underbrace{x_j \dots x_{l-1}}_{\gamma} \underbrace{\overbrace{x_l \dots x_{m-1}}^{w_k}}_{\backslash k} \underbrace{x_m \dots x_n}_{\delta}$$

$$w_k = x_i \dots x_{j-1} = x_l \dots x_{m-1}$$

Zložitejšie konštrukcie (2)

Lookahead ($? = \dots$) – nazeranie dopredu: $\alpha(? = \beta)\gamma$

$$w = \underbrace{x_1 \dots x_{i-1}}_{\alpha} \overbrace{x_i \dots x_j}^{\beta} \underbrace{x_{j+1} \dots x_n}_{\gamma}$$

Lookbehind ($? \leq \dots$) – nazeranie dozadu: $\alpha(? \leq \beta)\gamma$

$$w = \underbrace{x_1 \dots x_{i-1}}_{\alpha} \overbrace{x_i \dots x_j}^{\beta} \underbrace{x_{j+1} \dots x_n}_{\gamma}$$

Ich **negatívne formy** ($?! \dots$), ($?<! \dots$) \rightarrow opačná akceptácia (slová, ktoré sa v danom jazyku nenachádzajú)

- $(?(a^m) * \$)(a^{m+1})^*$

$$L_1 = \{a^{im(m+1)} \mid i \in \mathbb{N}_0\}$$

iný možný zápis $\rightarrow (a^{m(m+1)})^*$

- $(a^*)\#(?(\backslash 1) * \$)(a\backslash 1)^*$

$$L_2 = \{a^m \# a^{im(m+1)} \mid i \in \mathbb{N}_0\}$$

- $((?(a^{m+1}) * a\{1, m-1\}\$ \mid a^m\$) a^m) +$

$$L_3 = \{a^m, a^{2m}, \dots, a^{m^2}\}$$

Regex základné operácie a znaky $\cdot \wedge \$ (= \mathcal{R}$)

Eregex + spätné referencie

LEregex + lookahead, lookbehind

nLEregex + negatívny lookahead, negatívny lookbehind

Triedy jazykov: \mathcal{R} , \mathcal{L}_{ERE} , \mathcal{L}_{LERE} , \mathcal{L}_{nLERE}

Veta

\mathcal{L}_{LRE} je uzavretá na zreťazenie.

Dôkaz. Nech $\alpha, \beta \in LRE_{regex}$. Regex pre $L(\alpha)L(\beta)$ je

$$(\alpha \mid \beta) \mid \alpha \mid \beta$$

Veta

Trieda nad *Regex* s pozitívnym a negatívnym lookaroundom je \mathcal{R} .

Neobsahuje spätné referencie!

Hierarchia

$$\mathcal{R} \subsetneq \mathcal{L}_{ERE} \subsetneq \mathcal{L}_{LRE} \subseteq \mathcal{L}_{nLRE} \subsetneq \mathcal{L}_{CS}$$

\mathcal{L}_{LRE} a \mathcal{L}_{nLRE} sú neporovnateľné s \mathcal{L}_{CF}

Veta

Jazyk všetkých platných výpočtov Turingovho stroja patrí do \mathcal{L}_{LRE} .

Veta

Nech $\alpha \in LRegex$ nad unárnou abecedou $\Sigma = \{a\}$ taký, že neobsahuje lookahead s $\$$ ani lookbehind s \wedge vnútri iterácie $(*)$. Existuje konštanta N taká, že ak $w \in L(\alpha)$ a $|w| > N$, potom existuje dekompozícia $w = xy$ s nasledujúcimi vlastnosťami:

- (i) $|y| \geq 1$
- (ii) $\exists k \in \mathbb{N}, k \neq 0; \forall j = 1, 2, \dots : xy^{kj} \in L(\alpha)$

Nový formalizmus

- konfigurácia: $(r_1 \dots \lceil r_i \dots r_n, w_1 \dots \lceil w_j \dots w_m)$
- indexovateľné zátvorky
- krok výpočtu

$$(r_1 \dots (\dots) \lceil^k * \dots r_n, w_1 \dots w_a^k \dots w_b^{k'} \dots \lceil w_j \dots w_m)$$

$$(1) \vdash (r_1 \dots (\dots) * \lceil^k \dots r_n, w_1 \dots w_a^k \dots w_b^{k'} \dots \lceil w_j \dots w_m)$$

$$(2) \vdash (r_1 \dots (\lceil \dots) * \lceil^k \dots r_n, w_1 \dots w_a \dots w_b \dots \lceil^k w_j \dots w_m)$$

- akceptačný výpočet $(\lceil r, \lceil w) \vdash^* (r \lceil, w \lceil)$
- jazyk

Dĺžka akceptačného výpočtu

Lema 1

Nech $\alpha \in \mathcal{L}_{ERE}$ a $w \in L(\alpha)$. Potom existuje akceptačný výpočet, ktorý má najviac $3 \cdot |\alpha| \cdot |w|^2$ konfigurácií.

Počet všetkých možných konfigurácií $\alpha \in LERegex$ je najviac

$$(r + 1)(w + 2)^{2r+2} = O(rw^{2r+2})$$

Lema 2

Majme triedu regexov takú, že funkcia $f(r)$ určuje ich **maximálnu hĺbku vnorenia lookaroudov**, kde r je dĺžka regexu. Potom počet konfigurácií v akceptačnom výpočte pre slovo dĺžky w je najviac $O(rw^2(rw)^{f(r)})$.

Veta.

$$\mathcal{L}_{LRE} \subseteq NSPACE(\log n)$$

Dôkaz. Na páske si pamätáme informáciu z poschodových symbolov – vo forme adries

Zo Savitchovej vety vyplýva: $\mathcal{L}_{LRE} \subseteq DSPACE(\log^2 n)$

Veta

$$\mathcal{L}_{nLRE} \subseteq DSPACE(\log^2 n)$$

Dôkaz. Idea Savitchovej vety, ale s konfiguráciami z formalizmu. Pre každý negatívny lookaround spúšťame nový Turingov stroj s opačnou akceptáciou.

$$r = |regex|, w = |word|$$

Veta

$L(regex\#word) \in NSPACE(r \log w)$, $regex \in LERegex$

Dôkaz. Vyplýva z $\mathcal{L}_{LERE} \subseteq NSPACE(\log n)$, dĺžka regexu už viac nie je konštanta.

Veta

$L(regex\#word) \in DSPACE(n \log^2 n)$, kde $regex \in Eregex$.

Dôkaz. Idea Savitchovej vety, ale s konfiguráciami z formalizmu. Hĺbku rekúzie algoritmu vieme zhora ohraničiť pomocou lemy 1 ($3 \cdot |regex| \cdot |word|^2$).

Veta

$L(regex\#word) \in DSPACE(n \log^2 n)$, kde $regex \in Eregex$.

Dôsledok

Nech \mathcal{U} je trieda regexov, pre ktoré platí, že počet konfigurácií v akceptačnom výpočte pre regex $regex$ a slovo $word$ je najviac $f(n)$, kde $n = r + w + 1$. Potom pre $regex \in \mathcal{U}$ platí $L(regex\#word) \in DSPACE(n \cdot \log n \cdot \log(f(n)))$.

Dôsledok

Trieda regexov s hĺbkou vnorenia lookaroundov zhora ohraničenou konštantou h patrí do $DSPACE(n \log^2 n)$.

Ďakujem za pozornosť!