KONTEKSTNO-NEODVISNE GRAMATIKE ZA KODIRANJE IN STISKANJE PODATKOV

JANEZ PODLOGAR

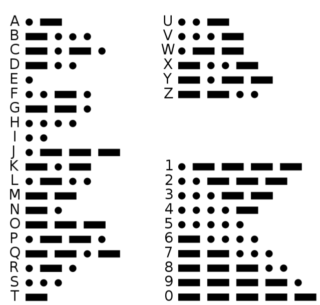
Povzetek. V delu predstavimo definicije in ilustritivne primere, ki motivirajo kodiranje in stiskanje podatkov s kontekstno-neodvisnimi gramatikami.

1. Kodiranje podatkov

Zapis informacije v neki obliki ni primeren za vsakršno rabo. Besedilo, zapisano z pismenkami, je neberljivo za slepe osebe, saj je komunikacijski kanal v tem primeru vid. Prav tako pisanega besedila v pravotni obliki ni moč poslati z telegrafom. V tem primeru je komunikacijski kanal žica in pismenke se po njej ne morejo sprehoditi. V obeh primerih je informacija, ki bi jo radi prenesli v neprimerni obliki. V prvem primeru je potrebno besedilo zapisati z Braillovo pisavo. V drugem primeru pa je potrebno besedil pretvoriti v električni signal. Spreminjanje zapisa sporočila pravimo *kodiranje*, sistemu pravil, po katerem se kodiranje opravi, pa *kod*.

Primer 1.1. *Morsejeva abeceda* je kodiranje črk, števil in ločil s pomočjo zaporedja kratkih in dolgih signalov:

- Dolžina kratkega signala je ena enota.
- Dolgi signal je trikrat daljši od kratkega signala.
- Razmik med signali znotraj črke je tišina dolžine kratkega signala.
- Razmik med črkami je tišina dolga tri kratke signale oz. en dolgi signal.
- Presledek med besedami je tišina dolga sedmih kratkih signalov.



Slika 1. Mednarodna Morsejeva abeceda

Prvotni namen Morsejeve abecede je komunikacija preko telegrama, saj komunikacijski kanal dovoljuje le električne signale in tišino med njimi. Kodiranje črk je takšno, da imajo črke z višjo frekvenco (v angleškem jeziku) krajši zapis, s tem se dolžina kodiranega sporočila skrajša in posledično tudi čas prenosa. ♢

Date: 7. november 2022.

Definicija 1.2. *Abeceda* je končna neprazna množica Σ. Elementom abecede pravimo *črke*. *Množica vseh končnih nizov abecede* Σ je

$$\Sigma^* = \{a_1 a_2 a_3 \cdots a_n \mid n \in \mathbb{N}_0 \land \forall i : a_i \in \Sigma\},$$

kjer za $n = 0$ dobimo prazen niz, ki ga označimo z ε. *Dolžino niza* w označimo z $|w|$ in je enaka številu črk v nizu $w \in \Sigma^*$. *Jezik na abecedi* Σ je poljubna podmnožica množice Σ^*.

Opomba 1.3. *Kleenejeva zvezdica* ali *Kleenejevo zaprtje* je operacija, ki abecedi Σ priredi najmanjšo nadmnožico Σ^*, ki vsebuje *prazen niz* ε in je zaprta za konkatenacijo oziroma veriženje. Z drugimi besedami, Σ^* je množica vseh končnih nizov, ki jih lahko generiramo z veriženjem črk abecede Σ.
Za abecedo Σ definirajmo

$$\Sigma^0 = \{\varepsilon\}$$
$$\Sigma^1 = \Sigma$$

ter za vsak $i > 0$ rekurzivno

$$\Sigma^{i+1} = \{wa \mid w \in \Sigma^i \text{ in } a \in \Sigma\},$$

potem je Kleenejeva zvezdico na Σ enaka

$$\Sigma^* = \bigcup_{i \geq 0} \Sigma^i$$

Primer 1.4. Naj bo $\Sigma = \{a, b, c\}$ abeceda, potem je

$$ab \in \Sigma^*$$
$$ccc \in \Sigma^*$$
$$cababcccababcccab \in \Sigma^*$$

$\diamondsuit$

Definicija 1.5. *Kodiranje nizov abecede* Σ je injektivna funkcija $\kappa\colon \Sigma^* \to \Sigma_c^*$, kjer je Σ_c *kodirna abeceda* in $\kappa(w)$ imenujemo *koda niza* w. *Dokodiranje kodiranja* κ je funkcija $\kappa^{-1}\colon C \subseteq \Sigma_c^* \to \Sigma^*$, da velja

$$\forall w \in \Sigma^* \colon \kappa^{-1}(\kappa(w)) = w$$

Opomba 1.6. Zožitev kodomene kodirne funkcije κ na $C \subseteq \Sigma_c^*$ je bijektivna funkcija.

Primer 1.7. Formalizirajmo Morsejevo abecedo iz Primera 1.1. Abecedi sta

$$\Sigma = \{\mathrm{A}, \mathrm{B}, \ldots, \mathrm{Z}\} \cup \{0, 1, \ldots, 9\} \cup \{\_\}$$
$$\Sigma_c = \{\cdot, -, \_\}$$

Definirajmo kodno funkcijo črk abecede $\kappa\colon \Sigma \to \Sigma_c^*$, ki vsakei črki iz abecede priredi niz črk kodirne abecede. Funkcija je definirana s tabelo iz Slike 1, dodatno presledek med besedami _ kodiramo v sedem kratkih enot tišine

$$\kappa(\_) = \_\_\_\_\_$$

Za niz $w = a_1 a_2 \ldots a_n \in \Sigma^*$ definiramo kodirno funkcijo K po črkah

$$K(w) = \kappa(a_1)\_\kappa(a_2)\_\cdots\kappa(a_n)$$

Poglejmo si dva primera kodiranja in v Morsejevi abecedi

$$K(\text{SOS}) = \cdots \;\; - - - \;\; \cdots$$
$$K(\text{AD HOC}) = \cdot - \;\; - \cdot\cdot \qquad \cdot\cdot\cdot\cdot \;\; - - - \;\; - \cdot - \cdot$$

Recimo, da smo dobili sporočilo, a se je pošiljatelj zmotil in je namesto kode $- - \cdot - \;\; \cdot \;\; - \cdot\cdot$, ki bi se dekodirala v

$$K^{-1}(- - \cdot - \;\; \cdot \;\; - \cdot\cdot) = \text{QED},$$

poslali kodo

$$- - \cdot - - \;\; \cdot \;\; - \cdot\cdot$$

Sporočila ne znamo dekodirati, saj se ne nahaja v domeni C dekodirne funkcije K^{-1}, natančneje ne obstaja dekodiranje $\kappa^{-1}(- - \cdot - -)$.

$\diamondsuit$

2. Stiskanje podatkov

Eden izmed namen kodiranja je tudi doseči čim večjo ekonomičnost zapisa. Želimo, da bi bilo naše sporočilo čim krajše. Kodiranje, ki skrajša zapis podatkov, imenujemo *stiskanje podatkov*.

Definicija 2.1. *Stiskanje* je kodiranje K za katerega velja

$$\forall w \in \Sigma^* \colon \; |K(w)| \ll |w|$$

Primer 2.2. Za abecedo vzemimo $\Sigma = \{a, b, c\}$ in poglejmo niz

$$w = cababcccababcccab$$

Opazimo, da se nam v nizu w večkrat ponovita vzorca ab in ccc. Zato uvedemo novi spremenljivki $A_1 = ab$ in $A_2 = ccc$. Sedaj lahko zapišemo w kot

$$w = cA_1A_1A_2A_1A_1A_2A_1$$

Ponovno se nam pojavi vzorec, tokrat $A_1A_1A_2$. Uvedemo novo spremeljivko $A_3 = A_1A_1A_2$ in zapišemo w kot

$$w = cA_3A_3A_1$$

Prvotni niz smo z novimi spremeljivkami skrajšali. Kot bomo videli, smo pretvorili niz w v kontekstno neodvisno gramatiko G_w s produkcijskimi pravili

$$\begin{aligned} &A_0 \to cA_3A_3A_1, \\ &A_1 \to ab, \\ &A_2 \to ccc, \\ &A_3 \to A_1A_1A_2 \end{aligned}$$

$\diamondsuit$

3. Kontekstno-neodvisne gramatike

Definicija 3.1. *Formalna gramatika* G so pravila, ki nam iz abecede Σ tvorijo jezik $L(G)$

Definicija 3.2. *Kontektsno-neodvisna gramatika* je četverica $G = (V, \Sigma, P, S)$, kjer je V končna množica *spremenljivk*, abeceda Σ množica *končnih simbolov* tako, da $\Sigma \cap V = \emptyset$, $P \subseteq V \times (V \cup \Sigma)^*$ relacija, ki ji pravimo *produkcijsko pravio* in $S \in V$ *začetna spremenljivka.*

Definicija 3.3. Naj bo $G = (V, \Sigma, P, S)$ kontekstno-neodvisna gramatika. Naj bodo α, β, $\gamma \in (V \cup \Sigma)^*$ nizi spremenljivk in končnih simbolov, $A \in V$ spremenljivka ter naj bo $(A, \beta) \in P$ produkcijsko pravilo, označimo ga z $A \to \beta$. Pravimo, da se $\alpha A \gamma$ *prepiše s pravilom* A v $\alpha\beta\gamma$, pišemo $\alpha A \gamma \Rightarrow \alpha\beta\gamma$. Pravimo, da α *porodi* β, če je $\alpha = \beta$ ali če za $k \geq 0$ obstaja zaporedje $\alpha_1, \alpha_2, \dots \alpha_n \in (V \cup \Sigma)^*$ tako, da

$$\alpha \Rightarrow \alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \beta$$

in pišemo $\alpha \overset{*}{\Rightarrow} \beta$.

Posledica 3.4. *Jezik kontekstno neodvisne gramatike G je*

$$L(G) = \{w \in \Sigma^* \mid S \overset{*}{\Rightarrow} w\}$$

Opomba 3.5. Ime kontekstno-neodvisna gramatika izvira iz oblike produkcijskih pravil. Na levi strani produkcijskega pravila mora vedno stati samo spremenljika. Torej vsebuje samo pravila oblike

$$A \to \alpha,$$

kjer je $A \in V$ in $\alpha \in (V \cup \Sigma)^*$. Ne sme pa vsebovati pravila oblike

$$\alpha A \gamma \to \alpha\beta\gamma,$$

kjer je $A \in V$ in so $\alpha, \beta, \gamma \in (V \cup \Sigma)^*$, saj je pravilo odvisno od predhodnega konteksta. Odvisno je od tega ali pred njim stoji α in za njim β.

Primer 3.6. Formalizirajmo gramatiko iz Primera 2.2, ki smo jo generirali z nizom $w = cababcccababcccab$. Označimo jo z $G_w = (V, \Sigma, P, S)$, kjer je

$$\begin{aligned} V &= \{A_0, A_1, A_2, A_3\}, \\ \Sigma &= \{a, b, c\}, \\ P &= \{A_0 \to cA_3A_3A_1, A_1 \to ab, A_2 \to ccc, A_3 \to A_1A_1A_2\}, \\ S &= A_0 \end{aligned}$$

Vidimo, da G_w ustreza naši definiciji kontekstno-neodvisne gramatike in res kodira w, saj je

$$L(G_w) = \{w\}$$

$\diamondsuit$

$$x \to G_x \to B(G_x)$$