KONTEKSTNO-NEODVISNE GRAMATIKE ZA KODIRANJE IN STISKANJE PODATKOV

JANEZ PODLOGAR

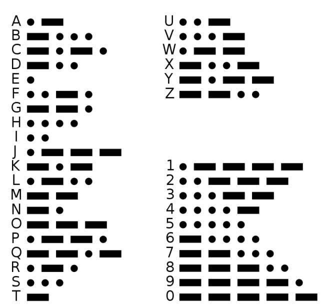
Povzetek. V delu podamo definicje kodiranja, dekodiranja in stiskanja podtakov ter predstavimo primere, ki motivirajo stiskanje podatkov s kontekstno-neodvisnimi gramatikami.

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 prvotni obliki ni mogoče poslati s 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, zapisana 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 imenujemo *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 oziroma 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. Tako se koda sporočila skrajša in posledično tudi čas njegovega prenosa.

$\diamondsuit$

Date: 15. november 2022.

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

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

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 operacijo stikanje, ki ji pravimo tudi konkatenacijo oziroma veriženje. Element ε je nevtralni element za stikanje. Z drugimi besedami, Σ^* je množica vseh končnih nizov, ki jih lahko generiramo z veriženjem črk abecede Σ.
Za abecedo Σ definirajmo

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

ter za vsak $\ell > 0$ rekurzivno

$$\Sigma^{\ell+1} = \{wa \mid w \in \Sigma^\ell \text{ in } a \in \Sigma\}.$$

Potem je Kleenejeva zvezdico na Σ enaka

$$\Sigma^* = \bigcup_{\ell \geq 0} \Sigma^\ell.$$

Omenimo še, da je Σ^ℓ množica vseh nizov abecede Σ dolžine ℓ.

Primer 1.4. Naj bo $\Sigma = \{a, b, c\}$ abeceda, potem so ab, ccc in $cababcccababcccab$ končni nizi abecede Σ in potemtakem elementi Σ^*.

$\diamondsuit$

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

$$\forall w \in \Sigma^* : \delta(\kappa(w)) = w.$$

Opomba 1.6. Funkcijo κ imenujemo *kodna funkcija*, funkcijo δ pa *dekodna funckija*.

Opomba 1.7. Zožitev kodomene kodne funkcije κ na $C \subseteq \Sigma_c^*$ je bijektivna funkcija.

Primer 1.8. Formalizirajmo Morsejevo abecedo iz Primera 1.1. Abecedi sta

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

kjer je $\textvisiblespace$ presledek in $\square$ ena kratka enota tišine. Definirajmo kodno funkcijo črk abecede $\kappa_s\colon \Sigma \to \Sigma_c^*$, ki vsaki črki iz abecede Σ_s priredi niz črk kodne abecede Σ_c. Predpis funkcije κ_s je določen s tabelo iz Slike 1, dodatno presledek $\textvisiblespace$ kodiramo v štiri kratkih enot tišine

$$\kappa_s(\textvisiblespace) = \square\square\square\square.$$

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

$$\kappa(w) = \kappa_s(a_1)\square\square\square\kappa_s(a_2)\square\square\square \cdots \kappa_s(a_n).$$

Poglejmo si dva primera kodiranja v Morsejevi abecedi

$$\kappa(\mathrm{SOS}) = \cdot\square\cdot\square\cdot\square\square\square-\square-\square-\square\square\square\cdot\square\cdot\square\cdot,$$

$$\kappa(\mathrm{AD\textvisiblespace HOC}) = \cdot\square-\square\square\square-\square\cdot\square\cdot\square\square\square\square\square\square\square\cdot\square\cdot\square\cdot\square\cdot\square\square\square-\square-\square-\square\square\square-\square\cdot\square-\square\cdot.$$

Recimo, da smo prejeli sporočilo, a se je pošiljatelj zmotil in je namesto kode

$$-\square - \square \cdot \square - \square\square\square \cdot \square\square\square - \square \cdot \square\cdot,$$

ki bi se dekodirala v

$$\delta(-\square - \square \cdot \square - \square\square\square \cdot \square\square\square - \square \cdot \square\cdot) = \text{QED},$$

poslali kodo

$$-\square - \square \cdot \square - \square - \square\square\square \cdot \square\square\square - \square \cdot \square \cdot .$$

Sporočila ne znamo dekodirati, saj se ne nahaja v domeni C dekodne funkcije δ.

$\diamondsuit$

2. Stiskanje podatkov

Eden izmed namenov kodiranja je tudi doseči čim večjo ekonomičnost zapisa. Želimo, da bi bil zapis sporočila čim krajšo. Kodiranje z namenom krajšanja kode imenujemo *stiskanje*.

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

$$\exists n \in \mathbb{N}\ \forall w \in \Sigma^* \colon |w| \geq n \implies |\kappa(w)| \ll |w|\,.$$

Opomba 2.2. Ločimo *kodiranje brez izgube*, kjer velja

$$\forall w \in \Sigma^* \colon \delta(\kappa(w)) = w$$

in *kodiranje z izgubo*, kjer kodiranje ni levo obrnljiv proces in v grobem velja

$$\forall w \in \Sigma^* \colon \delta(\kappa(w)) \approx w.$$

Definicija 2.3. Definirajmo slučajno spremenljivko $X \colon \Sigma^* \to \mathbb{R}$ z predpisom $X = \frac{|w|}{|\kappa(w)|}$. *Stopnja stiskanja* je enaka $\mathbb{E}[X]$.

Primer 2.4. 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 oznaki $A = ab$ in $B = ccc$. Sedaj lahko zapišemo w kot

$$w = cAABAABA.$$

Ponovno se nam pojavi vzorec, tokrat AAB. Uvedemo novo oznako $C = AAB$ in zapišemo w kot

$$w = cCCA.$$

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

$$\begin{aligned} S &\to cCCA, \\ A &\to ab, \\ B &\to ccc, \\ C &\to AAB. \end{aligned}$$

$\diamondsuit$

3. Kontekstno-neodvisne gramatike

Definicija 3.1. f

Kontektsno-neodvisna gramatika je četverica $G = (V, \Sigma, P, S)$, kjer je V končna množica *nekončnih simbolov*; abeceda Σ množica *končnih simbolov* tako, da $\Sigma \cap V = \emptyset$; $P \subseteq V \times (V \cup \Sigma)^*$ celovita relacija, elementu relacije pravimo *produkcijsko pravilo*; in $S \in V$ *začetni simbol*.

Opomba 3.2. Relacija $P \subseteq A \times B$ je celovita, če velja

$$\forall x \in A\ \exists y \in B \colon (x, y) \in P$$

Definicija 3.3. Naj bo $G = (V, \Sigma, P, S)$ kontekstno-neodvisna gramatika. Naj bodo α, β, $\gamma \in (V \cup \Sigma)^*$ nizi nekončnih in končnih simbolov, $A \in V$ nekončni simbol 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 *izpelje* β, če je $\alpha = \beta$ ali če za $k \geq 0$ obstaja zaporedje $\alpha_1, \alpha_2, \ldots \alpha_n \in (V \cup \Sigma)^*$ tako, da

$$\alpha \Rightarrow \alpha_1 \Rightarrow \alpha_2 \Rightarrow \ldots \Rightarrow \alpha_n \Rightarrow \beta,$$

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 spremenljivka. 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 možnost uporabe pravila odvisno od konteksta nekončnega simbola A. Kontekst določa niza α in β, ki se nahajata neposredno pred in po nekončnim simbolom A.

Primer 3.6. Formalizirajmo gramatiko iz Primera 2.4, ki smo jo generirali z nizom $w = cababcccababcccab$. Označimo jo z $G_w = (V, \Sigma, P, S)$, kjer je

$$V = \{S, A, B, C\},$$
$$\Sigma = \{a, b, c\},$$
$$P = \{S \to cCCA, A \to ab, B \to ccc, C \to AAB\},$$
$$S = S.$$

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

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

Dolžina gramatike G_w je enaka številu črk kodne abecede $\Sigma_c = \{S, A, B, C, \to\}$, ki smo jih porabili za opis gramatike in je enaka $|P| = 20$. Vidimo, da niza w nismo skrajšali, saj je $|w| = 17$. Niz w je bil prekratek, da bi ga lahko zares stisnili. Naj bo sedaj

$$w = cababcccababcccabababccc.$$

Gramatika, ki jo generira novi niz se od prejšnje gramatike razlikuje le v

$$P = \{S \to cCCAC, A \to mathitab, B \to ccc, C \to mathitAAB\}.$$

Sedaj smo stisnili w z G_w, saj je

$$|w| = 24 > 21 = |P|.$$

$\diamondsuit$

Pri stiskanju z kontekstno-neodvisnimi gramatikami poiščemo gramatiko G_w, ki generira enojec $\{w\}$ za svoj jezik. Med njimi poiščemo “najmanjšo”in jo kodiramo. Ker gramatike G_w generira w in je “majhna” jo bomo kodirali v “kratko” kodo. Tako bomo preko gramatike “dobro” stisnili niz w.