

Základy matematiky a statistiky pro humanitní obory I

Pavel Rychlý Vojtěch Kovář

Fakulta informatiky, Masarykova univerzita
Botanická 68a, 60200 Brno, Czech Republic
{pary, xkovar3}@fi.muni.cz

část 7

Obsah přednášky

Formální lingvistika

Formální gramatika

Konečný automat

Formální lingvistika

- ▶ **Matematické modely jazyka**
 - ▶ jazyk = množina slov nad nějakou abecedou
 - ▶ prvky abecedy mohou být znaky, slova, ...
 - ▶ původně navrženy k popisu přirozených jazyků
 - ▶ dnes rozlišujeme tzv. **formální jazyky**
- ▶ **Cíl přednášky**
 - ▶ seznámit se se základními konstrukcemi teorie formálních jazyků
 - ▶ → schopnost používat je v dalších kurzech

Formální lingvistika – základní pojmy

- ▶ **abeceda**
 - ▶ množina symbolů Σ (např. $\{a, b\}$)
- ▶ **slovo**
 - ▶ libovolná konečná posloupnost prvků Σ
 - ▶ např. *aabab*
- ▶ **délka slova $|v|$**
 - ▶ počet prvků této posloupnosti
 - ▶ např. $|aabab| = 5$
- ▶ **prázdné slovo ϵ**
 - ▶ slovo nulové délky

Formální lingvistika – základní pojmy (II)

- ▶ **množina Σ^***
 - ▶ množina všech slov nad abecedou Σ
 - ▶ např. $\{a, b\}^* = \{\epsilon, a, b, aa, bb, ab, ba, aab, abb, \dots\}$
- ▶ **operace zřetězení slov „ \cdot ”**
 - ▶ pro slova u, v : $u \cdot v = uv$
 - ▶ např. $aab \cdot ab = aabab$
- ▶ **mocnina slova u^i**
 - ▶ definována induktivně: $u^0 = \epsilon$; $u^{i+1} = u \cdot u^i$
 - ▶ např. $(ab)^3 = ababab$
- ▶ **Jazyk**
 - ▶ množina (některých) slov nad danou abecedou
 - ▶ pro každý jazyk L platí $L \subseteq \Sigma^*$

Formální lingvistika – základní pojmy (III)

- ▶ **zřetězení jazyků**
 - ▶ $L_1 \cdot L_2 = \{u \cdot v \mid u \in L_1 \wedge v \in L_2\}$
- ▶ podobně i další operace nad jazyky

Formální gramatika

- ▶ **Čtveřice (N, Σ, P, S)**
 - ▶ N – množina neterminálů
 - ▶ Σ – množina terminálů (symbolů abecedy)
 - ▶ $\rightarrow N \cap \Sigma = \emptyset$
 - ▶ $\rightarrow N \cup \Sigma$ označíme V (množina symbolů)
 - ▶ $P \subseteq (V^* \cdot N \cdot V^*) \times (V^*)$ – množina pravidel
 - ▶ S – počáteční symbol gramatiky
- ▶ **Pravidla gramatiky**
 - ▶ (α, β) zapisujeme jako $\alpha \rightarrow \beta$
 - ▶ α, β jsou slova nad V (řetězce terminálů a neterminálů)
 - ▶ kde α obsahuje alespoň jeden neterminál

Odvození z gramatiky

- ▶ **Gramatika je model, který generuje jazyk**
 - ▶ začneme počátečním neterminálem
 - ▶ používáme pravidla gramatiky jako přepisovací systém
 - ▶ \rightarrow tj. levou stranu pravidla nahradíme pravou
 - ▶ přepisujeme tak dlouho, dokud nedostaneme řetězec terminálů
- ▶ **Vztah jazyka a gramatiky**
 - ▶ **gramatika G generuje jazyk L** , pokud existuje odvození každého slova jazyka L z gramatiky G
 - ▶ značíme $L(G)$

Odvození z gramatiky – příklad

- ▶ Gramatika
 - ▶ $\Sigma = \{a, b\}, N = \{S, A\}$
 - ▶ $P = \{ S \rightarrow A, A \rightarrow AA, A \rightarrow a \}$
- ▶ Příklady odvození
 - ▶ $S \Rightarrow A \Rightarrow a$
 - ▶ $S \Rightarrow A \Rightarrow AA \Rightarrow aA \Rightarrow aAA \Rightarrow aaA \Rightarrow aaa$
 - ▶ kolik slov obsahuje jazyk generovaný touto gramatikou?

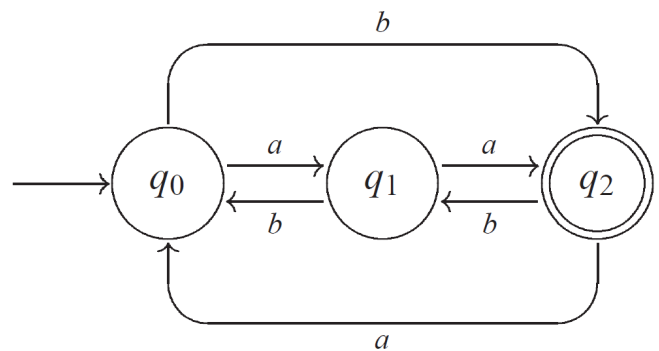
Chomského hierarchie gramatik

- ▶ Typy gramatik podle omezení na pravidla
- ▶ typ 0
 - ▶ žádná omezení
- ▶ typ 1
 - ▶ pro každé pravidlo $\alpha \rightarrow \beta$ je $|\alpha| \leq |\beta|$
 - ▶ též **kontextová gramatika**
- ▶ typ 2
 - ▶ každé pravidlo je tvaru $A \rightarrow \beta$ ($A \in N$)
 - ▶ též **bezkontextová gramatika**
- ▶ typ 3
 - ▶ každé pravidlo je tvaru $A \rightarrow aB$ nebo $A \rightarrow a$ ($A, B \in N; a \in \Sigma$)
 - ▶ též **regulární gramatika**

Konečný automat

- ▶ Jiný model charakterizující jazyky
- ▶ Pětice $(Q, \Sigma, \delta, q_0, F)$
 - ▶ Q – neprázdná konečná množina stavů
 - ▶ Σ – konečná množina vstupních symbolů (abeceda)
 - ▶ $\delta : Q \times \Sigma \rightarrow Q$ – přechodová funkce
 - ▶ q_0 – počáteční stav
 - ▶ F – množina koncových stavů
- ▶ Automat necháváme běžet nad vstupním slovem
 - ▶ začneme v počátečním stavu
 - ▶ podle dalšího symbolu na vstupu a aktuálního stavu se přesuneme do jiného stavu
 - ▶ opakujeme, dokud není slovo dočteno do konce

Konečný automat



Automaty a jazyky

- ▶ Automaty a jazyky
 - ▶ automat akceptuje slovo, pokud po jeho zpracování skončí v akceptujícím stavu
 - ▶ automat akceptuje jazyk, pokud akceptuje právě slova jazyka
- ▶ Automaty a gramatiky
 - ▶ pro každou regulární gramatiku G existuje automat, který akceptuje jazyk $L(G)$ (důkaz existuje :))
 - ▶ platí i naopak → **ekvivalentní formalismy**
- ▶ Co se nevešlo
 - ▶ existují i další typy automatů
 - ▶ některé ekvivalentní s jinými typy gramatik
 - ▶ např. zásobníkový automat, Turingův stroj