Neuromaty
- akceptory regulárnych jazykov

Neurónové siete

September 30, 2024

Výpočtová sila neurónových sietí

- Aj keď sú neurónové siete výpočtovými modelmi motivovanými predstavami o fungovaní mozgu, ich výpočtovú silu a efektívnosť možno študovať a porovnávať s inými modelmi v rámci informatiky.
- Znamená to porovnať výpočtovú silu neurónových sietí s tradičnými výpočtovými modelmi, ako sú konečno – stavové automaty a Turingove stroje.

Motivácia

Neurónové siete (NS) – neurónové akceptory jazykov (nad bin. abecedou).
Sú analyzované dva typy vstupných protokolov:

1. Neurónová sieť, v ktorej je len jeden neurón pre vstupné slovo (vstupný neurón postupne číta bity vstupného slova).
   Tu sa pracuje s diskrétnými alebo analógovými modelmi NS.
2. Existuje postupnosť neurónových sietí – pre každú dĺžku vstupného slova existuje jedna neurónová sieť s odpovedajúcim počtom vstupných neurónov.

V oboch prípadoch, **stav výstupného neurónu** vyjadruje, či vstupné slovo patrí do jazyka alebo nie.
Budeme pracovať s prvým typom vstupu.

Deterministické konečno – stavové automaty (DKA) a regulárne jazyky

- Abeceda je konečná množina symbolov.
- Reťazec nad abecedou Σ je konečná postupnosť symbolov z abecedy Σ.
- Dĺžka reťazca s, označená $|s|$, je celkový počet symbolov v s.
- Prázdny reťazec, označený ϵ, je reťazec bez symbolov, $|\epsilon| = 0$,.
- Ak s a t sú dva reťazce, tak ich konkatenácia (zreťazenie) je reťazec st.

DKA and regulárne jazyky

Definition

- Jazyk v abecede Σ je množina reťazcov nad abecedou Σ.
- Nech sú dané dva jazyky L_1 a L_2. Jazyk $L_1.L_2$, nazvaný konkatenácia jazykov L_1 a L_2, je $\{st|s \in L_1, t \in L_2\}$.
- Nech je daný jazyk L.
  Definujeme $L^0 = \{\epsilon\}$ a $L^n = L.L^{n-1}$ pre $n \geq 1$.
  Iterácia L, označená $L^\star$, je jazyk $L^\star = \bigcup_{n=0}^{\infty} L^n$.
- Podobne pozitívna iterácia $L^+ = \bigcup_{n=1}^{\infty} L^n$. $\odot$

Definition

Deterministický konečno – stavový automat je 5-tica $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$:

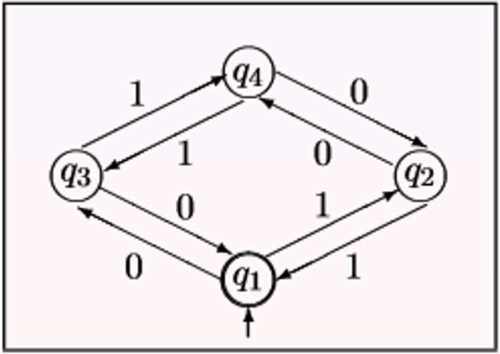
- $Q \neq \emptyset$, konečná množina stavov automatu,
- Σ je vstupná abeceda (v našom prípade $\Sigma = \{0, 1\}$),
- $\delta : Q \times \Sigma \to Q$ je prechodová funkcia,
- $q_0 \in Q$ je počiatočný stav automatu, a
- $F \subseteq Q$ je množina akceptujúcich stavov.

Zovšeobecnená prechodová funkcia DKA $\delta^* : Q \times \Sigma^* \to Q$ je definovaná takto:

1. $\delta^*(q, \epsilon) = q$ pre $q \in Q$,
2. $\delta^*(q, wa) = \delta(\delta^*(q, w), a)$ pre $q \in Q, w \in \Sigma^*, a \in \Sigma$. $\odot$

Príklad

Je daný DKA $A1 = (Q, \Sigma, \delta, q_1, q_1)$. Aký jazyk akceptuje?



Je možné použiť neurónovú sieť namiesto DKA (akceptora jazyka)?

Deterministický konečno – stavový automat (DKA) – akceptuje regulárne jazyky.
Ukážeme, že

- K DKA vieme skonštruovať neurónovú sieť, ktorá bude realizovať činnosť automatu.
- Nepoužijeme pritom algoritmus na trénovanie váh, ale váhy a prahy neurónov vhodne nastavíme.
- Veľkosť neurónovej siete je lineárne závislá vzhľadom na počet stavov DKA.

Konštrukcia neurónovej siete k DKA

Nech je daný DKA $A = (Q, \Sigma, \delta, q0, F), \Sigma = \{0, 1\}$
Architektúra siete:

- Každý stav $q \in Q$ je reprezentovaný 2 neurónmi, označme ich $(q, 0)$ a $(q, 1)$
- V neurónovej sieti vytvoríme prepojenia tak, že v čase t neurón (q, i) bude aktivovaný vtedy a len vtedy, keď originálny automat A v čase t je v stave q a dostáva vstup i.
- K sieti týchto neurónov je potrebné pridať jeden výstupný neurón.

Teda celkový počet neurónov je $2 * m + 1$, ak $m = |Q|$

Nastavenie váh

- Pre ľubovoľné dva stavy q_j a q_k, platí

$$w[(q_j, i), (q_k, 0)] = w[(q_j, i), (q_k, 1)] \in \{0, 1\}$$

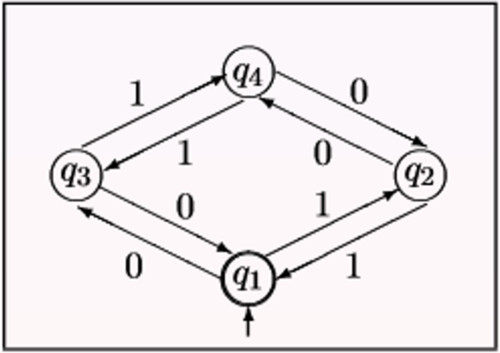
- $w[(q_j, i), (q_k, 0)] = w[(q_j, i), (q_k, 1)] = 1 \Leftrightarrow$ keď v DKA A je prechod zo stavu q_j do stavu q_k cez vstup i.
- Pre vstupné váhy platí, $w0(q, 0) = -1$ a $w0(q, 1) = +1$ pre všetky q.
- Pre výstupné váhy, $w[(q, i), 2m + 1] = \tau(q, i)$ (τ je výstupná funkcia).
- Všetky tu nezmienené váhy sú rovné 0.
- Prahy sú nastavené: $h(q, 1) = 2, h(q, 0) = 1$ a prah výstupného neur'onu $h(2m + 1) = 1$.

Je zrejmé, že

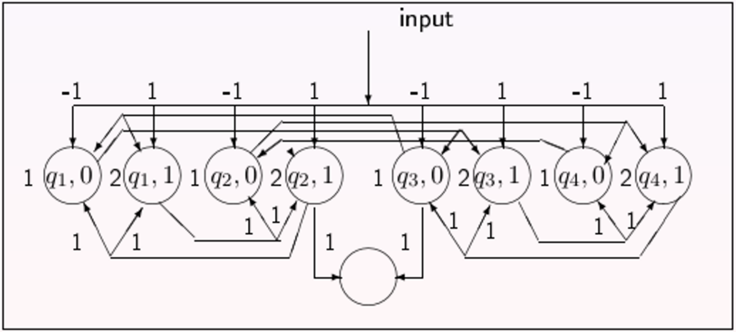
- v čase t = 0 presne jeden z neurónov $(q, i), q \in Q, i = 0, 1,$ je aktívny (nastavenie je potrebné riadiť podľa prvého písmena spracovávaného slova),
- v ľubovoľnom čase $t \geq 1$ v závislosti od vstupu práve jeden z neurónov bude aktívny a dynamika siete bude presne taká istá ako u DKA A,
- akceptovanie prázdneho slova sieť rieši o jeden krok neskôr.

Príklad

$A1 = (Q, \Sigma, \delta, q_1, q_1)$ je DKA, ktorý akceptuje všetky slová a len tie slová s párnym počtom núl a párnym počtom jednotiek.



Neurónová sieť simulujúca prácu automatu A1



Aktivačná funkcia neurónov je $F(x) = 1$, ak $x \geq 0$, inak $F(x) = 0$.

Table: Výpočet pre slovo 10101

Vst	$(q_1, 0)$	$(q_1, 1)$	$(q_2, 0)$	$(q_2, 1)$	$(q_3, 0)$	$(q_3, 1)$	$(q_4, 0)$	$(q_4, 1)$	Výst
	0	1	0	0	0	0	0	0	0

Nastavenie stavu neurónu $(q_1, 0)$ alebo $(q_1, 1)$ podľa prvého symbolu slova.

Table: Výpočet pre slovo 10101

Vst	$(q_1, 0)$	$(q_1, 1)$	$(q_2, 0)$	$(q_2, 1)$	$(q_3, 0)$	$(q_3, 1)$	$(q_4, 0)$	$(q_4, 1)$	Výst
0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
1	0	0	0	0	1	0	0	0	1
0	0	1	0	0	0	0	0	0	0

Výstup: akceptovanie slova je oneskorené o jeden krok, lebo sieť má 2 vrstvy.
Akceptuje slovo 1010

DKA realizovaný pomocou neurónovej siete

- Alon [1] - každý m-stavový deterministický konečný automat môže byť realizovaný pomocou diskrétnej neurónovej siete s $O(m^{\frac{3}{4}})$ neurónmi a aspoň $\Omega((m \log m)^{\frac{1}{3}})$ neurónov je nutných pre túto konštrukciu.
- Tieto hranice boli vylepšené: $O(m^{\frac{1}{2}})$ neurónov postačuje a neurónová sieť vyžaduje najviac $\Omega(m^{\frac{1}{2}})$ neurónov, keď hodnoty váh sú polynomiálne vzhľadom na veľkosť siete.
- Pripomíname výsledok z Hopcrofta a Ullmana:
Jazyk L je regulárny $\Leftrightarrow$ ak je rozpoznaný nejakým konečno – stavovým automatom.

Definition (Regulárne výrazy:)

Množina $\mathcal{RE}$ regulárnych výrazov nad abecedou $\Sigma = \{0, 1\}$ je minimálny jazyk nad abecedou $\{0, 1, \emptyset, \epsilon, +, ., \star, (,)\}$, ktorý spĺňa nasledujúce podmienky:

1. $\emptyset, \epsilon, 0, 1 \in \mathcal{RE}$
2. if $\alpha, \beta \in \mathcal{RE}$ then $(\alpha + \beta), (\alpha.\beta), \alpha^\star \in \mathcal{RE}$.

Množina $\mathcal{RL} = \{[\alpha] | \alpha \in \mathcal{RE}\}$ je množina regulárnych jazykov $[\alpha]$, ktoré sú vyjadrené regulárnym výrazom α nasledovne:

- $[\emptyset] = \emptyset$, $[\epsilon] = \{\epsilon\}$, $[0] = \{0\}$, $[1] = \{1\}$,
- ak $\alpha, \beta \in \mathcal{RE}$, tak $[\alpha + \beta] = [\alpha] \cup [\beta], [\alpha.\beta] = [\alpha].[\beta]$, $[\alpha^\star] = [\alpha]^\star$

Regulárny výraz α^+ zodpovedajúci $[\alpha]^+$ je definovaný výrazom $[\alpha^\star] = [\epsilon + \alpha^+]$. $\odot$

Neuromaty - neurónové akceptory jazykov

Sú známe nasledujúce tri výsledky [5]:

1. Ľubovoľný jazyk $L = L(\mathcal{N})$ rozpoznávaný neuromatom $\mathcal{N}$ je regulárny.
2. Pre každý regulárny jazyk L vyjadrený regulárnym výrazom $[\alpha]$ existuje neuromat $\mathcal{N}$ veľkosti $O(|\alpha|)$, ktorý rozpoznáva jazyk L.
3. Existujú regulárne jazyky $L_n = [\alpha_n], n \geq 1$ také, že ľubovoľný neuromat $\mathcal{N}$, ktorý rozpoznáva jazyk $L_n = L(\mathcal{N})$ vyžaduje aspoň $\Omega(|\alpha_n|)$ neurónov.

Pre prvý typ vstupu definujeme neuromat - akceptor regulárnych jazykov.

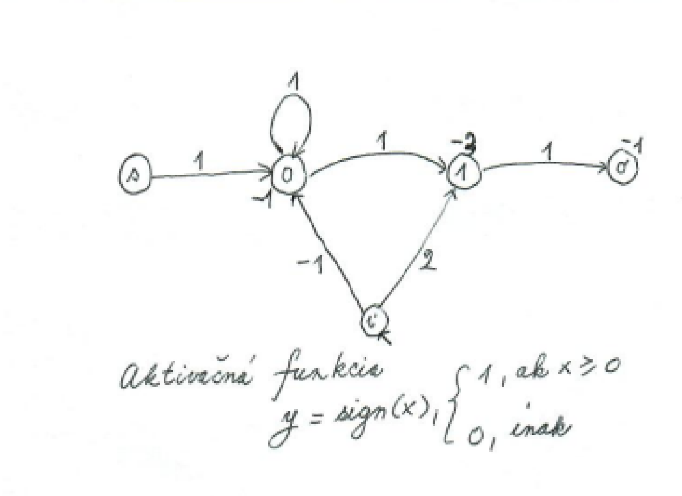
Definition (Neuromat)

Neurónový akceptor *(skrátene, neuromat)* je 7-ica
$\mathcal{N} = (V, inp, out, E, w, h, s_{init})$, kde

- V je množina neurónov obsahujúca aj vstupný $inp \in V$ a výstupný neurón $out \in V$,
- $E \subseteq V \times (V - inp)$ množina orientovaných prepojení (hrán),
- $w : E \longrightarrow Z$ (Z je množina celých čísel) je váhová funkcia (skrátene $w(\langle i, j \rangle) = w_{ij}$),
- $h : V - \{inp\} \longrightarrow \{0, 1\}$ je prahová funkcia (skrátene $h(i) = h_i$), a
- $s_{init} : V - \{inp\} \longrightarrow \{0, 1\}$ je počiatočné nastavenie siete.

- Graf (V, E) sa nazýva **architektúra neuromatu** $\mathcal{N}$ a $n = \mid V \mid$ je veľkosť neuromatu.
- Počet bitov potrebných na reprezentáciu celého neuromatu (váh a prahov,...) sa nazýva **popisná zložitosť neuromatu**. $\odot$

Príklad



Example

- $V = \{s, i, o, 0, 1\}$
- $inp = i$
- $out = o$
- $E = (s, 0), (0, 0), (0, 1), (1, o), (i, 0), (i, 1)$
- $w(s, 0) = 1, w(0, 0) = 1, w(0, 1) = 1, w(1, o) = 1, w(i, 0) = -1, w(i, 1) = 2$
- $(h(0) = -1, h(1) = -3, h(o) = -1$
- $sinit = (1, 0, 0, 0)$

Pripomíname výsledok, ktorý chceme potvrdiť

- Pre každý regulárny jazyk L vyjadrený regulárnym výrazom $[\alpha]$ existuje neuromat $\mathcal{N}$ veľkosti $O(|\alpha|)$, ktorý rozpoznáva jazyk L.

Pre jazyk $L = [\alpha]$, skonštruujeme neuromat

$$\mathcal{N}_\alpha = (V, inp, out, E, w, h, s_{init})$$

veľkosti $O(|\alpha|)$ tak, že $L = L(\mathcal{N}_\alpha)$.

Konštrukcia neuromatu

Šíma [5].

- Najprv vytvoríme architektúru (V, E) neuromatu $\mathcal{N}_\alpha$ rekurzívne vzhľadom na štruktúru regulárneho výrazu α.
- Na tento účel definujeme postupnosť grafov

$$(V_k, E_k), k = 0, \ldots, p,$$

kde (V_0, E_0) má jediný vrchol odpovedajúci celému výrazu α, ktorý je rekurzívne rozdelený na kratšie regulárne podvýrazy tak, že (V_p, E_p) má vrcholy typu 0 or 1 podľa výrazu α.

1. $V_0 = \{s, \alpha, o\}, E_0 = \{\langle s, \alpha\rangle, \langle \alpha, o\rangle\}$
2. Predpokladajme, že $V_k, E_k, 0 \leq k \ < p$ už bol skonštruovaný a $\beta \in V_k$ je podvýraz v α rôzny od 0 a 1.
   Teda okrem prázdneho jazyka a prázdneho reťazca regulárny výraz β môže predstavovať zjednotenie, konkatenáciu alebo iteráciu podvýrazov v β.
3. Skonštruujeme nový graf (V_{k+1}, E_{k+1}).
   Odstránime vrchol β a pridáme nové vrcholy a dostaneme nový graf.
   Jeden z nových vrcholov je možné označiť znovu ako β.

Substitúcie:

- β je $\emptyset$: $V_{k+1} = V_k - \{\beta\}$, $E_{k+1} = E_k - \{\langle x, \beta\rangle, \langle \beta, y\rangle \in E_k\}$.
- β je ϵ: $V_{k+1} = V_k - \{\beta\}$, $E_{k+1} = (E_k - \{\langle x, \beta\rangle, \langle \beta, y\rangle \in E_k\}) \cup \{\langle x, y\rangle | \langle x, \beta\rangle, \langle \beta, y\rangle \in E_k - \{\langle \beta, \beta\rangle\}\}$.
- β má tvar $\beta + \gamma$: $V_{k+1} = V_k \cup \{\gamma\}$, $E_{k+1} = E_k \cup \{\langle x, \gamma\rangle, \langle \gamma, y\rangle | \langle x, \beta\rangle, \langle \beta, y\rangle \in E_k\} \cup \{\langle \gamma, \gamma\rangle | \langle \beta, \beta\rangle \in E_k\}$.
- β má tvar $\beta.\gamma$: $V_{k+1} = V_k \cup \{\gamma\}$, $E_{k+1} = (E_k - \{\langle \beta, y\rangle \in E_k\}) \cup \{\langle \beta, \gamma\rangle\} \cup \{\langle \gamma, y\rangle | \langle \beta, y\rangle \in E_k\}$.
- β má tvar β^+: $V_{k+1} = V_k, E_{k+1} = E_k \cup \{\langle \beta, \beta\rangle\}$.

Konštrukcia je ukončená po $p = O(|\alpha|)$ krokoch, keď V_p obsahuje len podvýrazy 0 alebo 1. Definujeme architektúru neuromatu nasledovne:

$$V = V_p \cup \{inp\}, E = E_p \cup \{\langle inp, \beta\rangle | \beta \in V_p - \{s, o\}\}. \tag{1}$$

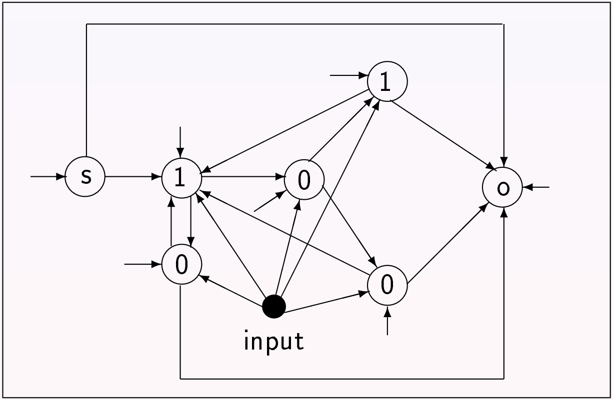
Pre $i \in V$ označíme $d(i) = |\{j \in V_p | \langle j, i\rangle \in E\}|$, počet hrán do i. Definujeme váhovú w a prahovú funkciu h:

- $i \in V$ je neurón typu 1: $w_{ji} = 1$ pre $\langle j, i\rangle \in E_p$ a $w_{inp,i} = d(i), h_i = d(i) + 1$.
- $i \in V$ je neurón typu 0: $w_{ji} = 1$ pre $\langle j, i\rangle \in E_p$ and $w_{inp,i} = -d(i), h_i = 1$.

- $s \in V$: $h_s = 1$.
- $o \in V$: $w_{o,j} = 1$ for $\langle j, o \rangle \in E_p, h_o = 1$.
- Počiatočný stav je definovaný
  $s_0(i) = 0$ pre $i \in V_p - \{s\}$ a $s_0(s) = 1$.

Množina V obsahuje 3 špeciálne neuróny inp, s, o a ďalšie neuróny typu 1 alebo 0 – jeden pre každý výskyt 1 alebo 0 v α; teda $|V| = O(|\alpha|)$. $\odot$

Neuromat pre jazyk $[(1(0+0(1+0)))^*]$

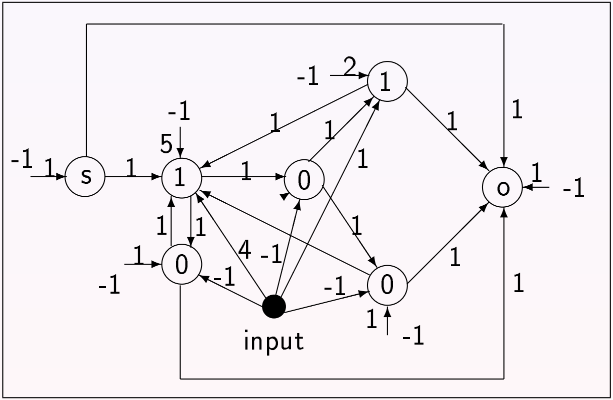


Výpočet váh a prahov

Pre $i \in V$ vypočítame $d(i) = |\{j \in V_p | [j, i] \in E\}|$.

Neurón	typ	d	prah	váhy zo vstupu
A	1	4	5	4
B	1	1	2	1
C	0	1	1	-1
D	0	1	1	-1
E	0	1	1	-1

Neuromat pre RE $[(1(0+0(1+0)))^{\star}]$ s váhami a prahmi



Neuromat pre RE $[(1(0+0(1+0)))^{\star}]$ – výpočet

```
Vypocet d
Vypis hodnot d
  0  4  0  4  1  1  1  1
Vypis prahov
  1  1  0  5  2  1  1  1
Vypis vah:

  0  0  0   0  0  0  0  0
  1  0  0   1  0  0  1  1
  0  0  0   0  0  0  0  0

  1  0  4   0  1  0  1  1
  0  0  1   0  0  1  0  0
  0  0 -1   1  0  0  0  0
  0  0 -1   0  0  1  0  0
  0  0 -1   1  0  0  0  0

Praca neuromatu pri rozpoznavani slov
=====================================

Stavy neuronov
 Krok    -2   -1    0    1    2    3    4    5  Vystup  | Sprac. slovo
 -----------------------------------------------------------------
   0      1    0    0    0    0    0    0    0  Nastavenie
1
   0      0    1    1    1    0    0    0    0       1  |
0
   1      0    1    0    0    0    1    0    1       1  | 1
1
   2      0    1    1    1    1    0    0    0       1  | 10
1
   3      0    1    1    1    0    0    0    0       1  | 101
```

Tretí výsledok ukazuje dolné hranice počtu neurónov, ktoré sú v najhoršom prípade potrebné na rozpoznávanie regulárnych jazykov, ktoré sú vyjadrené regulárnymi výrazmi dĺžky n.
Z toho vyplýva, že vyššie uvedená konštrukcia neuromatu je optimálna.
Dolné hranice by mala preukázať množina jazykov:

$$L_n = \Big(\sum_{i=0}^{n-3}(1(\epsilon+0))^i 10 + (1(\epsilon+0))^{n-1}(1+0)\Big)^* \tag{2}$$

$$\Pi_k = \left[(1(\epsilon+0))^k\right], \quad P_n = \cup_{k=0}^{n-1}\Pi_k. \tag{3}$$

Je zrejme, že $P_n, n \geq 1$ je množina prefixov jazyka L_n.

- Regulárny výraz, ktorý definuje jazyk P_n má v skutočnosti dĺžku $O(n^2)$.
  Je možné skonštruovať regulárny jazyk α_n lineárnej dĺžky vyjadrujúci ten istý jazyk L_n.
  Jazyky Π_n a P_n sú použité pri konštrukcii.
- Z toho vyplýva výsledok našich úvah a predpokladov, že každý neuromat $\mathcal{N}$ rozpoznávajúci jazyk L_n vyžaduje aspoň $\Omega(n)$ binárnych neurónov.

Diskusia

- Siegelmann a Sontag [4] dokázali, že sa dajú simulovať všetky Turingove stroje rekurentnými neurónovými sieťami prvého rádu, to znamená siete konečnej veľkosti, ktoré majú prepojenia synchrónne pracujúcich neurónov.
- Každý neurón aktualizuje svoj stav aplikáciou "sigmoidálnej" funkcie na lineárne kombinácie predchádzajúcich stavov všetkých neurónov. Teda je možné simulovať akýkoľvek multi-zásobníkový Turingov stroj v reálnom čase a existuje sieť s 886 neurónmi, ktorá počíta univerzálne parciálne rekurzívne funkcie.

- Alon, N., Dewdney, A. K., Teunis, J. O.: Efficient Simulation of Finite Automata by Neural Nets, Journal of ACM, Vol. 38, No. 2, April 1991, pp.495-514.
- Hassoun, M. H.: Fundamentals of artificial neural networks. MIT Press, Cambridge, 1995, pp. 511.
- Orponen, P.: Computational complexity of neural networks: A survey, NeuroCOLT Tech. Report Series, NC-TR-94-010, Royal Holloway University of London, 1994, pp. 20.
- Siegelmann, H. T., Sontag, E. D.: On the computational power of neural nets, Journal of computer and system sciences, Vol. 50, No. 1, 1995, p. 132-150.
- Šíma, J., Neruda, R.: Theoretical Questions on Neural Networks, MatfyzPress, Prague, 1996.

Bonusové úlohy, každá 1.5 bodu

Je daný regulárny výraz:

$$[(0(0+1)^{+}0+10(1+0))^{*}]$$

1. Vytvorte k nemu DKA a potom NS;
2. Vytvorte neuromat.

V oboch prípadoch je potrebné ukázať výpočet v tabuľke.