

Klasifikácia čiernobielych obrázkov pomocou neurónovej siete ART

Martin Veselovský
xvesel60@stud.fit.vutbr.cz

December 10, 2014

1 Úvod

Klasifikácia čiernobielych obrázkov je prípadom klasifikácie binárnych vektorov, pričom túto úlohu je možné efektívne riešiť neurónovou sieťou ART. Cieľom tohto procesu je zoznam zhlukov, pričom v rámci jedného zhľuku sa nachádzajú obrázky, ktoré obsahujú podobné obrazce. Dokument pojednáva o teoretickom pozadí neurónovej siete, aplikáčnom návrhu riešenia konkrétného problému spolu s jeho implementáciou. Posledná kapitola obsahuje stručný užívaľský manuál aplikácie.

2 Neurónová sieť ART

Adaptívna rezonančná teória (ART) bola vyvinutá k rozpoznávaniu obrazcov pre modelovanie mohutnej paralelnej architektúry. Jedná sa o asociatívnu sieť bez učiteľa, pričom hlavnou vlastnosťou siete ART je jej schopnosť naučiť sa novú informáciu bez poškodenia skôr uloženej informácie, tzn. sieť ART dokáže riešiť problém premennej stability (stability-plasticity dilemma).

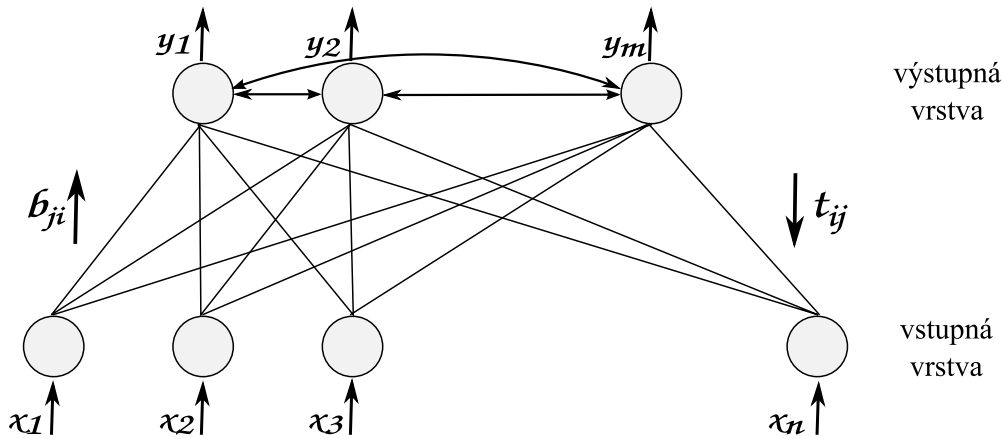


Figure 1: Neurónová sieť ART

Sieť ART je dvojvrstvová neurónová sieť. Tieto vrstvy sa označujú ako vstupná (porovnávací) a výstupná (rozpoznávací), pričom každý neurón vstupnej vrstvy je vzájomne prepojený s každým neurónom výstupnej vrstvy váhami b_{ji} a t_{ij} , $i = 1, 2, \dots, n$, $j = 1, \dots, m$, kde n je počet vstupných neurónov, ktorý je určený počtom elementov vstupného vzoru (dĺžkou vstupného vektora) a m je počet výstupných neurónov, ktorý zodpovedá počtu tried, pričom počet tried je variabilný. Na začiatku výpočtu môže sieť obsahovať iba jeden výstupný neurón a v priebehu výpočtu sa ich počet môže zvyšovať, tj. adaptívna vlastnosť siete [3]. Váhy b_{ij} sa označujú ako dopredné váhy (bottom-up) a sú to čísla s pohyblivou desatinnou čiarkou. Váhy t_{ji} sa označujú ako spätné váhy (top-down) a sú to čísla binárne (0/1).

Prvou verziou siete ART bola tzv. ART1, ktorá pracuje s binárnymi hodnotami vzorov. Konkrétne práve algoritmus ART1 je v projekte použitý. Okrem iných existujú napr. siete ART2 pre prácu s reálnymi hodnotami alebo ARTMAP pre učenie s učiteľom.

Učenie siete ART prebieha trvale¹ a má niekoľko fáz. Hlavným znakom siete je opakovaná výmena vstupného vektora medzi vstupnou a výstupnou vrstvou cez príslušné váhy. Táto výmena sa dá označiť ako rezonancia, pretože vzory medzi oboma vrstvami vymieňame tak dlho, pokiaľ nie je nájdený stabilný stav. Jednotlivé fáze siete sú [2, str. 229-234]:

Inicializačná Dopredné váhy b_{ji} sú nastavené na konštantnú hodnotu podľa vzťahu

$$b_{ji} = \frac{1}{1+n}$$

kde n je počet počet vstupných neurónov. Pre váhy t_{ij} platí $t_{ij} = 1$.

Rozpoznávacia Hodnoty na vstupoch sú transformované cez dopredné váhy k výstupným neurónom podľa vzťahu

$$y_j = \sum_{i=1}^n b_{ji} x_i$$

kde y_j je výstup j -tého neurónu. Následne je vybraný víťazný neurón j^* , tj. neurón s najvyššou výstupnou hodnotou y_j . Výber víťazného neurónu je na obr. 1 zobrazený ako prepojenie výstupných neurónov tzv. *postrannou inhibíciou*, tzn. že ostatné neuróny budú potlačené a podporený bude len ten najlepší. Vzor, ktorý je zakódovaný v spätných váhach t_{ij^*} tohto neurónu je odoslaný do vstupnej vrstvy.

Porovnávacia Na vstupnej vrstve sú prítomné dva vektory, vstupný vektor a vektor váh t_{ij^*} . Podobnosť týchto vektorov je vypočítaná podľa vzťahu

$$\frac{\sum_{i=1}^n t_{ij^*} x_i}{\sum_{i=1}^n x_i}$$

Výsledok následne porovnáme voči parametru ρ , ktorý sa nazýva parametrom ostrážitosti. Ak je podobnosť vektorov väčšia ako ρ tak nastáva adaptačná fáza.

Vyhľadávacia Ak vektor víťazného neurónu j^* neprešiel testom ostrážitosti, tak je potrebné nájsť iný víťazný neurón, ktorý bude lepšie reprezentovať vstupný vektor. Nastáva opäť rozpoznávacia fáza. Pokiaľ testu ostrážitosti nevyhoví ani jeden výstupný neurón, tak sa v sieti vytvorí nový výstupný výstupný neurón r s váhami

$$b_{ri} = \frac{x_i}{0.5 + \sum_{i=1}^n x_i}, \quad t_{ir} = x_i$$

Adaptačná Ak vektor víťazného neurónu j^* prešiel testom ostrážitosti, tak sú dopredné aj spätné váhy neurónu aktualizované podľa nasledovných vzťahov

$$b_{j^*i}^{nová} = \frac{t_{ij^*} x_i}{0.5 + \sum_{i=1}^n t_{ij^*} x_i}, \quad t_{ij^*}^{nová} = t_{ij^*} x_i$$

¹Trvalé učenie siete môže byť potlačené explicitným rozdelením pracovania siete na dve časti: 1. klasifikácia s učením (adaptácia váh), 2. klasifikácia bez učenia (váhy sa nemenia)

3 Návrh

Neurónová sieť dostane na vstupe množinu čiernobielych obrázkov a má rozhodnúť, ktoré obrázky sú si podobné. Takéto obrázky sú klasifikované do rovnakej triedy, pričom reprezentácia triedy je takisto binárnym vektorom (prototyp triedy), a teda je možné zobraziť aj grafickú reprezentáciu triedy, tzn. z vektoru späť vytvoríme obrázok.

Dôležitým parametrom neurónovej siete ART je parameter ostrážitosti, nakoľko zmena hodnoty tohto parametru výrazne ovplyvňuje chovanie siete. Pri veľkých hodnotách je vytvorených len málo nových výstupných neurónov (tried), naopak pri malých hodnotách je možné dospieť k stavu, kedy je pre každý vstup vytvorený nový výstupný neurón.

3.1 Vstup

Uvažujeme obrázky vo formáte PNG (grafický model RGBA), ktoré sú striktné čier-nobiele. Aby sme mohli porovnávať podobnosť, tak dĺžka vstupného vektoru musí byť rovnaká, inými slovami, môžeme porovnávať len obrázky rovnako veľké.

3.2 Výstup

Aplikácia umožňuje zobraziť grafickú reprezentáciu výstupného neurónu a uložiť ju do súboru. Hlavným cieľom je ale získať rozdelenie obrázkov podľa podobnosti. K tomuto účelu slúži grafické užívateľské rozhranie ale aj možnosť uložiť výstup do súboru, a to v jednoduchom formáte

j-neuron: x1.png x2.png ...
k-neuron: x3.png x4.png ...

3.3 Testovacia sada

Pre účely demonštrácie a testovania siete ART som vyrobil dve testovacie sady čiernobielych obrázkov. Prvá obsahuje 8 obrázkov veľkosti 16x16 pixelov obsahujúce jednoduché útvary (kolekcia simple). Druhá obsahuje 17 obrázkov veľkosti 128x128 pixelov a obsahuje jednoduché znázornenia dopravných prostriedkov (kolekcia vehicles). Súbor v druhej kolekcii sú pomenované podľa toho, aký prostriedok je na nich znázornený.

4 Implementácia

Aplikácia je implementovaná v programovacom jazyku Java a ponúka grafické užívateľské rozhranie pre pohľadné odvládanie a vizualizáciu chovania siete ART. Základom je voľba adresára obsahujúceho čiernobiele obrázky vo formáte PNG.

Po inicializačnom procese je možné algoritmus krokovať po jednom obrázku, alebo spustiť pre celú kolekciu, pričom priebežné výsledky sú vypisované v reálnom čase. Program sa skladá z triedy ART1, kde je implementovaný algoritmus neurónovej siete, triedy GUI kde je implementované užívateľské rozhranie a nakoniec triedy Main, ktorá je vstupným bodom aplikácie.

Praktickou možnosťou aplikácie je nahliadnutie na obrázok priradený k nejakej triede. Toto je možné uskutočniť presunutím kurzoru nad názov dotyčného súboru, kde sa následne zobrazí tooltip obsahujúci daný obrázok.

4.1 Vektorizácia obrázkov

V modeli RGBA sú pre každý pixel vyhradené 4 byty. Obrázok je prevedený na vektor takým spôsobom, že program spracuje len každý štvrtý byte. Tento byte vyjadríme ako

datový typ `integer` a pokiaľ je toto číslo rovné nule, tak sa jedná o čiernu farbu. Ak je číslo rovné -1, tak sa jedná o farbu bielu². Žiadne iné hodnoty v bytovej reprezentácii obrázku nie sú akceptované³.

4.2 Neurónová sieť

Neuróny sú implementované ako zoznamy ich váh [1]

b - zoznam n -rozmerných polí pre datový typ `double` - dopredné váhy siete

t - zoznam n -rozmerných polí pre datový typ `integer` - spätné váhy siete

*Napr. pre prístup k dopredným váham neurónu **k** pristúpime k položke v zozname **b** na indexe **k** (`b.get(k)`).*

V rámci rozpoznávacej fáze sú indexy všetkých výstupných neurónov uložené do množiny **A**. Výsledok porovnania vektoru víťazného neurónu a vstupného vektoru je podrobený testu ostrážitosti, tzn. je porovnaný voči premennej **vigilance**, ktorú je možné dynamicky meniť. Ak si vektory neboli dostatočne podobné, index momentálne spracovaného výstupného neurónu je z množiny **A** odstránený. Potom sa z množiny vyberie nový víťazný neurón. V prípade, že testu ostrážitosti nevyhoví žiadny výstupný neurón, pridáme do zoznamu **b**, resp. **t** nový prvok, tj. ďalšie n -rozmerné pole reprezentujúce dopredné, resp. spätné váhy nového neurónu.

5 Užívateľský manuál

Program nevyžaduje žiadne neštandardné knižnice. Pre preklad a spustenie programu je možné použiť príkaz **ant** s priloženým súborom **build.xml**.

Príkaz **ant** bez parametrov použitý v hlavnom adresári programu vykoná nasledovné: vyčistí výstupný adresár **out**, preloží súbory v jazyku Java (`out/production`) a následne vytvorí a spustí súbor **art.jar** (`out/jar`). Príkaz pre samostatný preklad je **ant compile**, pre samostatné spustenie **ant run**.

5.1 Použitie

Aplikáciu používame vždy v nasledovnom poradí: zvolíme vstupný adresár, inicializujeme neurónovú sieť a počítame buď po jednom obrázku alebo priamo pre všetky obrázky v adresári. V aplikácii sa nachádzajú nasledovné tlačítka a komponenty:

open - zvolíme vstupný adresár obsahujúci čiernobiele obrázky vo formáte PNG a vytvoríme v ňom výstupný adresár **out**, do ktorého budú uložené grafické reprezentácie výstupných neurónov⁴

init - inicializujeme neurónovú sieť (vytvoríme n vstupných neurónov, kde n je dĺžka vstupného vektoru)

step - spracovanie aktuálneho vstupu sieťou ART. ART vypočíta príslušnosť k danej triede, príp. vytvorí nový výstupný neurón reprezentujúci novú triedu, a uloží jej grafickú reprezentáciu ako súbor **k-neuron.png** do adresáru **out**. Zmena grafickej reprezentácie neurónu sa prekreslí.

²Program číslo -1 interne prevedie na jeho absolútnu hodnotu, tj. 1

³Pracujeme s čiernobielymi obrázkami. Ostatné obrázky aplikácia informatívne zobrazí ale sieť s nimi nepracuje

⁴Ak výstupný adresár už existuje tak jeho obsah bude vymazaný

Ak má aktuálne spracovávaný obrázok nesprávny formát alebo iné rozmery ako prvý obrázok pre ktorý bola sieť inicializovaná, tak tento preskočíme.

run - spracovanie všetkých obrázkov vo zvolenom adresári (výsledky sa zobrazujú v reálnom čase). Spracovanie je možné ovplyvniť tlačítkami stop/continue.

save - uloženie aktuálne spracovaných obrázkov do súboru **results.txt** v adresári **out**

V grafickom užívateľskom rozhraní sa nachádza aj **posuvník parametru ostráživosti**, ktorý je možné upravovať za behu programu. Zaškrtnuté pole **images** slúži pre zobrazenie obrázkov výstupných neurónov, takže ak požadujeme len textový výstup, je možné túto funkciu vypnúť. Zaškrtnuté pole **log** povoľuje výpis dodatočných informácií na terminál⁵.

5.2 Ukážka aplikácie

Testovanie aplikácie prebiehalo na pripravených kolekciách obrázkov (viď. 3.3). Na obrázku nižšie sú zobrazené hlavné rysy aplikácie.

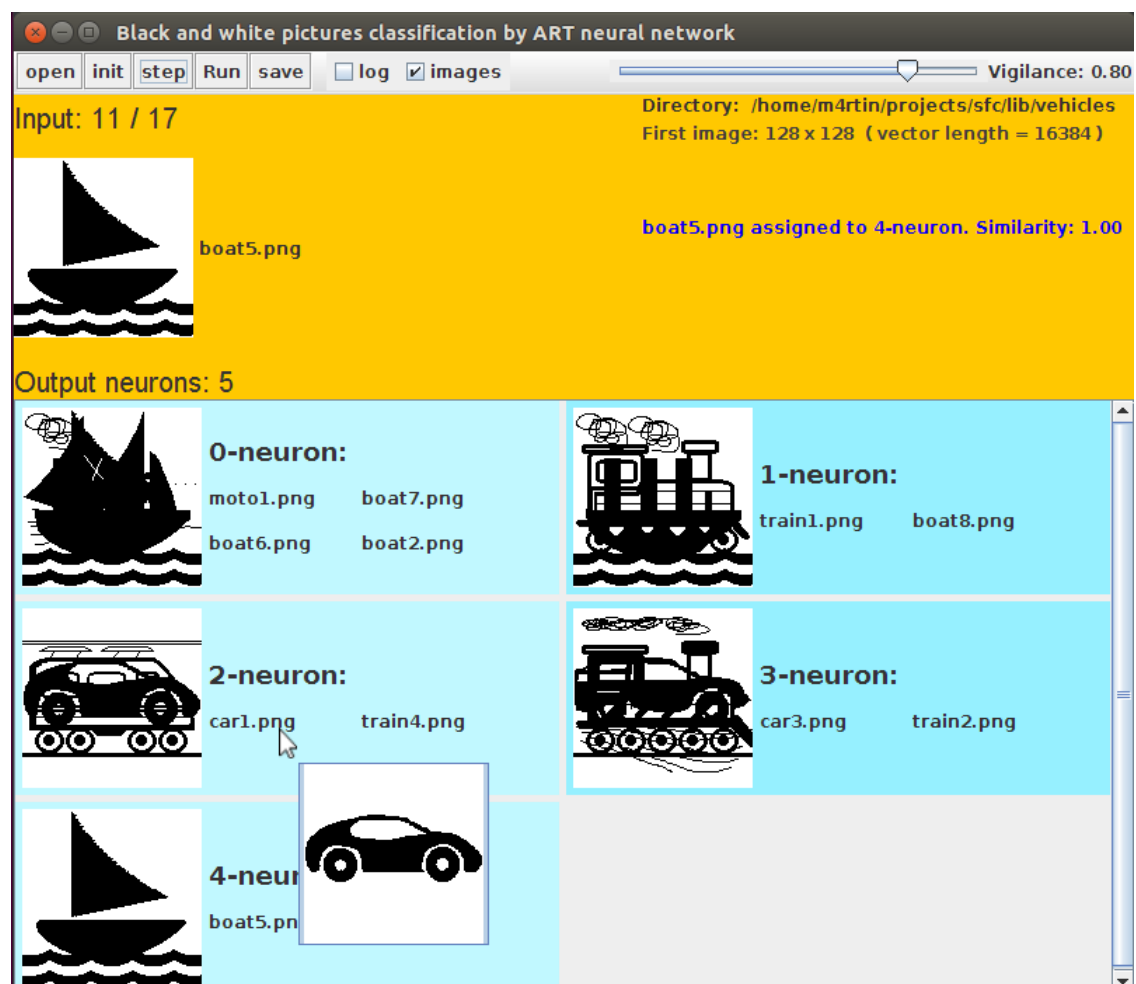


Figure 2: Ukážka spustenej aplikácie

Vo vrchnom paneli je aktuálne spracovávaný obrázok spolu s informáciami o poradí v rámci adresára, umiestnenia adresára, veľkosti obrázku, pre ktorý bola sieť inicializovaná a priebežného stavu výpočtu vzhľadom na celú kolekciu.

⁵Logovanie nie je vhodné pre príliš veľké obrázky nakoľko vypisuje všetky váhy všetkých neurónov

V spodnom panely sú zobrazené jednotlivé neuróny. Ich výstupné vektory (spätne váhy t_{ij}) sú zmesou všetkých priradených vstupných vektorov, takže ich grafická reprezentácia vyzerá tak, ako keby boli všetky obrázky naskladané na sebe. Na obrázku 2 je tiež znázornená možnosť nahliadnuť na obrázok priradeného vstupného vektoru (obrázku) a to tak, že presunieme kurzor nad názov⁶.

6 Záver

Klasifikácia neurónovou sieťou ART je efektívna a jednoduchá. Dôležitým faktom je, že sieť je citlivá na parameter ostrážitosti a nie je presný postup ako tento parameter nastaviť, takže jeho hodnotu je potrebné vybrať empiricky na základe experimentov s konkrétnymi vstupnými vektormi.

Spätne váhy výstupných neurónov tvoria ťažisko zhľuku (prototyp triedy), ktorý je možné zobraziť ako obrázok a ponúknuť tak nielen textovú ale aj vizuálnu odozvu klasifikácie pomocou neurónovej siete ART.

References

- [1] Graupe, D.: *Principles of artificial neural networks*. Singapore ; Hackensack, N.J: World Scientific, druhé vydání, 2007.
- [2] Mařík, V.; Štěpánková, O.; Lažanský, J.: *Umělá inteligence. (4)*. Praha: Academia, první vydání, 2003.
- [3] Mehrotra, K.; Mohan, C. K.; Ranka, S.: *Elements of artificial neural networks*. Cambridge, Mass: MIT Press, 1997.

⁶Pri väčších obrázkoch môže načítanie zobrazenia chvíľu trvať.