

# Deep learning

## v prostredí MATLAB

Michal BLAHO, Humusoft, s.r.o.

Deep learning je jedno z najprogressívnejších odvetví strojového učenia so schopnosťou riešiť úlohy, ktoré boli ešte pred desiatimi rokmi nemysliteľné. Uplatnenie deep learningu zasahuje veľa odborov ľudskej činnosti od systémov počítačového videnia cez vyhľadávajúce, diagnostické a asistenčné systémy až po autonómne vozidlá a bezpečnosť. MATLAB, ako vývojové prostredie pre vedecko-technické výpočty, ponúka v oblasti strojového učenia a deep learningu množstvo algoritmov a zjednodušuje ich aplikáciu pri riešení praktických úloh.

### Čo je to deep learning

Deep learning je metóda strojového učenia, kde je s pomocou mnohovrstvových výpočtových modelov získaná užitočná informácia priamo z dát. V klasifikačných úlohách dosahujú deep learning modely vysokú presnosť, ktorá môže prekonať ľudské schopnosti. Modely sú trénované s využitím veľkého množstva označených dát a ich architektúra je postavená na hĺbkových neurónových sieťach (deep neural networks). Deep learning sa stáva kľúčovou technológiou v mnohých oblastiach a úlohách, ktoré, riešia. Napríklad ide o detekciu objektov v obraze, detekciu rakovinových buniek, prípadne rozpoznávanie reči.

### Princíp deep learningu

Ako sme už spomenuli, metódy deep learningu využívajú architektúru neurónových sietí. Tradičné neurónové siete zvyčajne využívajú iba niekoľko skrytých vrstiev, u deep neural networks sú to desiatky. V súčasnosti najpopulárnejšie deep neural networks sú známe pod pojmom konvolučné neurónové siete (Convolutional Neural Networks - CNN). Výpočtový algoritmus CNN je založený na konvulúcii medzi vstupnými dátami a filtermi, ktoré reprezentujú hľadané charakteristické črty. CNN využíva dvojrozmerné (2-D) konvolučné vrstvy a je teda vhodná pre spracovanie 2-D dát, akými sú typicky obrázky.

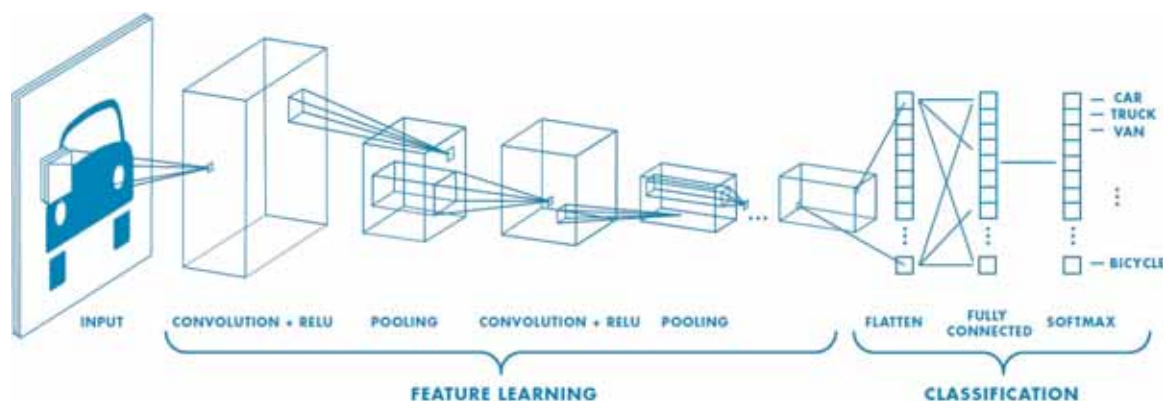
Dáta pretekajú naprieč neurónovou sieťou a dochádza k ich konvulúcii s naučenými filtermi (koeficienty filtrov sú naučené váhy neurónovej siete).

Výstup z konvulcie v jednej vrstve je vstupom do nasledujúcej vrstvy. Filtry môžu začínať ako veľmi jednoduché črty, typu svetlých a tmavých miest či hrán, a postupne prechádzajú do zložitejších tvarov v nasledujúcich vrstvách. Cieľom konvulcie je napríklad extrahovať z obrazových dát informácie, na základe ktorých možno objekty na snímkach rozpoznať. Konvolučné vrstvy sú často doplnené vrstvami ReLU a Pooling, ktoré výstupy upravujú a zmenšujú. Za sadou konvulčných vrstiev bývajú napojené klasifikačné vrstvy, ktoré pracujú s extrahovanými črtami zo vstupných obrázkov. Výstupom klasifikačných vrstiev môže byť pravdepodobnostné rozdelenie pre priradenie klasifikačnej triedy (hľadaného objektu) k vstupným obrázkom.

### Učenie deep learning modelov

Presnosť deep learning modelov závisí z veľkej časti na množstve dát použitých na ich učenie. Z tohto dôvodu nebolo možné až do posledných rokov úspešne pracovať s teóriou, ktorá je známa už z osemdesiatych rokov minulého storočia. Najpresnejšie modely vyžadujú veľké množstvá popísaných vzoriek. Napríklad autonómne riadenie vozidiel vyžaduje milióny obrázkov a tisíce hodín videa. Učenie tohto modelu môže trvať veľmi dlhú dobu. S príchodom výkonných grafických kariet (GPU) s paralelnou architektúrou sa práca s deep learningom stala efektívnejšia. Čas naučenia siete sa zredukoval z týždňov na hodiny. Naučený model je možné nasadiť následne do prevádzky v reálnom čase.

Obr. 1  
Princíp fungovania  
konvolučnej  
neurónovej siete



## Deep learning v MATLABe

MATLAB je inžiniersky nástroj a interaktívne prostredie pre vedecké a technické výpočty, analýzu dát, vizualizáciu, vývoj algoritmov, modelovanie a simulácie. MATLAB je nástroj ako pre pohodlnú interaktívnu prácu, tak i pre vývoj širokého spektra aplikácií. V oblasti deep learningu poskytuje MATLAB dva základné nástroje - autoenkodéry a konvolučné neurónové siete. V prostredí MATLAB sa s konvolučnými sieťami pracuje veľmi pohodlne. Existujú pripravené funkcie pre vytvorenie jednotlivých vrstiev siete, u ktorých môže používateľ nastavovať parametre. Následne vrstvy pospája, sieť natrénuje a využije ku klasifikácii s novými dátami. Na urýchlenie trénovanie siete možno využiť GPU. Typická CNN sa môže skladať z vrstvy pre vstup obrázkov, jednej alebo niekoľko za sebou zapojených skupín vrstiev convolution2d + relu + maxPooling2d, za ktorými nasledujú klasifikačné vrstvy fullyConnected + softmax + classification.

```
vrstvy = [imageInputLayer([28 28 1]);
convolution2dLayer(5,20);
reluLayer();
maxPooling2dLayer(2,'Stride',2);
fullyConnectedLayer(10);
softmaxLayer();
classificationLayer()];
nastavenie = trainingOptions('sgdm');
cnn = trainNetwork(trenovaciObr,...
    vrstvy,...
    nastavenie);
vysledok = classify(cnn,novyObr);
```



Obr. 2 Rozpoznávanie objektov pomocou CNN v prostredí MATLAB

## Učenie metódou transfer learning

Naučenie celej konvolučnej neurónovej siete z počiatočného nastavenia vyžaduje ohromné množstvo trénovacích dát, pretože je treba nastaviť milióny váhových koeficientov. Alternatívnym prístupom k učeniu „od nuly“ je využitie už predtrénovanej siete, ktorá už sama extrahuje charakteristické črty zo vstupných dát. K prispôbeniu siete konkrétnemu problému postačí doučiť iba niekoľko posledných vrstiev, ktoré len rozlišujú medzi konkrétnymi objektmi. Tento prístup, označovaný termínom transfer learning, je pohodlnou cestou aplikácie deep learningu bez nutnosti využitia rozsiahleho množstva dát a dlhšej doby učenia. MATLAB ponúka stiahnutie a využívanie obľúbených predtrénovaných sietí ako sú AlexNet, VGG-16, VGG-19 a GoogLeNet. Napríklad sieť AlexNet, je naučená s 1,2 miliónmi obrázkov, má 23 vrstiev, klasifikuje objekty do 1 000 kategórií. V MATLABe je tiež možné importovať predtrénované siete z Caffe.

## Konvolučné siete a klasická klasifikácia

Strojové učenie poskytuje množstvo metód a modelov, ktoré môžete zvoliť v aplikáciách vzhľadom na veľkosť dát a typ problému. Konvolučné neurónové siete nemusia byť vždy využívané, vrátane klasifikačných vrstiev. Je možné využívať iba konvolučné vrstvy pre extrakciu charakteristických črt z dát. Získané črty sú potom vstupom do klasických klasifikačných algoritmov strojového učenia, ako sú rozhodovacie stromy, support vector machines, k-nearest neighbors, discriminant analysis a iné metódy, ktoré sú v MATLABe k dispozícii.

## Vizualizácia konvolučnej siete

Dá sa jednoduchým spôsobom kontrolovať, či je sieť naučená správne? Okrem testovania s dátami, u ktorých dopredu poznáme správny výsledok, môžeme vykonať optickú kontrolu vizualizačnou technikou Deep Dream. MATLAB poskytuje funkciu deepDreamImage syntetizujúcu umelý obrázok, ktorý silne aktivuje vrstvy danej siete pre zvolenú kategóriu obrázkov. Opticky môžeme skontrolovať, či naučené vlastnosti zodpovedajú hľadanému tvaru. Na obrázku je vizualizácia siete AlexNet pre kategóriu kvetináča.



Obr. 3 Vizualizácia CNN metódou Deep Dream

## Konvolučné siete v počítačovom videní

Jednu z najčastejších úloh v oblasti počítačového videnia je detekcia špecifických objektov v snímanej scéne. Typickým príkladom môže byť rozpoznávanie dopravných značiek pri jazde automobilom. Pri detekcii objektov nejde o zaradenie celej snímky do danej kategórie, ale o rozpoznávanie objektu, ktorý pokrýva iba jej časť. V ostatných častiach sa potom nachádzajú ďalšie objekty, ktoré nás nezaujímajú, a preto nie je možné klasifikovať snímku ako celok. Ako preniesť silu a presnosť CNN do tejto úlohy?

V MATLABe sú k dispozícii algoritmy typu R-CNN (Regions with CNN), kde je konvolučná neurónová sieť aplikovaná na klasifikáciu výrezov v rámci danej snímky. R-CNN detektor spracováva iba výrezy, kde je výskyt hľadaného objektu pravdepodobný. To významnou mierou znižuje výpočtové nároky. V MATLABe sú k dispozícii detektory R-CNN, Fast R-CNN a Faster R-CNN. Výhodou posledne menovaného detektora (Faster R-CNN) je jeho rýchlosť, čo uľahčuje jeho nasadenie.

## Nasadenie konvolučnej siete

Posledným krokom pri práci s konvolučnými neurónovými sieťami by malo byť ich nasadenie na koncové (embedded) zariadenie. S príchodom MATLABu R2017b bola množina knižníc rozšírená o GPU Coder. GPU Coder generuje optimalizovaný CUDA kód z programu MATLABu, takže je rýchlejší ako bežne dostupné nástroje. Generovaný kód využíva optimalizované NVIDIA CUDA knižnice vrátane cuDNN, cuSolver a cuBLAS. Do vášho projektu môže byť integrovaný ako zdrojový kód, statická alebo dynamická knižnica. Pri nasadení GPU ako sú NVIDIA Tesla alebo NVIDIA Tegra je využívaný na prototypovanie. V spolupráci s Embedded Coderom umožňuje GPU Coder verifikovať správanie generovaného kódu pomocou software-in-the-loop (SIL) testovania.

Preniknite do tajov deep learningu v MATLABe prostredníctvom nového bezplatného online kurzu - Deep Learning Onramp (<https://matlabacademy.mathworks.com/>)

Kontakt na distribútora softvéru: HUMUSOFT, s.r.o., [www.humusoft.sk](http://www.humusoft.sk)



HUMUSOFT, s.r.o.

Cabanova 13/D, 841 02 Bratislava Slovensko

Tel.: +421 905 478 990

[info@humusoft.sk](mailto:info@humusoft.sk)

[www.humusoft.sk](http://www.humusoft.sk)