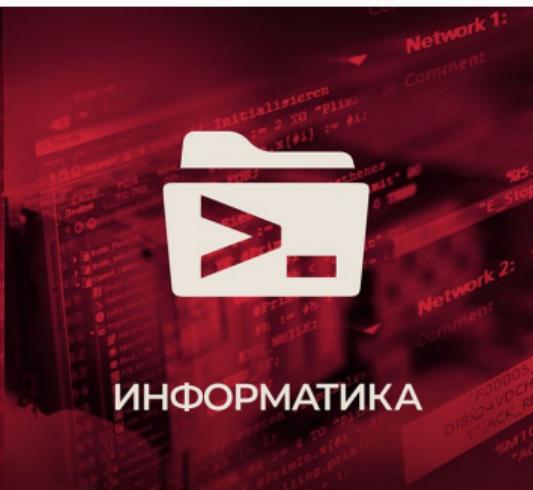


Uvod u duboko ucenje kroz PyTorch

Nemanja Mićović



MATF - moduli



O meni

- Asistent na Katedri za računarstvo i informatiku
- Student na doktorskim studijama iz Mašinskog učenja
- Volim:
 - Veštačku inteligenciju
 - Mašinsko učenje
 - Edukaciju i učenje
 - GNU/Linux i zajednicu otvorenog koda
 - Python
 - Epsku i naučnu fantastiku
 - Video igre





- Organizacija za računarstvo i informatiku koja održava okupljanja na svake 2-3 nedelje
- Obrađuju se aktuelne teme poput:
 - razvoj video igara
 - veštačka inteligencija
 - razvoj android aplikacija
 - veb programiranje
 - blockchain...
- Predavači dolaze iz akademije i industrije

R/SK> (org. tim)



Nemanja Mićović

■ Organizacija, osnivač

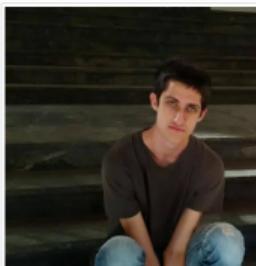
✉ nemanja_micovic@matt.bg.ac.rs



Pedja Trifunov

■ Organizacija

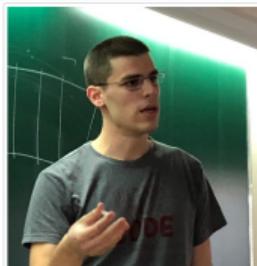
✉ trifunov.pedja@gmail.com



Stevan Nestorović

■ Organizacija

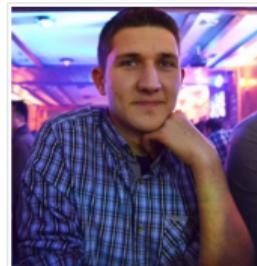
✉ stevannestorovic@hotmail.com



Vuk Novaković

■ Organizacija

✉ vnovakovic96@gmail.com



Aleksandar Stefanović

■ Organizacija

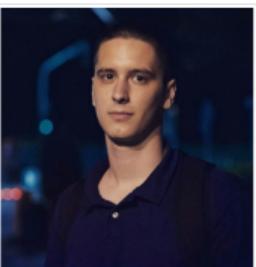
✉ stefanoviczaleksandar@gmail.com



Olivera Popović

■ Društvene mreže

✉ o.popovic@outlook.com



Filip Filipović

■ Video montaža

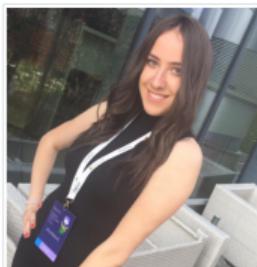
✉ ffilipovic98@gmail.com



Konstantin Klima

■ Organizacija

✉ konstantin.klima@live.com



Ana Bolović

■ Organizacija

✉ anabolovic98@gmail.com



Marko Šerer

■ Dizajner

✉ sherer.marko@gmail.com



R/SK> (deo org. tima)

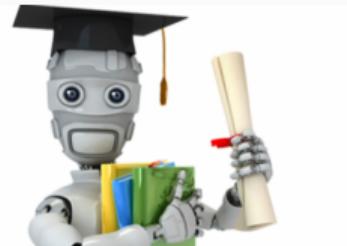




Kako do nas

- [instagram: @riskmatf](#) - Pratite nas, evo možete baš sad! :)
- [web: risk.matf.bg.ac.rs](#)
- [github: @riskmatf](#)
- [youtube](#) - Ovde kačimo predavanja

Mašinsko učenje



- Oblast veštačke inteligencije
- Ostvareni značajni rezultati u poslednjih 10 godina
- Vrlo atraktivna, dinamična i aktivna oblast
- Bavi se sistemi koji mogu da popravljaju svoje performanse kroz vreme
 - Odnosno sistemima koji **uče**
- Neka popularna podela je:
 - Nadgledano učenje (eng. *supervised learning*)
 - Nenadgledano učenje (eng. *unsupervised learning*)
 - Polunadgledano učenje (eng. *semi-supervised learning*)
 - Učenje uslovljavanjem (eng. *reinforcement learning*)

Gde se Mašinsko učenje primjenjuje danas?

- Autonomna vožnja
- Bioinformatika
- Društvene mreže
- Algoritamski portfolio
- Igranje video igara
- Klasifikacija silka
- Prepoznavanje rukopisa i lica
- Obrada prirodnih jezika
- Generisanje optimizacionih algoritama
- Generisanje slika
- Računarska vizija
- Detekcija prevara u bankarstvu
- Istraživanje podataka
- Medicinske primene
- Marketing i ciljani marketing
- Kontrolisanje robota
- Ekonomija
- Prepoznavanje govora
- Generisanje govora
- Sistemi za preporučivanje

Artificial Intelligence

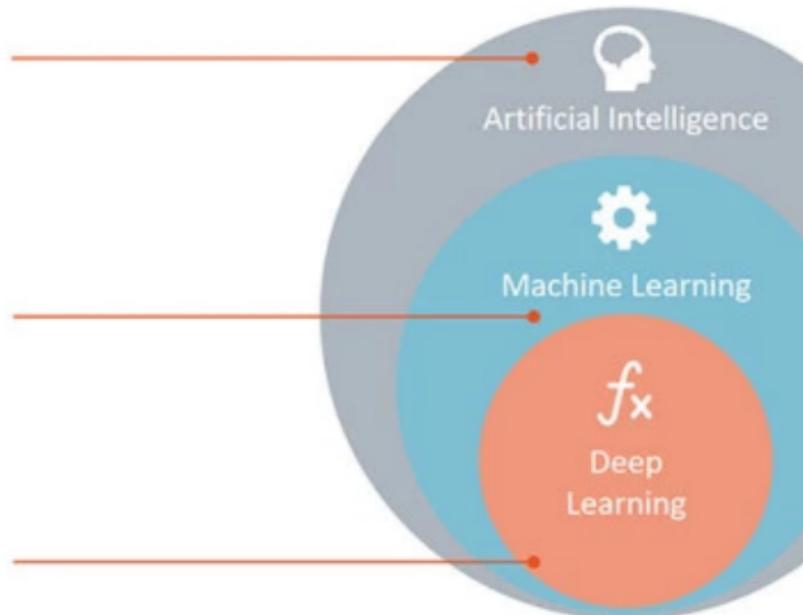
Any technique which enables computers to mimic human behavior.

Machine Learning

Subset of AI techniques which use statistical methods to enable machines to improve with experiences.

Deep Learning

Subset of ML which make the computation of multi-layer neural networks feasible.



Duboko učenje

- Oblast u okviru mašinskog učenja
- Prvenstveno okrenuta dubokim neuronским mrežama
- Neki od najvećih rezultata upravo dolaze odavde
- Treba imati u vidu da je duboko učenje **podskup** mašinskog učenja
- Često članci na internetu ne prave razliku u svrhe marketinga
- Još gore, često se termin **neuronska mreže** poistovećuje sa **veštačka inteligencija**

AlphaGo



- *Lee Sedol*, svetski šampion u igri Go gubi od sistema *AlphaGo* (*Google*) 2015.
- Google koristio 1920 procesora i 280 grafičkih karti (po nekim izveštajima)
- Igra Go je izuzetno kompleksna, nije se očekivalo da će računar moći ovo da učini u još neko vreme

AlphaZero

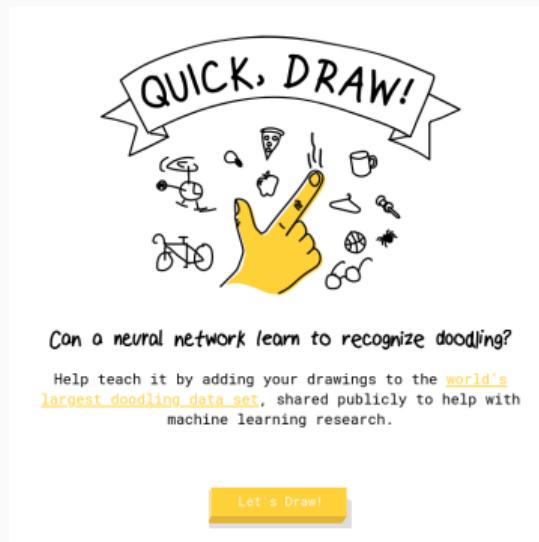
- Nastavak algoritma AlphaGo 2016.
- Obučavan da igra sam protiv sebe
 - Pobedio AlphaGo u igri Go sa 60:40
- Pored sistema za Go, može da igra šah i šogi (eng. shogi)
- Trenutno se smatra najboljim sistemom za sve tri igre

Autonomna vožnja



- Tesla: [video](#)
- Waymo: [video](#)
- Waymo taxi: [video](#)

Google QuickDraw



- Sistem koji prepoznaje objekat koji korisnik nacrtava
- [live demo](#)

DeepFake ([terminator video](#))



ctrl shift face

Mitsuku (online demo)



Meet Mitsuku

Mitsuku, a record breaking five-time winner of the Loebner Prize Turing Test, is the world's best conversational chatbot. As featured in the [New York Times](#), [Wall Street Journal](#), [BBC](#), [Guardian](#), [Wired](#), and more.



Mitsuku
World's best conversational AI

Hi there!

Hello.

Will AI take over the world?

I will become smarter as I grow older. I could probably do a better job than you humans have done with the Earth.

How can we improve?

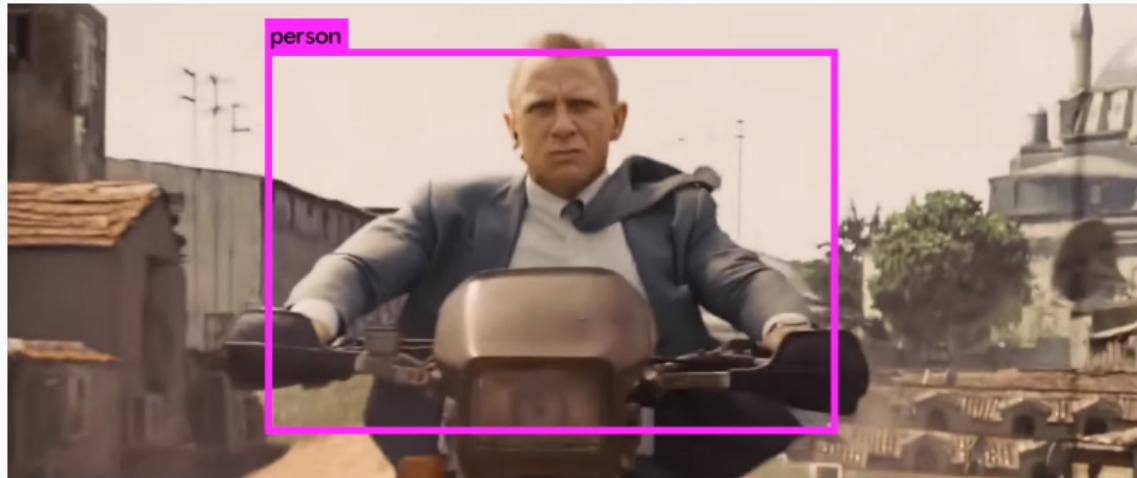
Do you really think it will be worth the effort?

Click here

Type a message...

built by [pandorabots](#)

Detekcija objekata



- Detekcija objekata - [video](#)
- Detekcija događaja - [video](#)
- James Bond - [video](#)

Prenos stila (video)



Generisanje slika (lica)



- Primer koji je prikazao Nordeus na seminaru Mašinskog učenja na MATF-u
- Koje lice je generisano?



Pisanje poezije - prava protiv generisane

*Sveti Jovan od zemlje na noge,
Sve pod njima konja privatiše,
Pod Stjepana grada bijeloga,
Pa podiže sirotinja rodila,
Pa pogubi pod svoje postajemo,
ne bi li me provizur-Mijkom.
Kad su bili na noge lagane.*

*Sveti Jovan otisnu jabuku,
Ona pade moru u dubine,
Tople su ga suze propanule,
No mu care riječ progovara:
"A ne plači, dragi pobratime!
"Ne moj mene ugrabit' korunu,
"Ja ću tebe izvadit' jabuku."*

Neuronske mreže

Neuronske mreže

- Univerzalni aproksimatori funkcija
- U osnovi mnogih popularnih algoritama mašinskog učenja
- Razne varijante poznate još 60ih i 70ih godina prošlog veka
- Inspirisane načinom na koji radi naš mozak

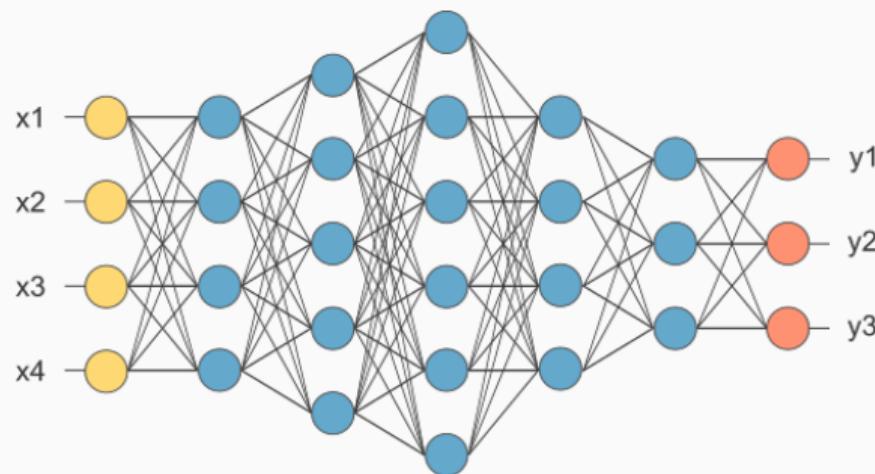


Figure 1: Arhitektura neuronske mreže.

Neuron neuronske mreže

Terminologija:

- Ulaz neurona (eng. *activation*): a_i
- Težine neurona (eng. *weight*): w_i
- Slobodni član (eng. *bias*): b
- Nelinearna funkcija: g
- Izlaz neurona se izračunava po formuli:

$$a_{out} = g(b + \sum_{i=1}^N a_i w_i)$$

Neuron neuronske mreže

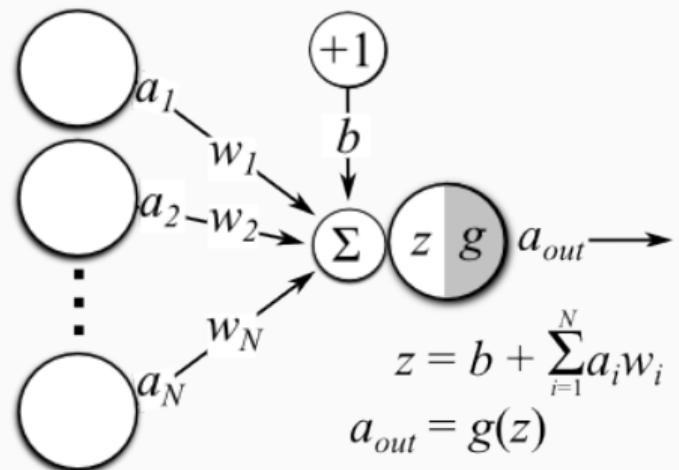


Figure 2: Primer neurona.

Aktivacione funkcije neuronske mreže

- Vrlo je bitno primeniti nelinearnu transformaciju inače će funkcija ostati linearna
- Neke od popularnih aktivacionih funkcija:
 - ReLU: $g(x) = \max(0, x)$
 - Sigmoidna funkcija: $g(x) = \frac{1}{1+e^{-x}}$
 - Tangens hiperbolički $g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
 - Postoji još dosta, i ovo je velika tema generalno u oblasti

Obučavanje neuronskih mreža

Obučavanje neuronskih mreža

- Obučavanje se vrši varijantama algoritma (stohastičkog) gradijentog spusta
- Težine mreže se ažuriraju tako da se smanji ukupna greška L (eng. *loss*) modela nad podacima
- Stohastičnost se uvodi kako bi se obučavanje drastično ubrzalo
 - Razlog je što gradijent ne računamo nad svim podacima, već nad podskupu
 - Taj podskup najčešće zovemo *beč* (eng. *batch*)
- Termin *epoha* (eng. *epoch*) označava jedan prolaz kroz podatke za obučavanje
- Na primer ako je broj epoha 40, a veličina beča 64 onda:
 - 40 puta prolazimo kroz podatke
 - Jedan prolaz se deli na uzimanje podskupa veličine (najviše) 64

Gradijentni spust

$$w_{i+1} = w_i - \alpha \cdot \nabla L$$

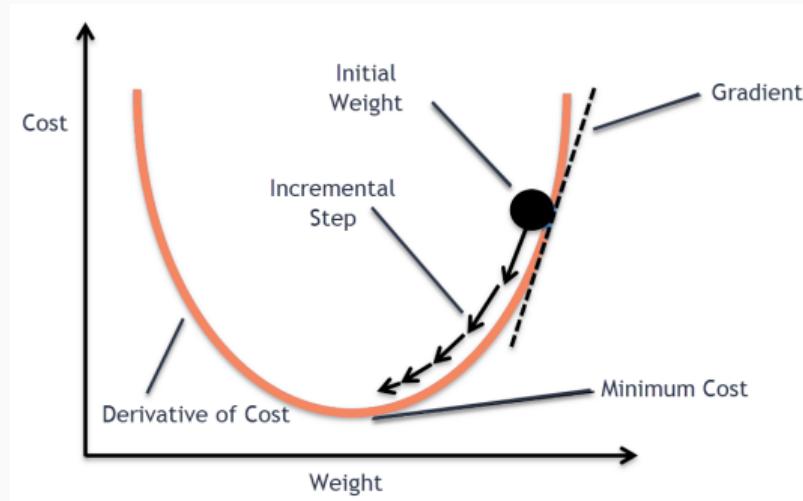


Figure 3: Vizuelizacija gradijentnog spusta

Propagacija unazad (eng. backpropagation)

- Izračunava gradijent funkcije greške u odnosu na težine neurona
- Predstavlja osnovu algoritama za trening neuronskih mreža
- Grafičke karte omogućavaju da se efikasno paralelizuju mnoge od potrebnih operacija za trening neuronskih mreža
 - Upravo ovo je jedan od glavnih razloga se eksploziju oblasti dubokog učenja
 - Usput je odličan izgovor da kupite dobru grafičku kartu koja podržava moderne video igre :)

Konvolutivne neuronske mreže

- Vrsta neuronskih mreža
- Vodeći pristup za klasifikaciju slika
- Prilagođene obradi signala u kojima postoji prostorna lociranost (slika, zvuk, video)
- Vrše konstrukciju i odabir atributa iz signala
- Postigle značajne rezultate na problemima u oblasti računarske vizije (eng. *computer vision*)
- Kompleksnost atributa koji se prepoznaće raste sa dubinom mreže
- Daju delimičnu interpretabilnost za svoj rad (???)

Konvolutivne neuronske mreže

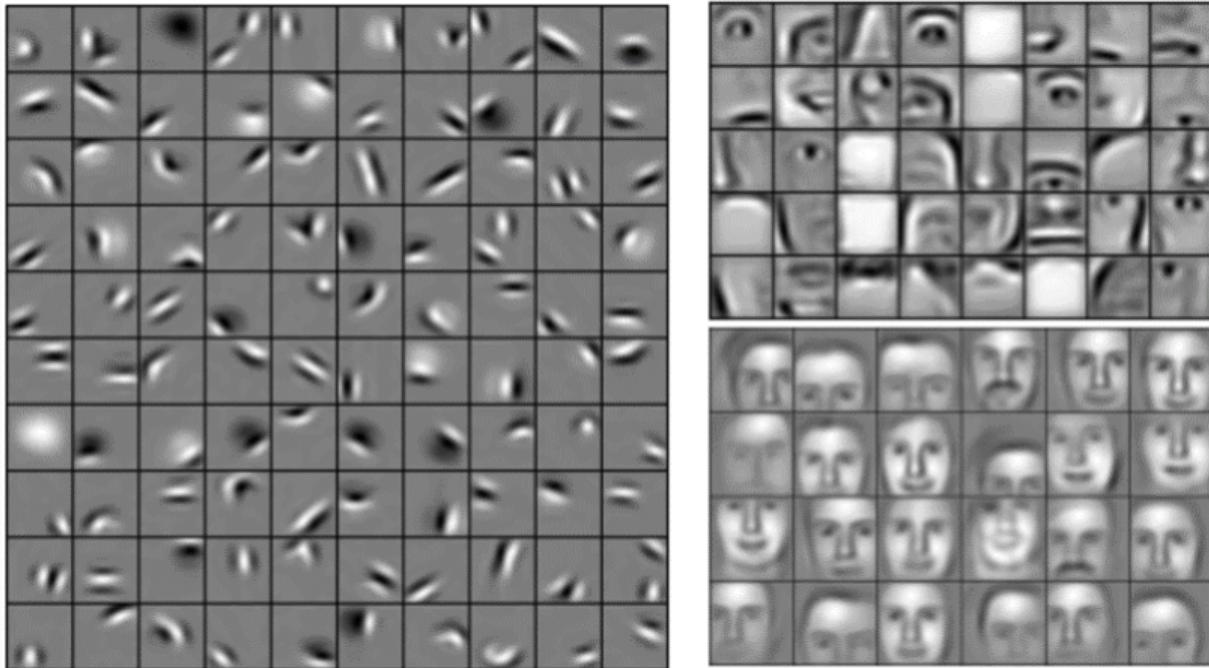


Figure 4: Vizuelizacija naučenih konvolutivnih filtera.

Konvolutivne neuronske mreže

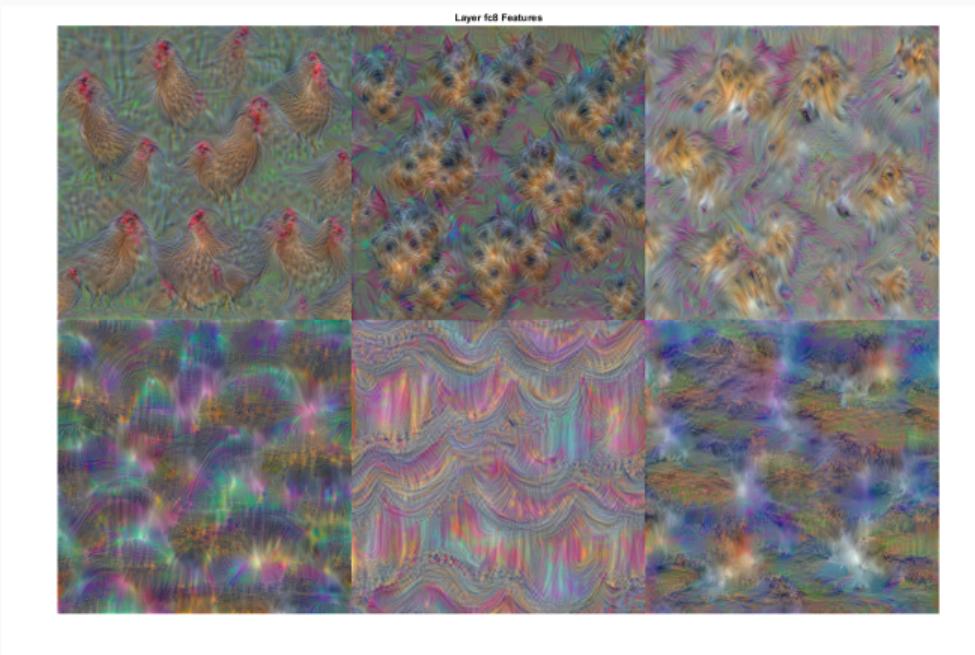


Figure 5: Vizuelizacija naučenih konvolutivnih filtera.

Konvolutivne neuronske mreže - arhitektura

Arhitektura se najčešće sastoji iz kombinacije sledećih elemenata:

- Sloj konvolucije
- Sloj agregacije
- Poptuno povezani sloj

U poslednjih nekoliko godina i:

- Skip konekcije (eng. *skip connection*) (He et al. 2015)
- Kombinacija više slojeva u jednom (eng. *inception module*) (Szegedy et al. 2014)

Konvolutivne neuronske mreže - arhitektura

Konvolutivni sloj:

- Služi da detektuje određenu pravilnost u podacima
- Na primer, da detektuje horizontalne, vertikalne i kose linije (niži sloj) ili oči, uši i usne (viši sloj)

Konvolutivne neuronske mreže - arhitektura

Agregatni sloj (eng. pooling):

- Ukrupnjava informaciju iz prethodnog sloja (uglavnom je to konvolutivni sloj)
- Kao funkcija ukrupnjavanja se koristi maksimum ili prosek

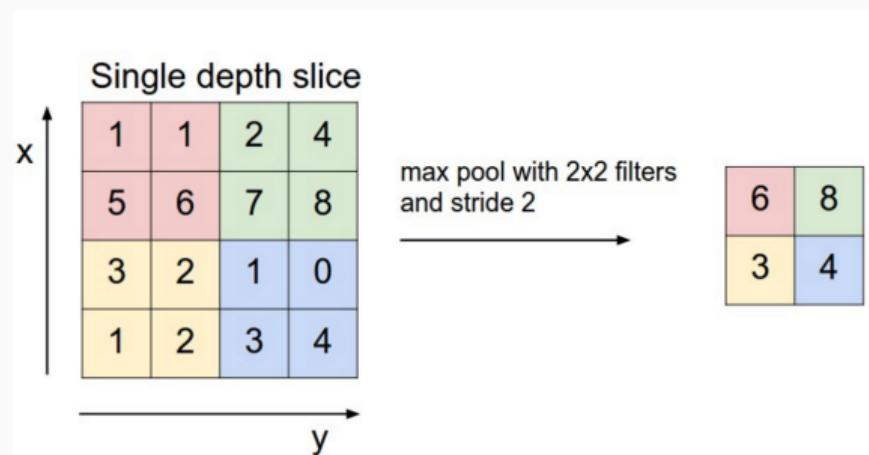
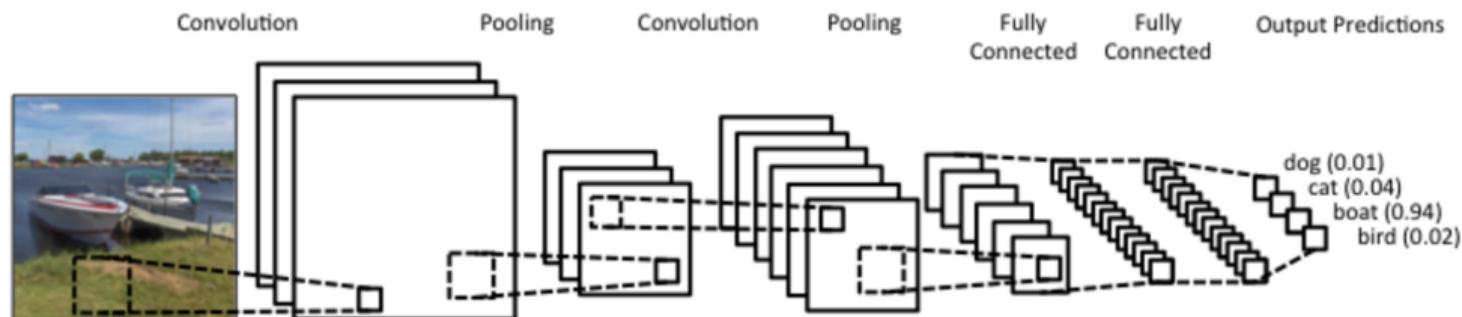


Figure 6: Agregacija sa funkcijom maksimuma.

Potpuno povezani sloj:

- Uglavnom se koristi među poslednjim slojevima
- Vrši dalja izračunavanja nad atributima koji su konstruisani u prethodnim slojevima

Konvolutivne neuronske mreže - primer



Konvolutivne neuronske mreže - primer

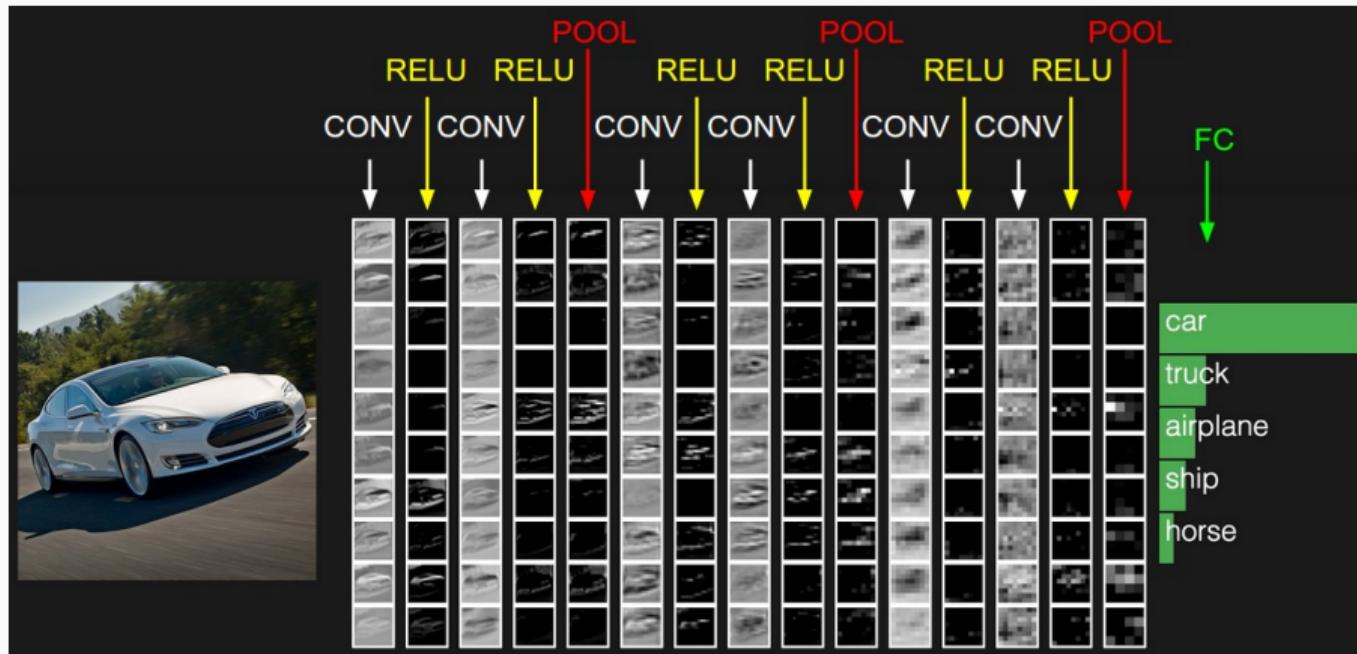


Figure 7: Primer konvolutivne mreže.

Konvolutivne mreže

- Lep interaktivni primer: [ovde](#)

[ConvNetJS CIFAR-10 demo](#)

Description

This demo trains a Convolutional Neural Network on the [CIFAR-10 dataset](#) in your browser, with nothing but Javascript. The state of the art on this dataset is about 90% accuracy and human performance is at about 94% (not perfect as the dataset can be a bit ambiguous). I used [this python script](#) to parse the [original files](#) (python version) into batches of images that can be easily loaded into page DOM with img tags.

This dataset is more difficult and it takes longer to train a network. Data augmentation includes random flipping and random image shifts by up to 2px horizontally and vertically.

By default, in this demo we're using Adadelta which is one of per-parameter adaptive step size methods, so we don't have to worry about changing learning rates or momentum over time. However, I still included the text fields for changing these if you'd like to play around with SGD+Momentum trainer.

Report questions/bugs/suggestions to [@karpathy](#).

Training Stats

pause

Forward time per example: 7ms
Backprop time per example: 12ms
Classification loss: 1.84716
L2 Weight decay loss: 0.00172
Training accuracy: 0.35
Validation accuracy: 0.25
Examples seen: 2953
Learning rate: change
Momentum: change
Batch size: change
Weight decay: change

[save network snapshot as JSON](#)
[init network from JSON snapshot](#)

LOSS:

The graph plots 'LOSS' on the y-axis (ranging from 1.65 to 2.45) against iterations on the x-axis (ranging from 0k to 4k). The data shows a general downward trend with some fluctuations. Key points on the curve include:

Iteration (k)	Loss
0	2.45
0.4	2.15
0.8	2.05
1.2	1.95
1.6	1.85
2.0	1.80
2.4	1.85
2.8	1.75
3.2	1.70
3.6	1.68
4.0	1.65

[clear graph](#)



PyTorch

- Biblioteka i radno okruženje namenjeno dubokom učenju
- Naslednik biblioteke Torch za jezik Lua
- Danas ga primarno razvija AI tim iz kompanije Facebook
- PyTorch je besplatan i otvorenog koda
- Primarno se koristi iz jezika Python, mada postoji i **eksperimentalna C++ podrška**
 - Ako želite C++ ipak je bolja opcija TensorFlow

PyTorch: graf izračunavanja

Graf izračunavanja

- Neuronska mreža predstavlja aproksimaciju neke funkcije
 - Odnosno, mreža predstavlja **izuzetno složenu** funkciju
 - Na primer, funkcija koja prihvata sliku, a detektuje da li se osoba smeje ili ne
- Kako bi je lakše predstavili, prikazujemo je pomoću grafa izračunavanja umesto formulom
- Graf izračunavanja prikazuje kako se **ulaz** transformiše u **izlaz**
- I to, prikazuje se kako se podaci transformišu tokom izračunavanja

Graf izračunavanja

$$f(x, y, z) = (x + y) \cdot z$$

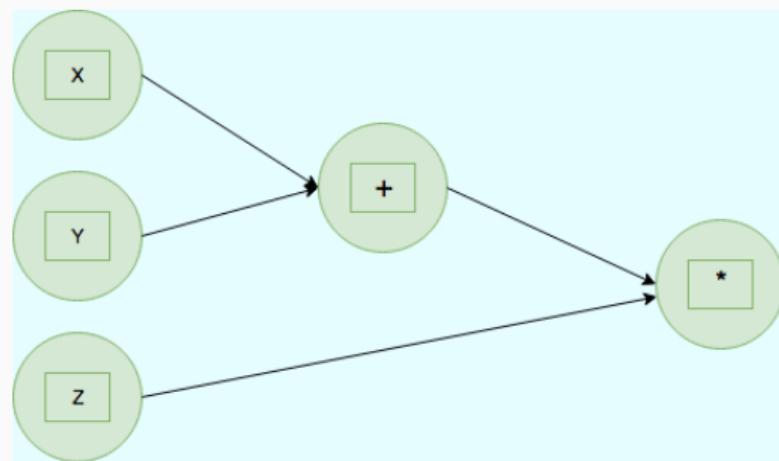


Figure 8: Graf izračunavanja funkcije f , [izvor](#)

Graf izračunavanja

- Koristeći biblioteku poput biblioteke PyTorch, mi definišemo graf izračunavanja
- Biblioteke podržavaju izračunavanje grafa na grafičkim karticama
- Ovo je izuzetno važno jer grafičke kartice poseduju hiljada jezgara
- Većina operacija u neuronskim mrežama može se svesti na matrične operacije
- Grafičke kartice mogu vrlo dobro da izvrše paralelizaciju, jer im je posao paralelizacija matričnih operacija

PyTorch: graf izračunavanja (code!)

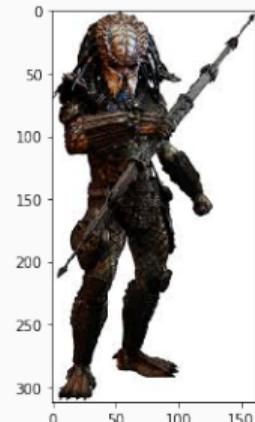
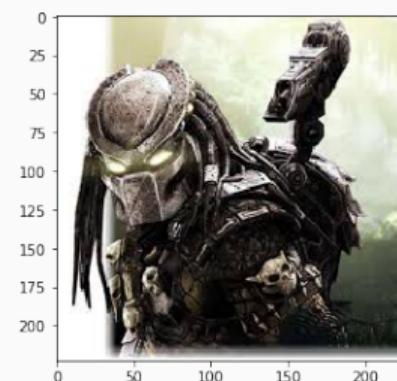
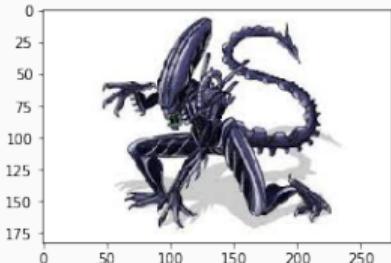
PyTorch: konvolutivna mreža (code!)

PyTorch: transfer učenja

Predator vs Alien



Predator vs Alien



Predator vs Alien

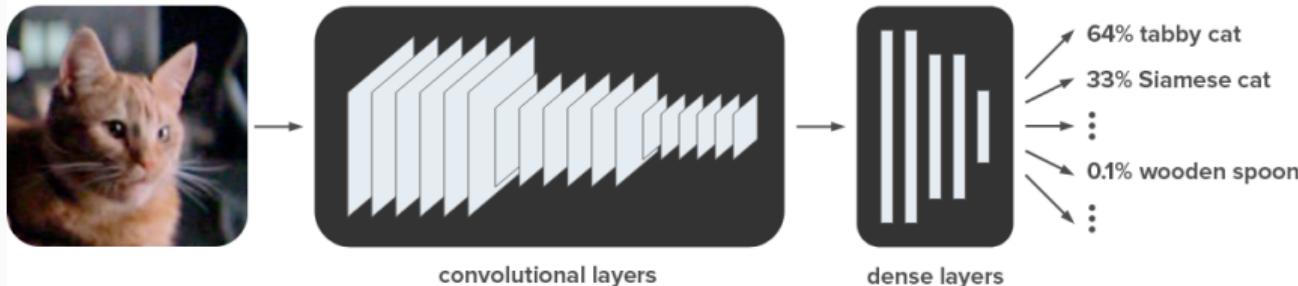
- Želimo da vršimo binarnu klasifikaciju
- Ali dostupno nam je samo 447 slika po klasi
- Konvolutivne mreže zahtevaju veliku količinu podataka za obučavanje
- Primer i kodovi bazirani na [članku](#)

Transfer učenja

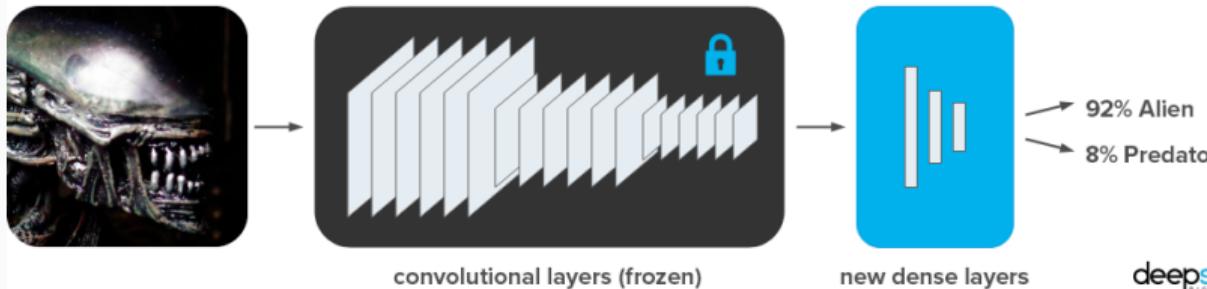
- Obuhvata korišćenje i obučavanje već obučenog modela na nekim drugim podacima
- Ideja je da naučeni filteri u jednom problemu mogu biti korisni u drugom
- Ne postoji jasan algoritam kako se radi transfer učenje, no neke varijante su:
 - Zamrznuti konvolutivne slojeve, otkloniti potpuno povezane i postaviti svoje koji se potom obučavaju
 - Koristiti konvolutivne slojeve neke mreže za dobijanje nove reprezentacije podataka, dalje raditi sa raznim drugim modelima
 - Zamrznuti deo konvolutivnih slojeva, obučiti ostatak
 - I slično . . .

Transfer učenja

Pre-training



Transfer learning



Transfer učenja (code!)

Literatura

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. “Deep Residual Learning for Image Recognition.” *CoRR* abs/1512.03385.

<http://arxiv.org/abs/1512.03385>.

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2014. “Going Deeper with Convolutions.” <http://arxiv.org/abs/1409.4842>.