practical deep learning

research and jobs

Alex Honchar
University of Verona

Day 5 goals

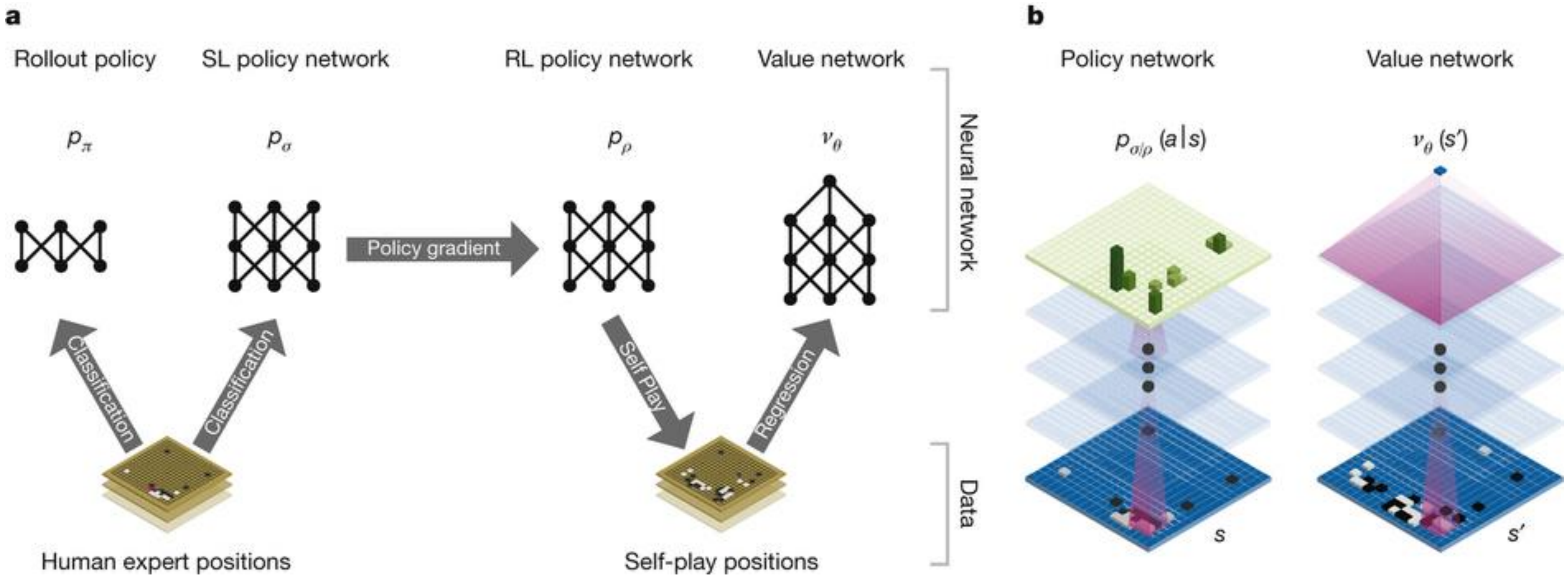
- You understand modern research trends in deep learning
- You understand how to get a job in machine learning area
- You can perform the full deep learning pipeline after making a final project

New technologies?

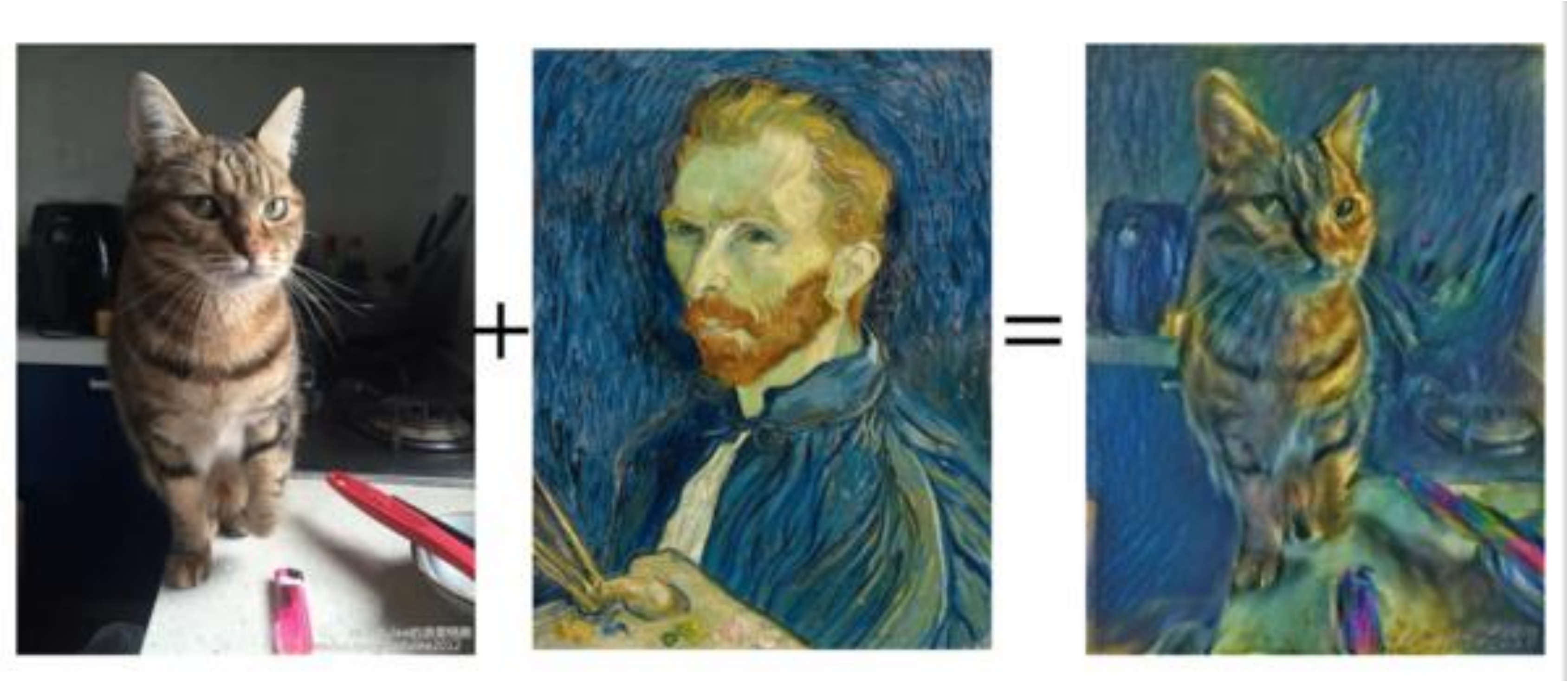
- Multilayer perceptron (1969)
- Convolutional neural networks (1980s)
- Recurrent neural networks (LSTM - 1997)

2k16

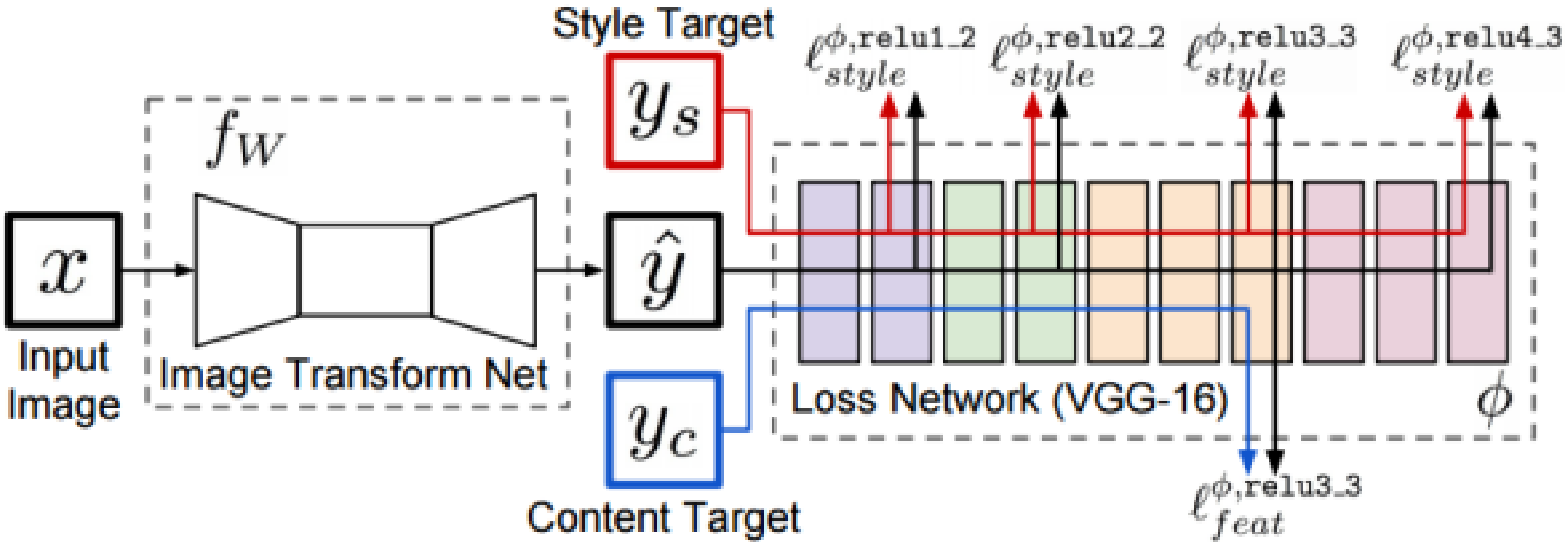
1. AlphaGo


a
Rollout policy
SL policy network
RL policy network
Value network
p_π
p_σ
p_ρ
v_θ
Policy gradient
Classification
Classification
Self Play
Regression
Human expert positions
Self-play positions
Neural network
Data
b
Policy network
Value network
$p_{\sigma/\rho}(a|s)$
$v_\theta(s')$
s
s′

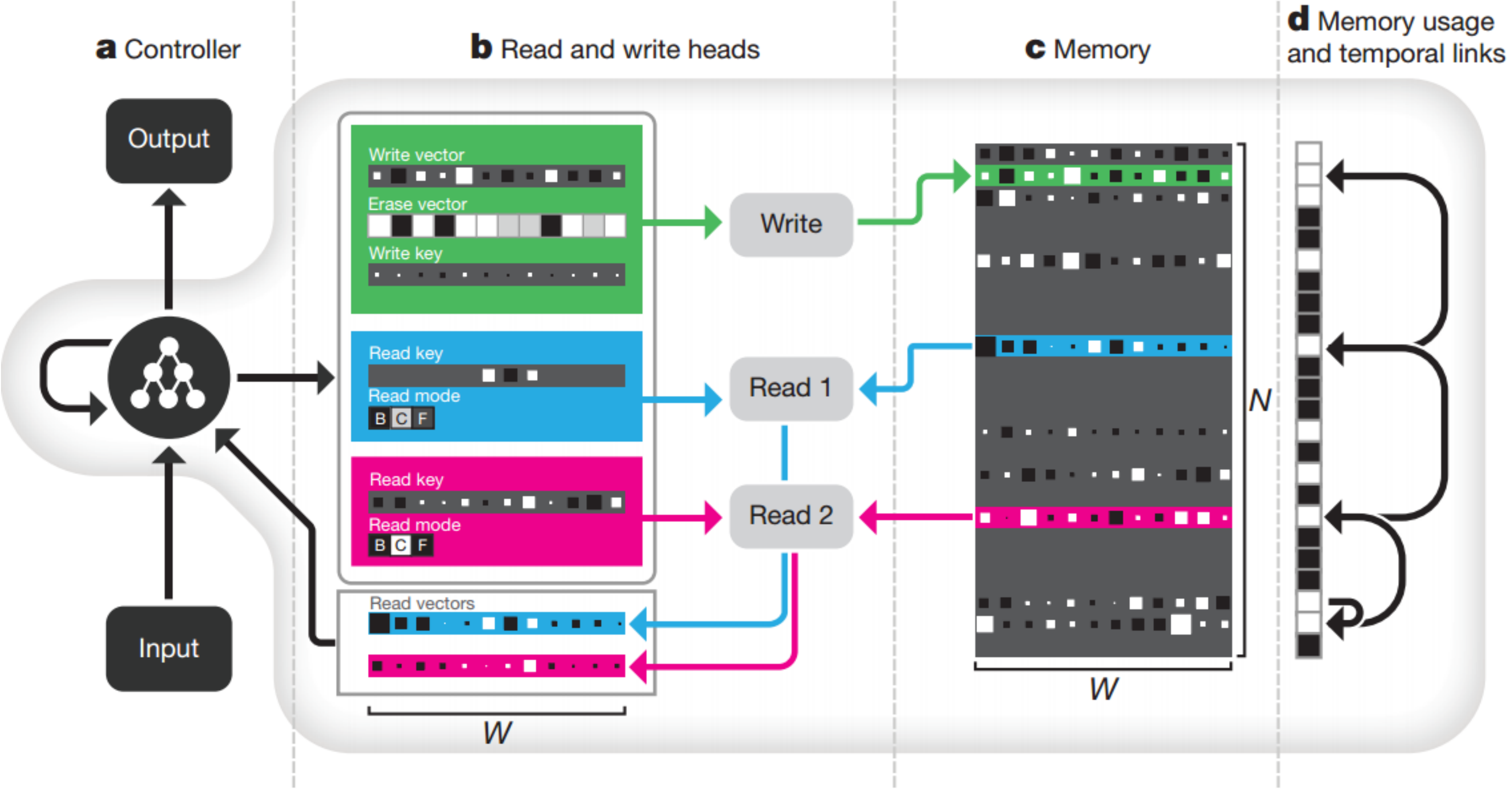
2. Image style transfer



2. Image style transfer

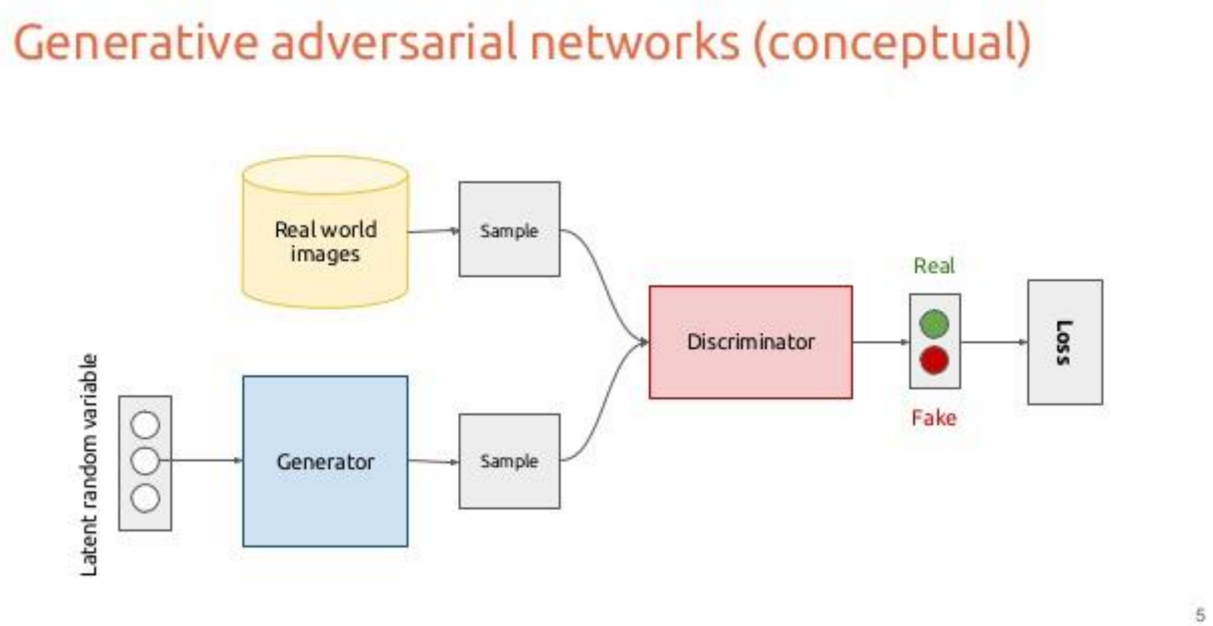

Style Target
$\ell_{style}^{\phi,relu1\_2}$
$\ell_{style}^{\phi,relu2\_2}$
$\ell_{style}^{\phi,relu3\_3}$
$\ell_{style}^{\phi,relu4\_3}$
y_s
f_W
x
$\hat{y}$
Input Image
Image Transform Net
y_c
Loss Network (VGG-16)
ϕ
$\ell_{feat}^{\phi,relu3\_3}$
Content Target

3. Differentiable neural computer



4. OpenAI

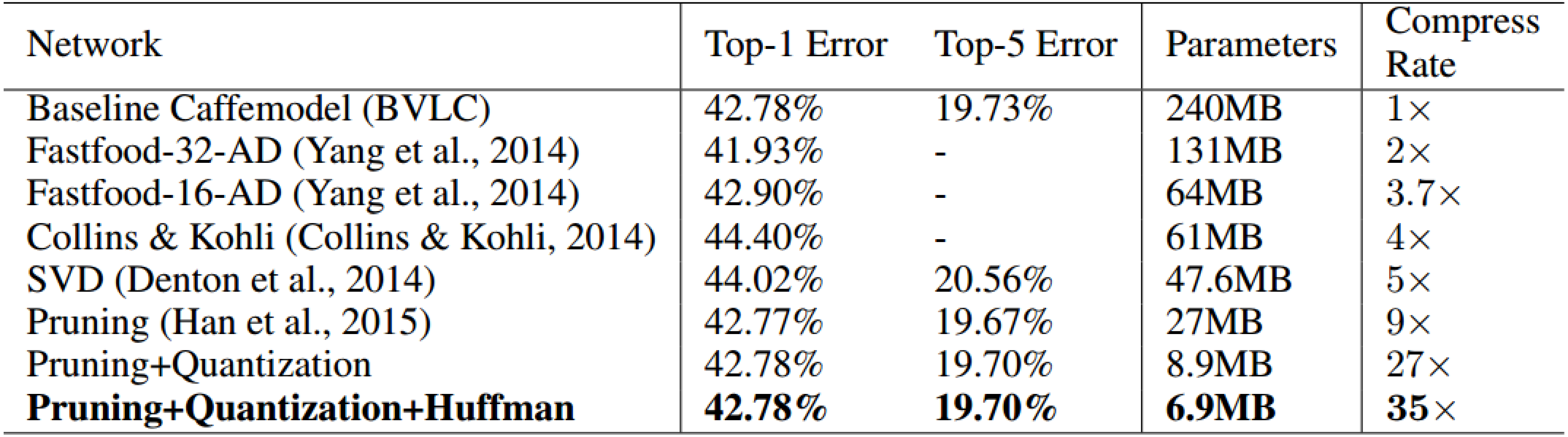
5. Generative models



Generative adversarial networks (conceptual)

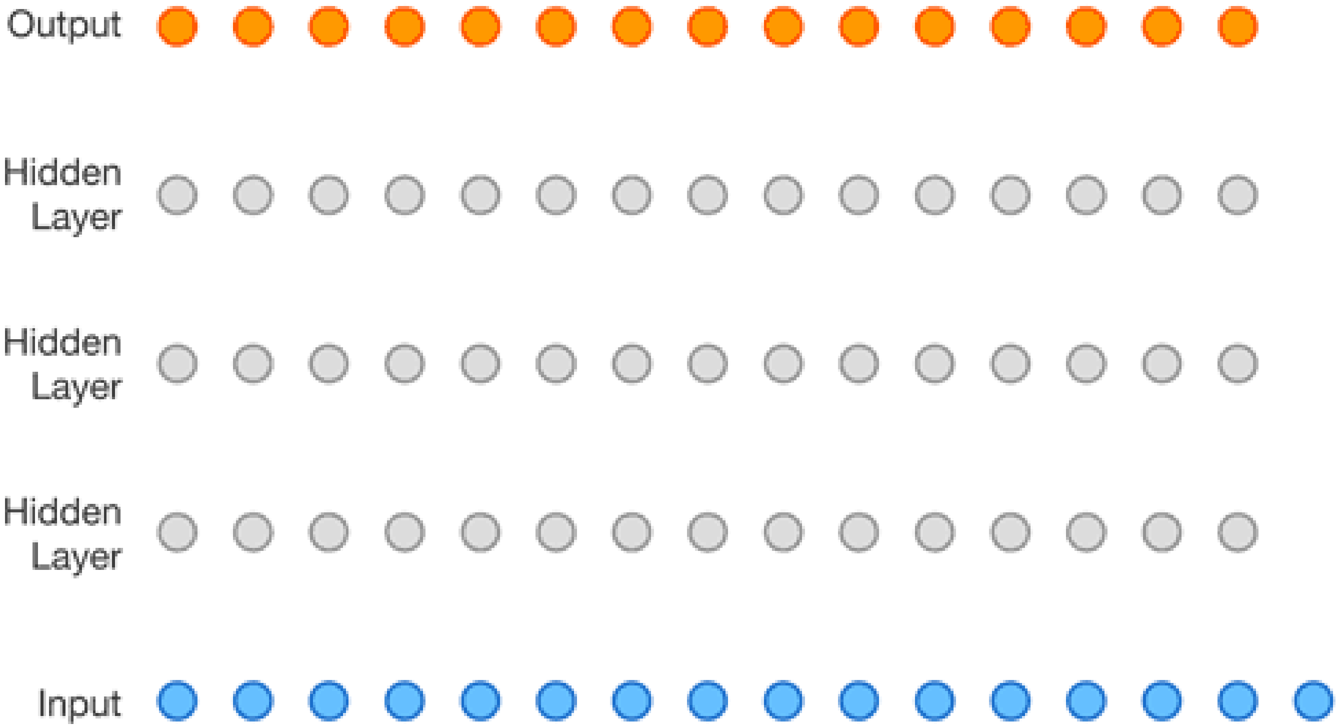
5

6. Neural networks compression



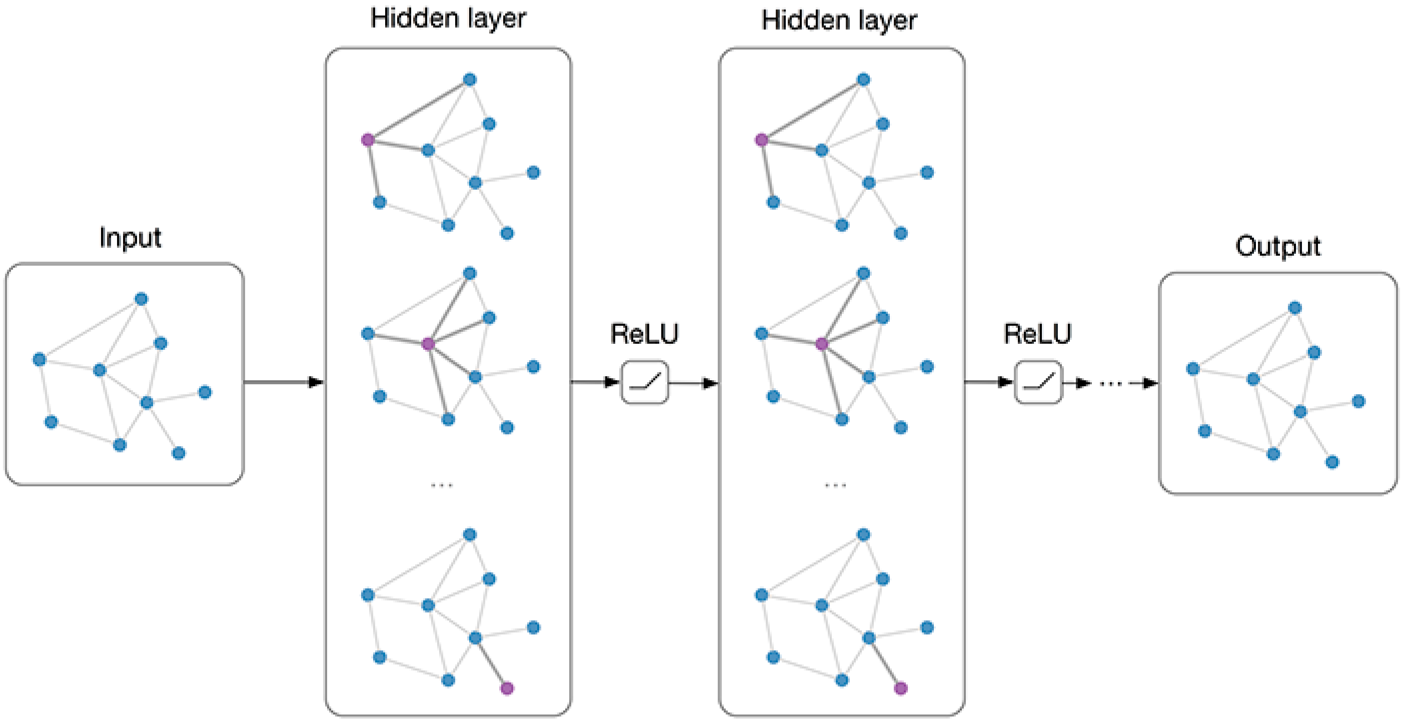
Network	Top-1 Error	Top-5 Error	Parameters	Compress Rate
Baseline Caffemodel (BVLC)	42.78%	19.73%	240MB	1×
Fastfood-32-AD (Yang et al., 2014)	41.93%	-	131MB	2×
Fastfood-16-AD (Yang et al., 2014)	42.90%	-	64MB	3.7×
Collins & Kohli (Collins & Kohli, 2014)	44.40%	-	61MB	4×
SVD (Denton et al., 2014)	44.02%	20.56%	47.6MB	5×
Pruning (Han et al., 2015)	42.77%	19.67%	27MB	9×
Pruning+Quantization	42.78%	19.70%	8.9MB	27×
Pruning+Quantization+Huffman	**42.78%**	**19.70%**	**6.9MB**	**35×**

7. Voice generation

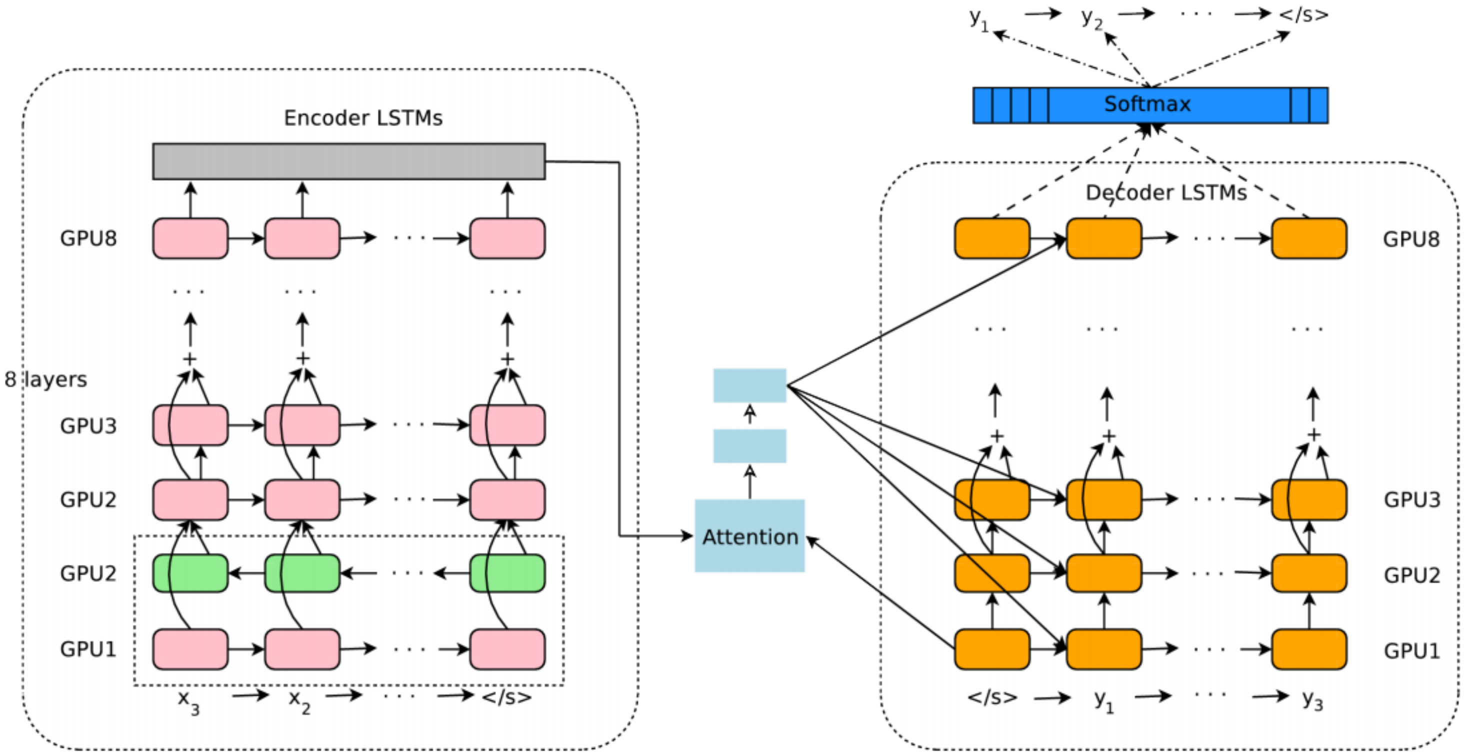


8. Donald Trump’s win

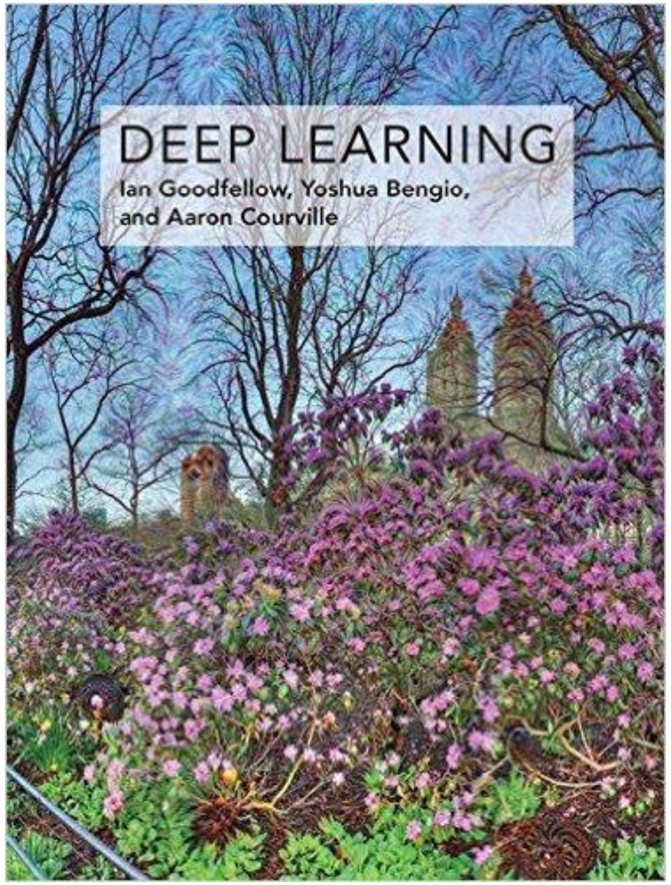
9. CNNs for graphs

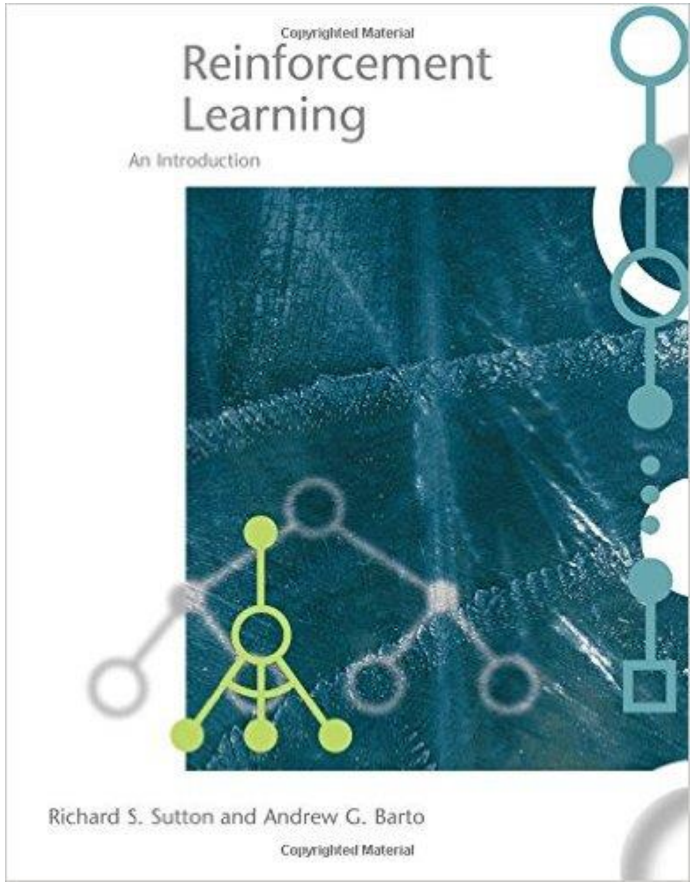


10. End-to-end machine translation



+ Good books


DEEP LEARNING
Ian Goodfellow, Yoshua Bengio, and Aaron Courville


Copyrighted Material
Reinforcement Learning
An Introduction
Richard S. Sutton and Andrew G. Barto
Copyrighted Material

jobs

Main professions

- Data Scientist
- Machine learning Engineer
- Researcher

Skills needed (computer science)

MUST:

- Programming skills (Python / C++ / Java / Matlab / Lua)
- Algorithms and data structures (discrete mathematics, graphs!)

IS A PLUS:

- Parallel computing
- Knowledge of GPU programming (CUDA)
- Backend development
- Data visualization skills

Skills needed (mathematics)

MUST (2-3 years of bachelors in math/CS/statistics):

- Linear algebra (all about matrixes, vector spaces, linear operators)
- Calculus (limits, partial derivatives, series analysis)
- Probability theory (random variables, distributions, mean, variance)
- Statistics (central limit theorem, statistics tests, regression, Bayesian stats)

IS A PLUS:

- Numerical analysis (interpolations, numerical derivatives, numerical optimization)
- Tensor algebra
- Stochastic processes
- Multivariate statistics
- Functional analysis (for reading and understanding latest white papers)

7 easy (no) steps to get a job

1. You must know and understand **basics** listed before
2. Complete one of the **machine learning** courses with certificate (Coursera Stanford course for example)
3. Make a submission on an easy **Kaggle** problem
4. Complete one or two courses / books in **deep learning**
5. Make a submission on intermediate / hard **Kaggle** problem
6. Implement one or two **white paper algorithms** from scratch using Theano / Tensorflow / Caffe and upload it to GitHub
7. **Compile a resume** and spam it on LinkedIn and head hunting sites

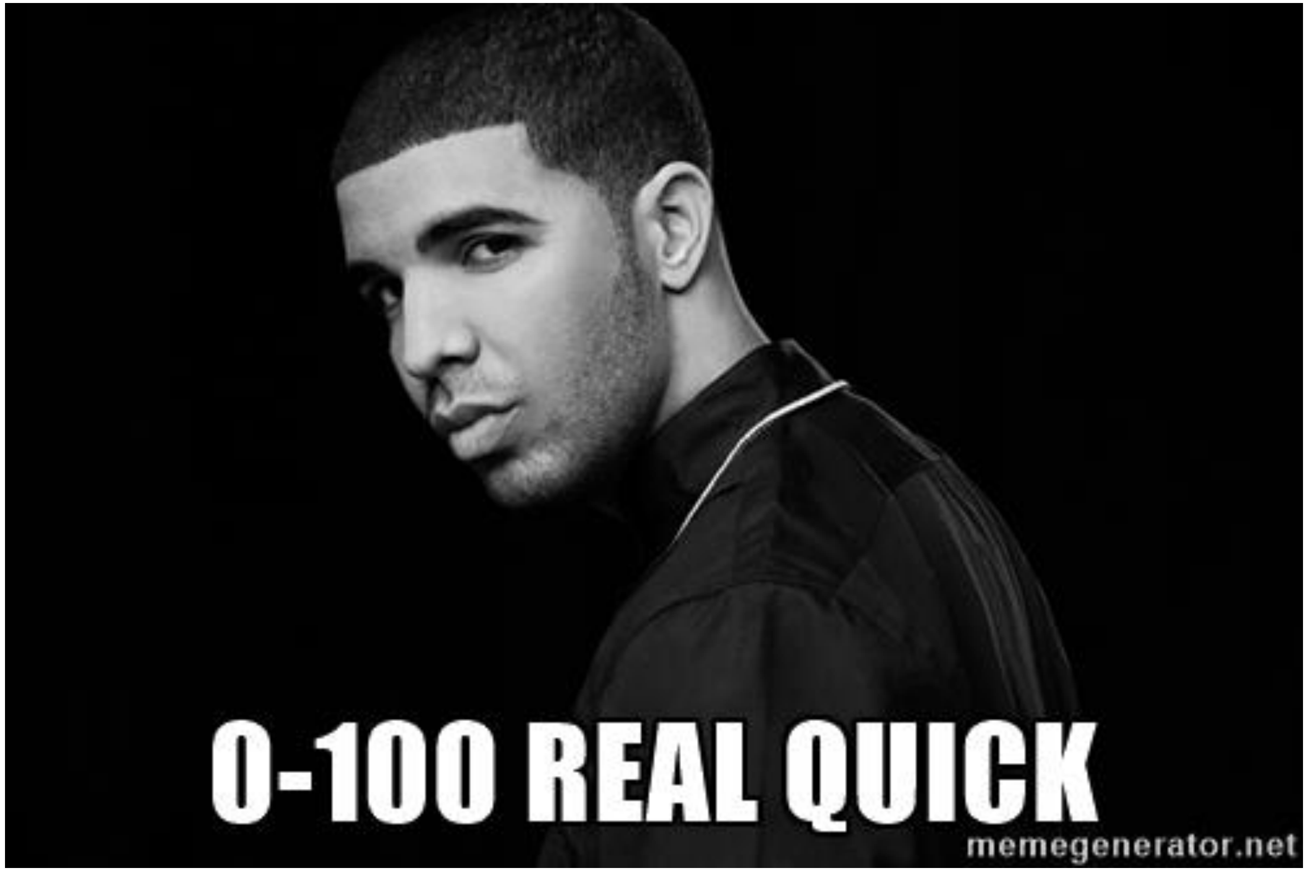
Steps MOOC mockup

- Python: https://www.codecademy.com/learn/python
- Algorithms: https://www.coursera.org/learn/introduction-to-algorithms
- Algebra: https://www.mooc-list.com/course/ut501x-linear-algebra-foundations-frontiers-edx
- Calculus: https://www.coursera.org/learn/calculus1
- Statistics: https://www.coursera.org/specializations/statistics
- Machine learning: https://www.coursera.org/learn/machine-learning
- Kaggle 1: https://www.kaggle.com/c/titanic
- Deep Learning 1 (computer vision): http://cs231n.stanford.edu/
- Deep Learning 2 (natural language processing): http://cs224d.stanford.edu/
- Kaggle 2: any from “featured” section
- White paper 1: http://arxiv.org/pdf/1603.08155v1.pdf (image style transfer project)
- White paper 2: https://arxiv.org/pdf/1609.08359v1.pdf (emoji project)

Next steps

1. Read white papers. Every week, better every day. Notice the trends, use them. (**arxiv, gitxiv, Twitter**)
2. Implement models (at least their parts) from scratch for educational purpose (**implement custom convolutional layer for Keras**)
3. Probably dive a bit more into theory
4. Practice with data: do all the “dirty job”
5. Publish papers / commit to Github

Self-study materials from zero



Find your PDF here: https://github.com/Rachnog/education

Project proposals

- Titanic survivors prediction
- House price prediction
- Fashion image classification
- Food image classification
- Arrhythmia ECG detection
- Plants images classification
- Adult content recognition
- Stock price prediction

- Face identification
- Motivational quotes generation
- Chat bots
- Speech recognition
- Machine translation
- Stealing ML models from API
- Your ideas

thank you for attention!

rachnogstyle@gmail.com
facebook |rachnogstyle