Neural Networks and Deep Learning

Lecture 1: introduction

Jan Chorowski
Instytut Informatyki
Wydział Matematyki i Informatyki Uniwersytet Wrocławski

2020-2021

Practical Information

Course Materials:

- Teams: lecture videoconferences and chat
- SKOS (basic course information and exams): https://skos.ii.uni.wroc.pl/course/view.php?id=376
- GitHub (lecture notes, assignment notebooks): https://github.com/janchorowski/dl_uwr
- USOS: grades

On-line Resources:

- https://www.deeplearningbook.org/

Extra reading

- http://cs229.stanford.edu/
- https://argmax.ai/ml-course/
- Bishop, Pattern Recognition and Machine Learning (PRML)

Machine Learning

Pattern Recognition

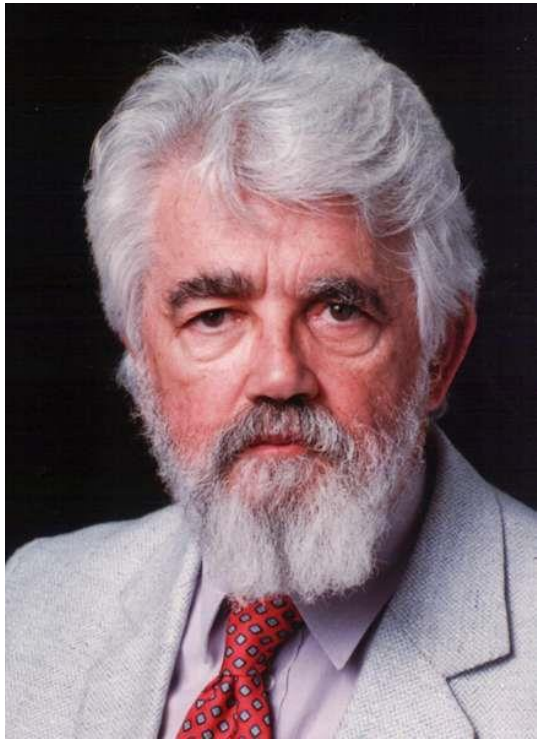
Artificial Intelligence

?

Data Mining

Deep Learning

Artificial Intelligence



John McCarthy, 1955

AI is the science and engineering of making intelligent machines.

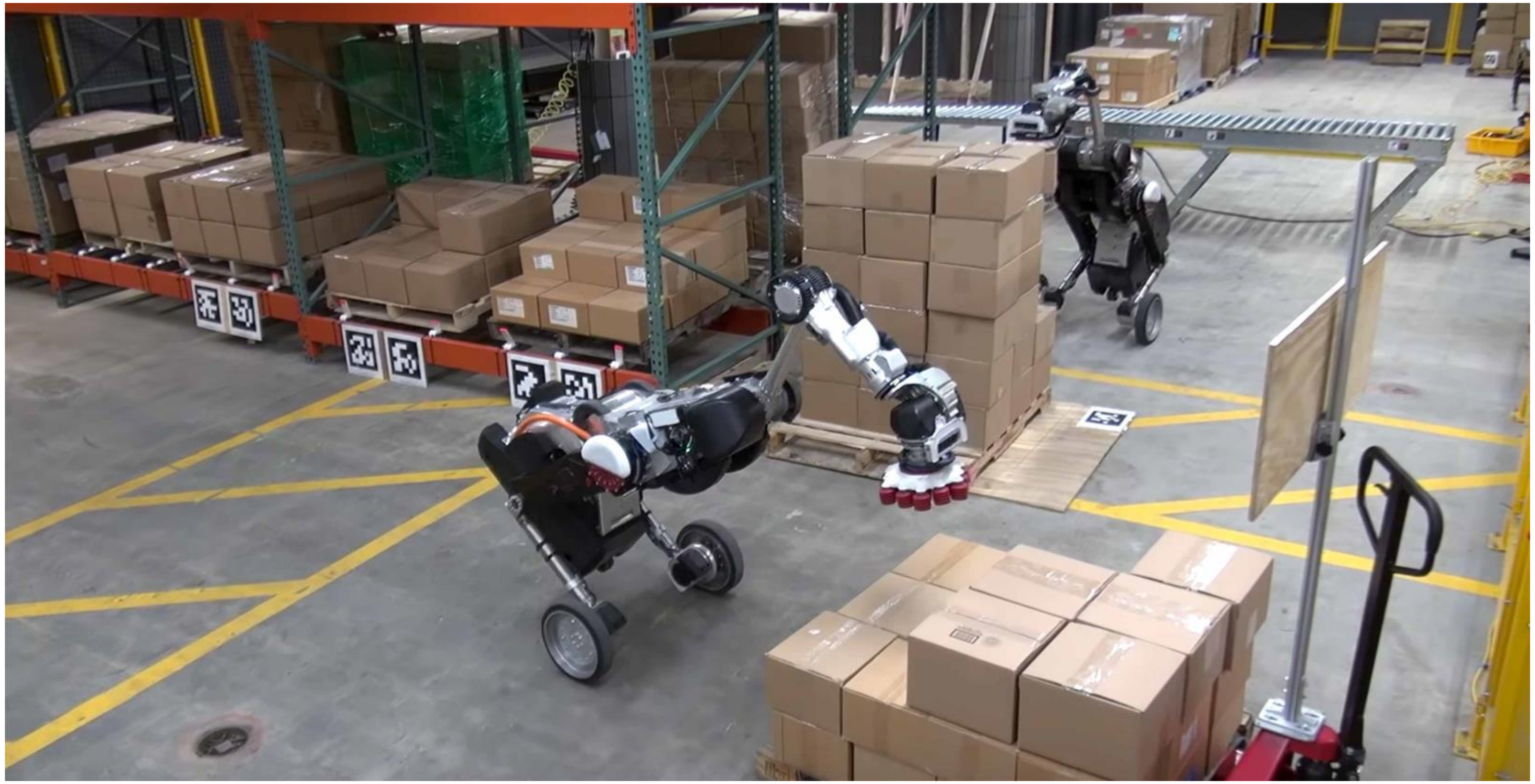
AI Paradox

Hans Moravec 1988

It is **comparatively easy** to make computers exhibit adult level performance on **intelligence tests** or **playing checkers**, and **difficult or impossible** to give them the skills of a one-year-old when it comes to **perception and mobility.**

Perception is difficult

XKCD 1425, ca 2014



Boston Dynamics 2019

Perception ≥ Pattern Matching

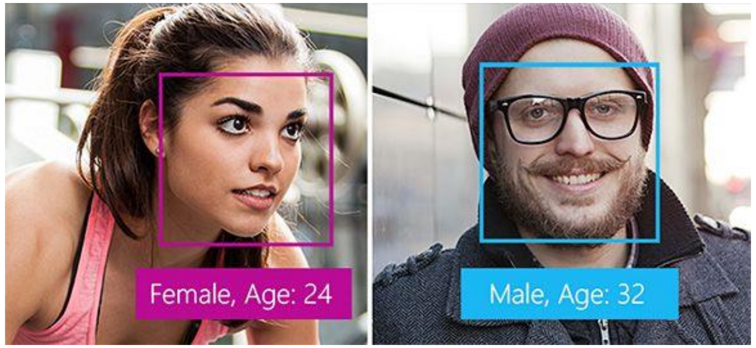
Horse is as everyone can see.

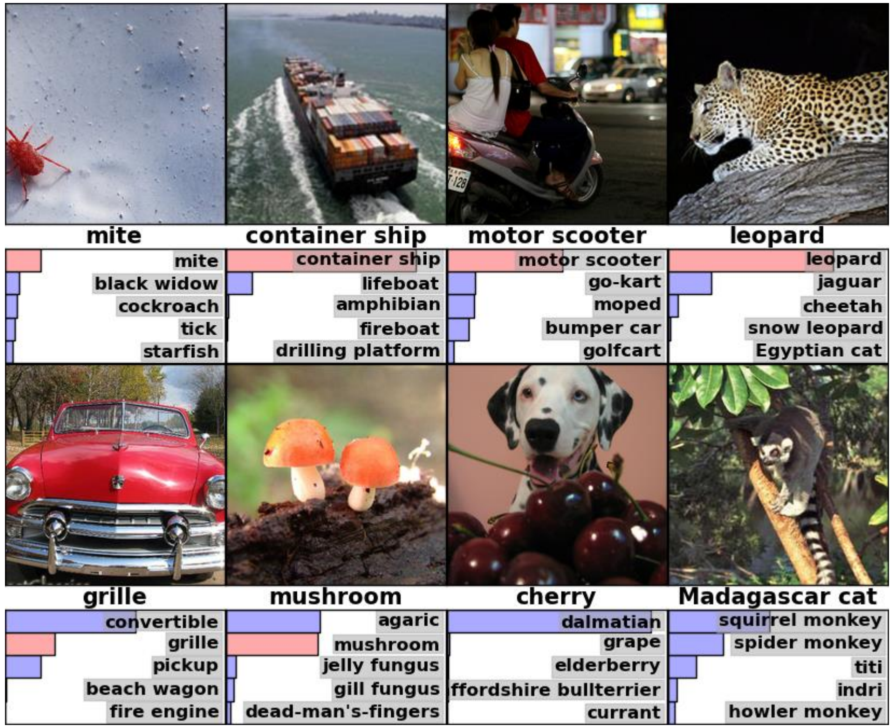
[B. Chmielowski, definition of "horse" from Nowe Ateny (New Athens, 1745), the first encyclopedia written in Polish]

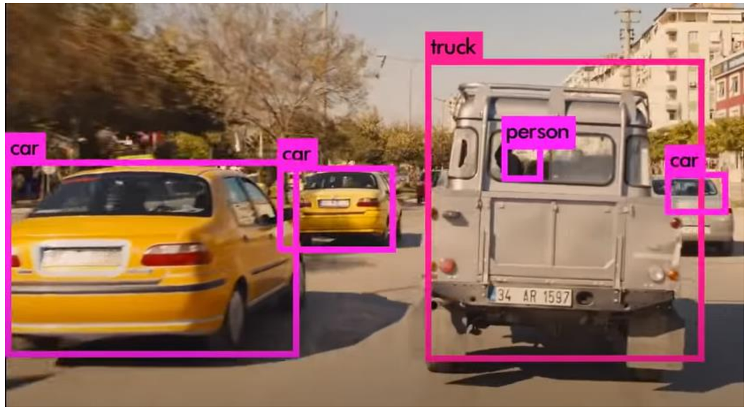




Revolution in perception

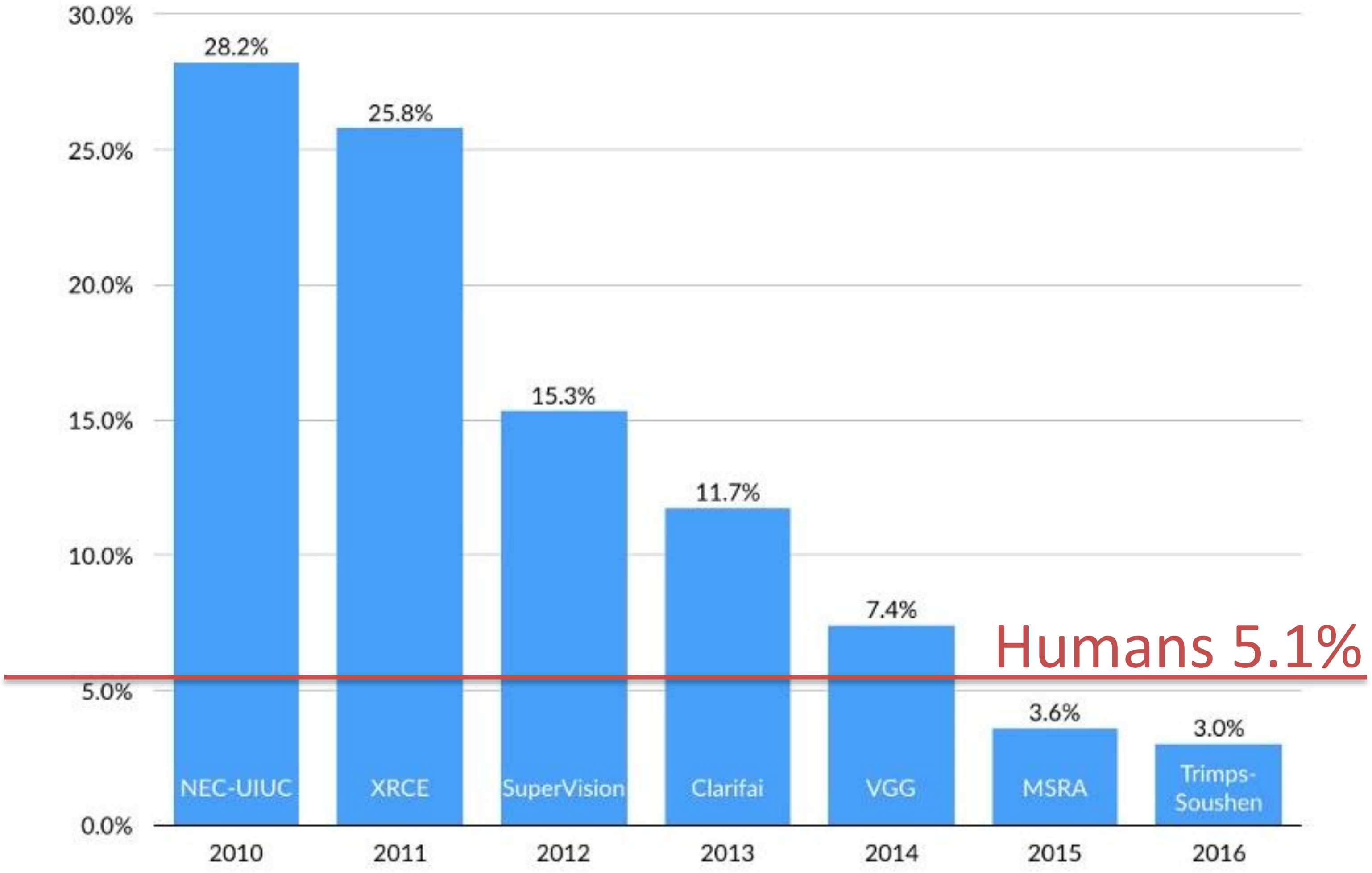






Sources: Microsoft and Apple face detection API documentations, wykop.pl, A Krizhevsky, YOLOv2 https://www.youtube.com/watch?v=VOC3huqHrss

Revolution in perception



Russakovsky et al. arXiv, 2014

Revolution in Games

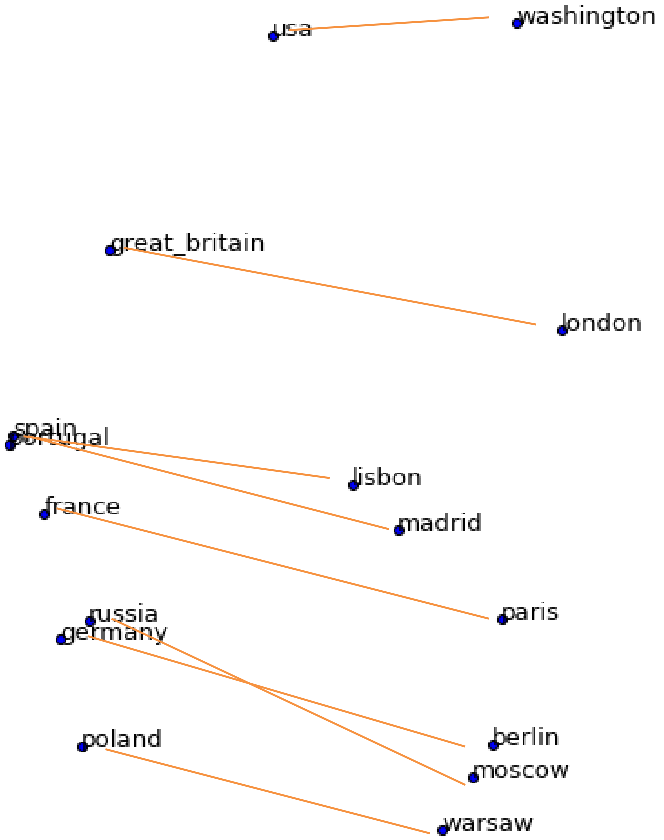
- A classical chess program considers 80e6 positions

- A human grandmaster – about 80

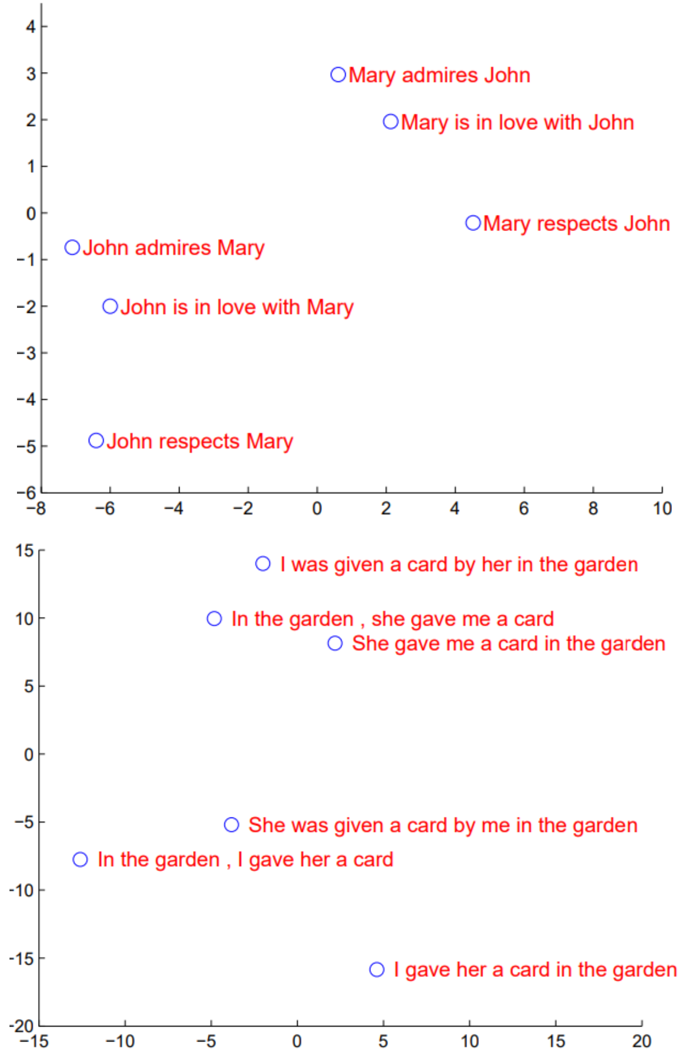
Revolution in Games

- A classical chess program considers 80e6 positions
- AlphaZero Neural net – 80e3
- A human grandmaster – about 80

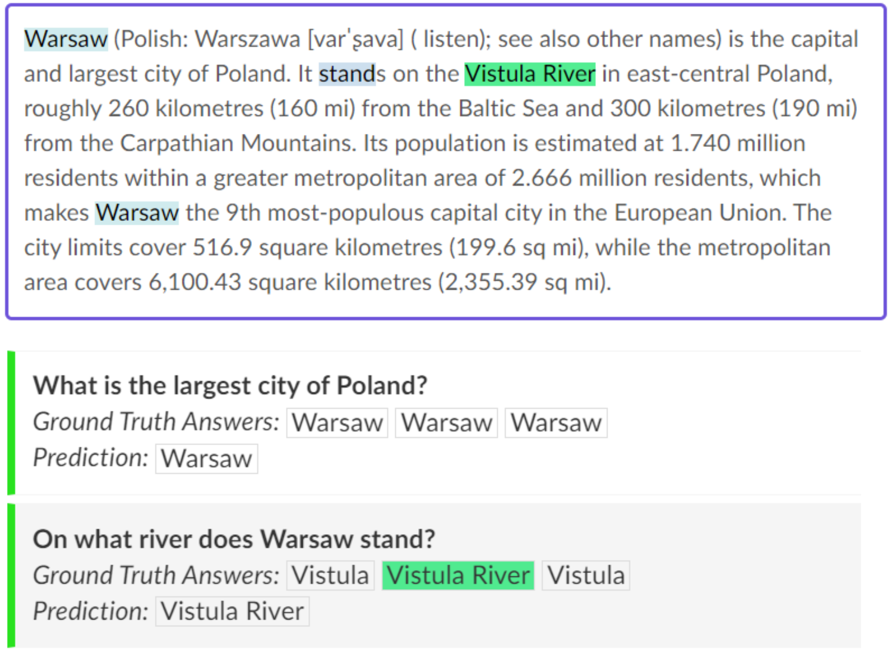
Revolution in language understanding



T. Mikolov et al. https://arxiv.org/abs/1310.4546

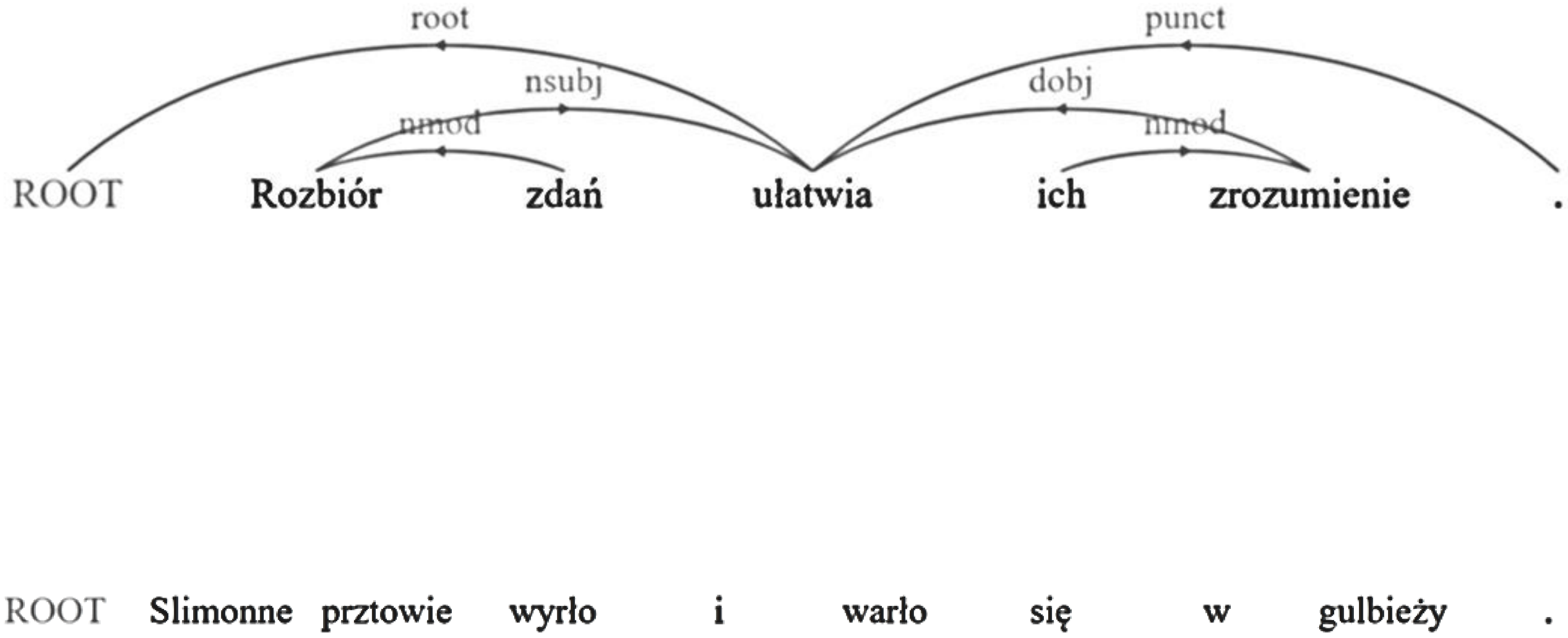


Sutskever et al., „Seq2Seq learning with Neural Networks"



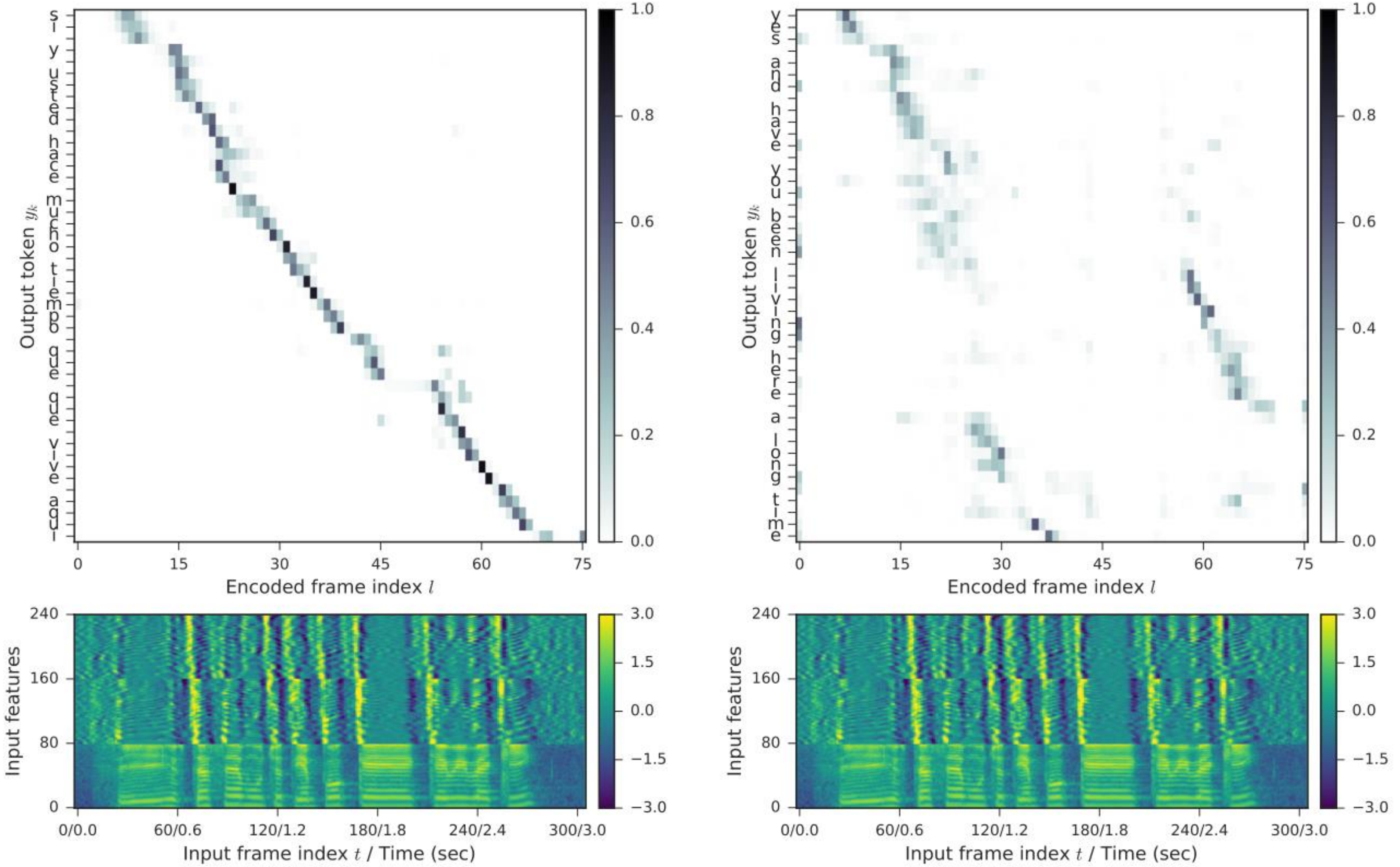
SQUAD v2 Explorer, https://rajpurkar.github.io/SQuAD-explorer/explore/v2.0/dev/Warsaw.html?model=BERT%20(single%20model)%20(Google%20AI%20Language)&version=v2.0

Revolution in language understanding



Chorowski, Zapotoczny, Rychlikowski
Read, Tag, and Parse All at Once, or Fully-neural Dependency Parsing

Revolution in speech processing



Weiss, Chorowski et al. Sequence-to-Sequence Models Can Directly Transcribe Foreign Speech

Revolution in synthesis

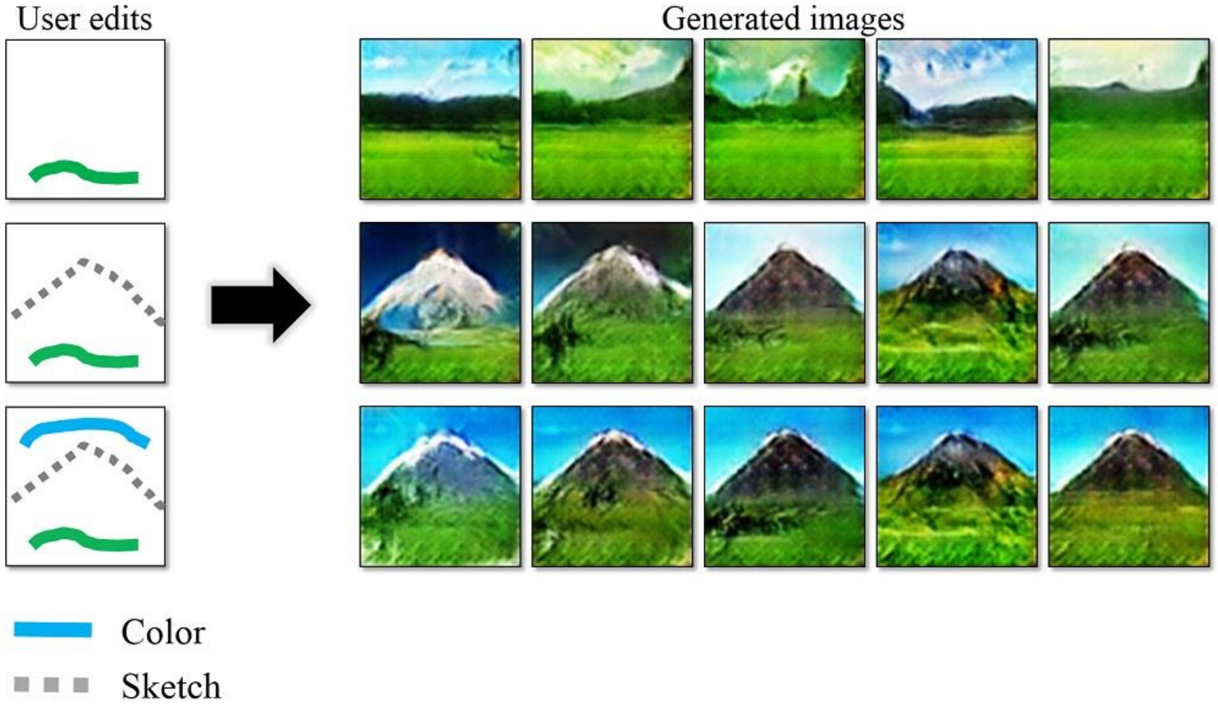
A Style-Based Generator Architecture for Generative Adversarial Networks
Tero Karras (NVIDIA), Samuli Laine (NVIDIA), Timo Aila (NVIDIA)
https://arxiv.org/abs/1812.04948

Some demos

- https://talktotransformer.com/
- https://openai.com/blog/glow/
- https://openai.com/blog/dall-e/

Image manipulation

Transform sketches into images:



https://www.youtube.com/watch?v=FDELBFSeqQs

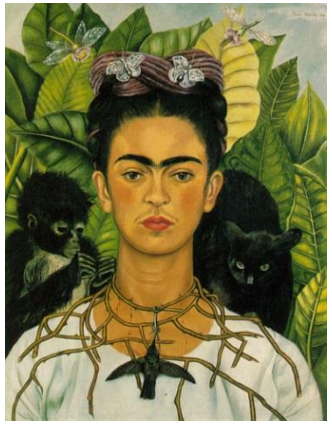
https://www.youtube.com/watch?v=9c4z6YsBGQ0

Image super-resolution

(e) Bicubic (f) SRCNN (g) A+ (h) RAISR

https://arxiv.org/pdf/1606.01299.pdf

+



=

Style transfer

Find image that takes content from image A and style from B

Gatys et al., „A Neural Algorithm of Artistic Style", 2015

Sample adaptation to videos:

https://www.youtube.com/watch?v=Khuj4ASldmU

Ruder et al., „Artistic Style Transfer For Videos"

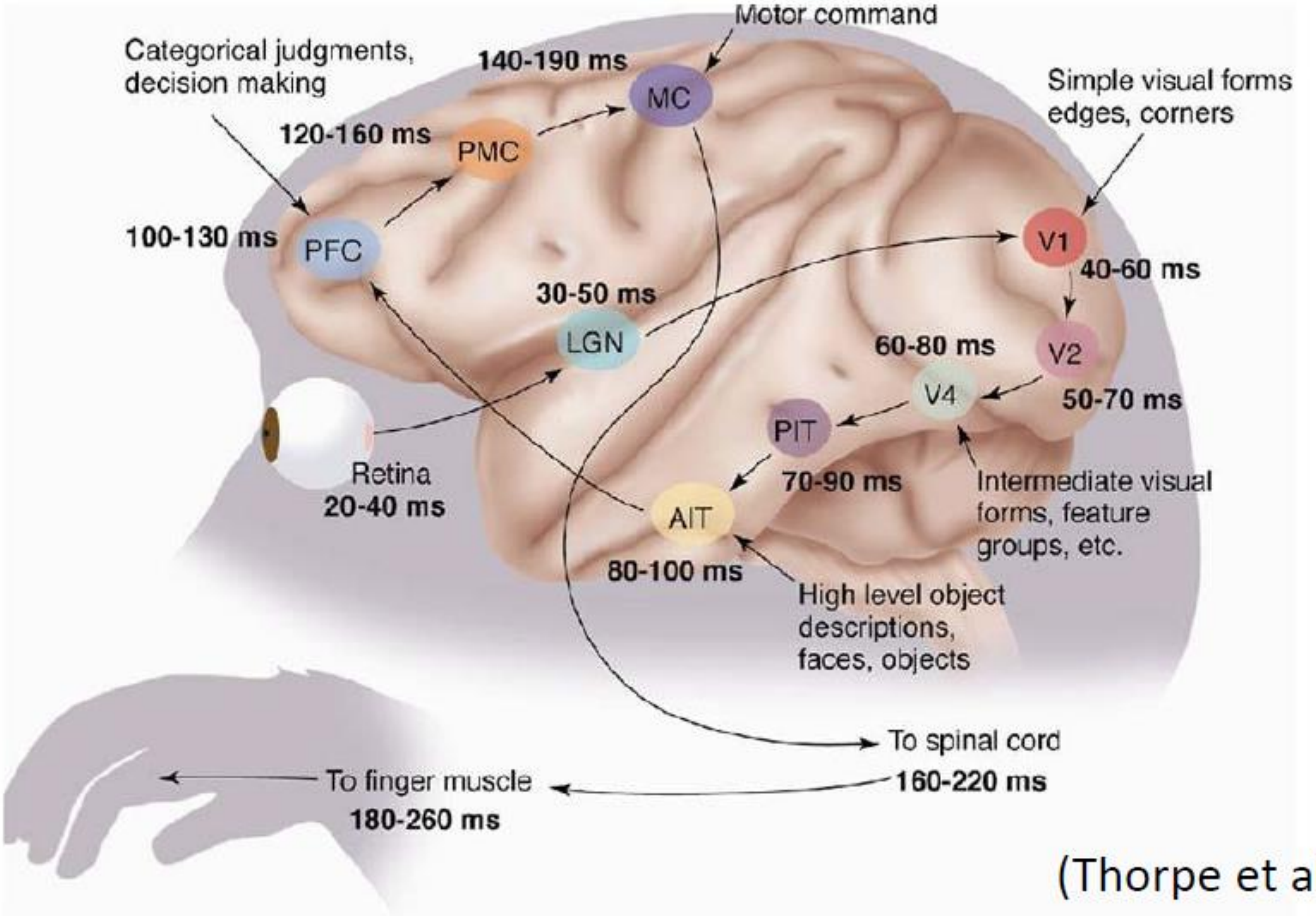
Change the image to see many eyes/buildings in it.

Inceptionism: Going Deeper into Neural Networks
http://googleresearch.blogspot.com/2015/06/inceptionism-going-deeper-into-neural.html

Grocery Trip: https://www.youtube.com/watch?v=DgPaCWJL7XI

NEURAL NET INTUITIONS

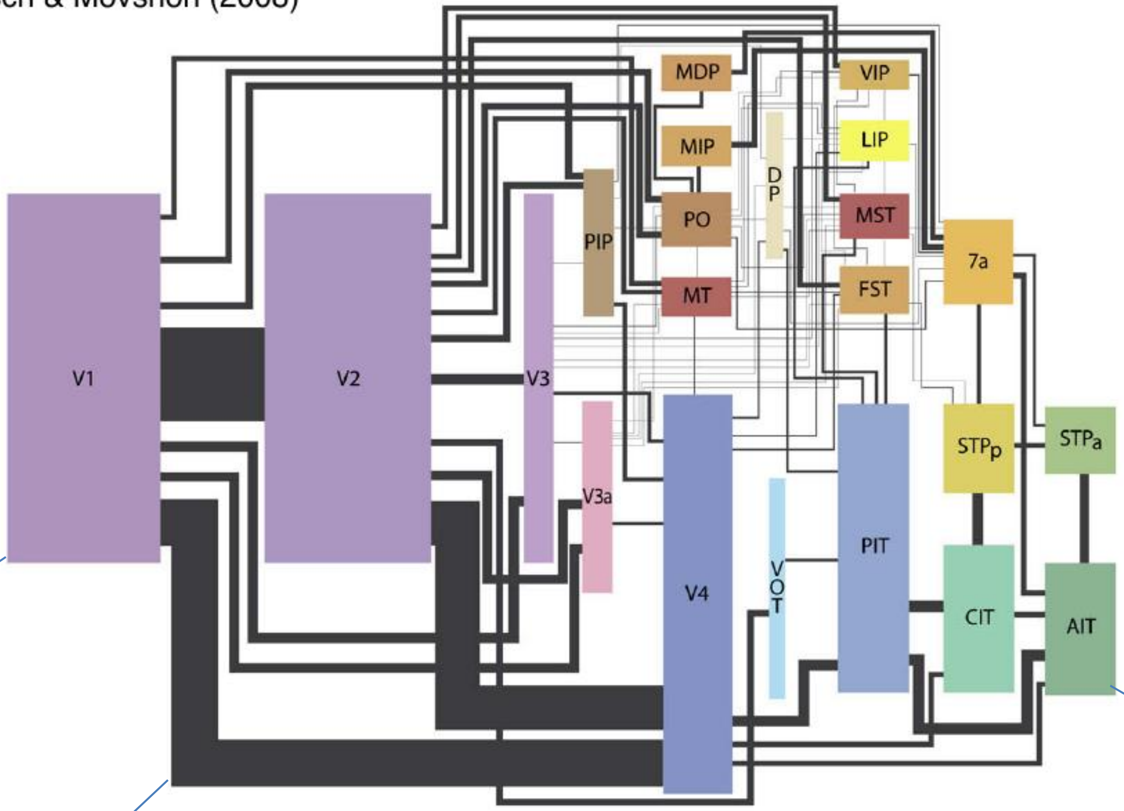
Human perception speed

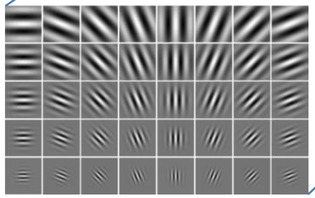


(Thorpe et al., 1995-...)

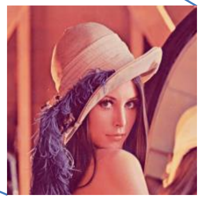
Visual Cortex Diagram

Wallisch & Movshon (2008)



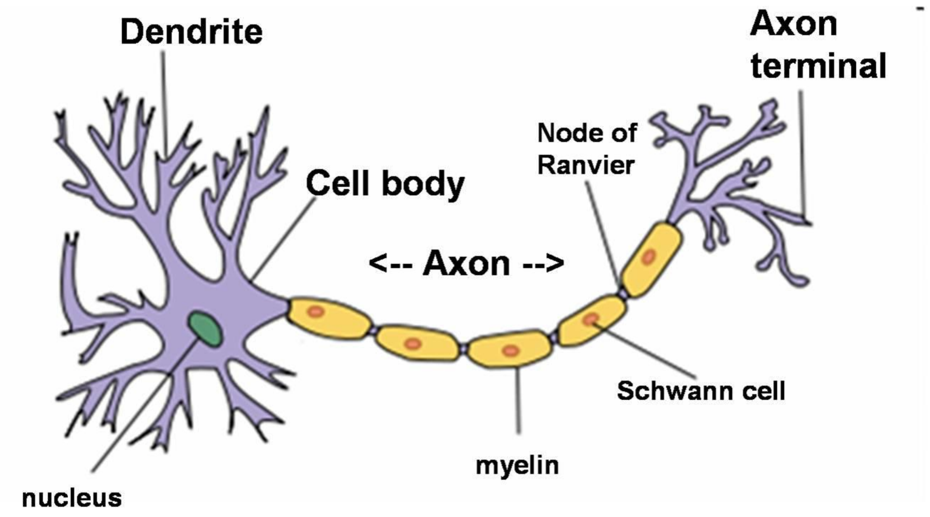


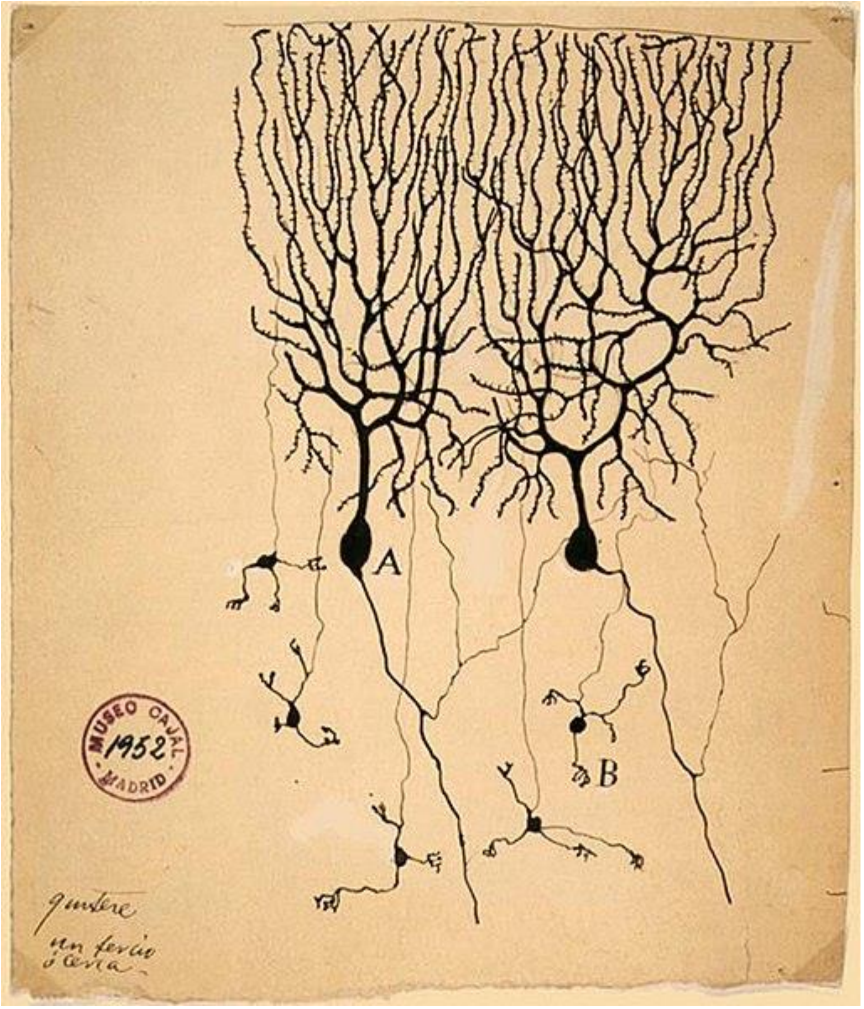
‘Gabor filters’ . . ? . . objects . faces



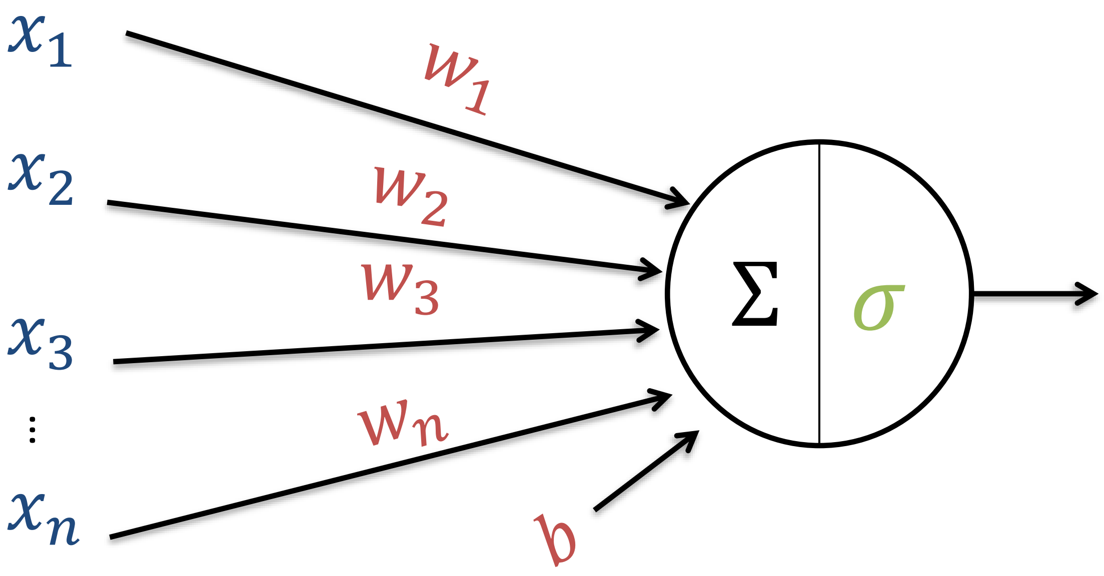
Neurons

Basic computational units in the brain





The artificial McCulloch-Pitts neuron (perceptron)



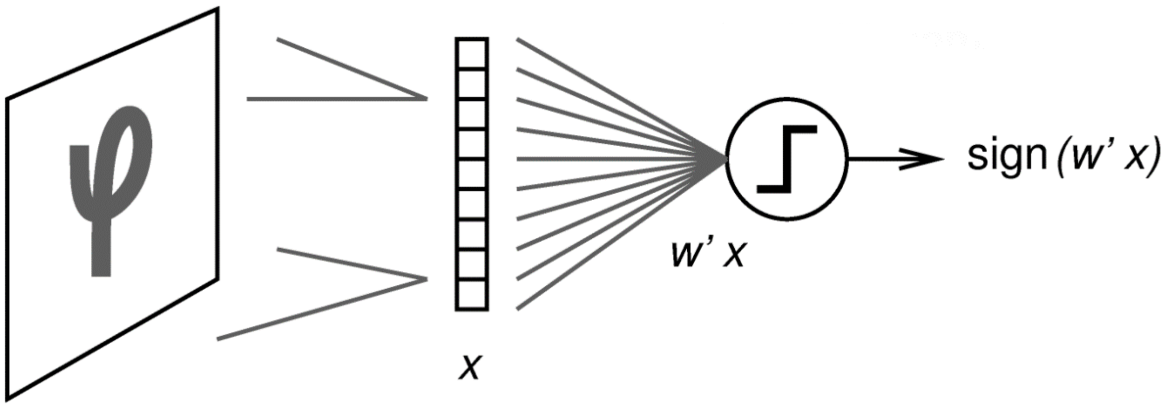
Output:

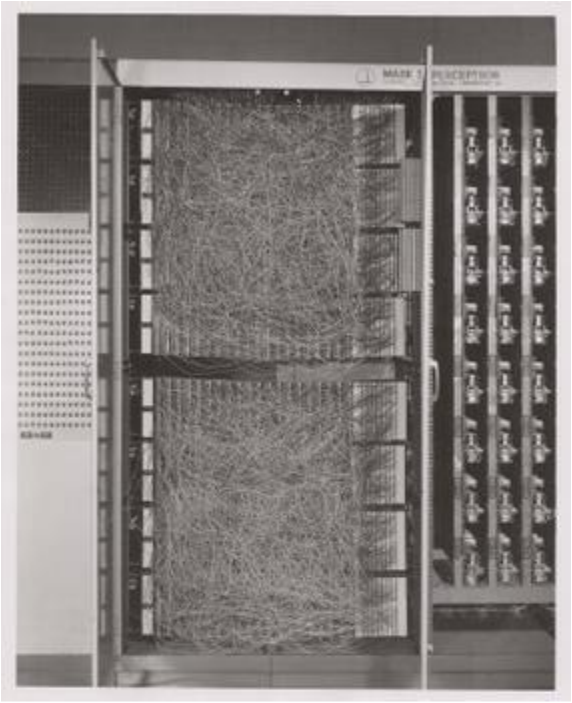
$$y = \sigma\left(\sum w_i x_i + b\right) = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b)$$

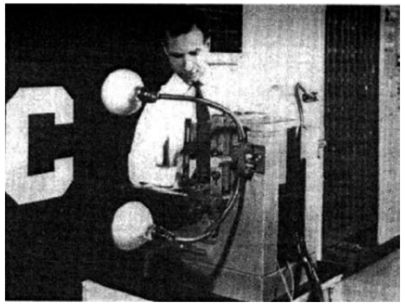
- x_i are the inputs
- w_i are the weights and b the bias
- Σ denotes the summation
- σ is a (nonlinear) activation function

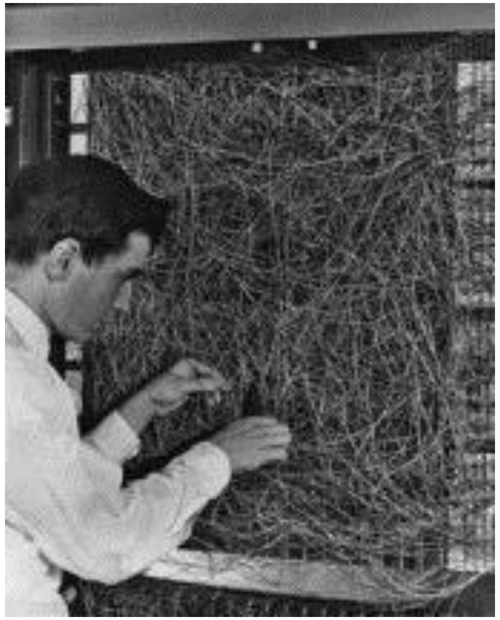
w_i, b are TUNABLE!!

Perceptron (1958)

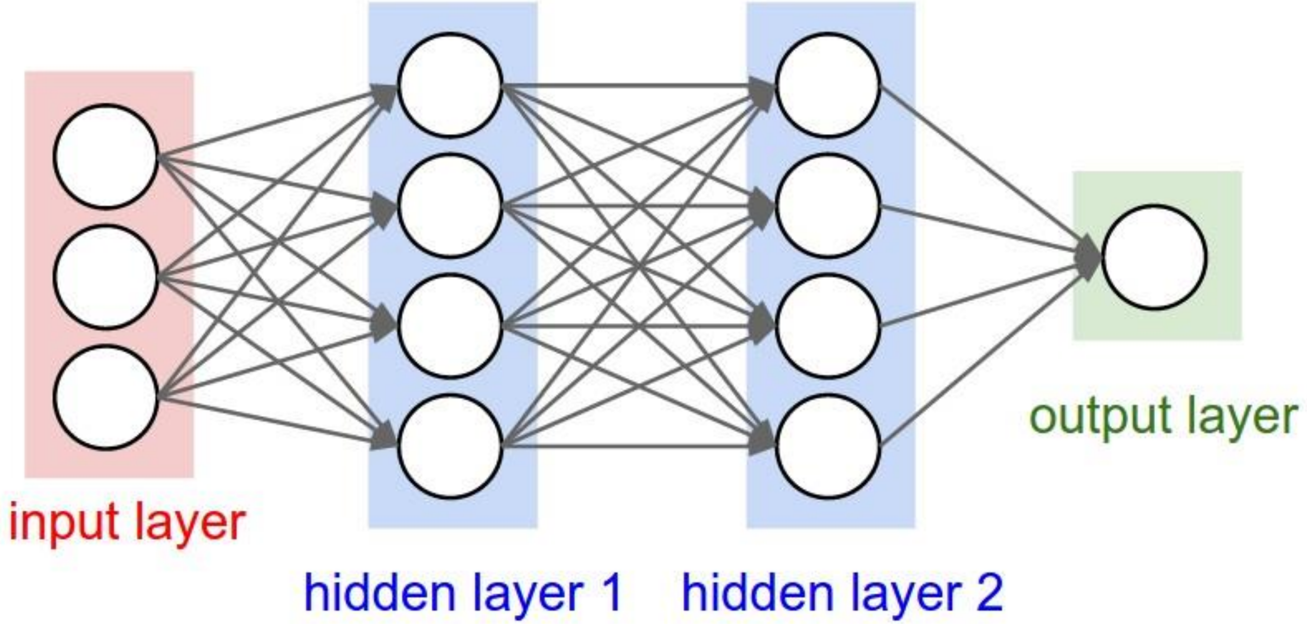








Neural networks



- A neuron detects some patterns in its inputs – combinations that cause it to fire
- When assembled into a network, neurons deep in the network react to patterns composed of more primitive parts

Demos and questions

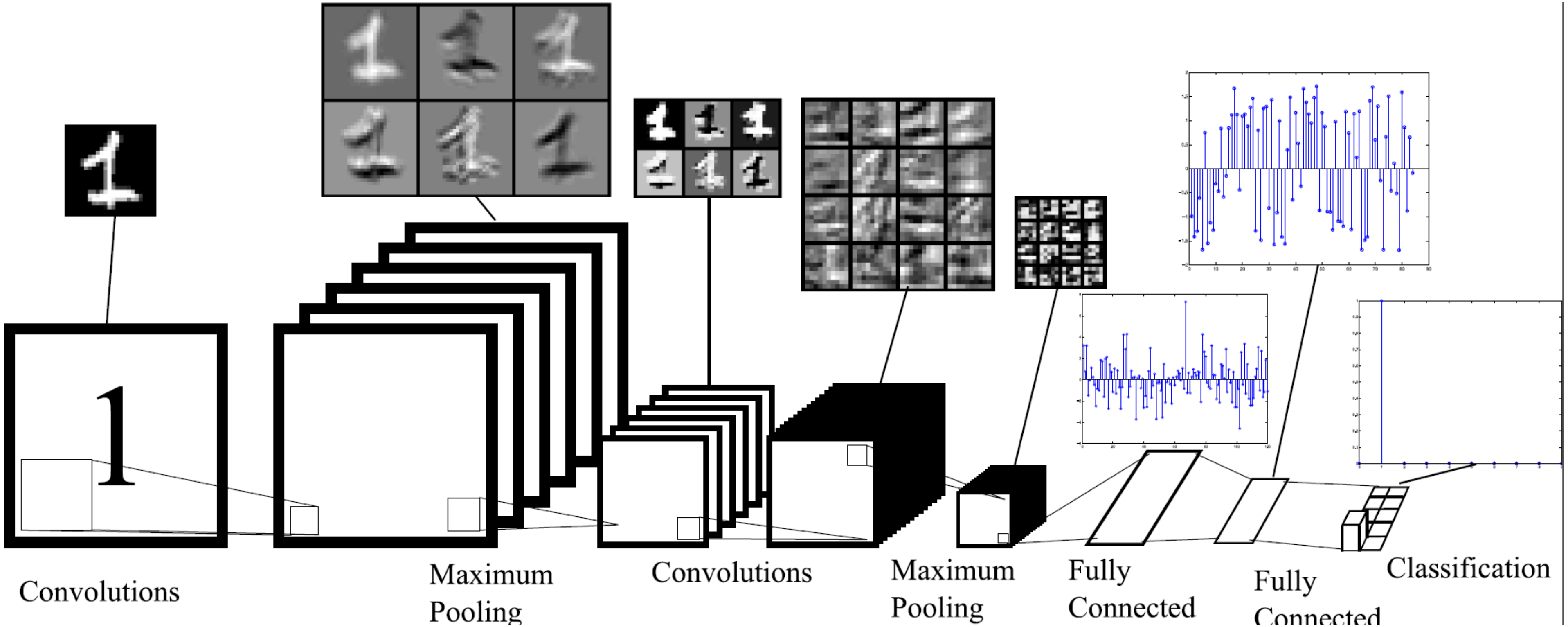
https://cs.stanford.edu/people/karpathy/convnetjs/

Q1: Is the nonlinearity necessary?

Q2: Is a net with a hidden layer really that different from a net with no hidden layers?

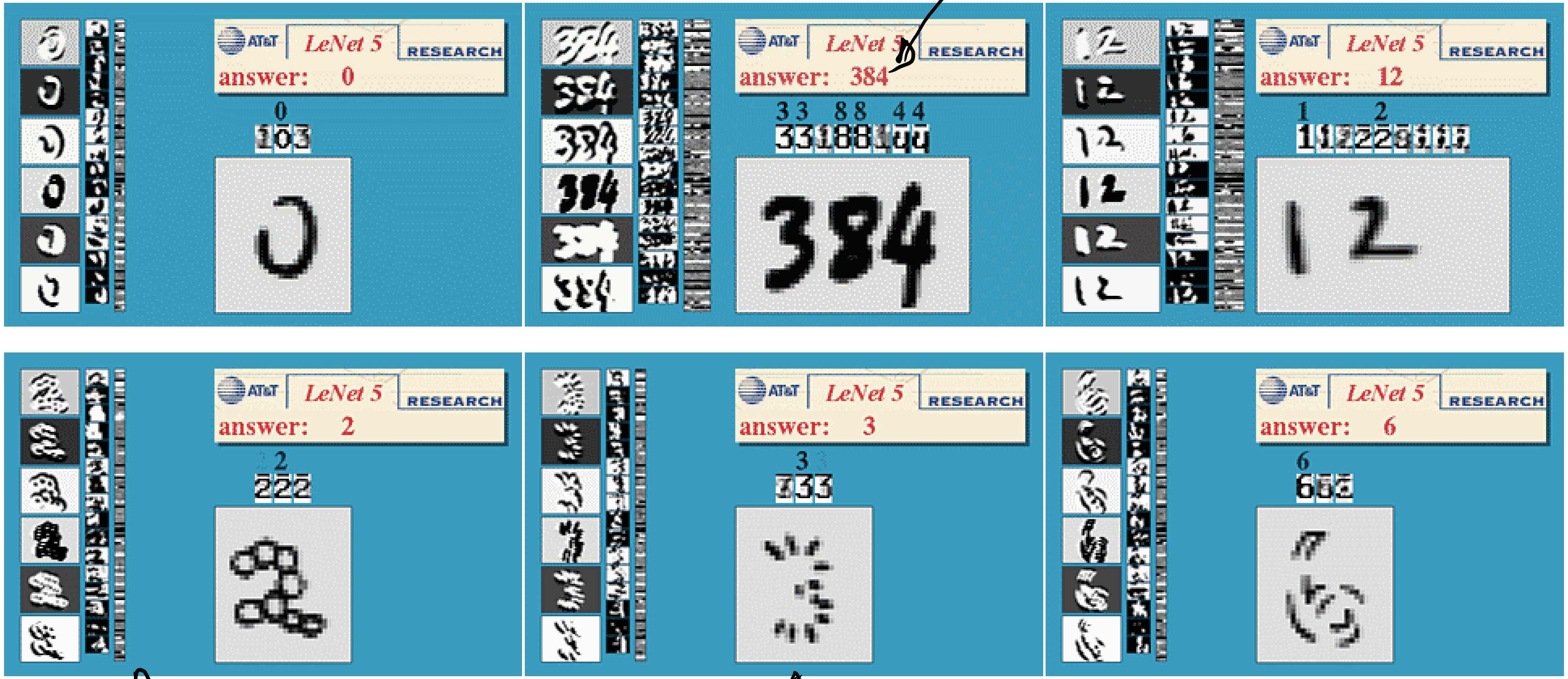
Convolutional networks (1998)

Best for images!



Convnets on digits

output.

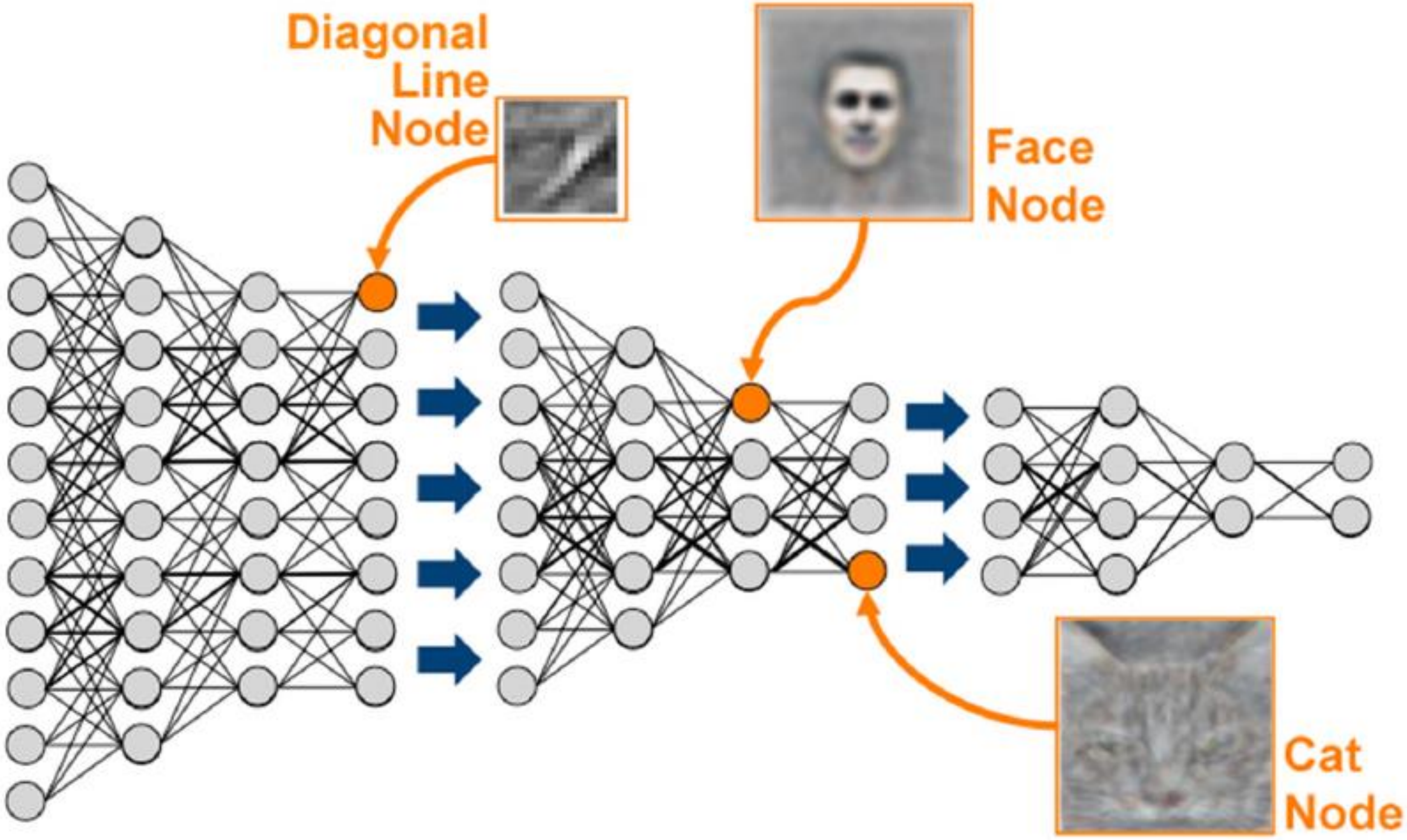


W1 W3 W5 input

Neural nets learn hierarchies!

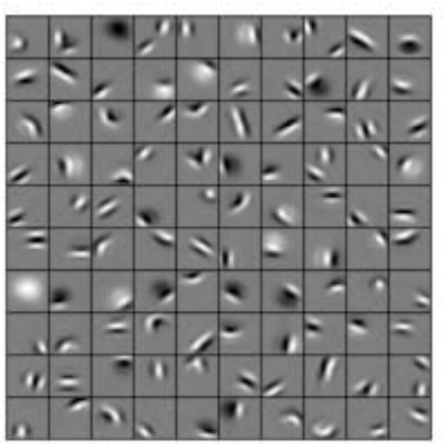
Google trained a network on YouTube videos. The net developed units detecting persons and cats!

Raw pixels!

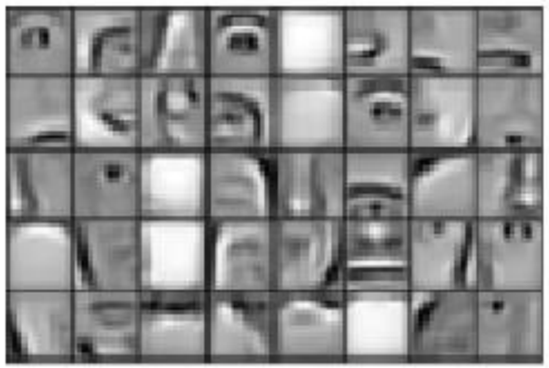


Q. Le et al. „Building high-level features using large scale unsupervised learning"

Neural nets learn hierarchies!



First layer



Second layer

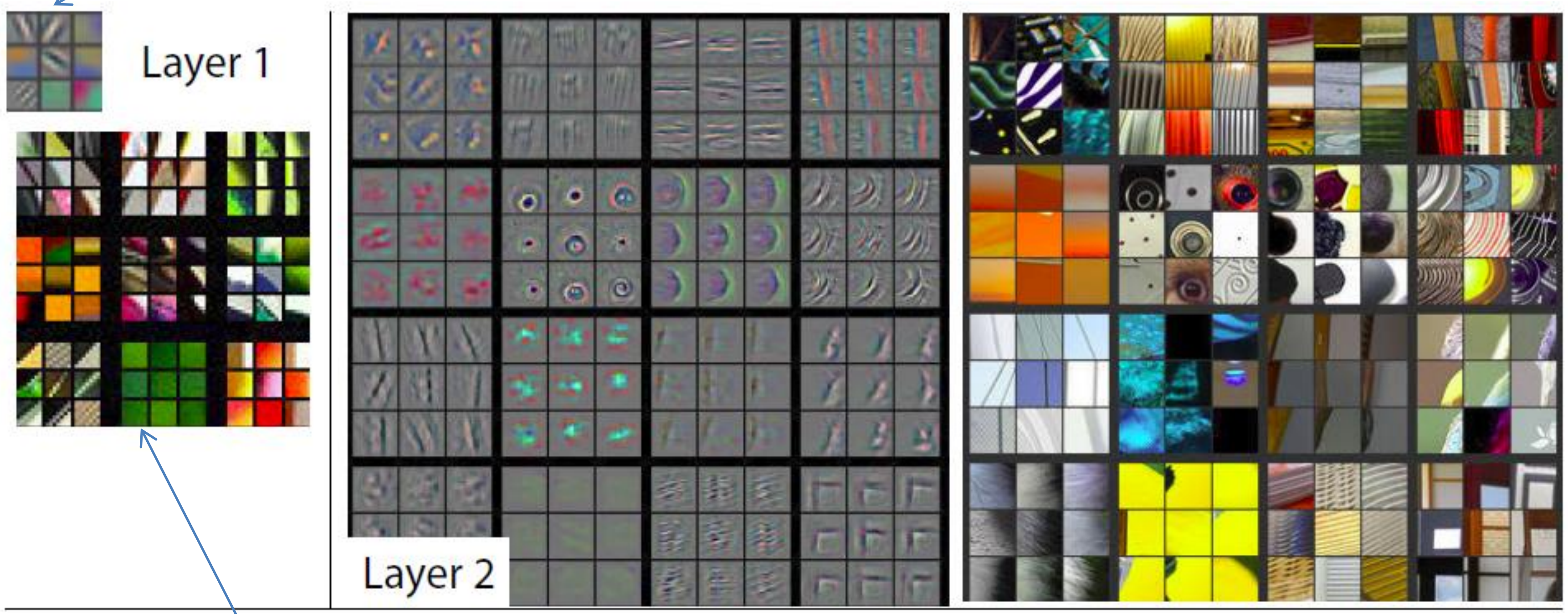
Third layer

Hierarchical features learned from a dataset of face images

(Lee et al., „Unsupervised Learning of Hierarchical Representations with Convolutional Deep Belief Networks”)

Low-level features

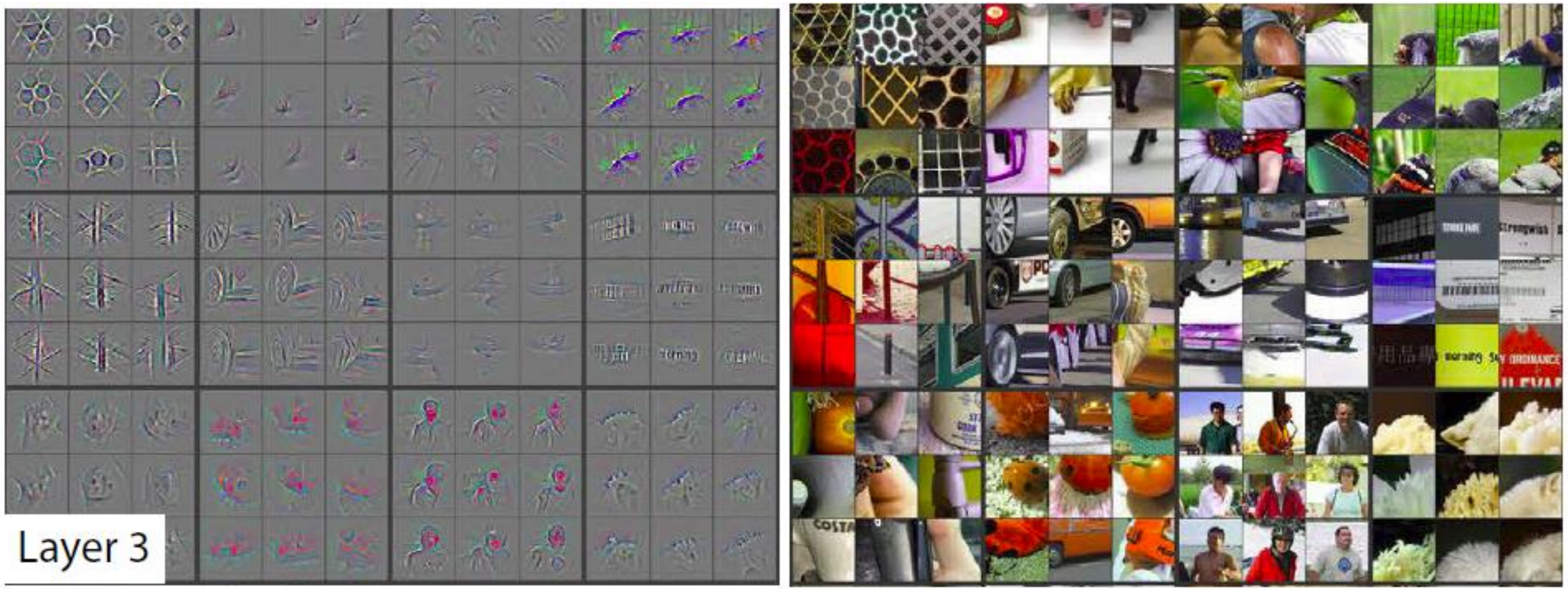
What the neuron (feature-detector looks for)



What images are selected by the neuron

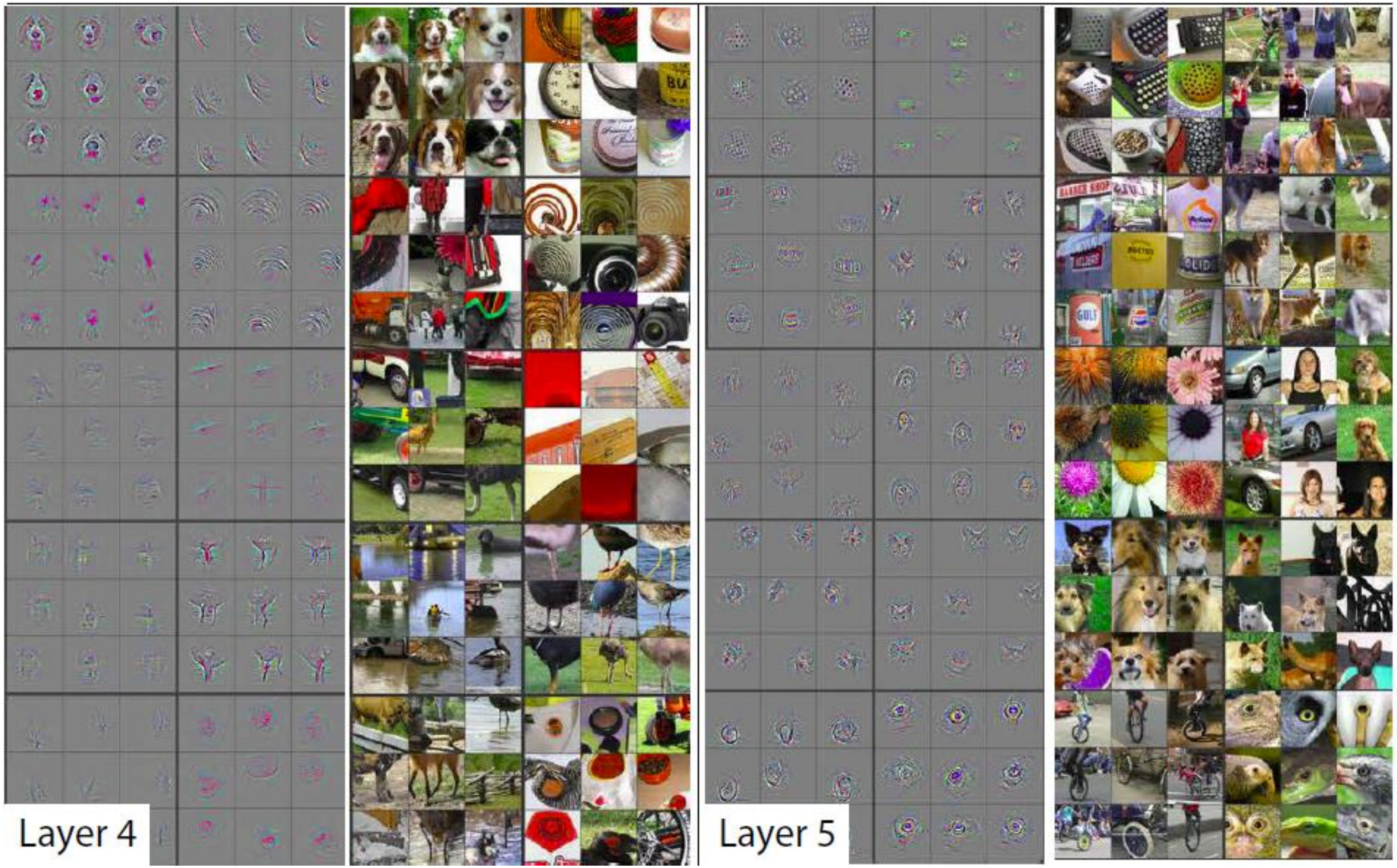
M. Zieler, “Visualizing and Understanding Convolutional Networks”

Mid-level features



M. Zieler, “Visualizing and Understanding Convolutional Networks”

High-level features

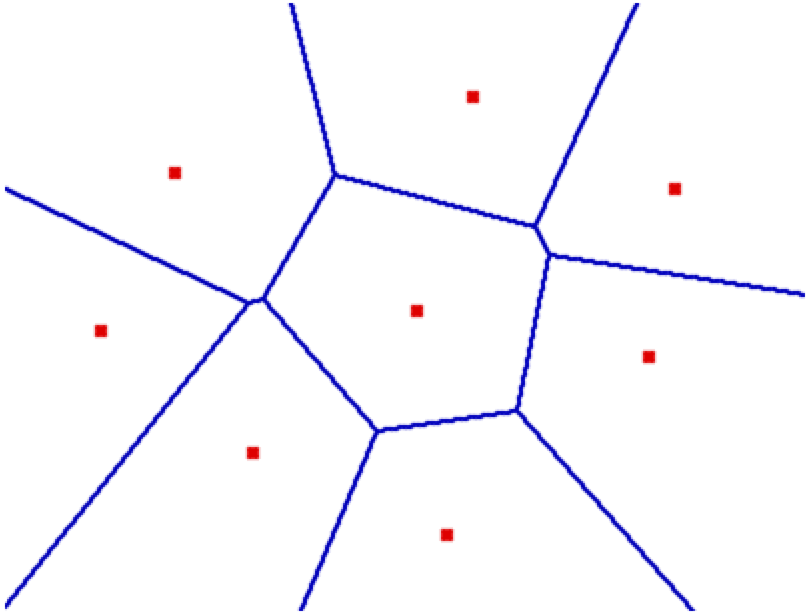


M. Zieler, “Visualizing and Understanding Convolutional Networks”

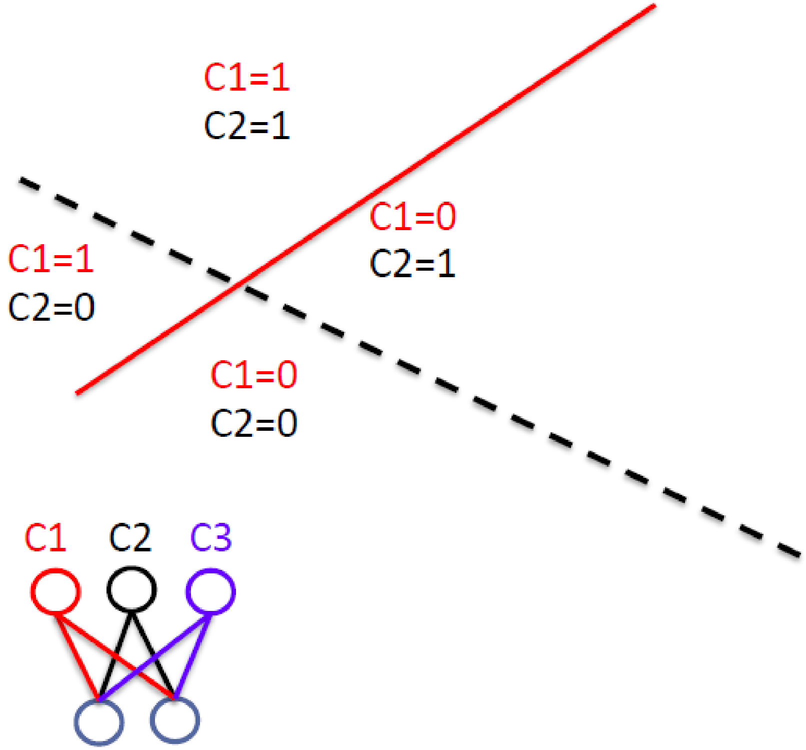
Depth is important

Nearest neighbors

- Look-up tables



Neural nets

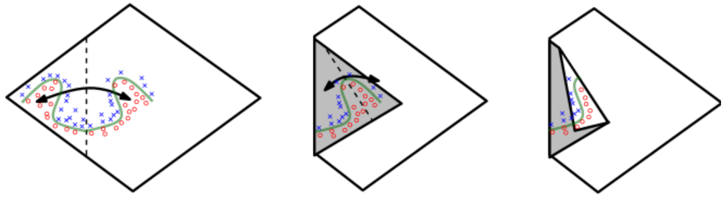


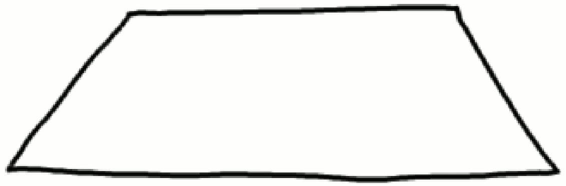
Bengio, 2009, Foundations and Trends in Machine Learning

Zalasky theorem (1975):

$$r(A_m) = \binom{m}{2} + m + 1$$

Deep nets





Guido Montufar, Razvan Pascanu, Kyunghyun Cho & Yoshua Bengio, *On the number of linear regions of Deep Neural Networks*, NIPS 2014

Deep Learning

Study of models that
solve tasks
in stages
whose exact function
emerges during training.

Deep Learning history: 1986

Learning representations by back-propagating errors

David E. Rumelhart*, Geoffrey E. Hinton†
& Ronald J. Williams*

* Institute for Cognitive Science, C-015, University of California, San Diego, La Jolla, California 92093, USA
† Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Philadelphia 15213, USA

We describe a new learning procedure, back-propagation, for networks of neurone-like units. The procedure repeatedly adjusts the weights of the connections in the network so as to minimize a measure of the difference between the actual output vector of the net and the desired output vector. As a result of the weight adjustments, internal 'hidden' units which are not part of the input or output come to represent important features of the task domain, and the regularities in the task are captured by the interactions of these units. The ability to create useful new features distinguishes back-propagation from earlier, simpler methods such as the perceptron-convergence procedure[1].

Internal „hidden" units which are not part of the input or output come to represent important features of the task domain

Neural networks are remarkably versatile

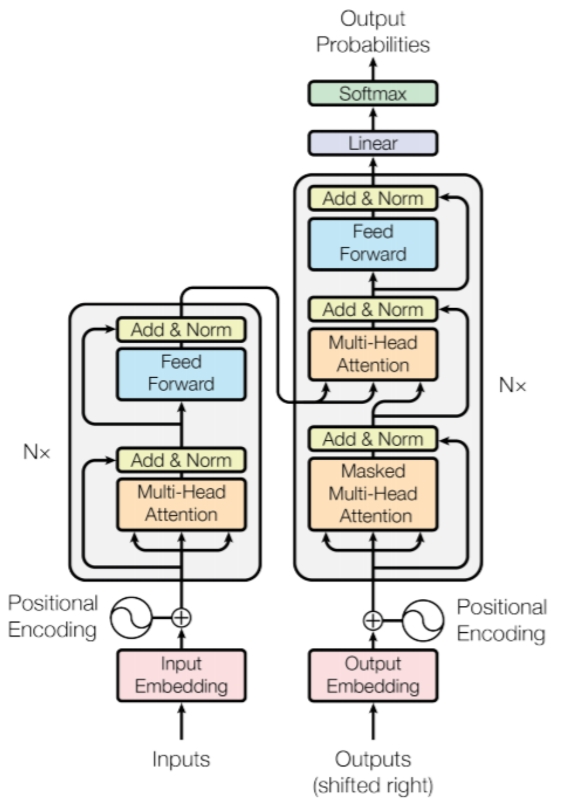


Figure 1: The Transformer - model architecture.

Attnetion is all you need,
https://arxiv.org/pdf/1706.03762.pdf

Translation (read sentence in language X, output in Y)

https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html

Neural networks are remarkably versatile

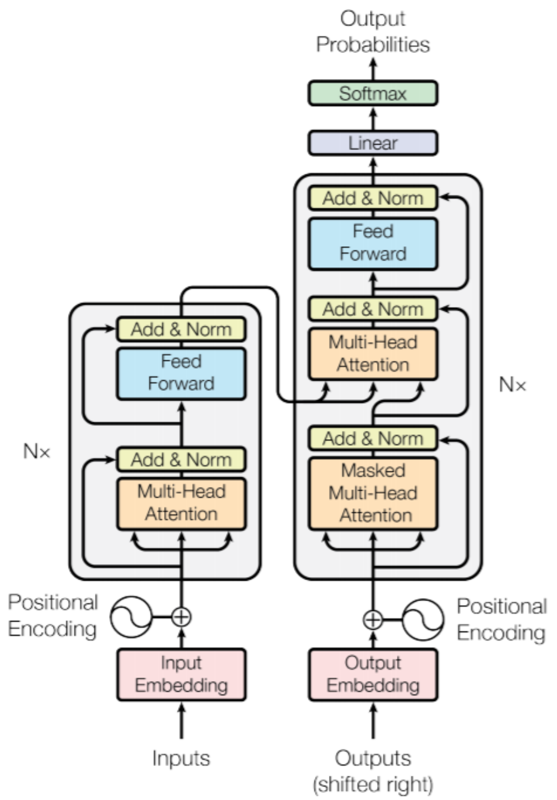


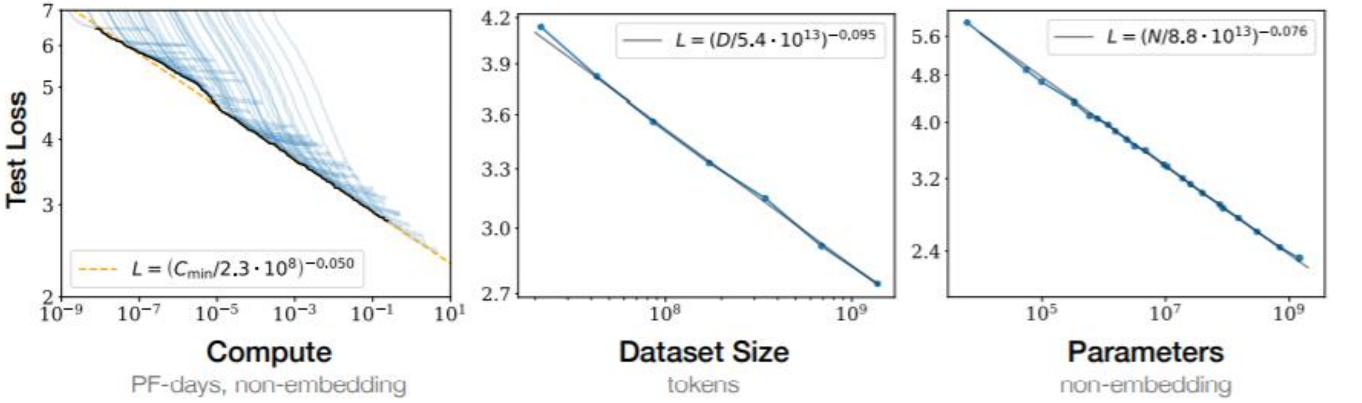
Figure 1: The Transformer - model architecture.

Attnetion is all you need, https://arxiv.org/pdf/1706.03762.pdf

- **Translation** (read sentence in language X, output in Y)
- **Text generation** (read all past words, output next)
- **Speech recognition** (read in waveform samples, output text)
- **Game playing** (read in previous screen pixels, output action)
- **Image generation** (read in previous pixels, output next)
- **Graph processing** (send messages between nodes over edges)

Sidenote: why now!?

- DL principles were known since a long time
- „Revolution" required:
  - Lots of data
  - Lots of compute
- It is as much a scientific, as engineering effort



Scaling Laws for Neural Language Models, arxiv.org/abs/2001.08361

- Still, awesome: backprop and gradient descent works better than anyone could hope

Goals for this course

1. Teach basics of NN implementation
2. Teach ConvNets
3. Teach about NNs in NLP, Speech
4. Show how to work without labels
5. Exotic topics: NNs on graphs, point-clouds, RL

Tentative course schedule

Weeks 2-4: How to properly train a deep net

Weeks 5-6: Convnets

Weeks 7-8: Recurrent Nets

Weeks 9-10: NLP

Weeks 11-12: Speech

Weeks 13-15: Unsupervised learning, other domains, maybe a bit of RL