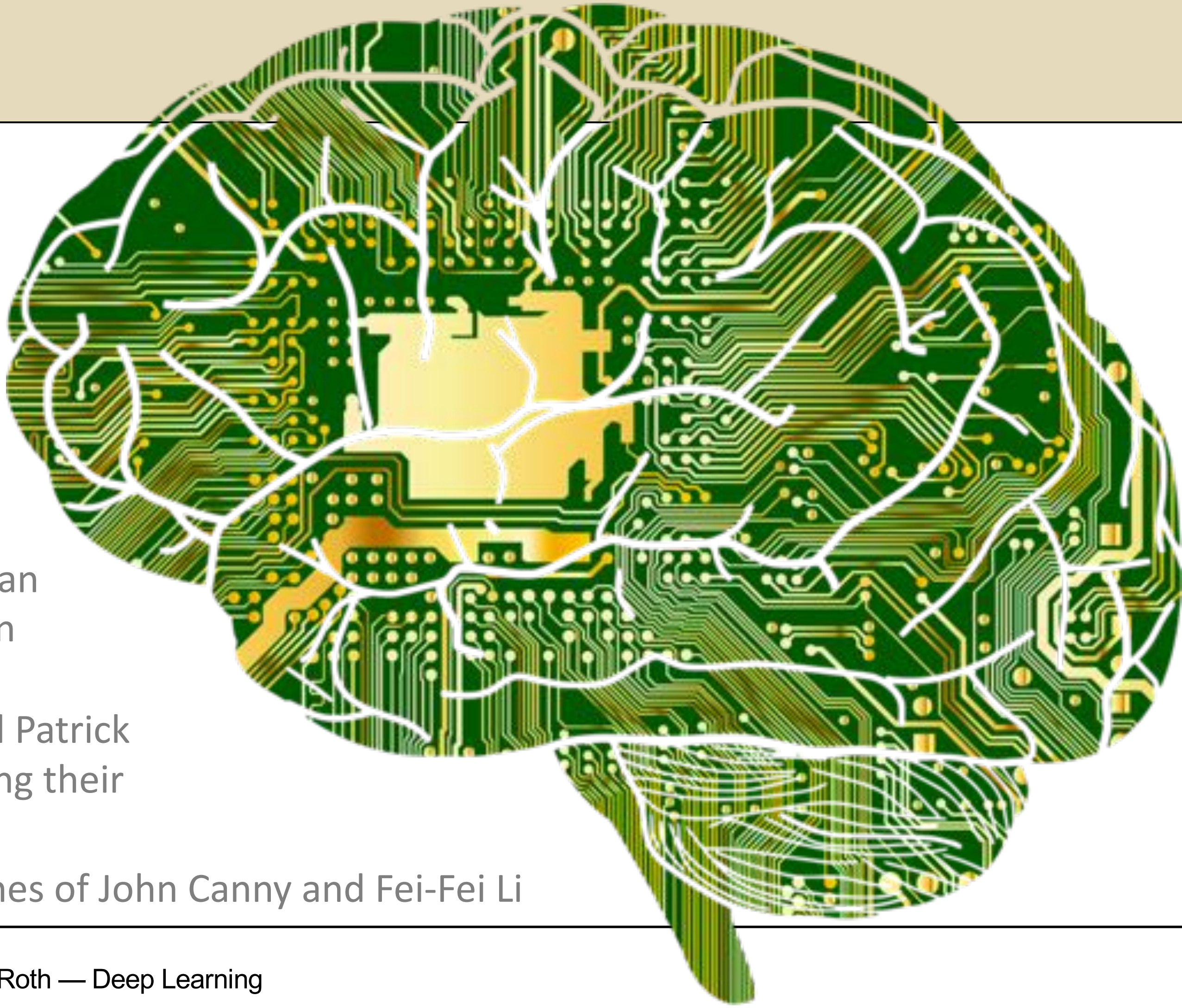
Deep Learning

Architectures and Methods

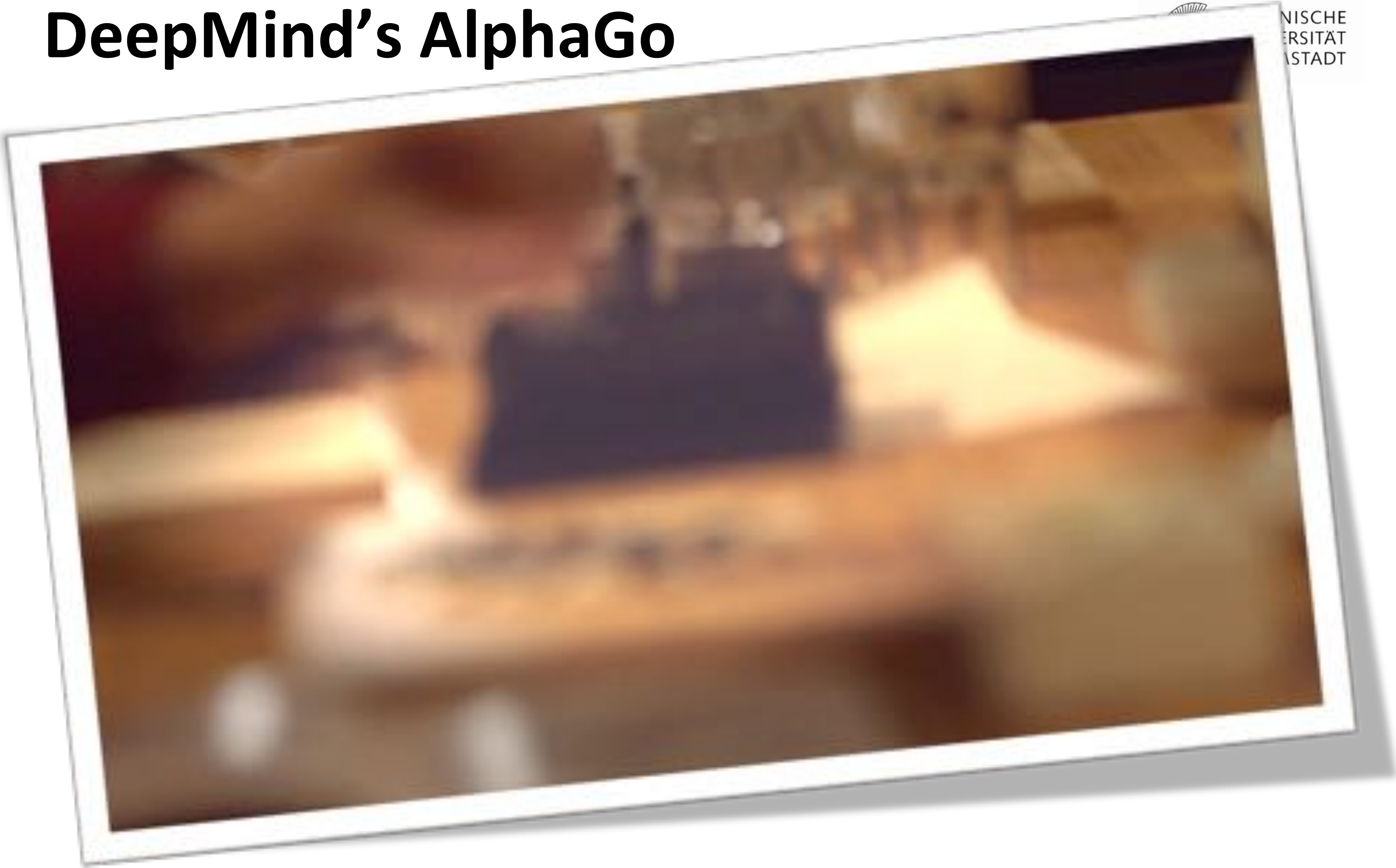
Introduction



Thanks to John Canny, Fei-Fei Li, Ian Goodfellow, Yoshua Bengio, Aaron Courville, Efstratios Gavves, Kirill Gavrilyuk, Berkay Kicanaoglu, and Patrick Putzky and many others for making their materials publically available.

The slides are mainly based on ones of John Canny and Fei-Fei Li

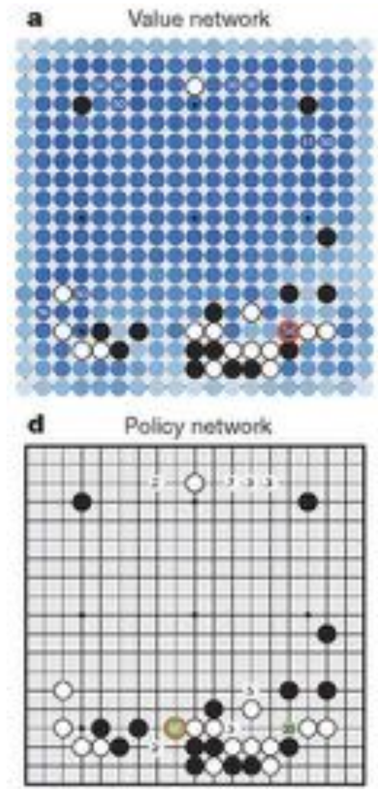
DeepMind's AlphaGo



Watch NATURE video at https://www.youtube.com/watch?v=g-dKXOlsf98

DeepMind’s AlphaGo

TECHNISCHE UNIVERSITÄT DARMSTADT



Deep policy network is trained to produce probability map of promising moves. The deep value network is used to prune the search tree (monte-carlo tree search); so there is a lot of classical AI machinery around the deep part.

And yes, the machine may also learn to play other games

Goal of Deep Architectures

High-level semenatical representations

To this aim most approaches use (stacked) neural networks

Edges, local shapes, object parts

Low level representation

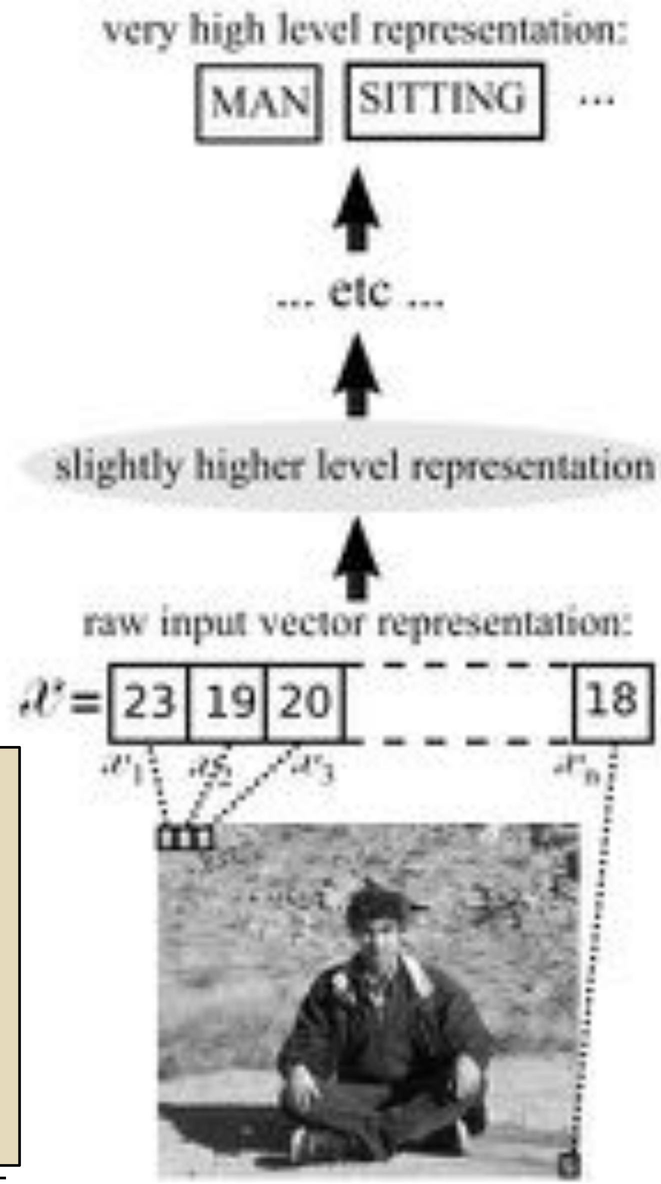


Figure is from Yoshua Bengio

Deep learning methods aim at

- **learning feature hierarchies**
- where features from higher levels of the hierarchy are formed by lower level features.

Deep Architectures

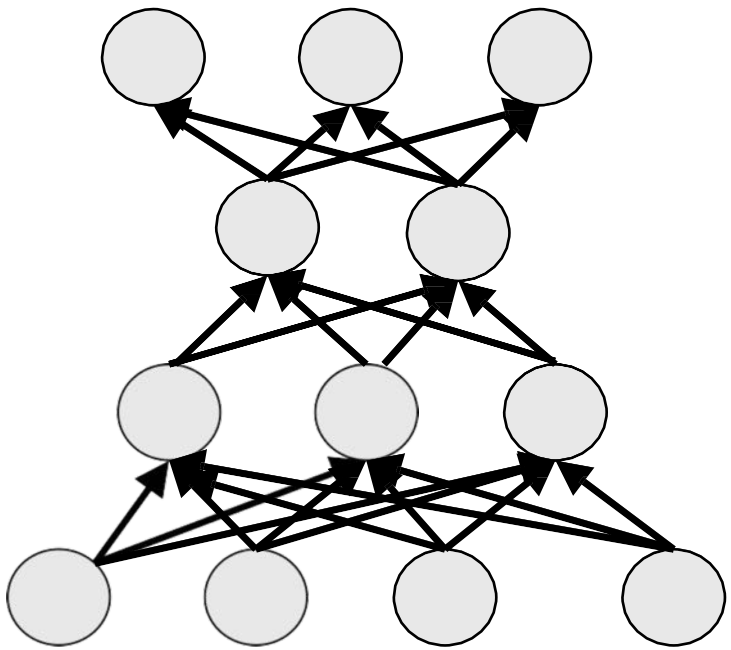
TECHNISCHE UNIVERSITÄT DARMSTADT

Deep architectures are composed of multiple levels of non-linear operations, such as neural nets with many hidden layers.

Output layer

Hidden layers

Input layer



Examples of non-linear activations:

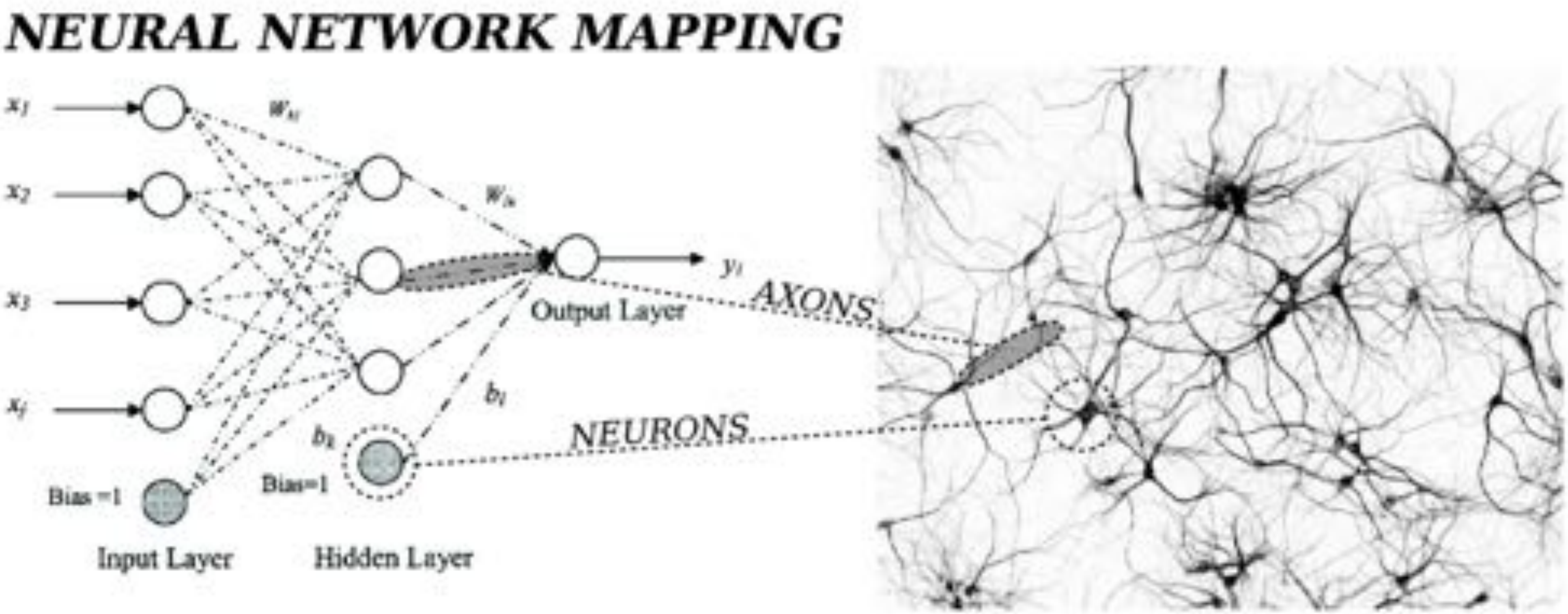
$$\tanh(x)$$

$$\sigma(x) = (1 + e^{-x})^{-1}$$

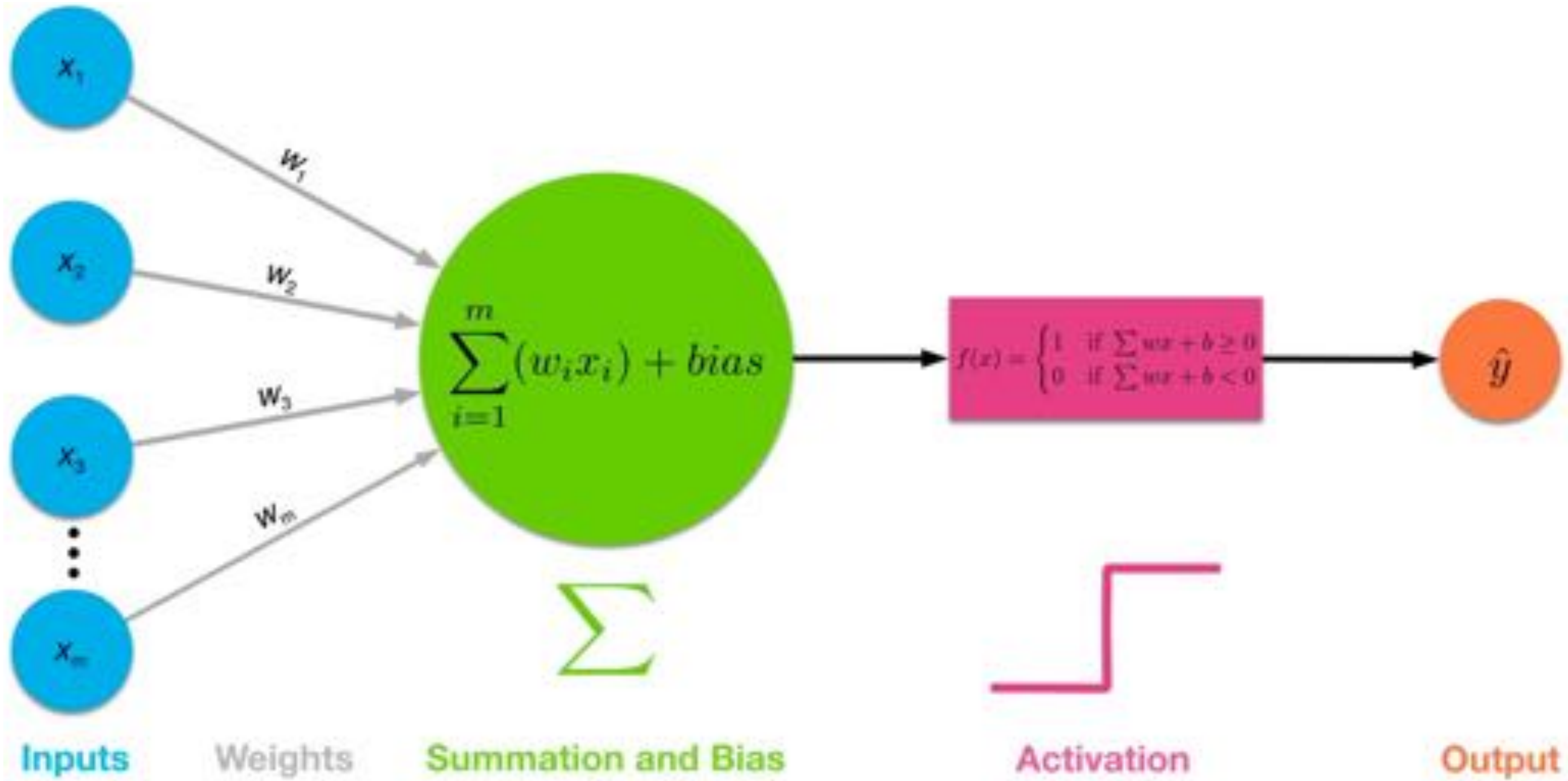
$$\max(0, x)$$

In practice, NN with multiple hidden layers work better than with a single hidden layer.

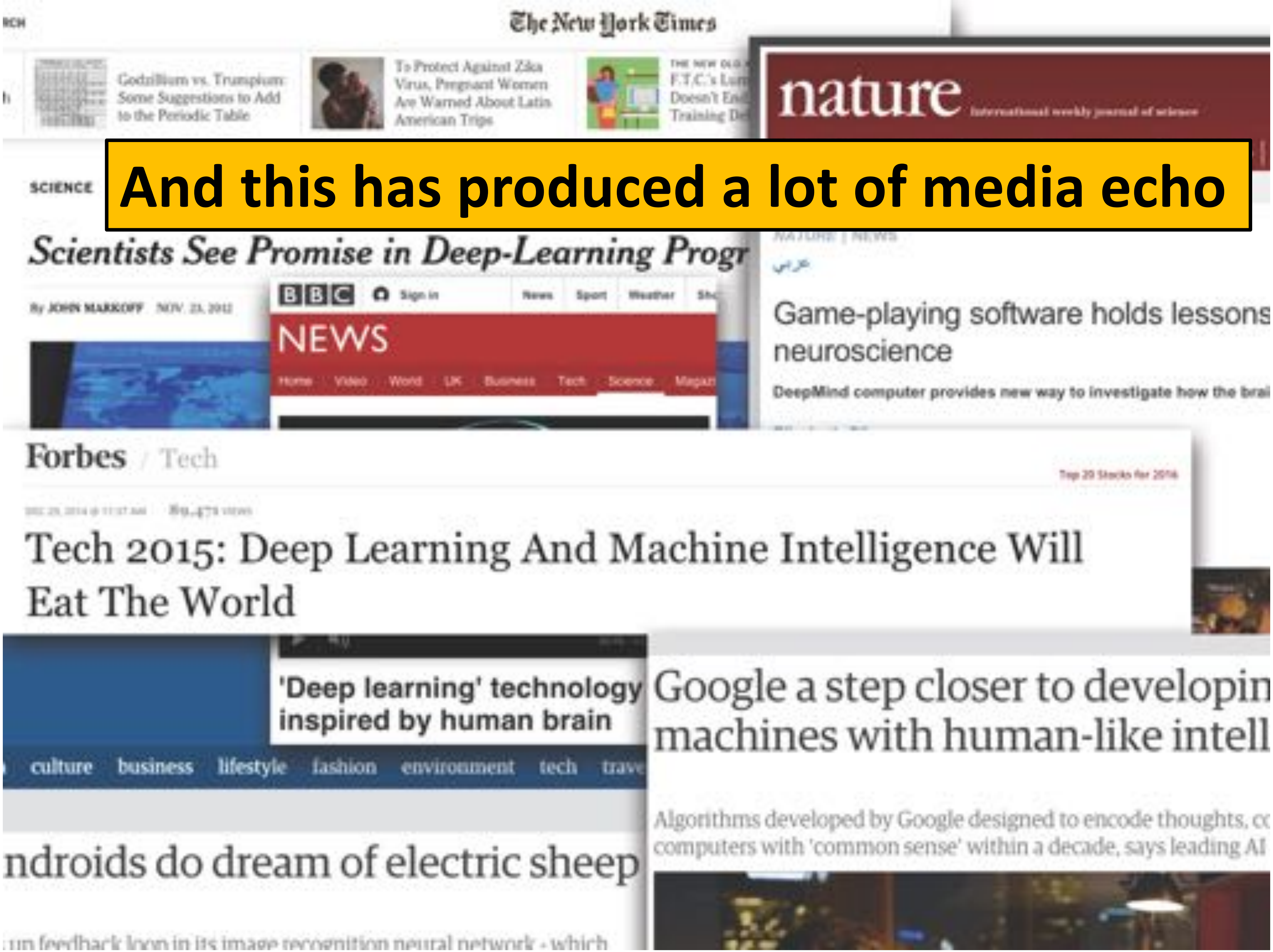
Artificial Neural Networks are inspired by neural networks



Abstract Neural Unit



And this has produced a lot of media echo



But this might not be the human way

TECHNISCHE UNIVERSITÄT DARMSTADT

Current Biology

Explore | Online Now | Current Issue | Archive | Journal Information | For Authors

< Previous Article | Volume 27, Issue 18, p2827–2832.e3, 25 September 2017 | Next Article >

REPORT

Humans, but Not Deep Neural Networks, Often Miss Giant Targets in Scenes

Miguel P. Eckstein, Kathryn Koehler, Lauren E. Welbourne, Emre Akbas

Switch to Standard View

Email Article

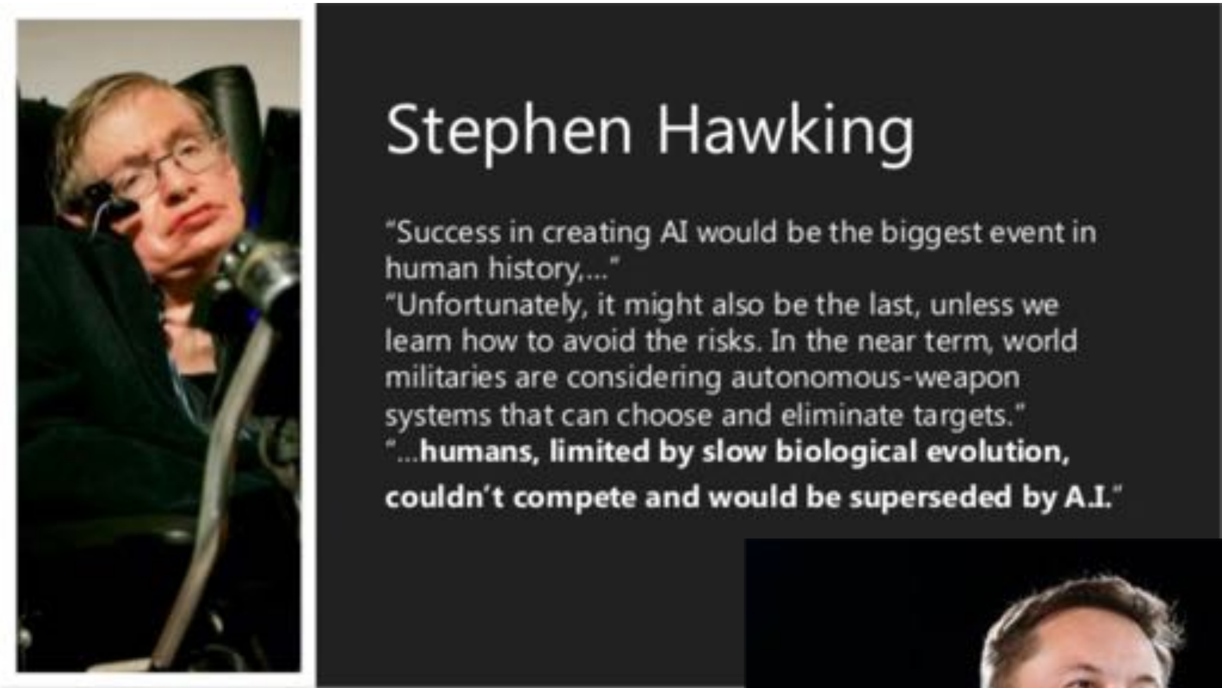
Add to My Reading List

Export Citation

Summary

Even with great advances in machine vision, animals are still unmatched in their ability to visually search complex scenes. Animals from bees [1, 2] to birds [3] to humans [4, 5, 6, 7, 8, 9, 10, 11, 12] learn about the statistical relations in visual environments to guide and aid their search for targets. Here, we investigate a novel manner in which humans utilize rapidly acquired information about scenes by guiding search toward likely target sizes. We show that humans often miss targets when their size is inconsistent with the rest of the scene, even when the targets were made larger and more salient and observers fixated the target. In contrast, we show that state-of-the-art deep neural networks do not exhibit such deficits in finding mis-scaled targets but, unlike humans, can be fooled by target-shaped distractors that are inconsistent with the expected target's size within the scene. Thus, it is not a human deficiency to miss targets when they are inconsistent in size with the scene; instead, it is a byproduct of a useful strategy that the brain has implemented to rapidly discount potential distractors.

Anyhow, deep neural learning fueled the discussion on AI



Artificial Intelligence and Humans

"How do we humans get so much from so little?" and by that I mean how do we acquire our understanding of the world given what is clearly by today's engineering standards so little data, so little time, and so little energy.

Josh Tenenbaum
“Bayesian Program Learning”

Lake, Salakhutdinov, Tenenbaum, Science 350 (6266), 1332-1338, 2015
Tenenbaum, Kemp, Griffiths, Goodman, Science 331 (6022), 1279-1285, 2011

Systems AI: the computational and mathematical modeling of complex AI systems.

TensorFlow™

PYTORCH

Deep Learning with PyTorch

OpenAI

APACHE mxnet™

Systems AI: the computational and mathematical modeling of complex AI systems.

Eric Schmidt, Executive Chairman, Alphabet Inc.: Just Say "Yes", Stanford Graduate School of Business, May 2, 2017.https://www.youtube.com/watch?v=vbb-AjiXyh0. But also see e.g. Soica et al. "A Berkeley View of Systems Challenges for AI", Technical Report No. UCB/EECS-2017-159, Oct. 16, 2017

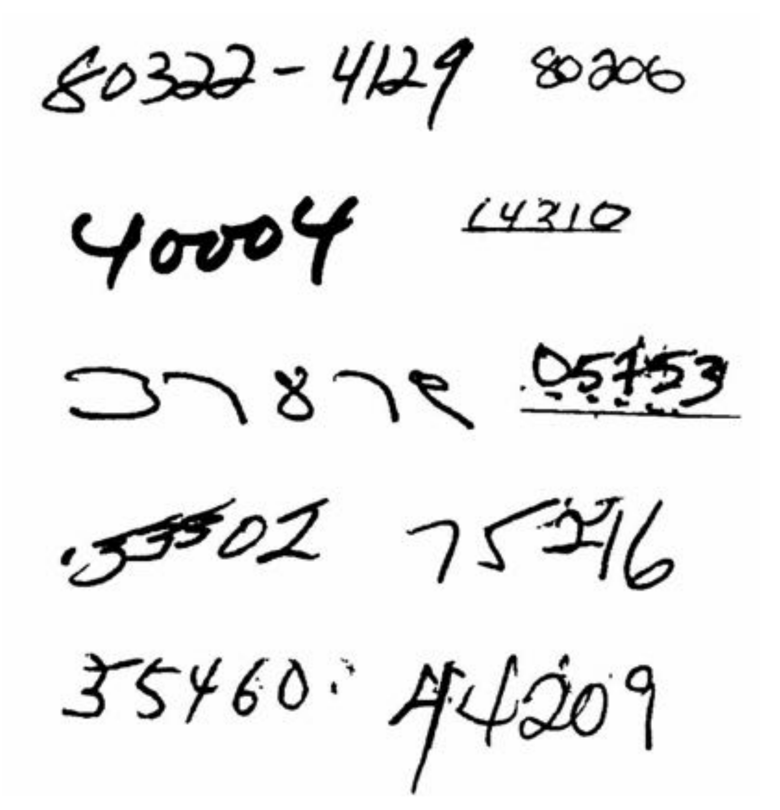
Deep Learning: Hype or Hope?

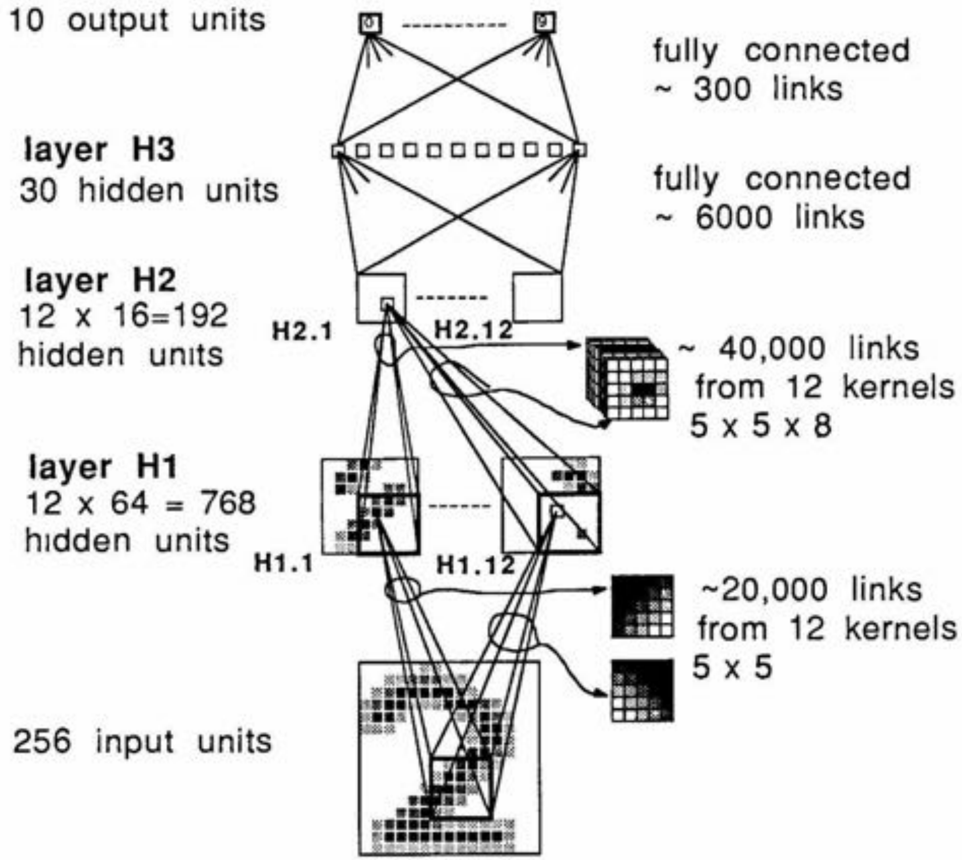
Hype: (n) “extravagant or intensive publicity or promotion”

Hope: (n) “expectation of fulfillment or success”

Milestones: Digit Recognition

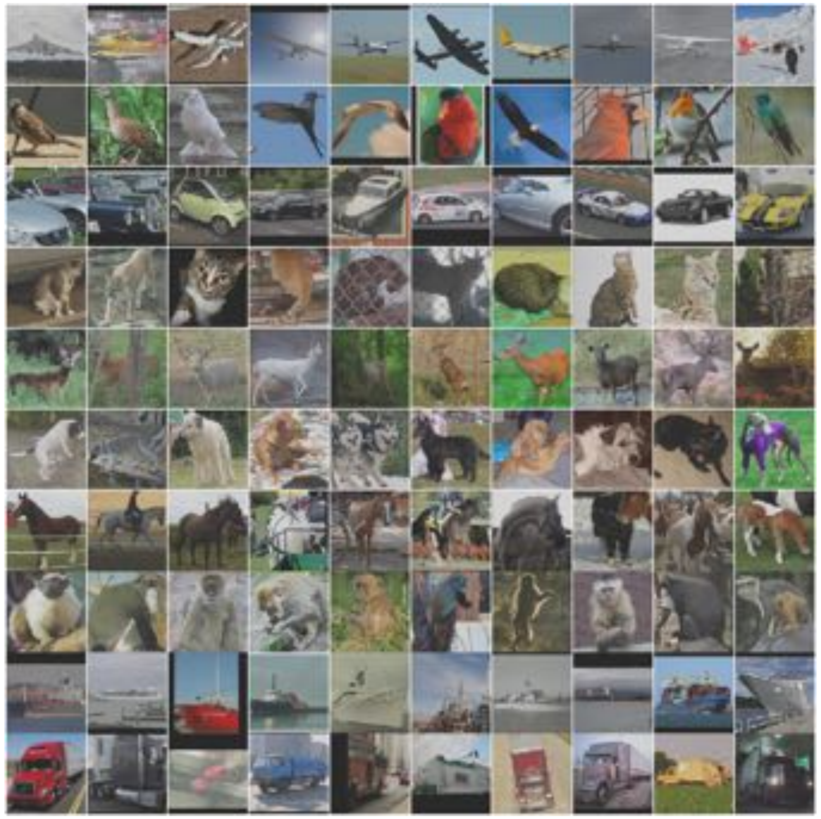
LeNet 1989: recognize zip codes, Yann Lecun, Bernhard Boser and others, ran live in US postal service

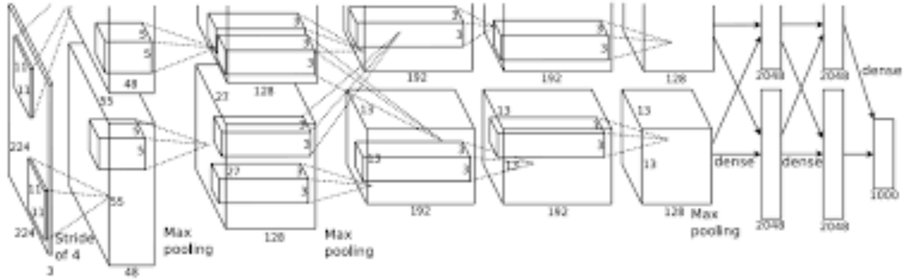


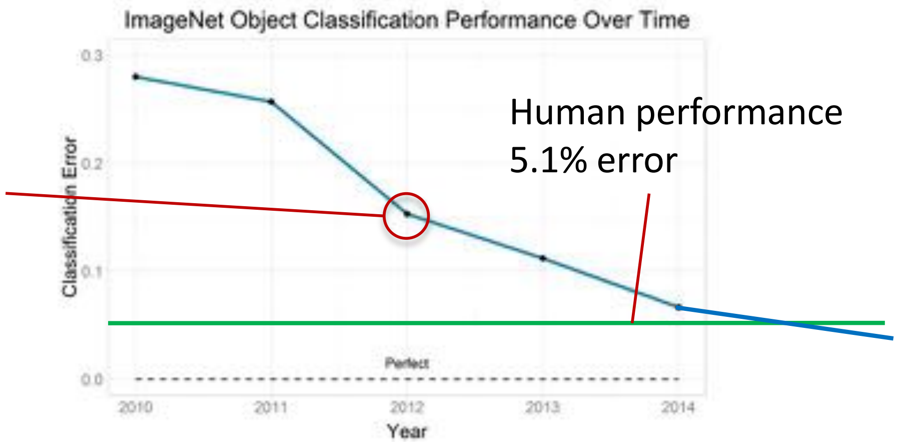


Milestones: Image Classification

Convolutional NNs: AlexNet (2012): trained on 200 GB of ImageNet Data

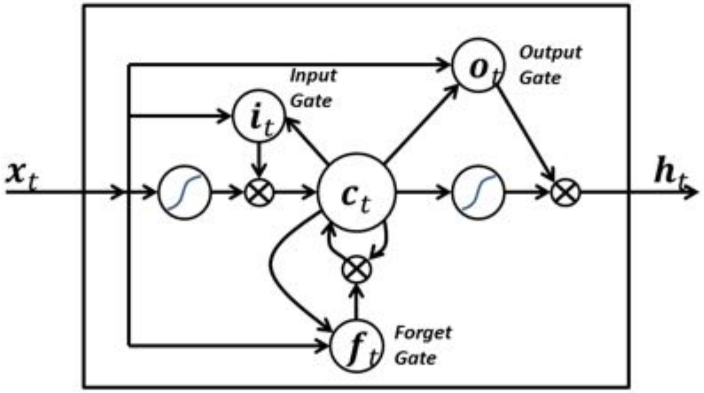


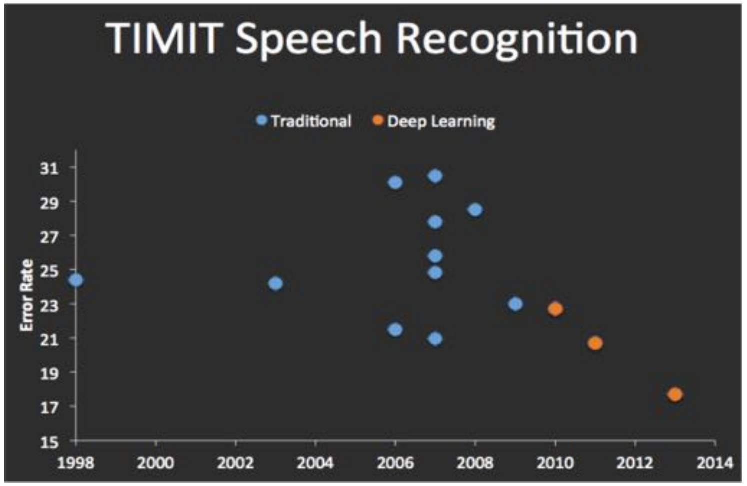


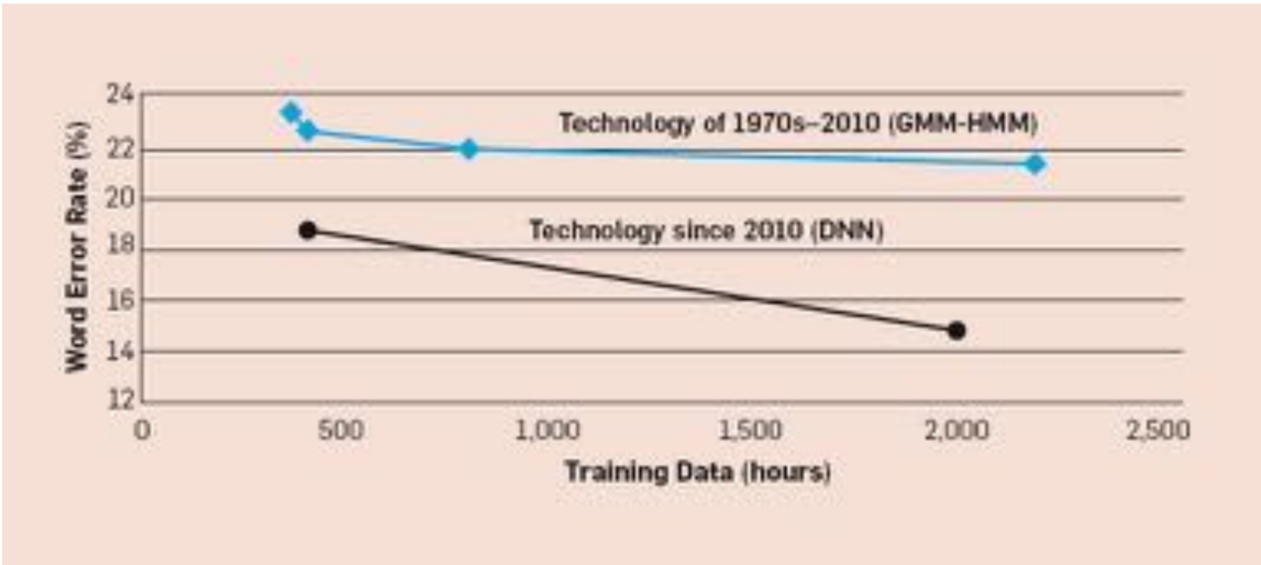


Milestones: Speech Recognition

Recurrent Nets: LSTMs (1997):

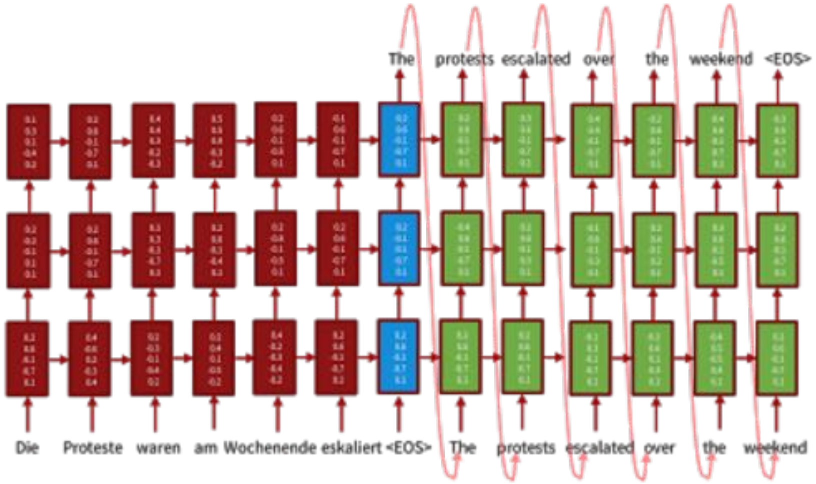


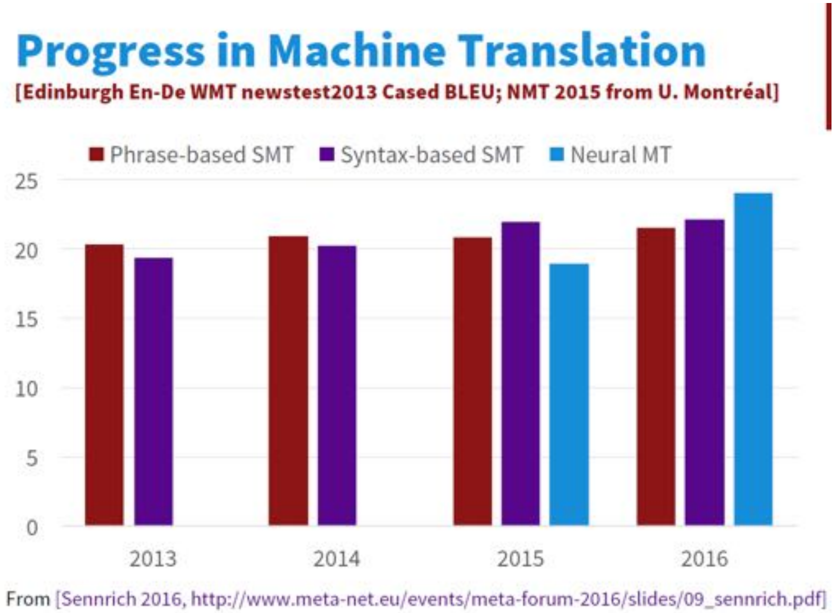




Milestones: Language Translation

Sequence-to-sequence models with LSTMs and attention:

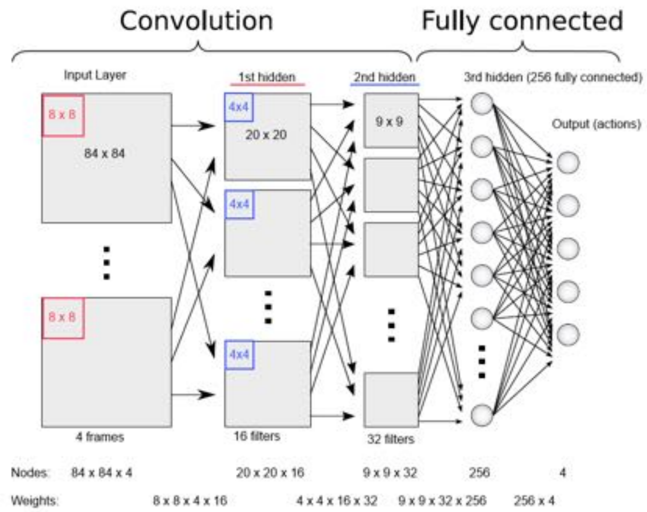


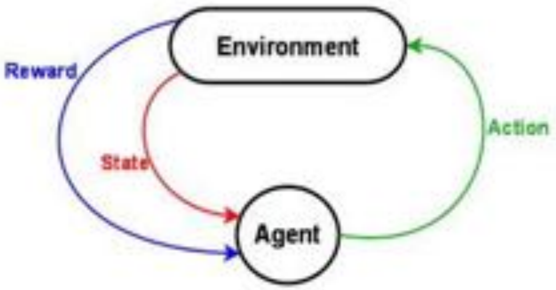


Source Luong, Cho, Manning ACL Tutorial 2016.

Milestones: Deep Reinforcement Learning

In 2013, Deep Mind's arcade player bests human expert on six Atari Games. Acquired by Google in 2014.





In 2016, Deep Mind's alphaGo defeats former world champion Lee Sedol

Deep Learning: Is it Hype or Hope?

Deep Learning: Is it Hype or Hope?

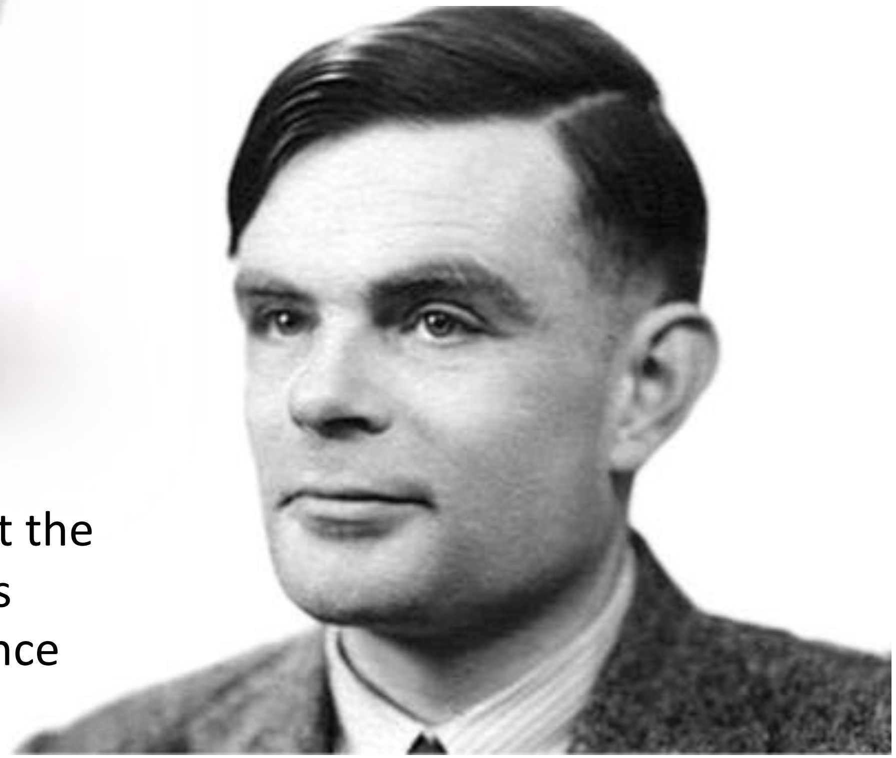
Yes!

But …

Godfathers of Deep Learning honored with Turing Award 2019

Yann LeCun, Geoffrey Hinton, and Yoshua Bengio

Turing Award = Nobel Prize for Computing



Named after Alan Turing, a British mathematician at the University of Manchester. Turing is often credited as being the key founder of theoretical computer science and AI.

Learning about Deep Neural Networks

Yann Lecun quote: **DNNs require: “an interplay between intuitive insights, theoretical modeling, practical implementations, empirical studies, and scientific analyses”**

I.e. there isn’t a framework or core set of principles to explain everything (c.f. graphical models for machine learning) yet. **We try to cover the ground in Lecun’s quote.**

This Course

Goals:

- Introduce deep learning to a broad audience.
- Review principles and techniques for understanding deep networks.
- Develop skill at designing networks for applications.

Other Deep Learning courses at TU

Deep Learning for NLP, UKP

Analyzing Software using Deep Learning, Software Lab

Deep Learning for Medical Imaging, GRIS

Deep Generative Models, IGD

Project Lab Deep Learning in Computer Vision, VisInf

…

This Course: a flipped classroom experiment

We will mainly follow John Canny’s (UC Berkeley) course

CS294-129 Designing, Visualizing and Understanding Deep Neural Networks

https://bcourses.berkeley.edu/courses/1453965/pages/cs294-129-designing-visualizing-and-understanding-deep-neural-networks

and Fei-Fei Li‘s (Stanford) course

CS231n: Convolutional Neural Networks for Visual Recognition

http://cs231n.stanford.edu/

This will be done as a flipped classroom experiment: **videos are used to 'deliver content' outside of the classroom**

This Course: a flipped classroom experiment

You will **watch the next video lecture prior to the lecture**, **prepare a list of 5 questions as main homework for that lecture**, and then we will try to **answer your question in class as a group**.

- We may also briefly recap the slides and also present our own stuff without videos. We may even ask you to read papers
- Next to the “5-questions” homework, there might also be textual exercises and you can propose and run a deep learning project.

This Course

We will check the bonus option. If so, roughly:

- Class Participation: 30%
- Questions: 30%
- Final Project (in groups): 40%

Audience: CS students

Final exam will be a written exam

Logistics

- Course Number: 20-00-1034-iv Summer 2020, TU
- Instructor: Kersting
- Time: Tuesday 11:40-13:20
- Location: S105/122
- Exercice: Wednesday 11:40-13:20 in S202/C205

Course Project

- More info later
- 2-pages expose to get a deal on a project
- Encourage “open-source” projects that can be archived somewhere.
- Talk also to other groups at TUDa

Tentative Outline

1. Intro, Computer Vision History, Classification, KNN
2. Linear Classification, Feature Selection, Optimization, Stochastic Gradient, Backpropagation
3. Training DNNs: Activation functions, initialization, gradient flow, batch normalization, parameter updates, ensembes, droupout
4. Convolutional Neural Networks
5. Recurent Networks, LSTMs

Tentative Outline

6. Deep NLP
7. Variational Autoencoder
8. Deep Probabilistic Models
9. Interpreting DNNs
10. DNNs for Games
11. Deep Reinforcement Learning
12. Project Presentations (prefinal, in the exercise session)
13. Final Project Presentations
14. Final Project Presentations

Some History

- Reading: the Deep Learning Book, Introduction

Phases of Neural Network Research

- 1940s-1960s: Cybernetics: Brain like electronic systems, morphed into modern control theory and signal processing.
- 1960s-1980s: **D**igital computers, automata theory, computational complexity theory: simple shallow circuits are very limited…
- 1980s-1990s: Connectionism: complex, non-linear networks, back-propagation.
- 1990s-2010s: Computational learning theory, graphical models: Learning is computationally hard, simple shallow circuits are very limited…
- 2006→: Deep learning: End-to-end training, large datasets, explosion in applications.

Citations of the “LeNet” paper

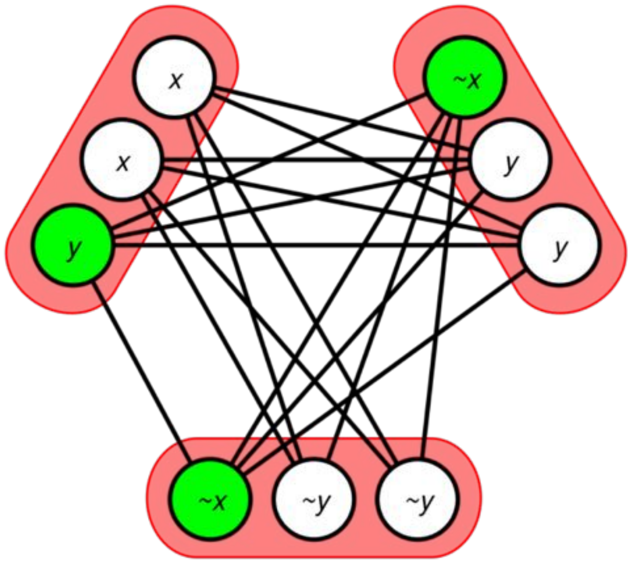
- Recall the LeNet was a modern visual classification network that recognized digits for zip codes. Its citations look like this:

Second phase Deep Learning “Winter” Third phase

- The 2000s were a golden age for machine learning, and marked the ascent of graphical models. But not so for neural networks.

Why the success of DNNs is surprising

- From both complexity and learning theory perspectives, simple networks are very limited.
  - Can’t compute parity with a small network.
  - NP-Hard to learn “simple” functions like 3SAT formulae, and i.e. training a DNN is NP-hard.



Why the success of DNNs is surprising

- The most successful DNN training algorithm is a version of gradient descent which will only find local optima. In other words, it's a greedy algorithm. Backprop:

$$\text{loss} = f(g(h(y)))$$

$$d\,\text{loss}/dy = f'(g) \times g'(h) \times h'(y)$$

- Greedy algorithms are even more limited in what they can represent and how well they learn.

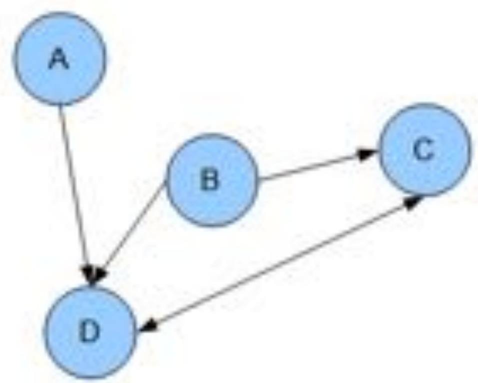
- If a problem has a greedy solution, its regarded as an "easy" problem.

Why the success of DNNs is surprising

- In graphical models, values in a network represent random variables, and have a clear meaning. The network structure encodes dependency information, i.e. you can represent rich models.

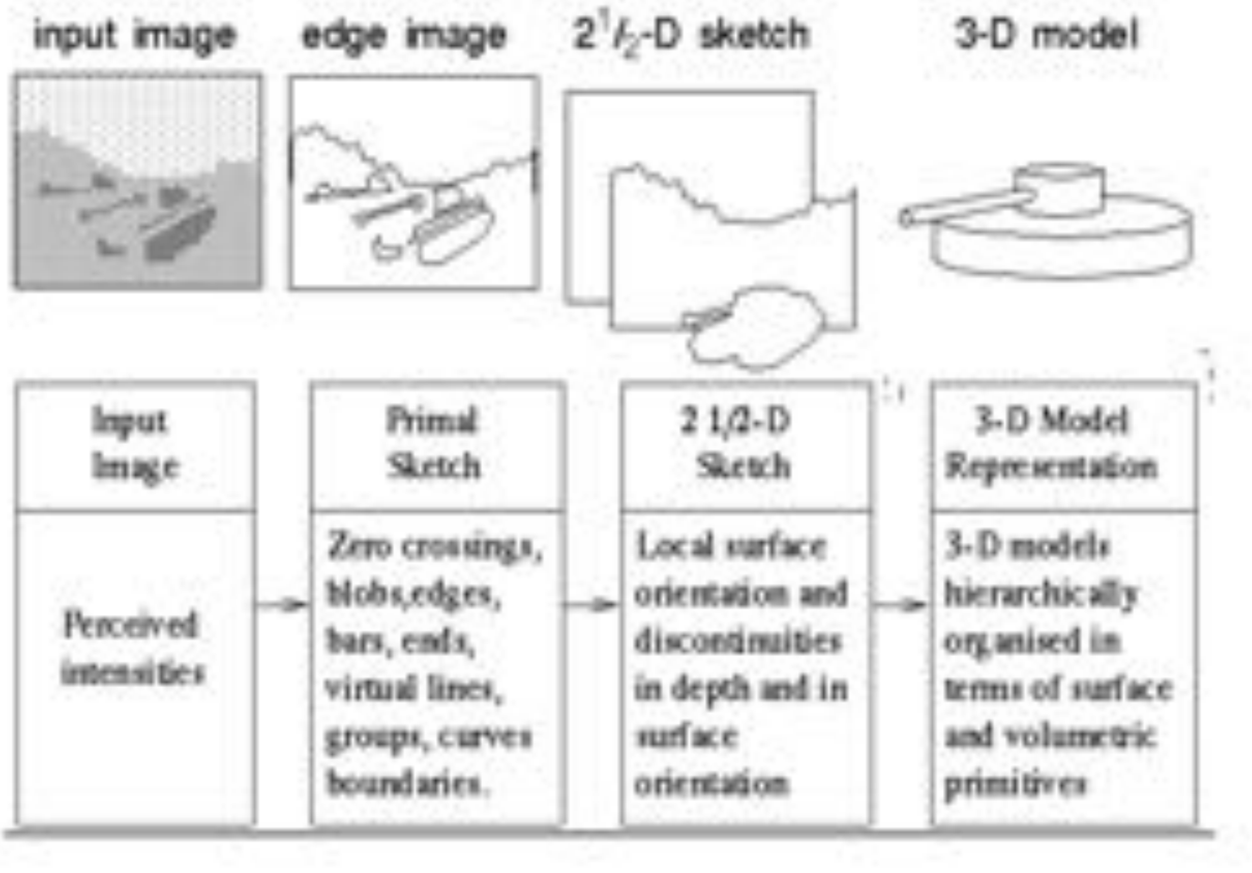
- In a DNN, node activations encode nothing in particular, and the network structure only encodes (trivially) how they derive from each other.

But more on the connection between DNNs and PGMs later in class



Why the success of DNNs is ~~surprising~~ obvious

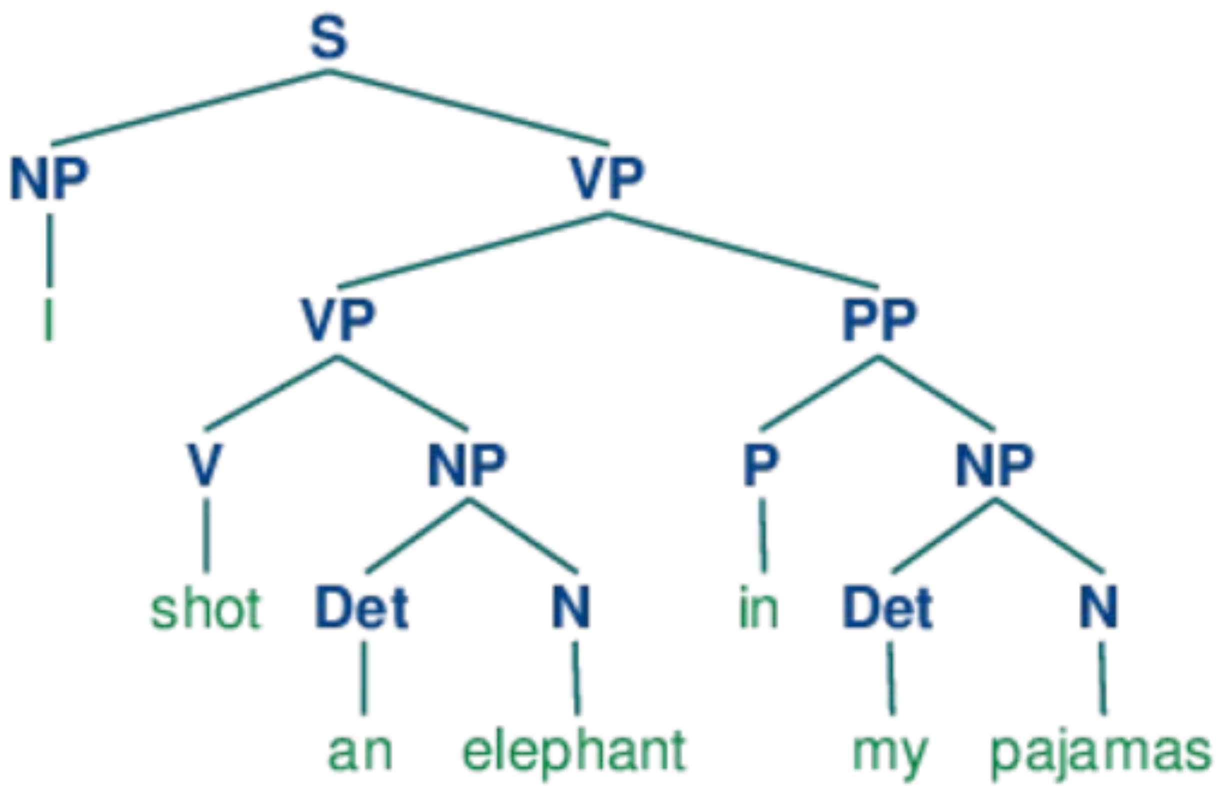
- Hierarchical representations are ubiquitous in AI. Computer vision:



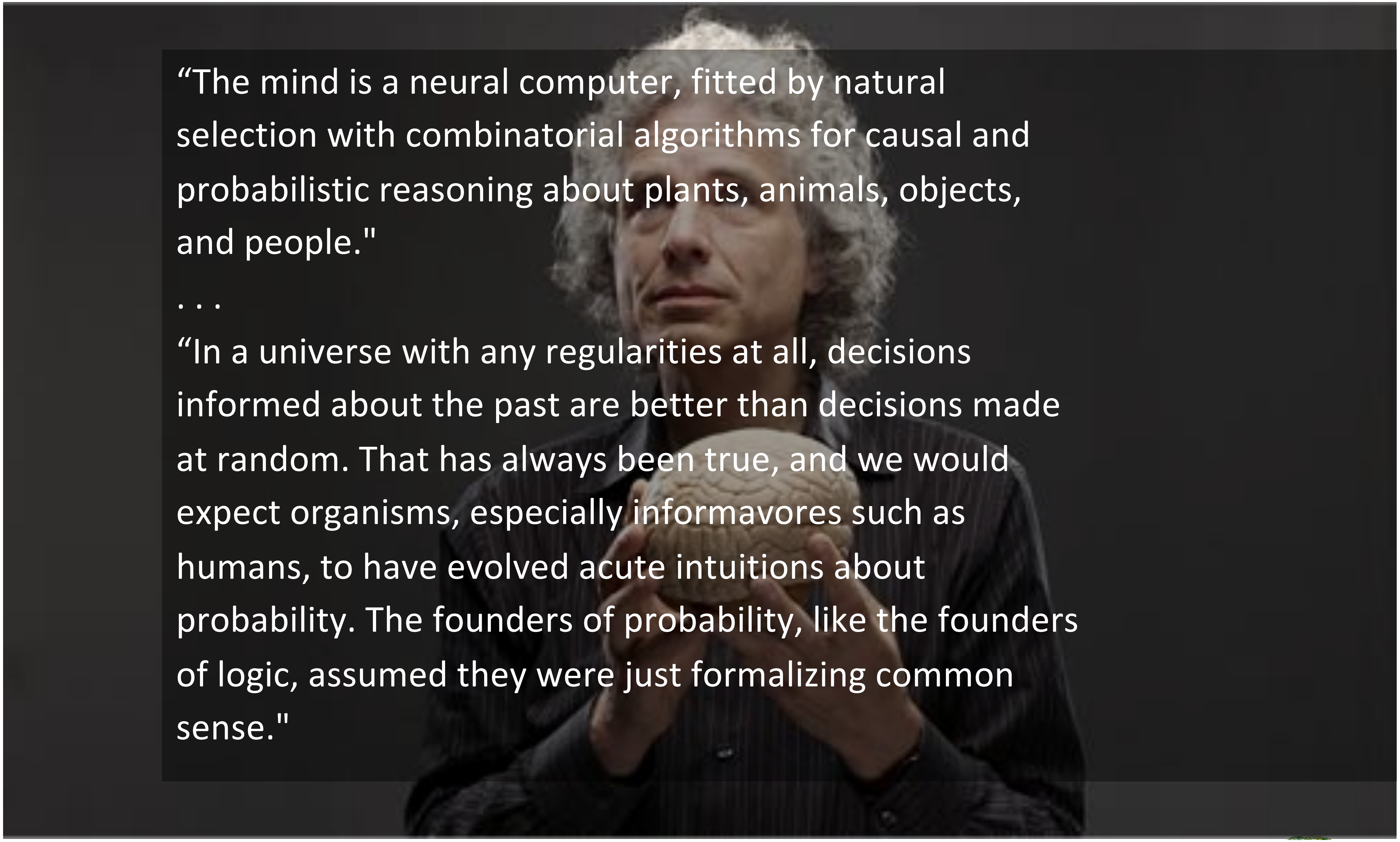
Stages of Visual Representation, David Marr, 1970s

Why the success of DNNs is ~~surprising~~ obvious

- Natural language:



Why the success of DNNs is ~~surprising~~ obvious



“The mind is a neural computer, fitted by natural selection with combinatorial algorithms for causal and probabilistic reasoning about plants, animals, objects, and people."

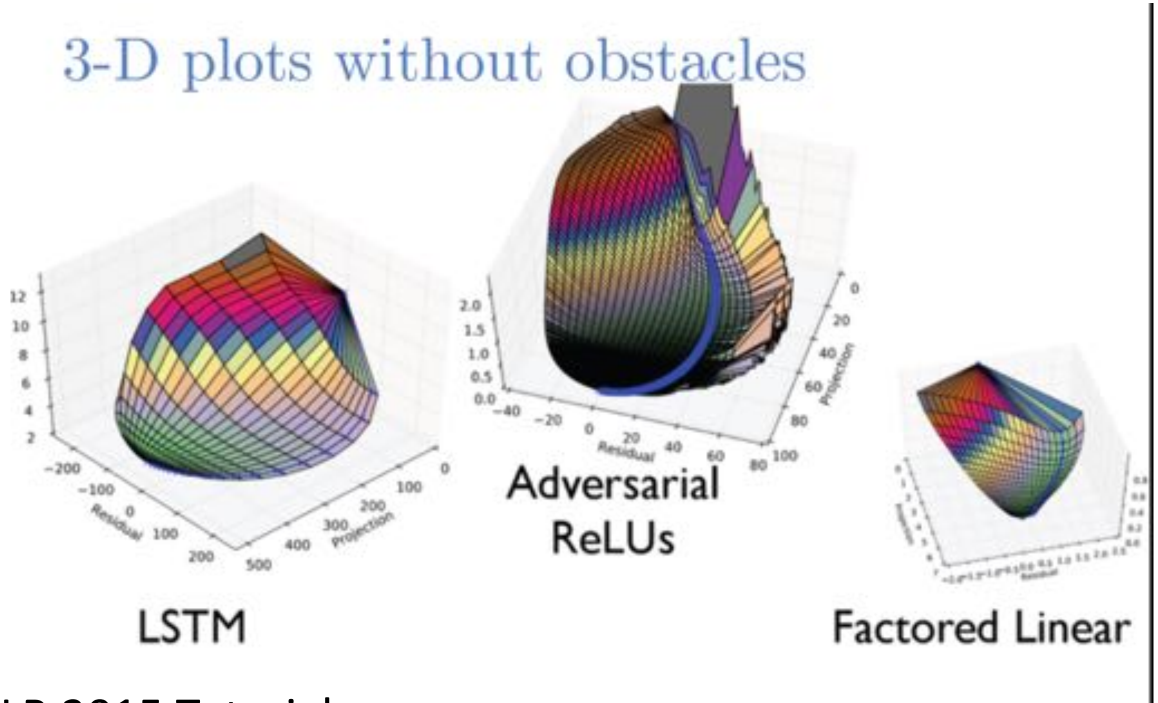
. . .

“In a universe with any regularities at all, decisions informed about the past are better than decisions made at random. That has always been true, and we would expect organisms, especially informavores such as humans, to have evolved acute intuitions about probability. The founders of probability, like the founders of logic, assumed they were just formalizing common sense."

-Steven Pinker, How the Mind Works, 1997, pp. 524, 343.

Why the success of DNNs is ~~surprising~~ obvious

- What about greedy optimization?
- Less obvious, but it looks like many learning problems (e.g. image classification) are actually "easy" i.e. have reliable steepest descent paths to a good model.



Ian Goodfellow – ICLR 2015 Tutorial

What have your learnt?

- Deep Learning is quite successful
- Deep Learning is hyped
- The goal of the course is to review principles and techniques for understanding deep networks.

Have fun!

- Watch the next video and prepare 5 questions! We start with the part on optimizaton. The part on KNN, linear models etc. is only a recap as it was covered in other lectures.

- Also, start to work on your projects? What are interesting topics? What would you enjoy? Prepare a two-pages expose (Latex including clean and complete references)