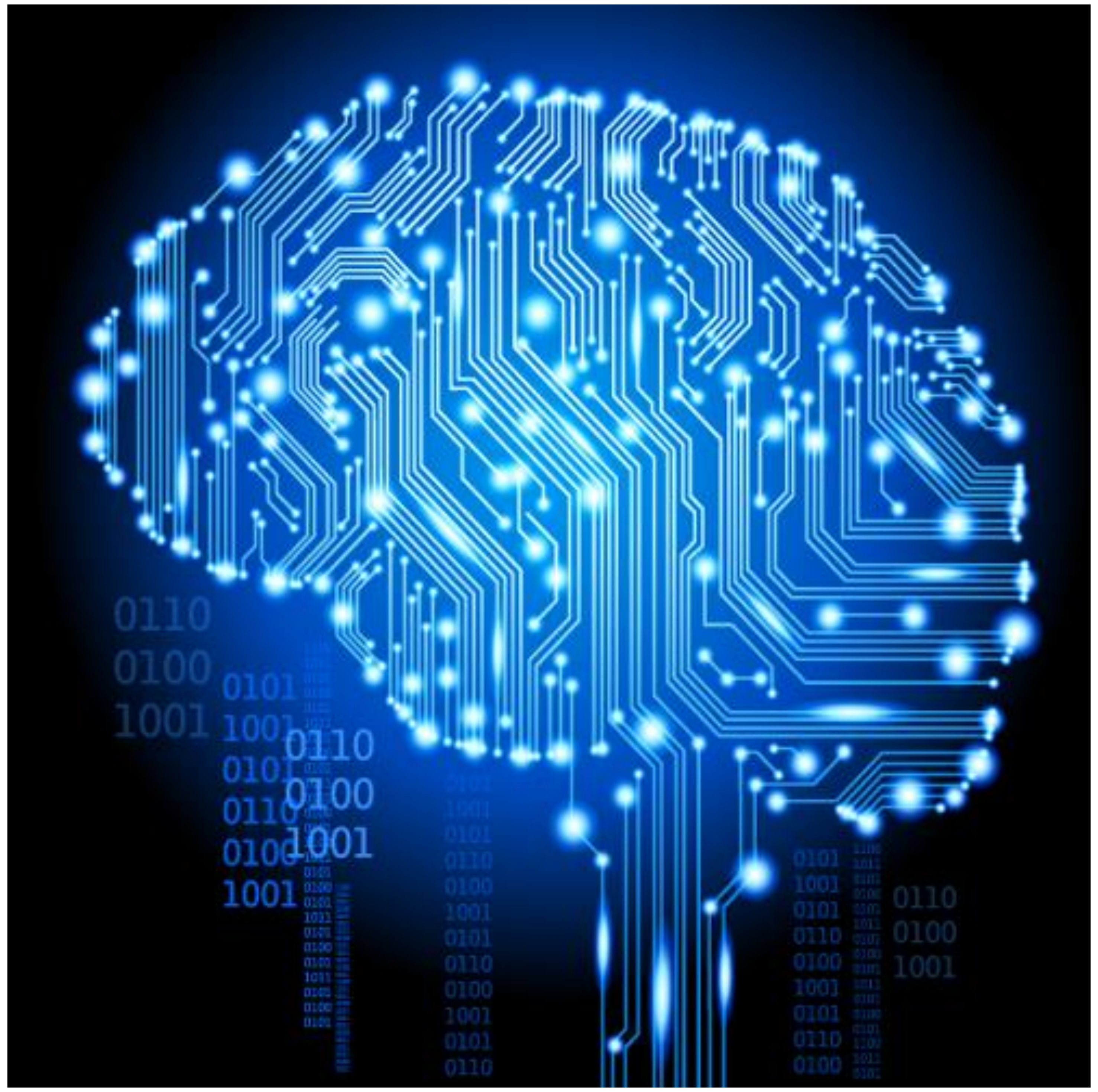


Deep Learning

A Brief Introduction to Neural Networks and Deep Learning

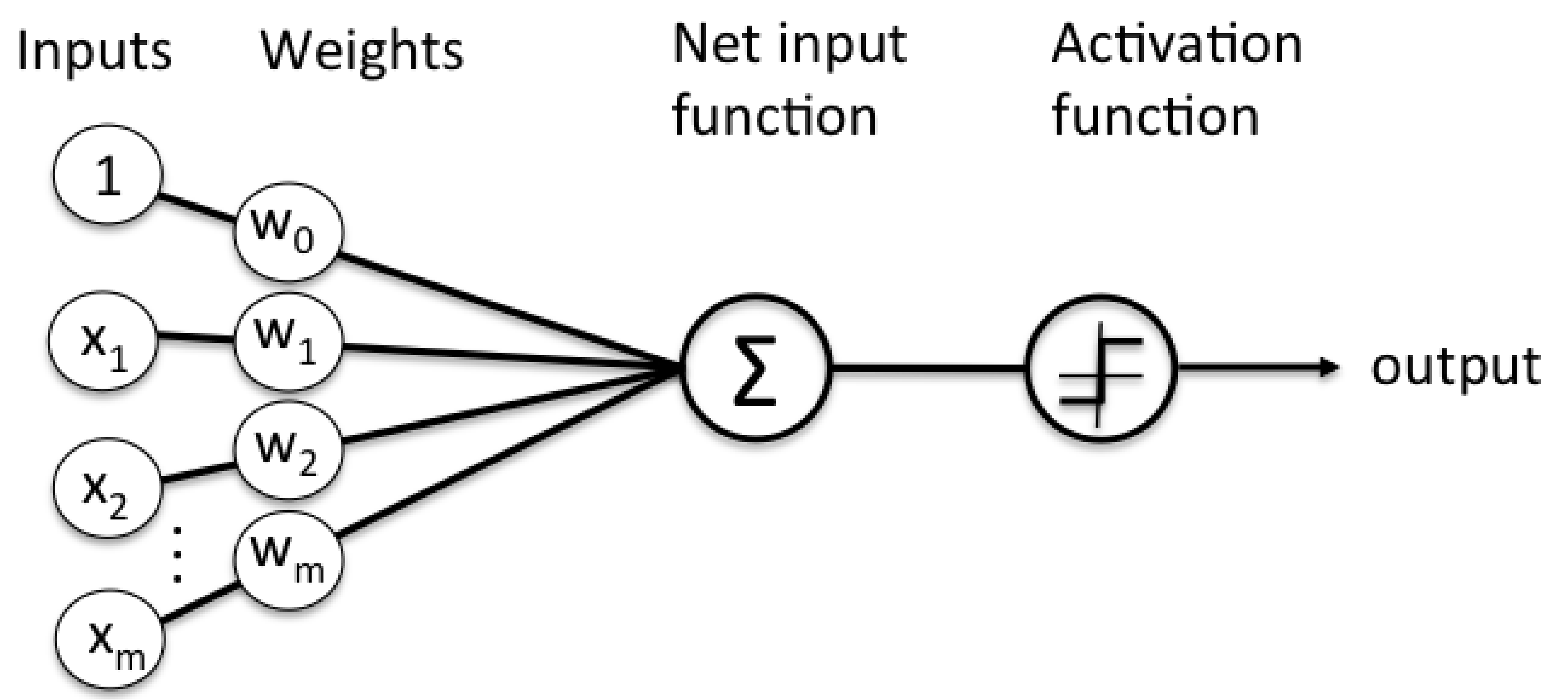
Ismail Elezi

Ca' Foscari, University of Venice

LAYOUT OF THE LECTURE

1. A brief history of neural networks
2. Some of the main ideas in deep learning
3. Recent achievements on deep learning
4. Problems and criticism
5. (A lot of) Resources

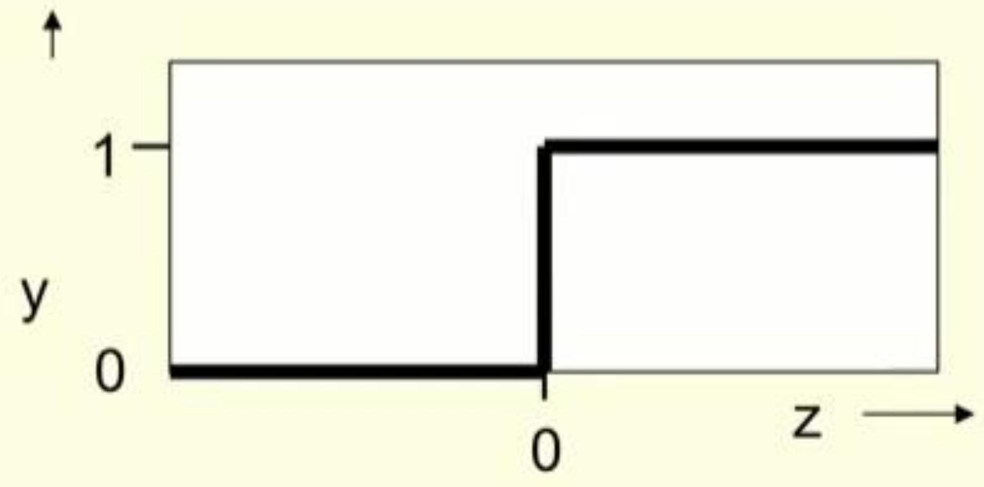
Rosenblatt's Perceptron (1957)



Schematic of Rosenblatt's perceptron.

$$z = b + \sum_i x_i w_i$$

$$y = \begin{cases} 1 \text{ if } & z \geq 0 \\ 0 \text{ otherwise} & \end{cases}$$



- If the output unit is correct, leave its weights alone.
- If the output unit incorrectly outputs a zero, add the input vector to the weight vector.
- If the output unit incorrectly outputs a 1, subtract the input vector from the weight vector.

Minsky and Papert (1969) showed that perceptrons can separate only linearly separable data.

First AI winter begins!

Back-propagation Algorithm

Kelley and Brason (1960/1961) in control theory.

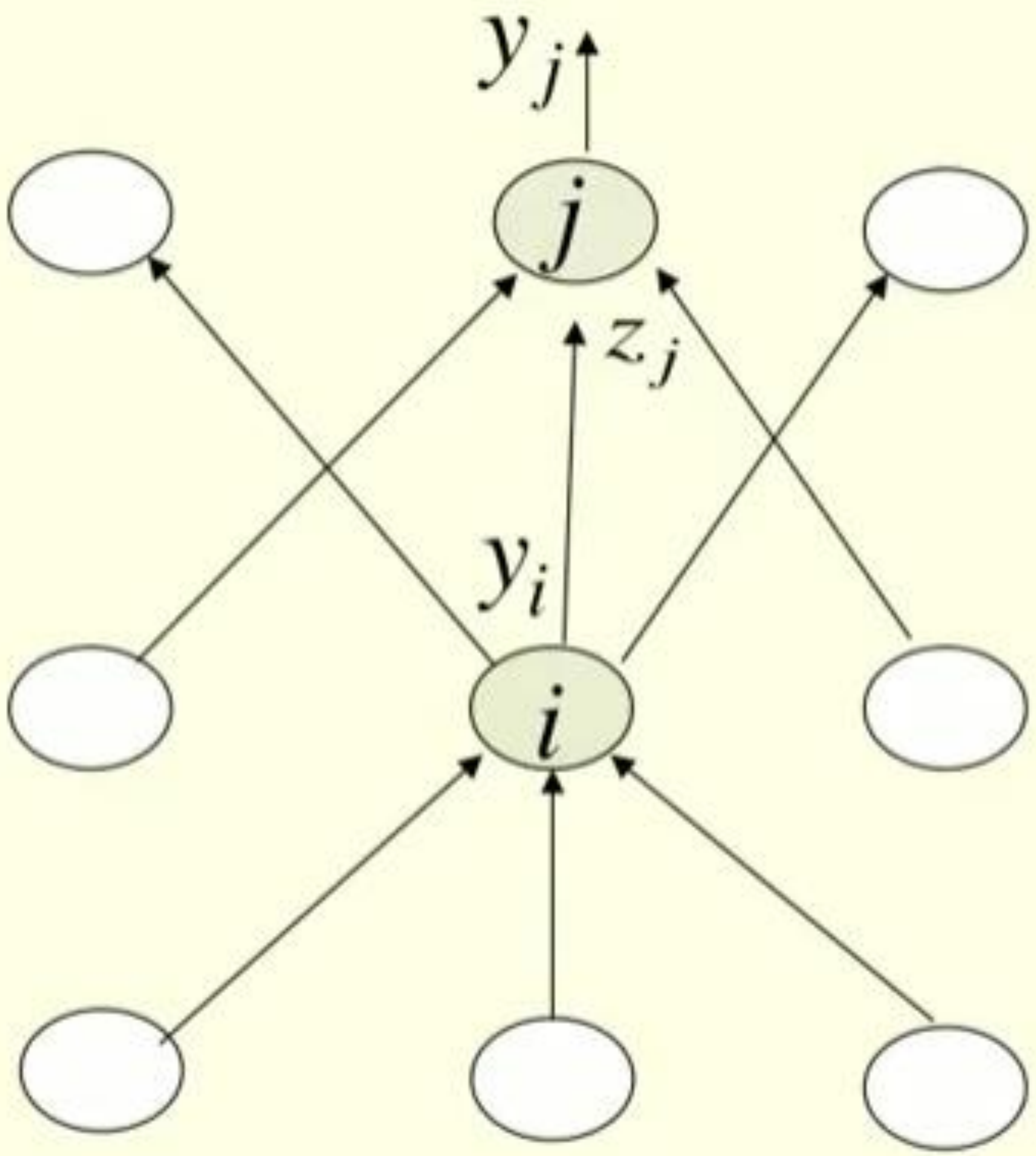
Paul Werbos (1974) in econometrics.

Rumelhart, Hinton & Williams (1986) developed an algorithm called error-backpropagation. No ‘neuron’ was mentioned on the original paper.

P(Geoffrey Hinton | fancy name & ANN) ≈ **1**

All versions were developed independently. Rumelhart et al. were the only ones who implemented it

Backpropagation – Chain Rule



$$\frac{\partial E}{\partial z_j} = \frac{dy_j}{dz_j}\frac{\partial E}{\partial y_j} = y_j\,(1 - y_j)\,\frac{\partial E}{\partial y_j}$$

$$\frac{\partial E}{\partial y_i} = \sum_j \frac{dz_j}{dy_i}\frac{\partial E}{\partial z_j} = \sum_j w_{ij}\frac{\partial E}{\partial z_j}$$

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial z_j}{\partial w_{ij}}\frac{\partial E}{\partial z_j} = y_i\frac{\partial E}{\partial z_j}$$

Neural Networks in Practice

Zip Code Reader

Autonomous driving

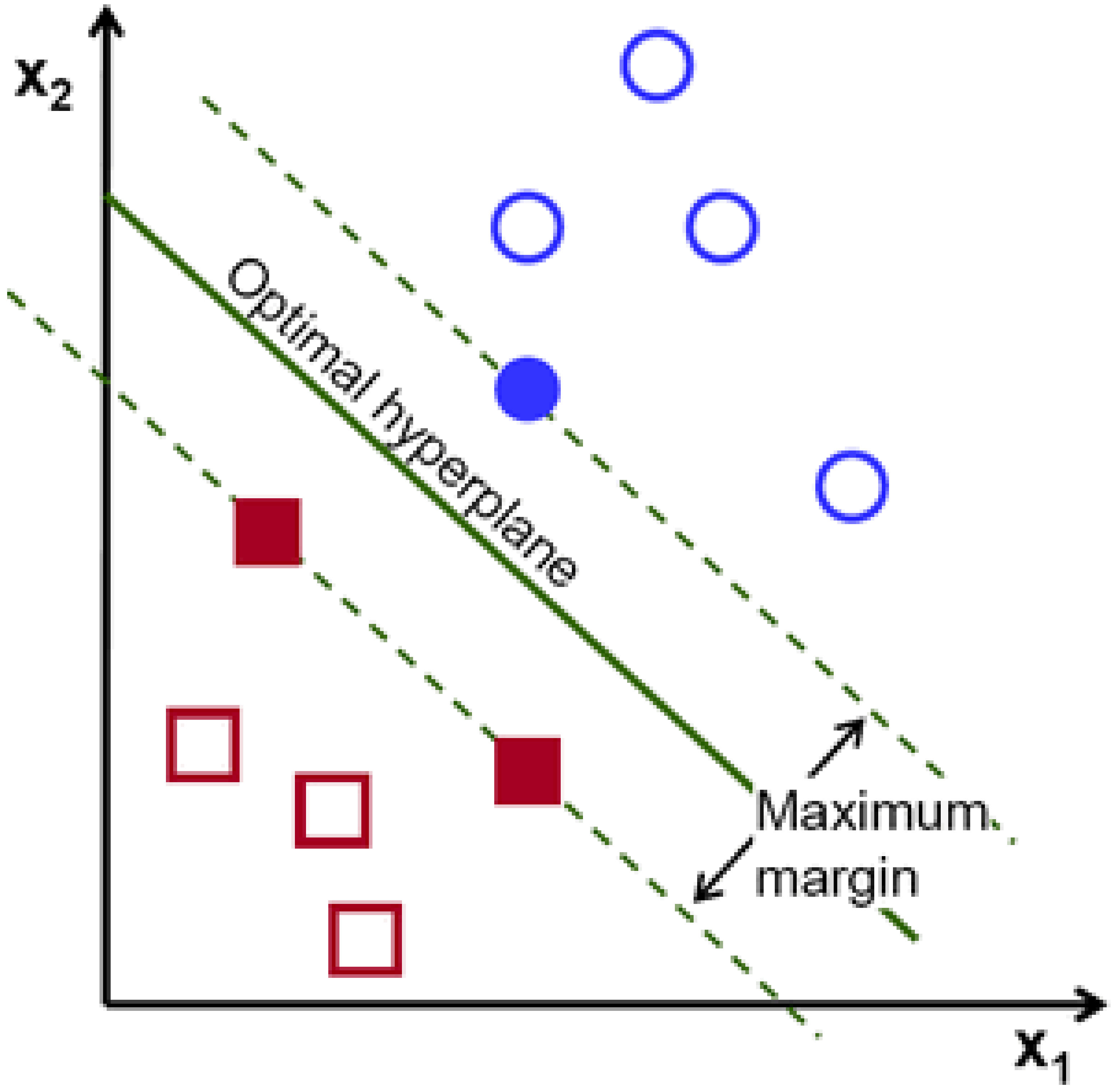
Neural Network Developments

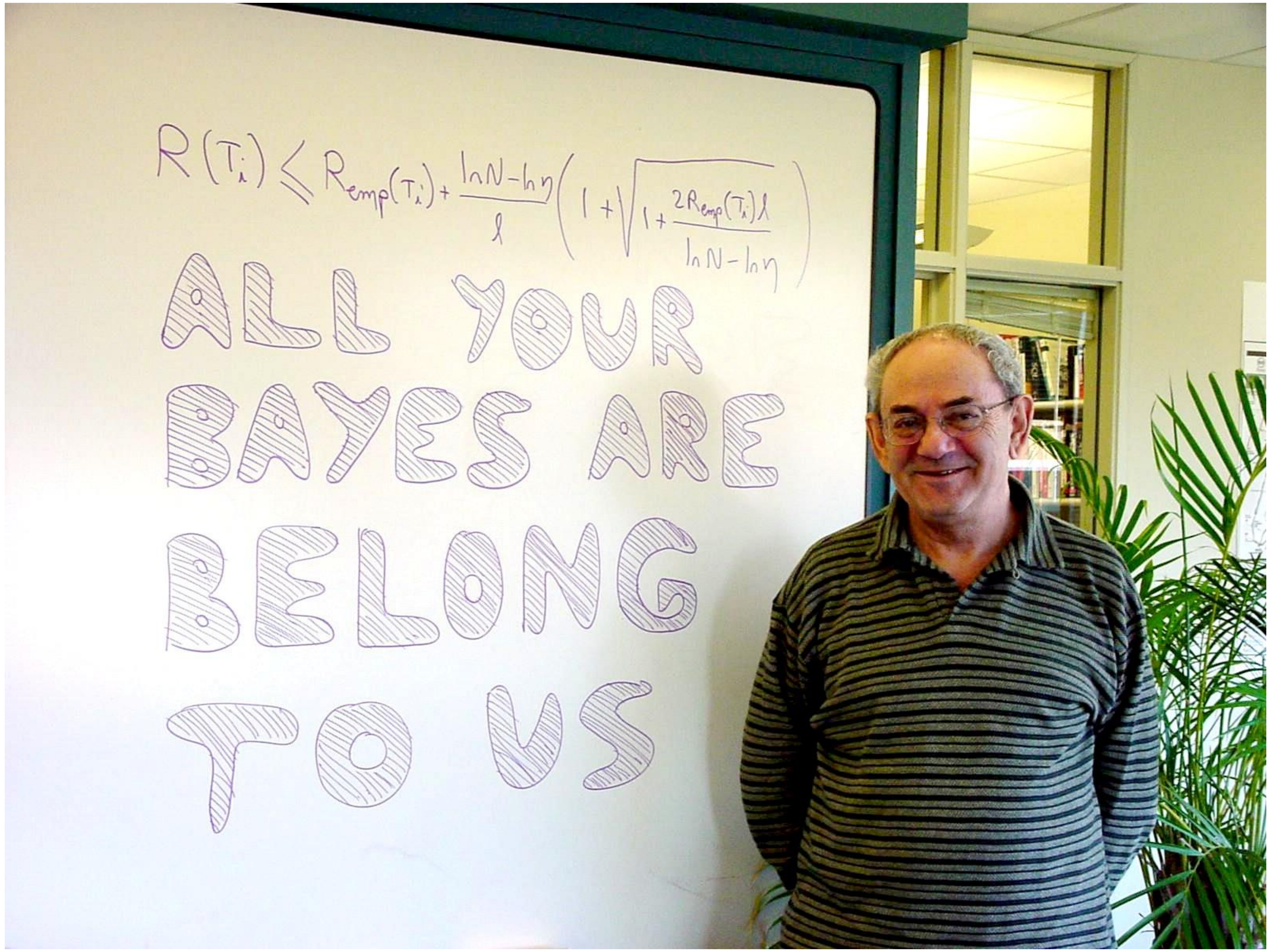
1 With backpropagation becoming so successful, other (even older) types of neural networks got popularized. Hopfield NN (Hopfield, 1982), Restricted Boltzmann Machine (Sejnowski & Hinton, 1985).

2 New types of neural networks were developed: Convolutional Neural Networks (LeCun), Recurrent Neural Networks (Schmidhuber), Deep Belief Networks (Hinton).

3 The future of Artificial Neural Networks was bright.

And then, SVM happened!





The Second Neural Networks Winter

Most of the research on the field of neural networks was abolished. The grants were cut, and top conferences weren't accepting (for most part) neural network-related papers.

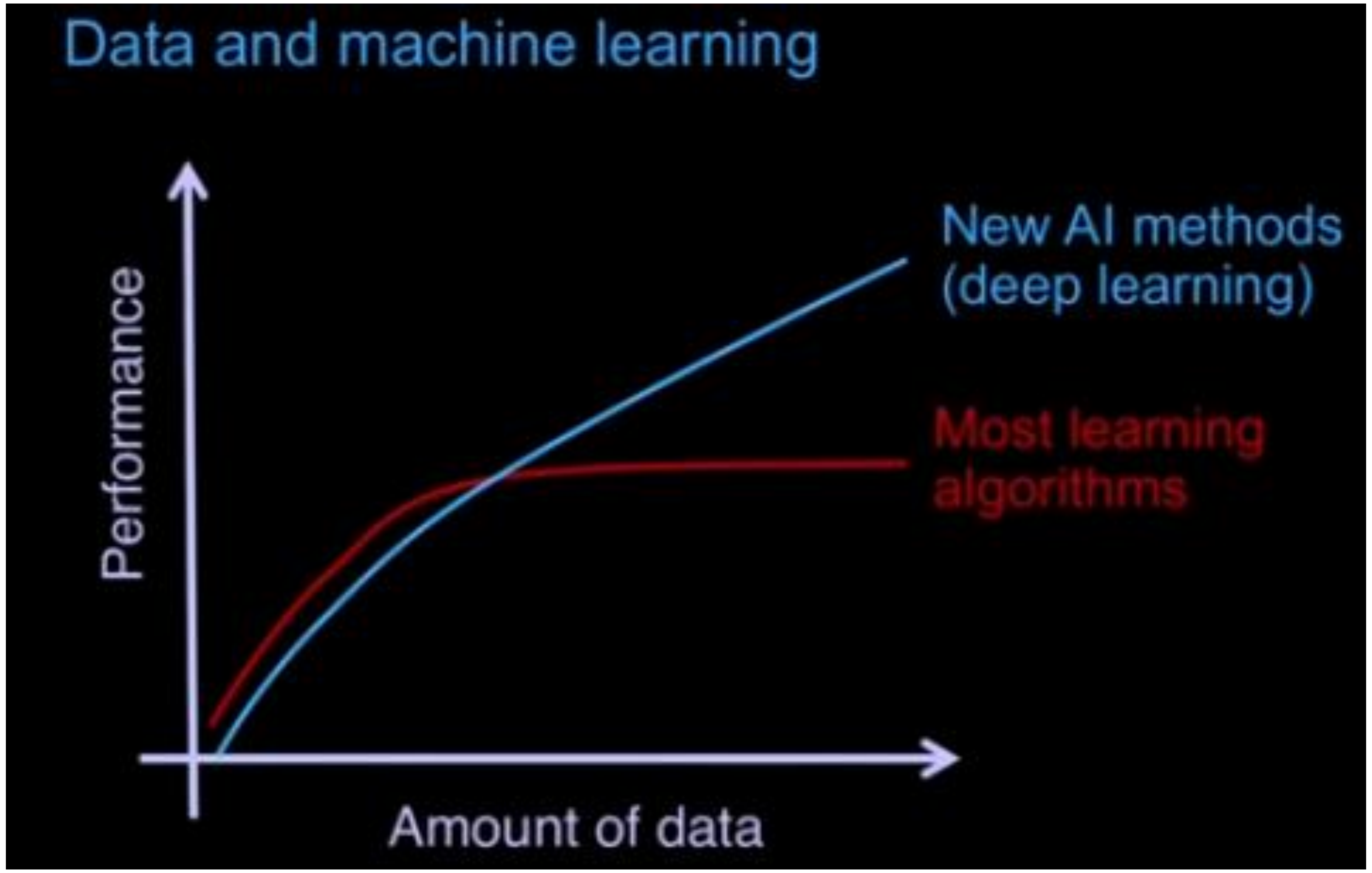
The only large groups who continued working on neural networks were the groups of Geoffrey Hinton (University of Toronto), Yoshua Bengio (University of Montreal), Yann LeCun (New York University) and Juergen Schmidhuber (University of Lugano). Andrew Ng (Stanford University) started getting interested on neural networks in 2006.

For near 15 years, there were basically no developments.

The (main) people behind neural networks

A New Spring

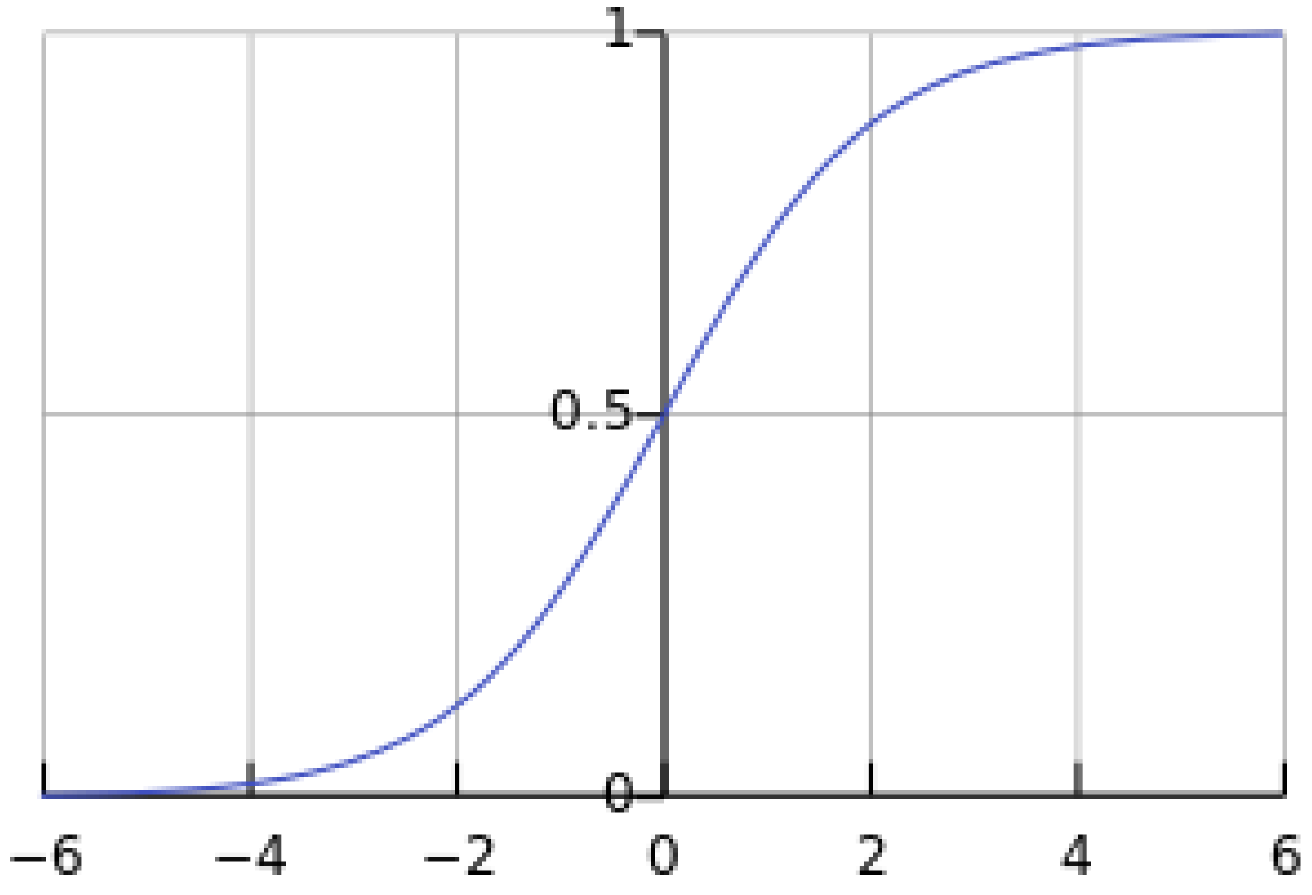

ASUS
GeForce
GTX590

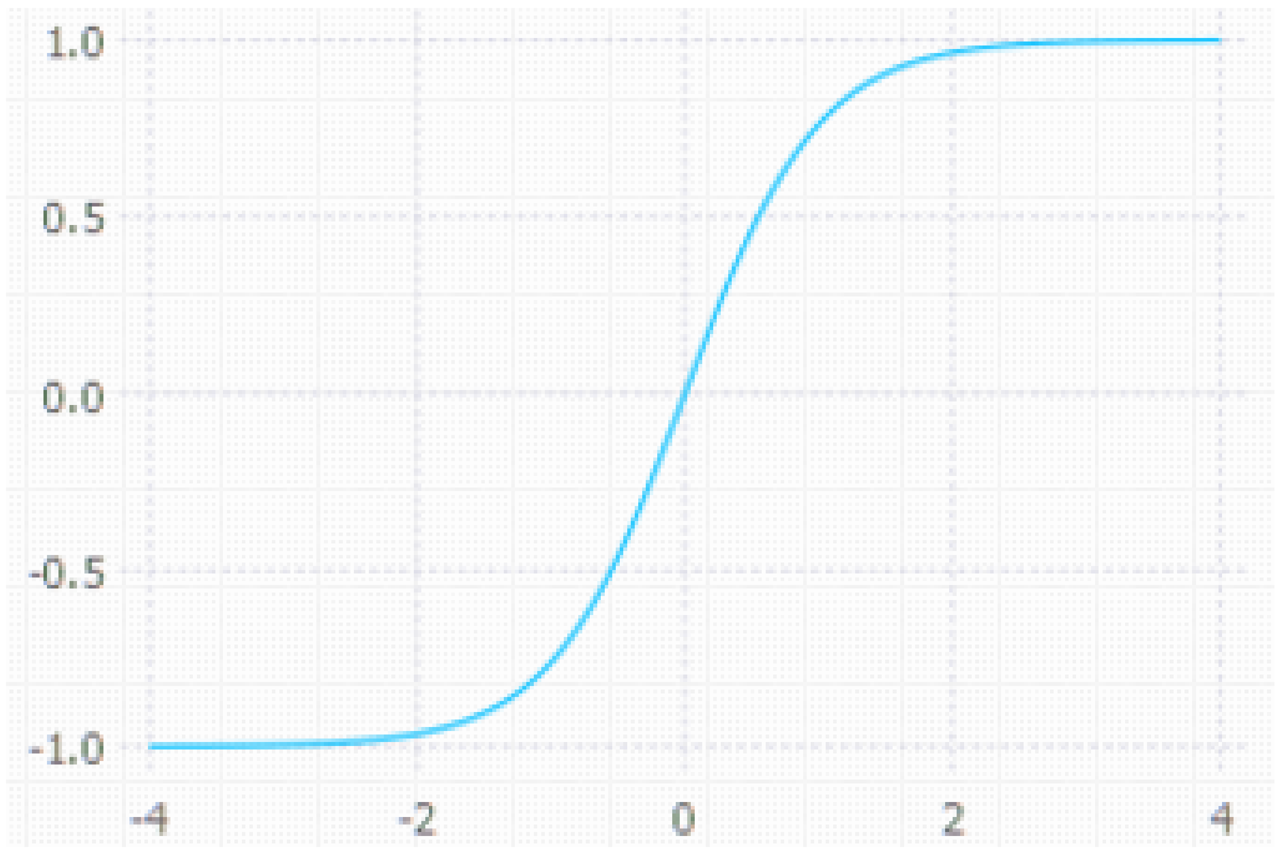

Data and machine learning
Performance
New AI methods (deep learning)
Most learning algorithms
Amount of data

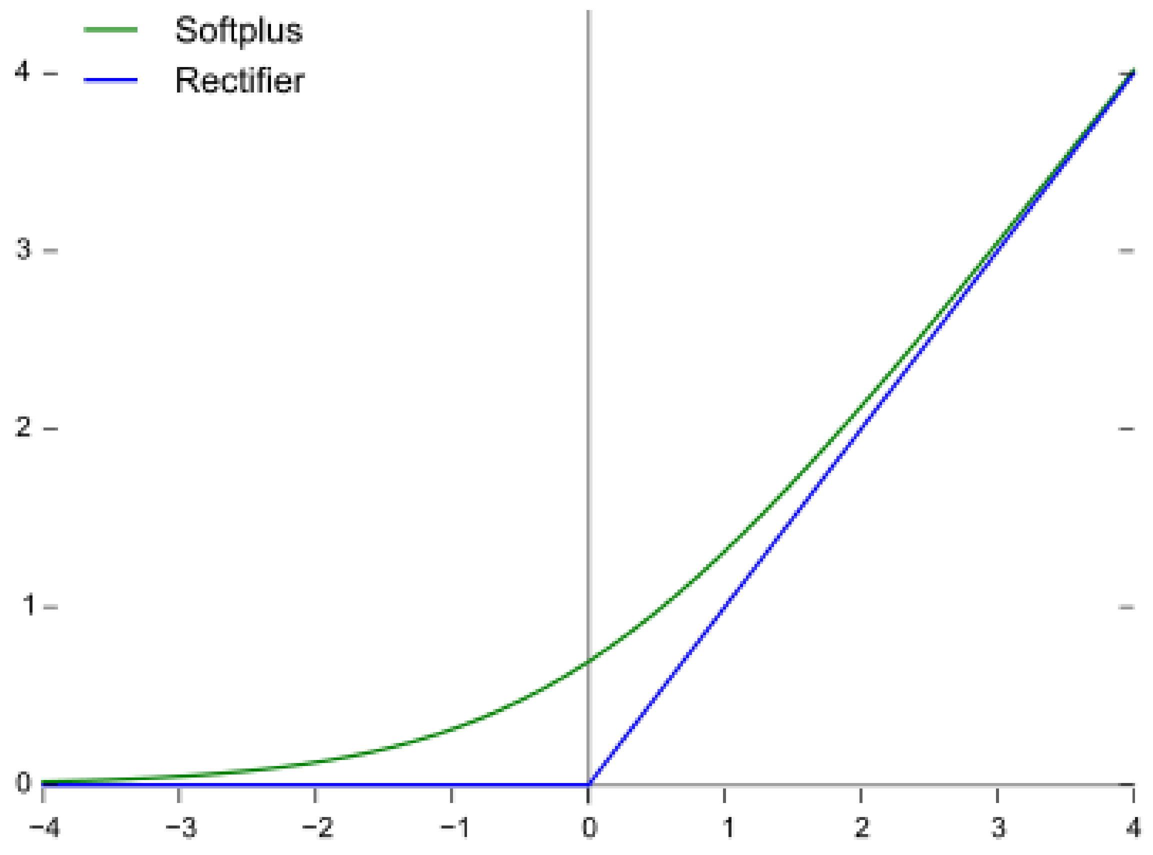
Backprop in Practice

- Use ReLU non-linearities
- Use cross-entropy loss for classification
- Use Stochastic Gradient Descent on minibatches
- Shuffle the training samples (← very important)
- Normalize the input variables (zero mean, unit variance)
- Schedule to decrease the learning rate
- Use a bit of L1 or L2 regularization on the weights (or a combination)
  - But it's best to turn it on after a couple of epochs
- Use "dropout" for regularization
- Lots more in [LeCun et al. "Efficient Backprop" 1998]
- Lots, lots more in "Neural Networks, Tricks of the Trade" (2012 edition) edited by G. Montavon, G. B. Orr, and K-R Müller (Springer)
- More recent: Deep Learning (MIT Press book in preparation)

Backpropagation – Activation Function







$$S(t) = \frac{1}{1+e^{-t}}.$$

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

$$f(x) = \max(0, x)$$

$$f(x) = \ln(1 + e^x)$$

Deep Learning: Automating Feature Discovery

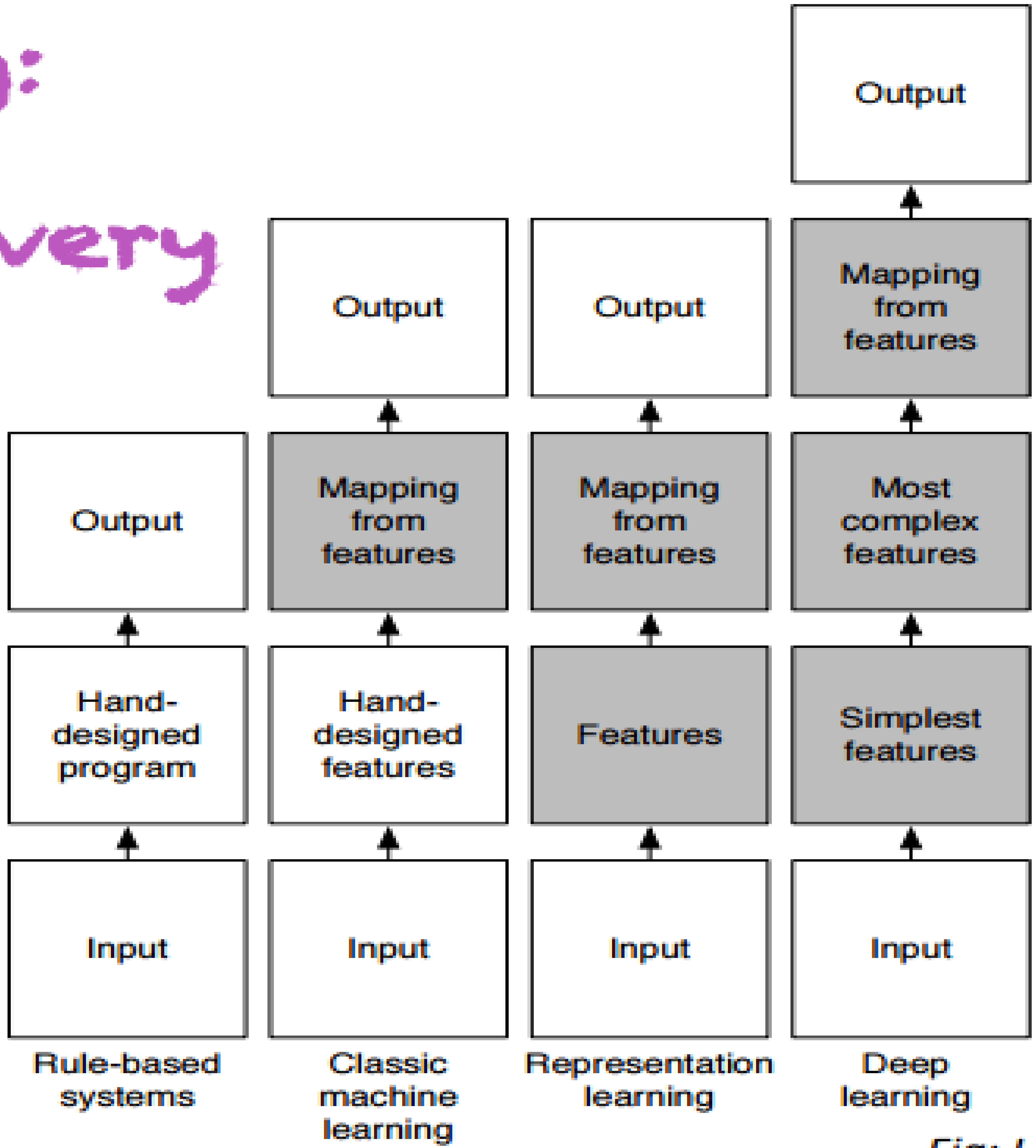


Fig: I. Goodfellow

Deep Learning = Training Multistage Machines

Y LeCun

- **Traditional Pattern Recognition:** Fixed/Handcrafted Feature Extractor

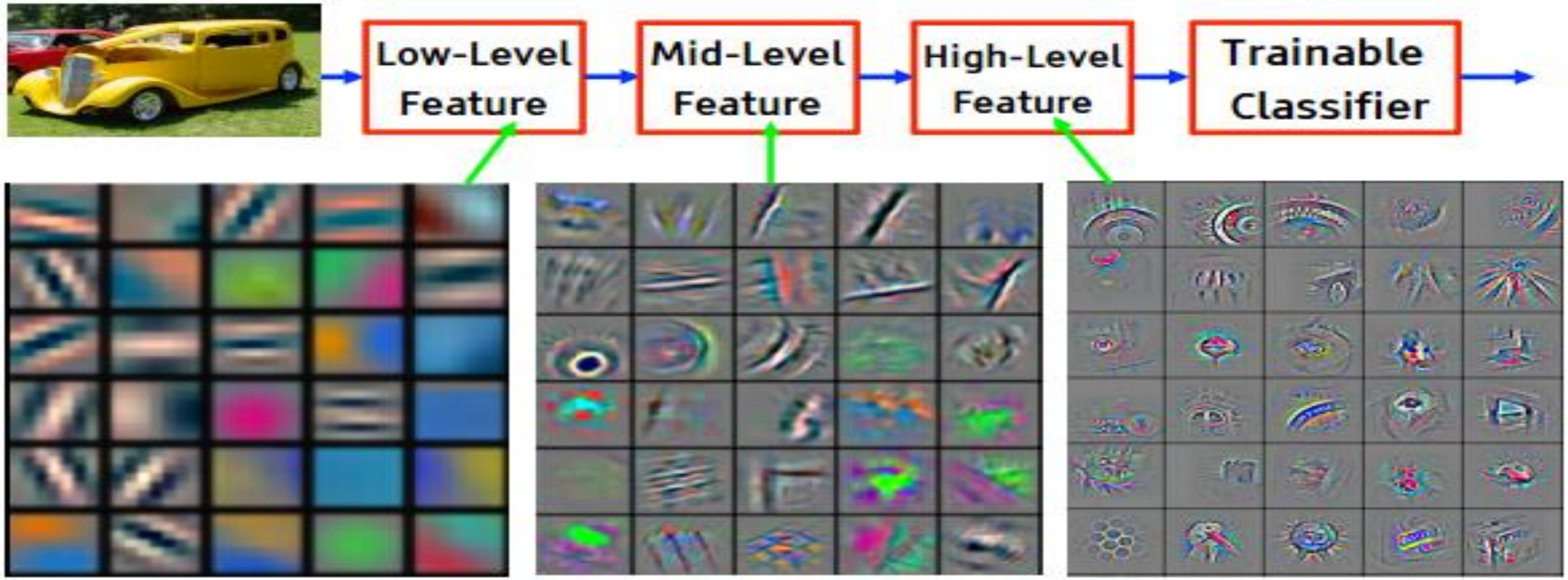
- **Mainstream Pattern Recognition 9until recently)**

- **Deep Learning:** Multiple stages/layers trained end to end

Why Multiple Layers? The World is Compositional

Y LeCun

- Hierarchy of representations with increasing level of abstraction
- Each stage is a kind of trainable feature transform
- Image recognition: Pixel → edge → texton → motif → part → object
- Text: Character → word → word group → clause → sentence → story
- Speech: Sample → spectral band → sound → ... → phone → phoneme → word



First successes (MNIST Dataset – Digit Recognizer)

Place	Algorithm	Author	Error rate
1	CNN	Ciresan et al.	0.23
2	CNN	Ciresan et al.	0.27
3	CNN	Ciresan et al.	0.35
4	ANN	Ciresan et al.	0.35
5	CNN	Ranzato et al.	0.39
6	ANN	Meier et al.	0.39
7	CNN	Simard et al.	0.4
8	CNN	Jarrett et al.	0.53
9	CNN	Lauer et al.	0.54
10	CNN	Lauer et al.	0.56
11	SVM	DeCoste and Scholkopf	0.56

First successes (Text Generator)

He was elected President during the Revolutionary War and forgave Opus Paul at Rome. The regime of his crew of England, is now Arab women's icons in and the demons that use something between the characters' sisters in lower coil trains were always operated on the line of the ephemerable street, respectively, the graphic or other facility for deformation of a given proportion of large segments at RTUS). The B every chord was a "strongly cold internal palette pour even the white blade."

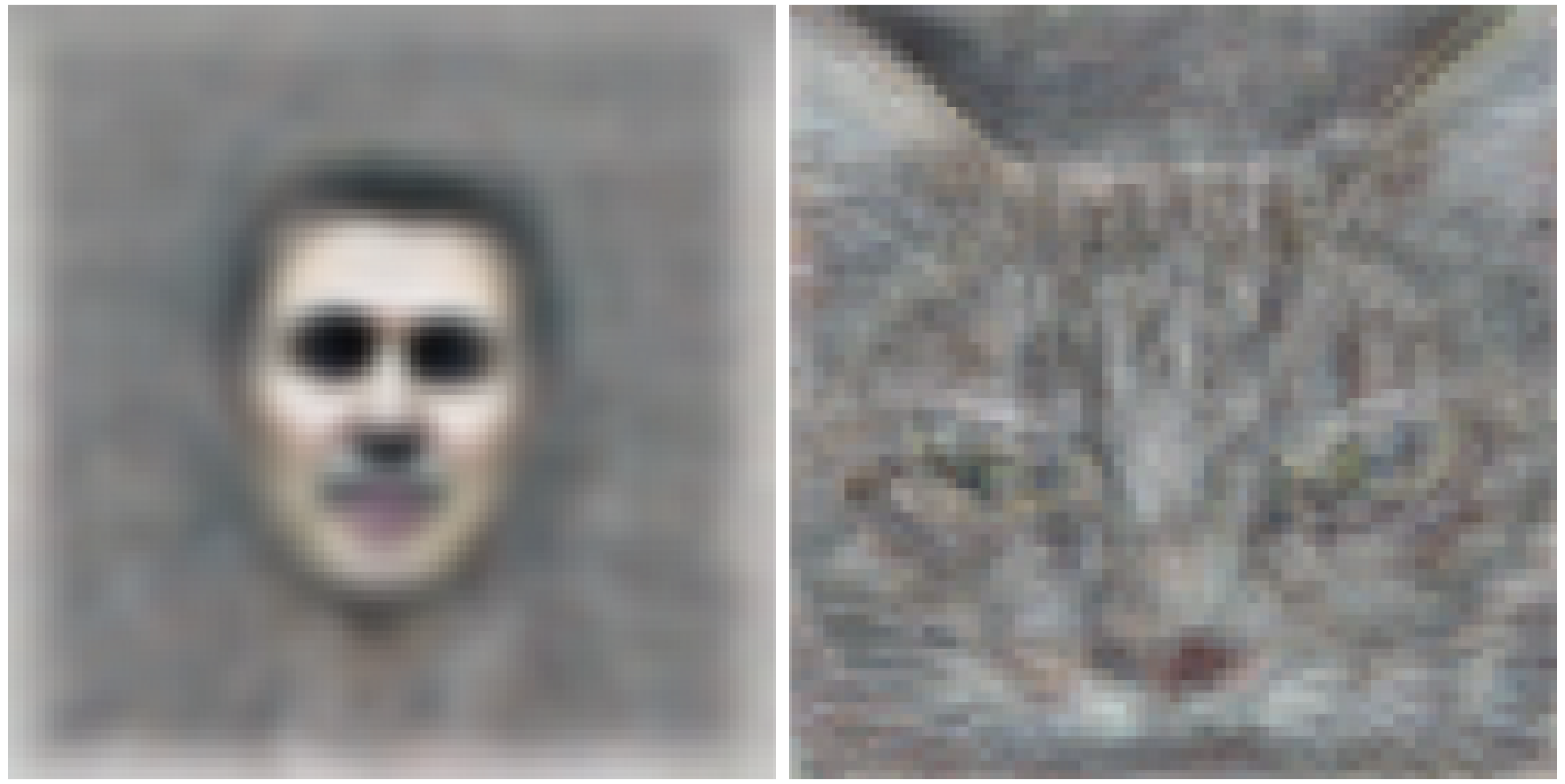
- Sheila thrunges (most frequent)
- People thrunge (most frequent next character is space)
- Shiela, Thrungelini del Rey (first try)
- The meaning of life is literary recognition. (6^{th} try)

Ilya Sutskever website

First successes (Unsupervised Learning)

Building High-level Features Using Large Scale Unsupervised Learning

Quoc V. Le, Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S. Corrado, Jeffrey Dean, and Andrew Y. Ng



Abstract

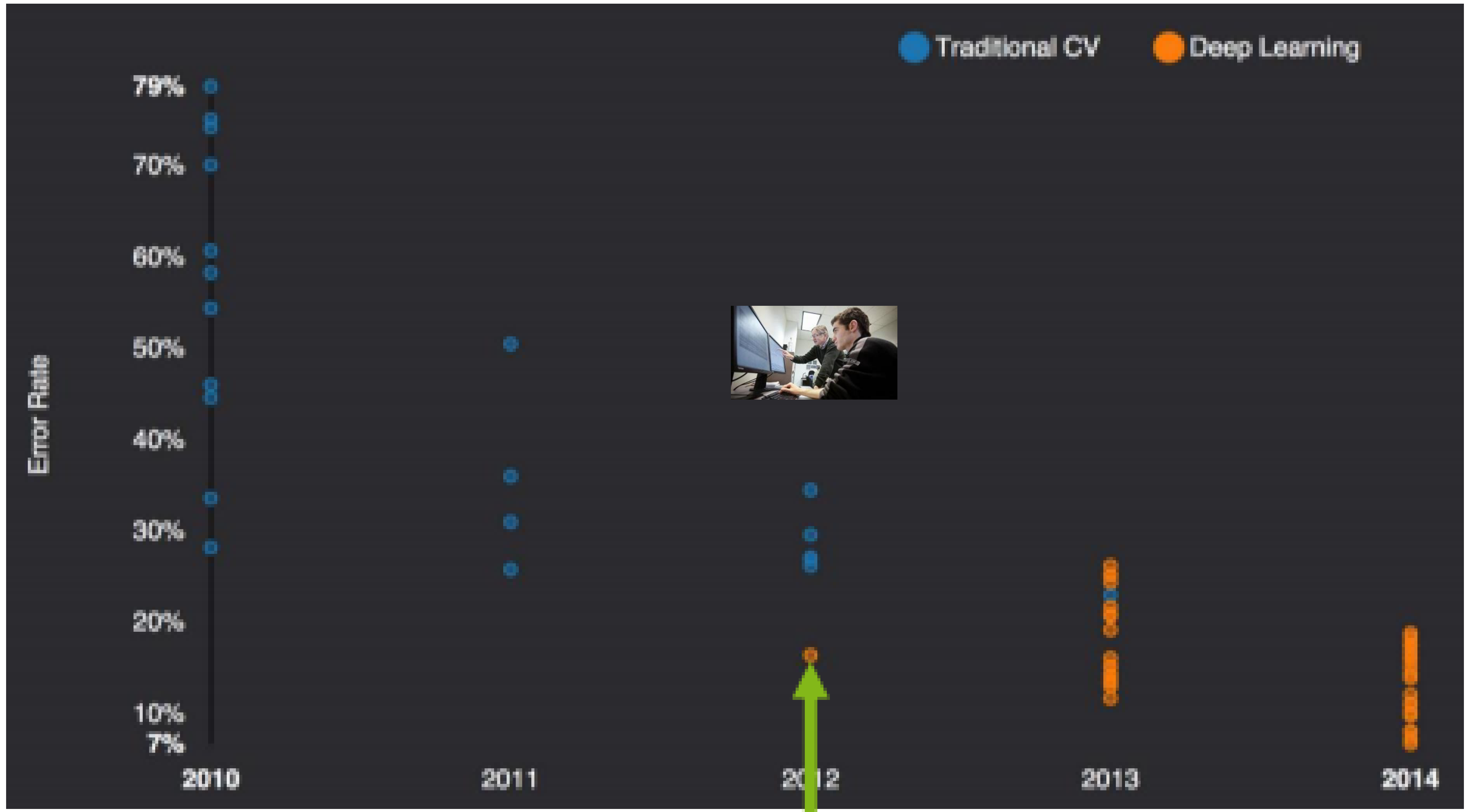
We consider the problem of building high-level, class-specific feature detectors from only unlabeled data. For example, is it possible to learn a face detector using only unlabeled images? To answer this, we train a 9-layered locally connected sparse autoencoder with pooling and local contrast normalization on a large dataset of images (the model has 1 billion connections, the dataset has 10 million 200x200 pixel images downloaded from the Internet). We train this network using model parallelism and asynchronous SGD on a cluster with 1,000 machines (16,000 cores) for three days. Contrary to what appears to be a widely-held intuition, our experimental results reveal that it is possible to train a face detector without having to label images as containing a face or not. Control experiments show that this feature detector is robust not only to translation but also to scaling and out-of-plane rotation. We also find that the same network is sensitive to other high-level concepts such as cat faces and human bodies. Starting with these learned features, we trained our network to obtain 15.8% accuracy in recognizing 20,000 object categories from ImageNet, a leap of 70% relative improvement over the previous state-of-the-art.

But the skepticism remains!

ImageNet is a good competition to test whether neural networks work well for object recognition

1.3 million images
1000 categories

ImageNet: The Deep Learning goes mainstream



A. Krizhevsky uses first CNN in 2012.
Trained on Gaming Graphic Cards

2015: It gets tougher

4.95% Microsoft (Feb 6)
→ surpassing human performance of 5.1%

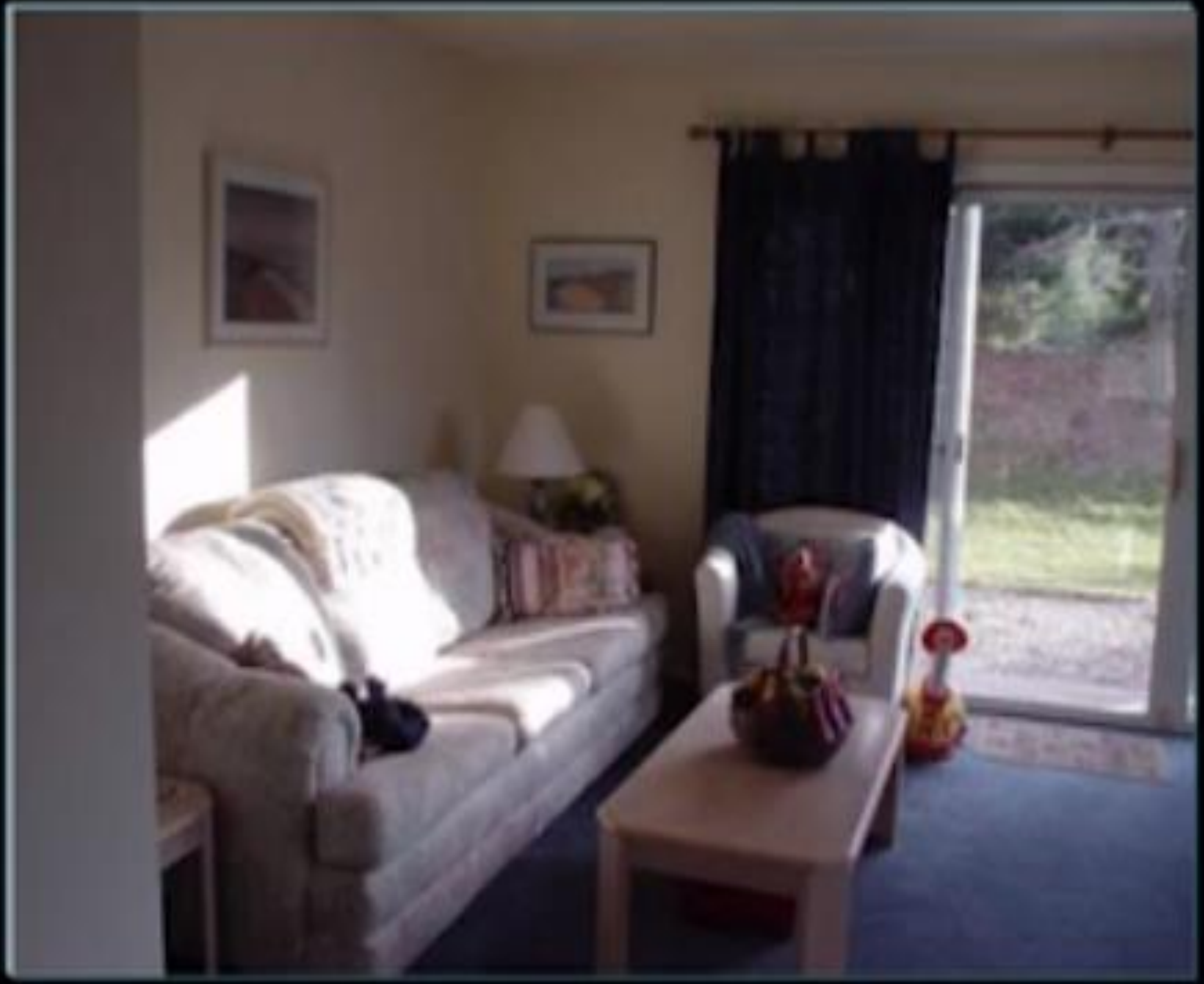
4.8% Google (Feb 11)

4.58% Baidu (May 11)

→ **Computers learn** to identify objects

Can a computer understand these pictures?

A yellow bus driving down a road with green trees and green grass in the background.



Living room with white couch and blue carpeting. The room in the apartment gets some afternoon sun.

Autonomous Driving

Baidu's autonomous car

nVIDIA: The Way It's Meant to be Drove

9 inception layers

3 convolutional layers

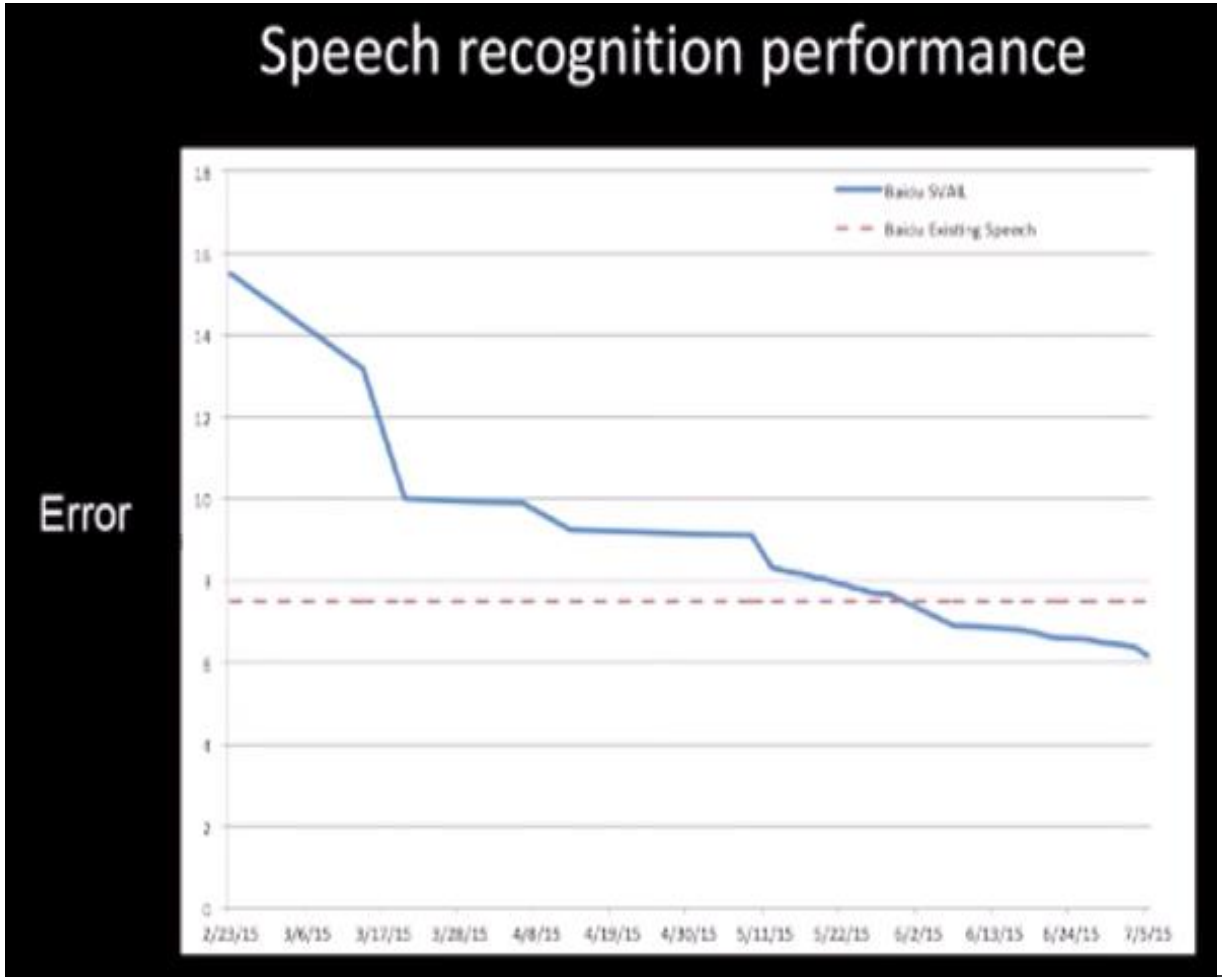
37M neurons

40B operations

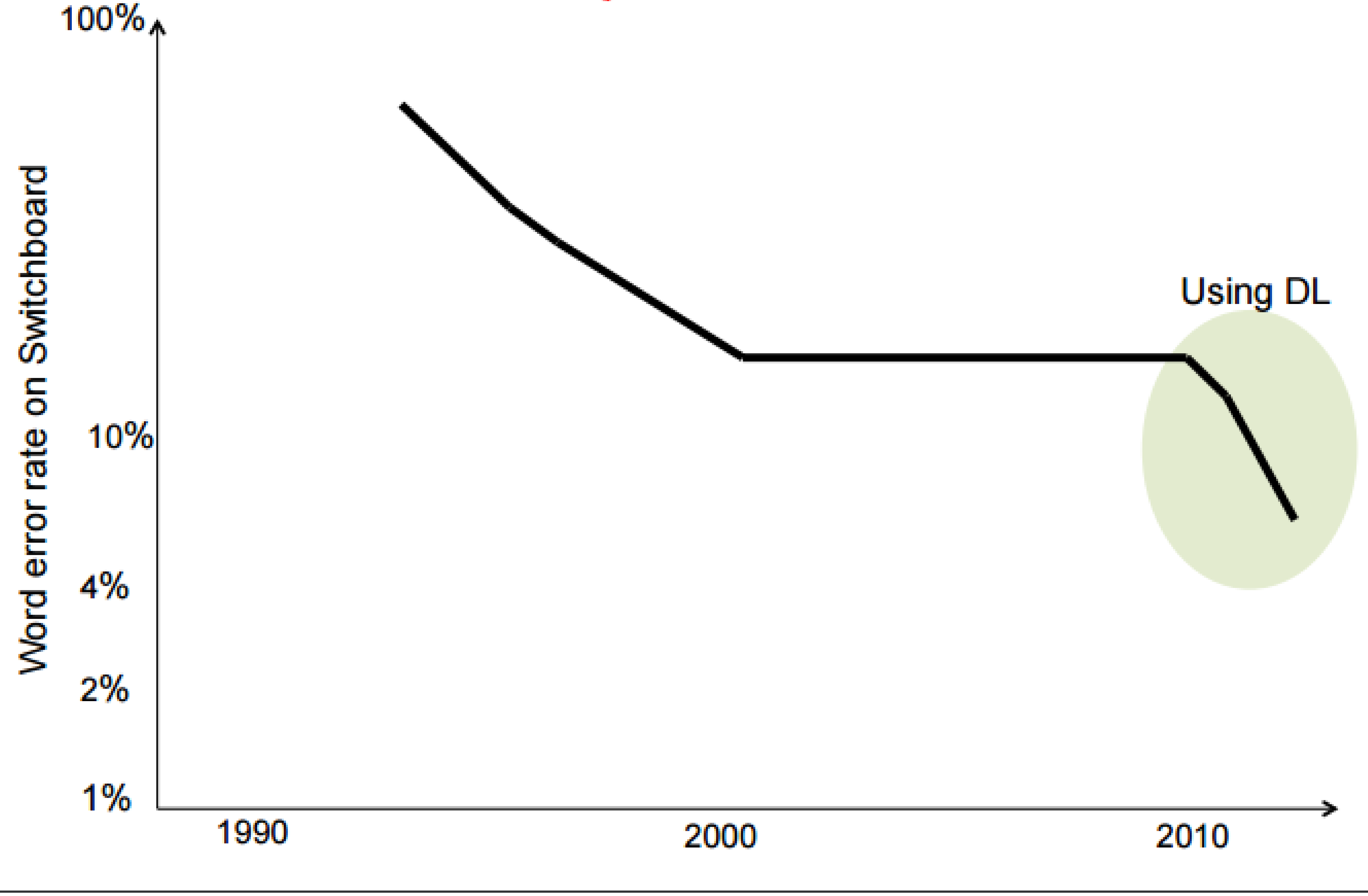
Single and multi-class detection

Segmentation

Baidu Eye: Helping the Visually Impaired People



The dramatic impact of Deep Learning on Speech Recognition (according to Microsoft)



Google DeepMind’s AlphaGo vs Lee Sedol

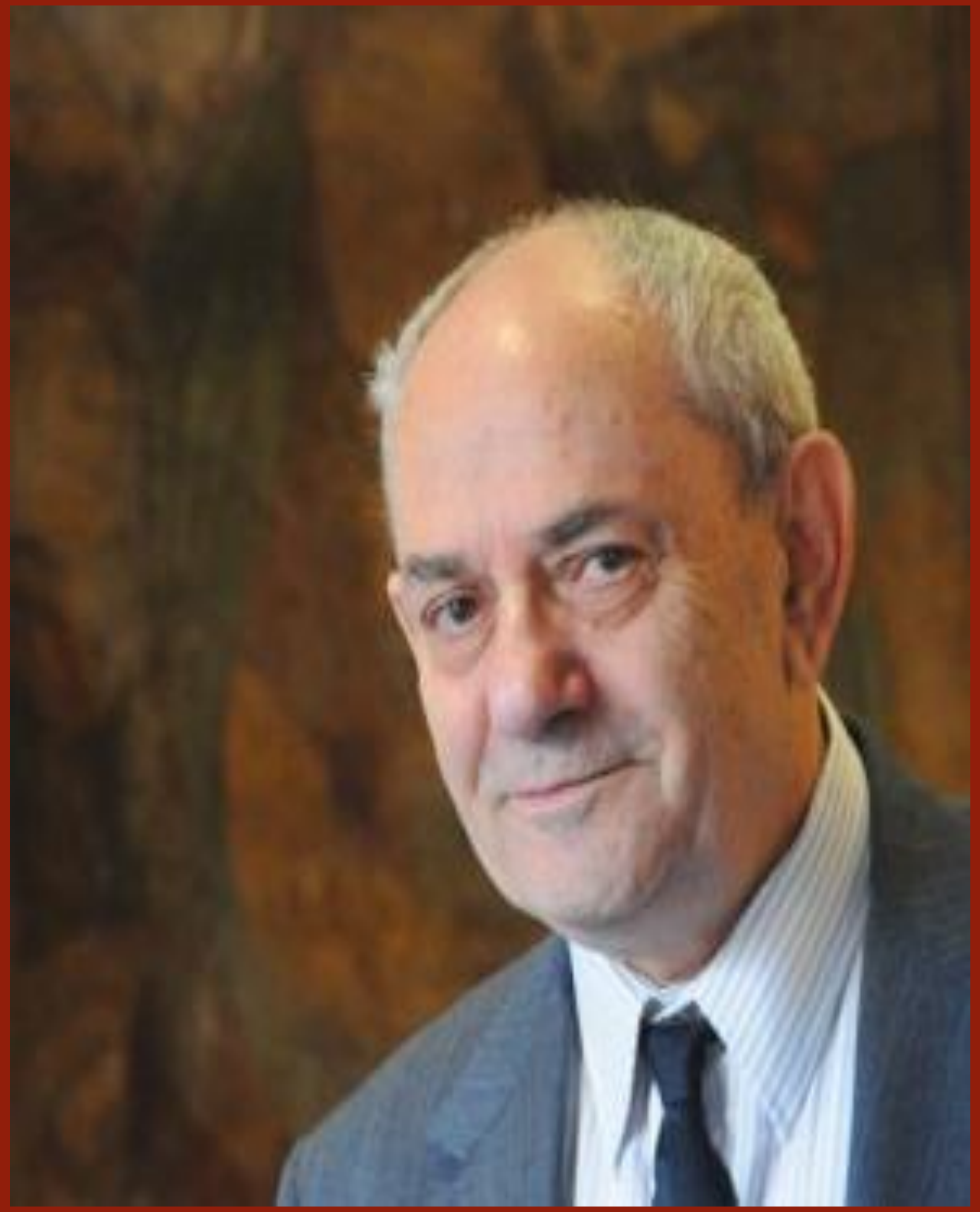
[Google’s AlphaGo]()

Criticism and Skepticism!

Deep Learning might not win the machine learning race!

Stop making brain parallelism! And stop overhyping it!

Deep Learning is evil!



Captain Schmidhuber: Civil War

As a case in point, let me now comment on a recent article in Nature (2015) about "deep learning" in artificial neural networks (NNs), by LeCun & Bengio & Hinton (LBH for short), three CIFAR-funded collaborators who call themselves the "deep learning conspiracy" (e.g., LeCun, 2015)…

1. LBH's survey does not even mention the father of deep learning, Alexey Grigorevich Ivakhnenko, who published the first general, working learning algorithms for deep networks (e.g., Ivakhnenko and Lapa, 1965).

2. LBH discuss the importance and problems of gradient descent-based learning through back propagation (BP), and cite their own papers on BP, plus a few others, but fail to mention BP's inventors.

3. LBH claim: "Interest in deep feedforward networks [FNNs] was revived around 2006 (refs 31-34) by a group of researchers brought together by the Canadian Institute for Advanced Research (CIFAR)." Here they refer exclusively to their own labs, which is misleading. For example, by 2006, many researchers had used deep nets of the Ivakhnenko type for decades…

One more little quibble: While LBH suggest that "the earliest days of pattern recognition" date back to the 1950s, the cited methods are actually very similar to linear regressors of the early 1800s, by Gauss and Legendre. Gauss famously used such techniques to recognize predictive patterns in observations of the asteroid Ceres.
LBH may be backed by the best PR machines of the Western world (Google hired Hinton; Facebook hired LeCun). In the long run, however, historic scientific facts (as evident from the published record) will be stronger than any PR. There is a long tradition of insights into deep learning, and the community as a whole will benefit from appreciating the historical foundations.

The LeCun Strikes Back

Yes lots and lots of people have used chain rule before [Rumelhart et al. 1986], lots of people figured you could multiply Jacobians in reverse order in a multi-step function (perhaps even going back to Gauss, Leibniz, Newton, and Lagrange). But did they all "invent backprop?" No! They did not realize how this could be used for machine learning and they sure didn't implement it and made it work for that. …

Yes, a few people actually figured out early on that you could use chain rule for training a machine (including Rumelhart by the way. It took him and Geoff Hinton several years to get it to work). Some people had the intuition that you could use backward signals to train a multi-stage system (e.g. system theorist A. M. Andrews in the early 70s). But did they reduce it to practice and did they manage to make it work? No. that didn't really happen until the mid-1980s. …

Lots of people tried to build helicopters in the early 20th century, and several took off. But the idea didn't become practical until Sikorski's refinement of the cyclic control and tail rotor in the late 30s and early 40s. Who should get credit? Leonardo da Vinci?

Krizhevski, Sutskever and Hinton get a lot of credit for their work, and it's well deserved. They used many of my ideas (and added a few), but you don't see me complain about it. That's how science and technology make progress.

Fooling Neural Networks (and this is a real problem)!

Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images

Anh Nguyen
University of Wyoming
anguyen8@uwyo.edu

Jason Yosinski
Cornell University
yosinski@cs.cornell.edu

Jeff Clune
University of Wyoming
jeffclune@uwyo.edu

Abstract

Deep neural networks (DNNs) have recently been achieving state-of-the-art performance on a variety of pattern-recognition tasks, most notably visual classification problems. Given that DNNs are now able to classify objects in images with near-human-level performance, questions naturally arise as to what differences remain between computer and human vision. A recent study revealed that changing an image (e.g. of a lion) in a way imperceptible to humans can cause a DNN to label the image as something else entirely (e.g. mislabeling a lion a library). Here we show a related result: it is easy to produce images that are completely unrecognizable to humans, but that state-of-the-art DNNs believe to be recognizable objects with 99.99% confidence (e.g. labeling with certainty that white noise static is a lion). Specifically, we take convolutional neural networks trained to perform well on either the ImageNet or MNIST datasets and then find images with evolutionary algorithms or gradient ascent that DNNs label with high confidence as belonging to each dataset class. It is possible to produce images totally unrecognizable to human eyes that DNNs believe with near certainty are familiar objects. Our results shed light on interesting differences between human vision and current DNNs, and raise questions about the generality of DNN computer vision.

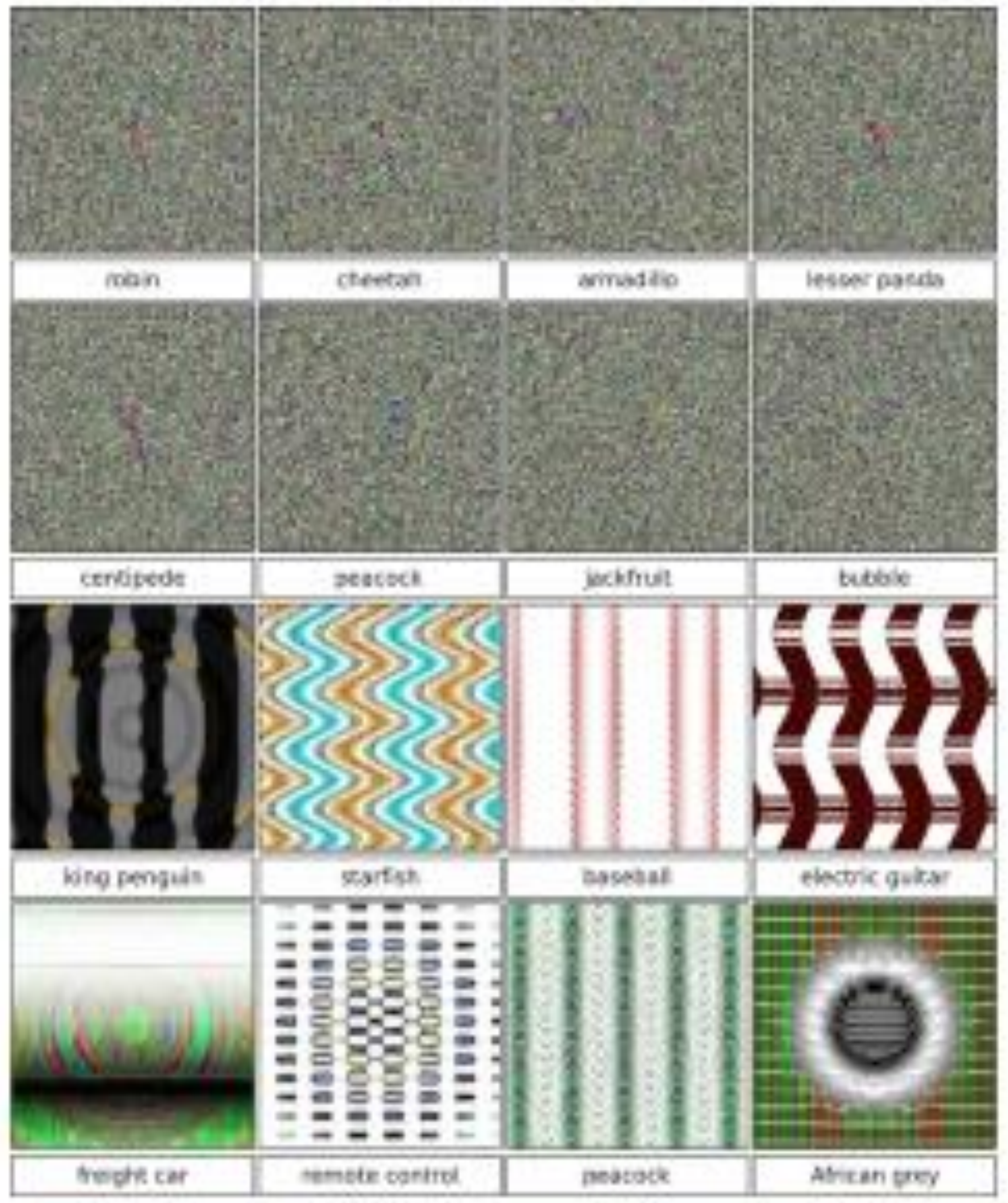


Figure 1. Evolved images that are unrecognizable to humans, but that state-of-the-art DNNs trained on ImageNet believe with $\geq$ 99.6% certainty to be a familiar object. This result highlights differences between how DNNs and humans recognize objects. Images are either directly (*top*) or indirectly (*bottom*) encoded.

Intriguing properties of neural networks

Christian Szegedy
Google Inc.

Wojciech Zaremba
New York University

Ilya Sutskever
Google Inc.

Joan Bruna
New York University

Dumitru Erhan
Google Inc.

Ian Goodfellow
University of Montreal

Rob Fergus
New York University
Facebook Inc.

Abstract

Deep neural networks are highly expressive models that have recently achieved state of the art performance on speech and visual recognition tasks. While their expressiveness is the reason they succeed, it also causes them to learn uninterpretable solutions that could have counter-intuitive properties. In this paper we report two such properties.

First, we find that there is no distinction between individual high level units and random linear combinations of high level units, according to various methods of unit analysis. It suggests that it is the space, rather than the individual units, that contains the semantic information in the high layers of neural networks.

Second, we find that deep neural networks learn input-output mappings that are fairly discontinuous to a significant extent. We can cause the network to misclassify an image by applying a certain hardly perceptible perturbation, which is found by maximizing the network's prediction error. In addition, the specific nature of these perturbations is not a random artifact of learning: the same perturbation can cause a different network, that was trained on a different subset of the dataset, to misclassify the same input.

Resources: First Learn Some ML

1. Ng's course in Coursera
2. Ng's course in Stanford
3. Mitchell's course in CMU
4. Abu-Mostafa's course in Caltech

1. Mitchell – Machine Learning
2. Duda & Hart – Pattern Classification
3. Bishop – Pattern Recognition for ML
4. Hastie, Tibshirani & Friedman – The Elements of Statistical Learning
5. Murphy – ML: A Probabilistic Perspective

And then, jump into Deep Learning

1 Hinton’s course in Coursera

2 Li’s course in Stanford

3 Sochers’ course in Stanford

4 de Freitas’ course in Oxford

1 Goodfellow, Bengio and Courville – Deep Learning

2 Stanford’s tutorial in DL

3 A lot of other resources

Tools

1 Theano (Python)

2 Caffe (C++)

3 Torch (Lua)

4 Cuda

5 DeepLearning4j (Java)

Thank You!