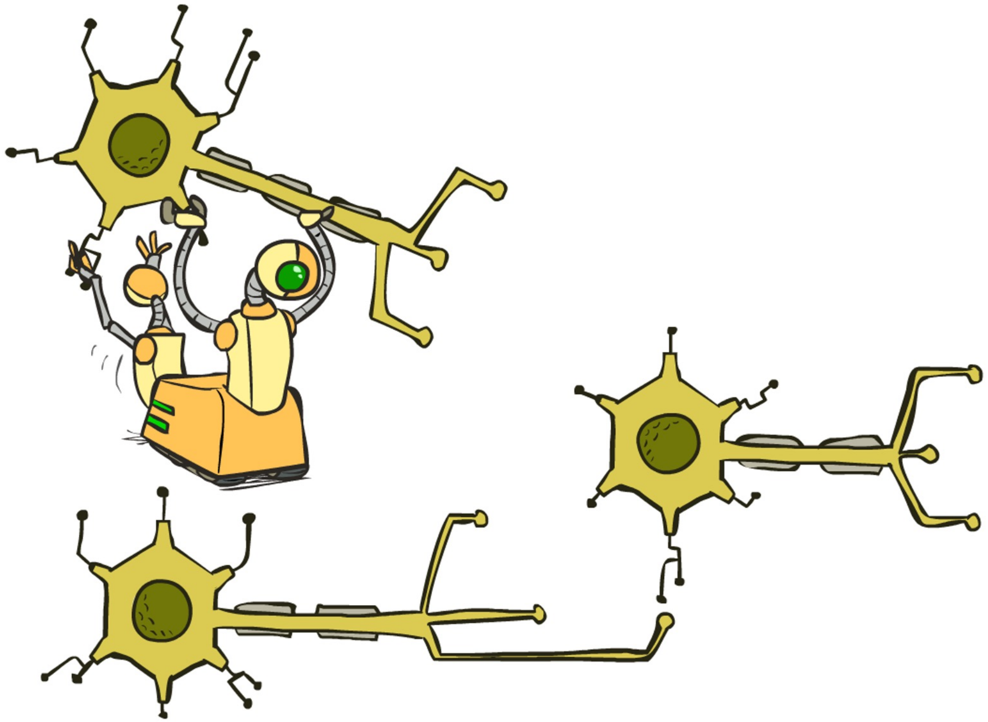
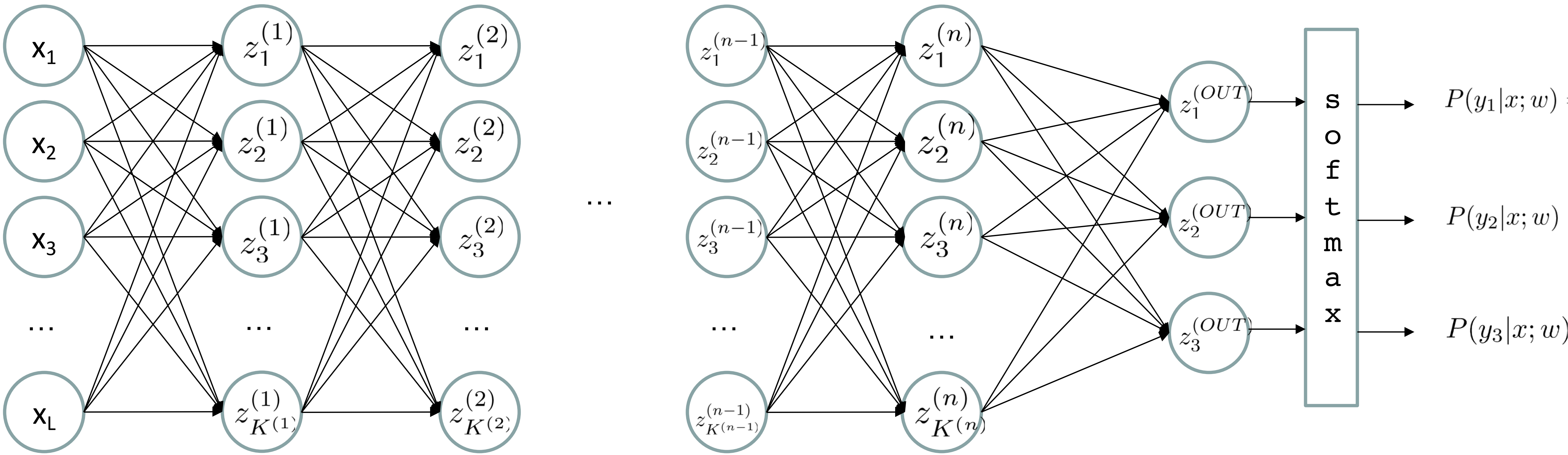
CS 188: Artificial Intelligence

Neural Networks



[These slides were created by Dan Klein, Pieter Abbeel, Anca Dragan for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at http://ai.berkeley.edu.]

Recall: Deep Neural Network



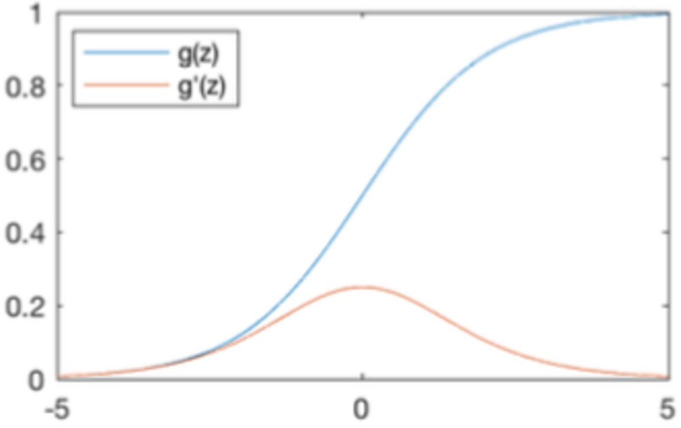
$$z_i^{(k)} = g\left(\sum_j W_{i,j}^{(k-1,k)} z_j^{(k-1)}\right)$$

g = nonlinear activation function

- Neural network with n layers
- $z^{(k)}$: activations at layer k
- $W^{(k-1,k)}$: weights taking activations from layer k-1 to layer k

Recall: Common Activation Functions

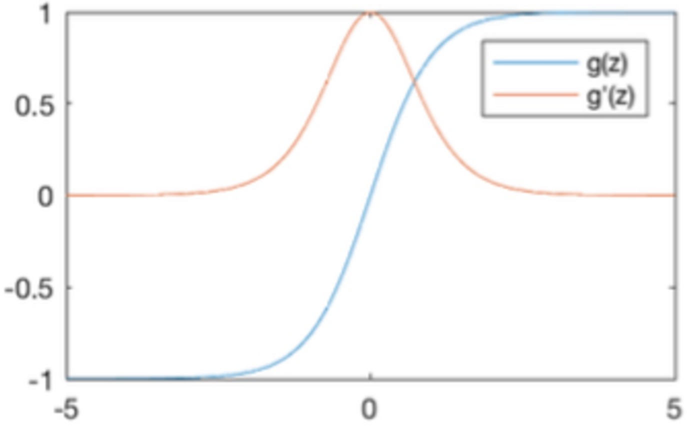
Sigmoid Function



$$g(z) = \frac{1}{1 + e^{-z}}$$

$$g'(z) = g(z)(1 - g(z))$$

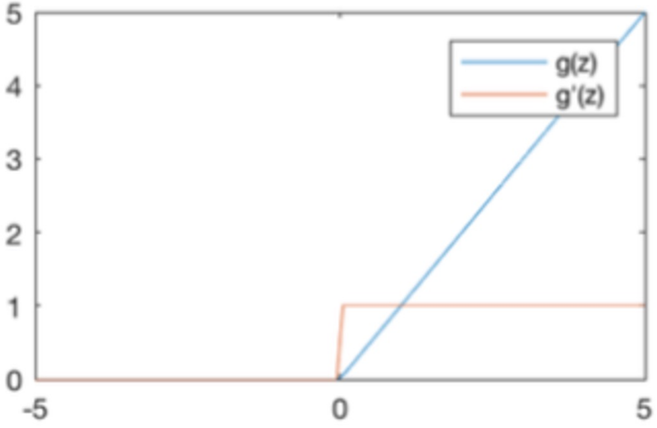
Hyperbolic Tangent



$$g(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$$

$$g'(z) = 1 - g(z)^2$$

Rectified Linear Unit (ReLU)

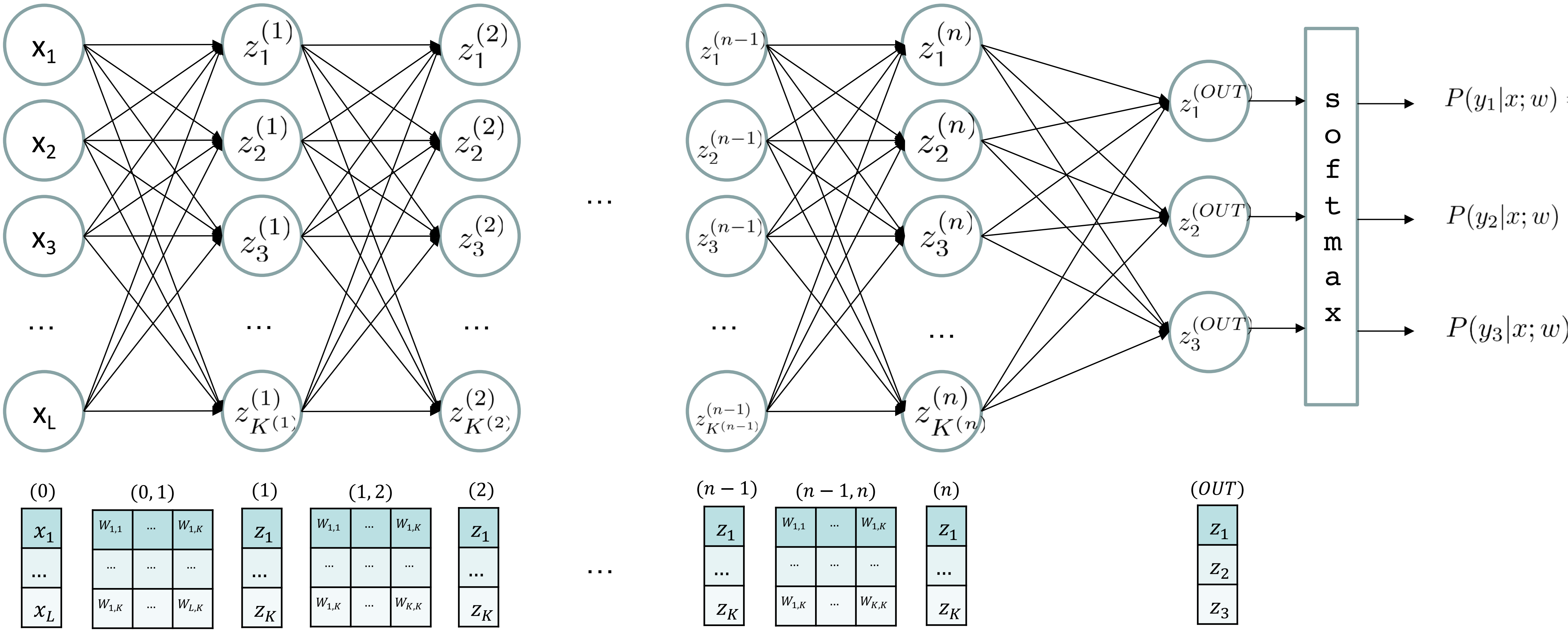


$$g(z) = \max(0, z)$$

$$g'(z) = \begin{cases} 1, & z > 0 \\ 0, & \text{otherwise} \end{cases}$$

[source: MIT 6.S191 introtodeeplearning.com]

Recall: Deep Neural Network



More compactly as matrix multiplication: $\boxed{z^{(k)} = g(W^{(k-1,k)} z^{(k-1)})}$

Recall: Deep Neural Network Training

Training the deep neural network is just like logistic regression:

$$\max_w \quad ll(w) = \max_w \quad \sum_i \log P(y^{(i)} | x^{(i)}; w)$$

just w tends to be a much, much larger vector ☺️

-> just run gradient ascent

+ stop when log likelihood of hold-out data starts to decrease

Batch Gradient Ascent on the Log Likelihood Objective

$$\max_w \quad ll(w) = \max_w \quad \underbrace{\sum_i \log P(y^{(i)}|x^{(i)};w)}_{g(w)}$$

```
init w
for iter = 1, 2, …
```

$$w \leftarrow w + \alpha * \sum_i \nabla \log P(y^{(i)}|x^{(i)};w)$$

Recall: How about computing all the derivatives?

- But neural net f is never one of those?
  - No problem: CHAIN RULE:

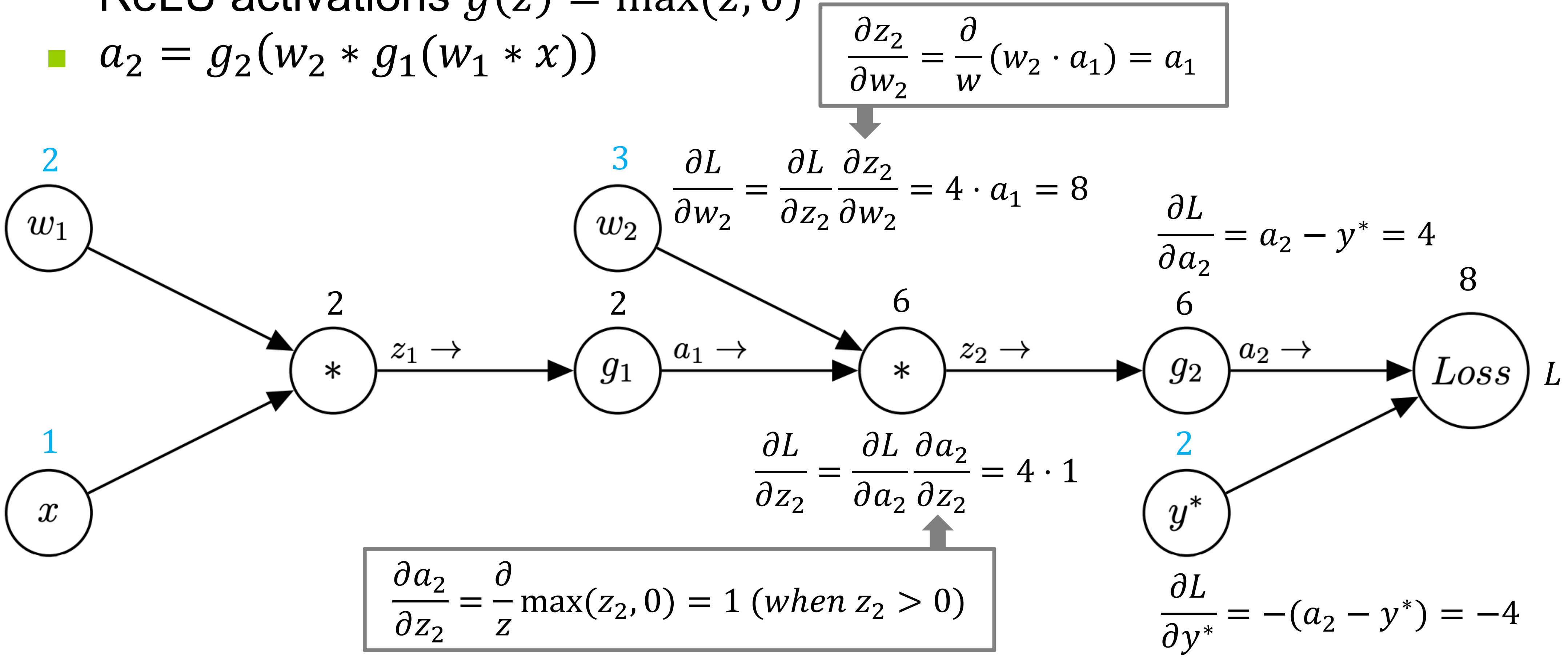
If $$f(x) = g(h(x))$$

Then $$f'(x) = g'(h(x))h'(x)$$

Derivatives can be computed by following well-defined procedures

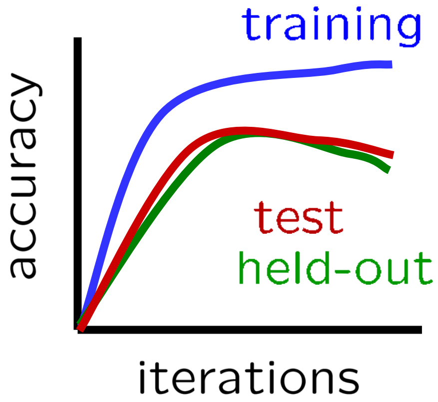
Example: Automatic Differentiation*

- Build a *computation graph* and use chain rule: $f(x) = g(h(x)) \quad f'(x) = g'(h(x))h'(x)$
- Example: neural network with quadratic loss $L(a_2, y^*) = \frac{1}{2}(a_2 - y^*)^2$ and ReLU activations $g(z) = \max(z, 0)$
- $a_2 = g_2(w_2 * g_1(w_1 * x))$



Preventing Overfitting in Neural Networks

Early stopping:



Weight regularization

Weight Regularization

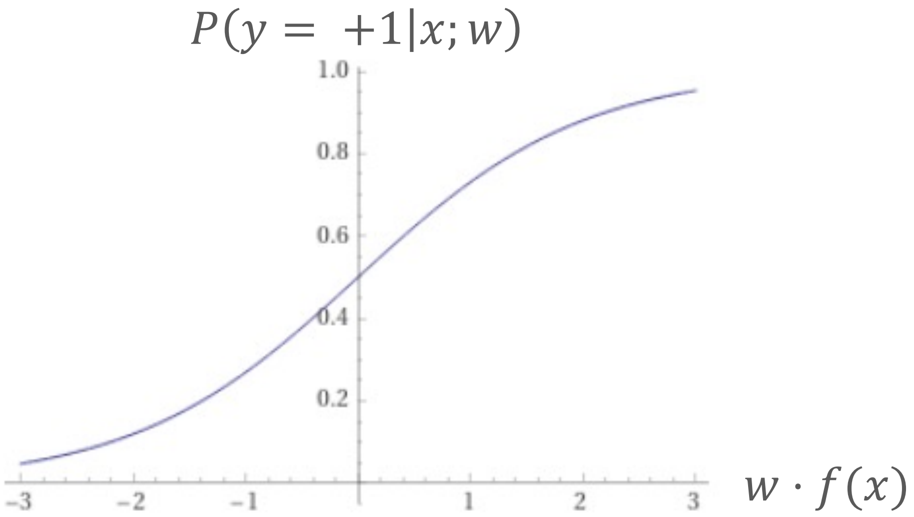
What can go wrong when we maximize log-likelihood?

Example: logistic regression

$$\max_w \sum_i \log P(y^{(i)}|x^{(i)};w)$$

- $P(y = +1|x;w) = \frac{1}{1+e^{-w\cdot f(x)}}$
- $P(y = -1|x;w) = 1 - \frac{1}{1+e^{-w\cdot f(x)}}$

w can grow very large and lead to overfitting and learning instability



Weight Regularization

What can go wrong when we maximize log-likelihood?

$$\max_{w} \quad \sum_{i} \log P(y^{(i)}|x^{(i)};w)$$

w can grow very large

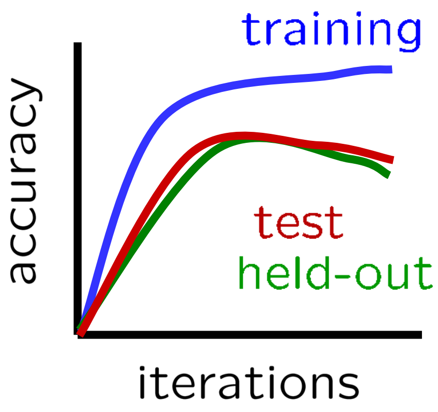
Solution: add an objective term to penalize weight magnitude

$$\max_{\mathrm{w}} \sum_{i} \log P(y^{(i)}|x^{(i)};w) - \frac{\lambda}{2}\sum_{j} w_j^2$$

λ is a hyperparameter (typically 0.1 to 0.0001 or smaller)

Preventing Overfitting in Neural Networks

Early stopping:



Weight regularization: $\max_{w} \sum_i \log P(y^{(i)}|x^{(i)}; w) - \frac{\lambda}{2}\sum_j w_j^2$

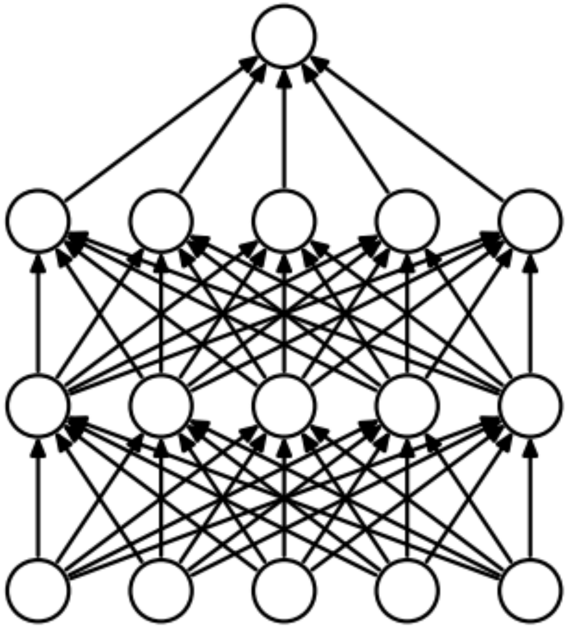
Dropout

Dropout*

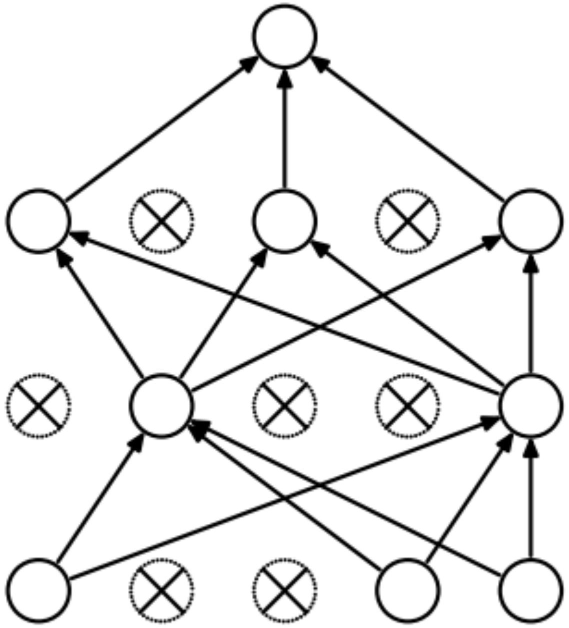
"Damage" the network during training to encourage redundancy

At each training step, with probability *(1-p)* set an activation to zero (drop it)

After training, don't drop, but multiply weights by *p*



(a) Standard Neural Net

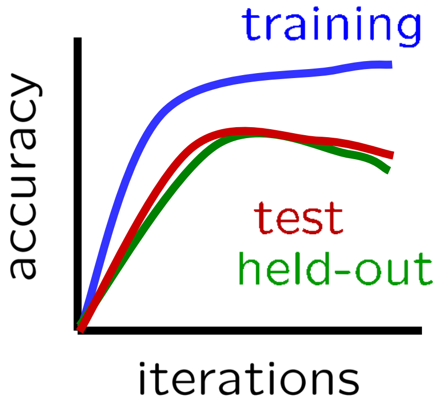


(b) After applying dropout.

Srivastava et al, 2014

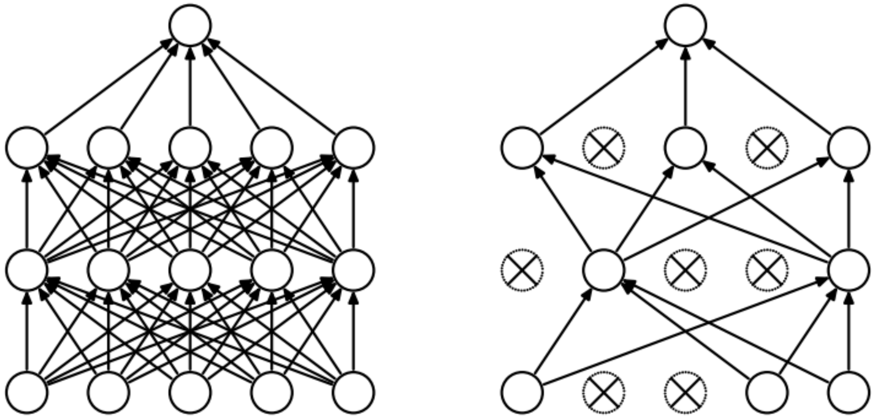
Preventing Overfitting in Neural Networks

Early stopping:



Weight regularization: $\max_w \sum_i \log P(y^{(i)} | x^{(i)}; w) - \frac{\lambda}{2} \sum_j w_j^2$

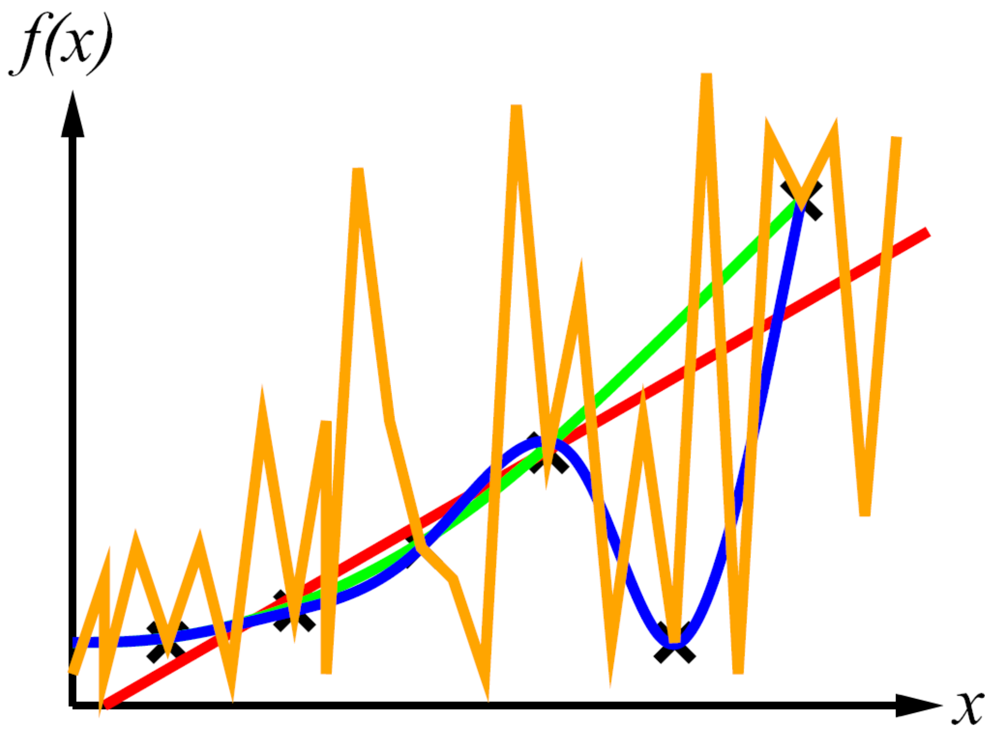
Dropout:



(a) Standard Neural Net (b) After applying dropout.

Consistency vs. Simplicity

- Example: curve fitting (regression, function approximation)



- Consistency vs. simplicity
- Ockham's razor

Consistency vs. Simplicity

- Fundamental tradeoff: bias vs. variance
- Usually algorithms prefer consistency by default (why?)
- Several ways to operationalize “simplicity”
  - Reduce the hypothesis/model space
    - Assume more: e.g. independence assumptions, as in naïve Bayes
    - Fewer features or neurons
    - Other limits on model structure
  - Regularization
    - Laplace Smoothing: cautious use of small counts
    - Small weight vectors in neural networks (stay close to zero-mean prior)
    - Hypothesis space stays big, but harder to get to the outskirts

Fun Neural Net Demo Site

Demo-site:

http://playground.tensorflow.org/

Summary of Key Ideas

Optimize probability of label given input

$$\max_w \quad ll(w) = \max_w \quad \sum_i \log P(y^{(i)}|x^{(i)}; w)$$

Continuous optimization

Gradient ascent:

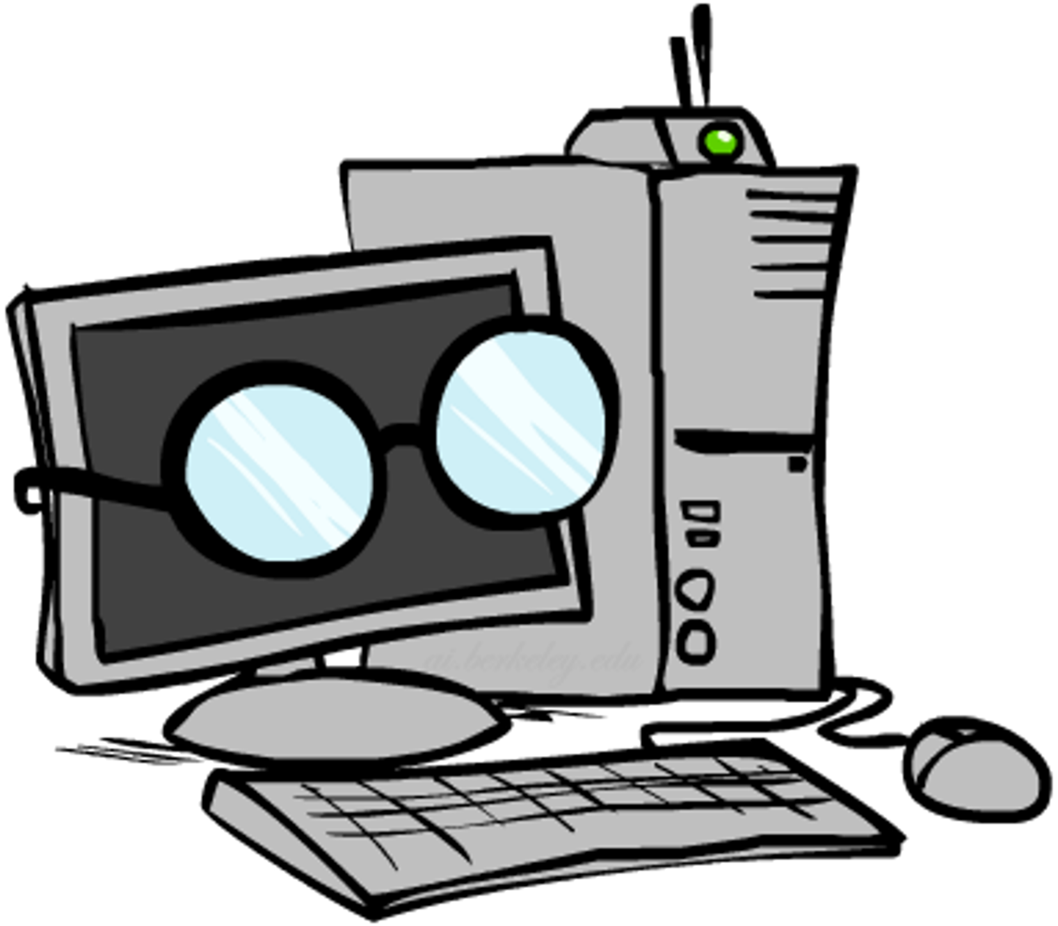
- Compute steepest uphill direction = gradient (= just vector of partial derivatives)
- Take step in the gradient direction
- Repeat (until held-out data accuracy starts to drop = "early stopping")

Deep neural nets

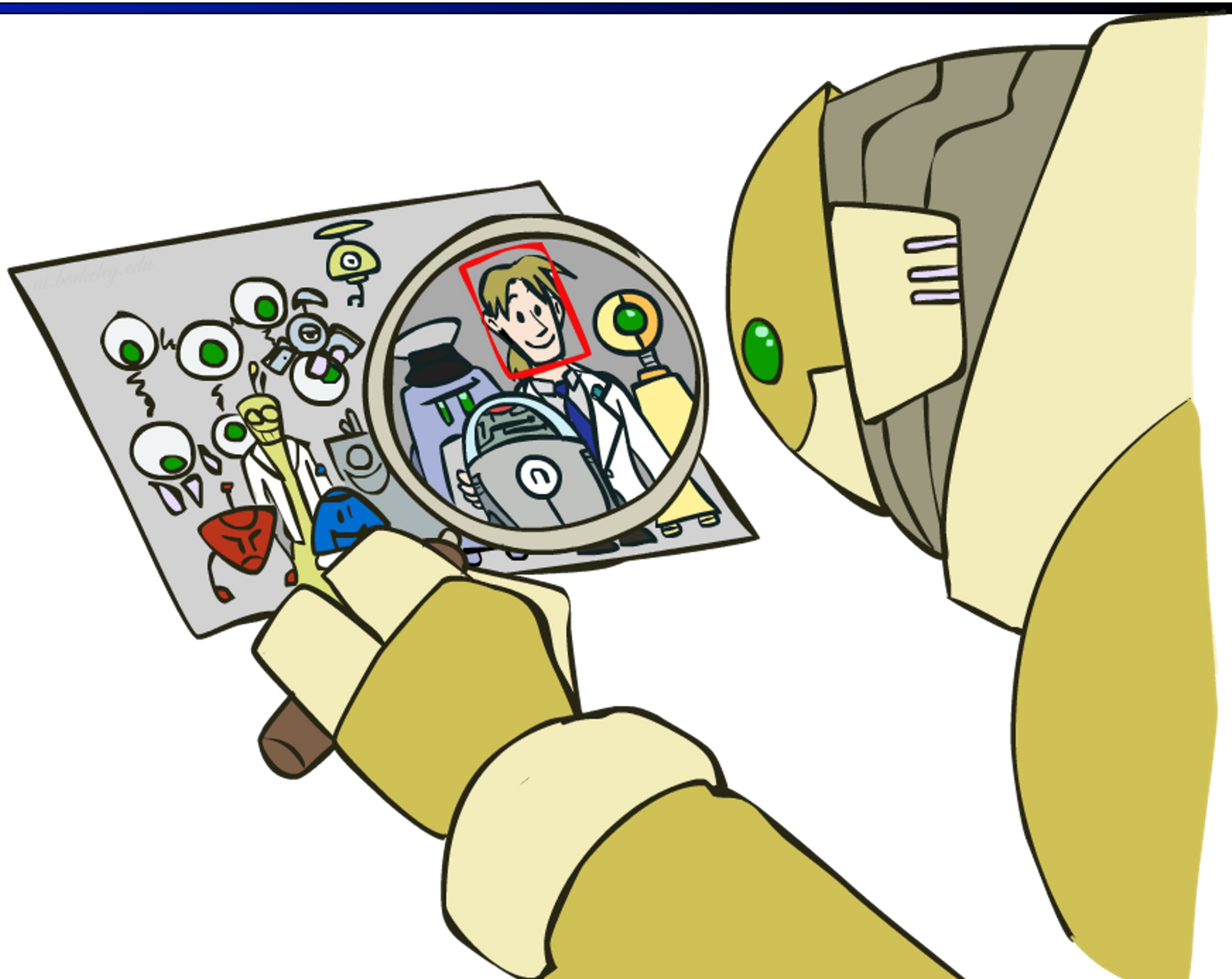
- Last layer = still logistic regression
- Now also many more layers before this last layer
  - = computing the features
  - the features are learned rather than hand-designed
- Universal function approximation theorem
  - `If` neural net is large enough
  - `Then` neural net can represent any continuous mapping from input to output with arbitrary accuracy
  - But remember: need to avoid overfitting / memorizing the training data ? early stopping!
- Automatic differentiation gives the derivatives efficiently (how? = outside of scope of 188)

How well does deep learning work?

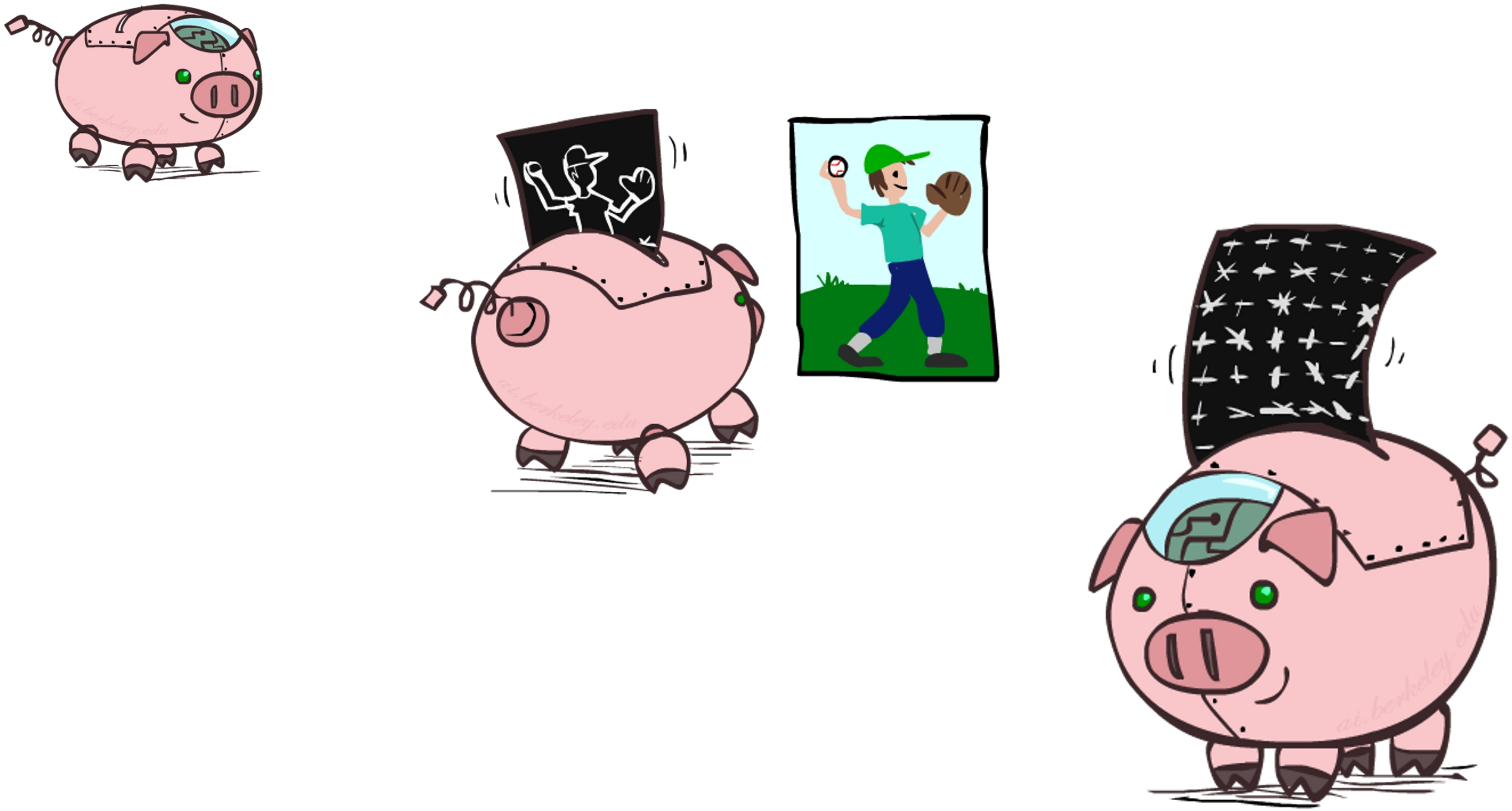
Computer Vision



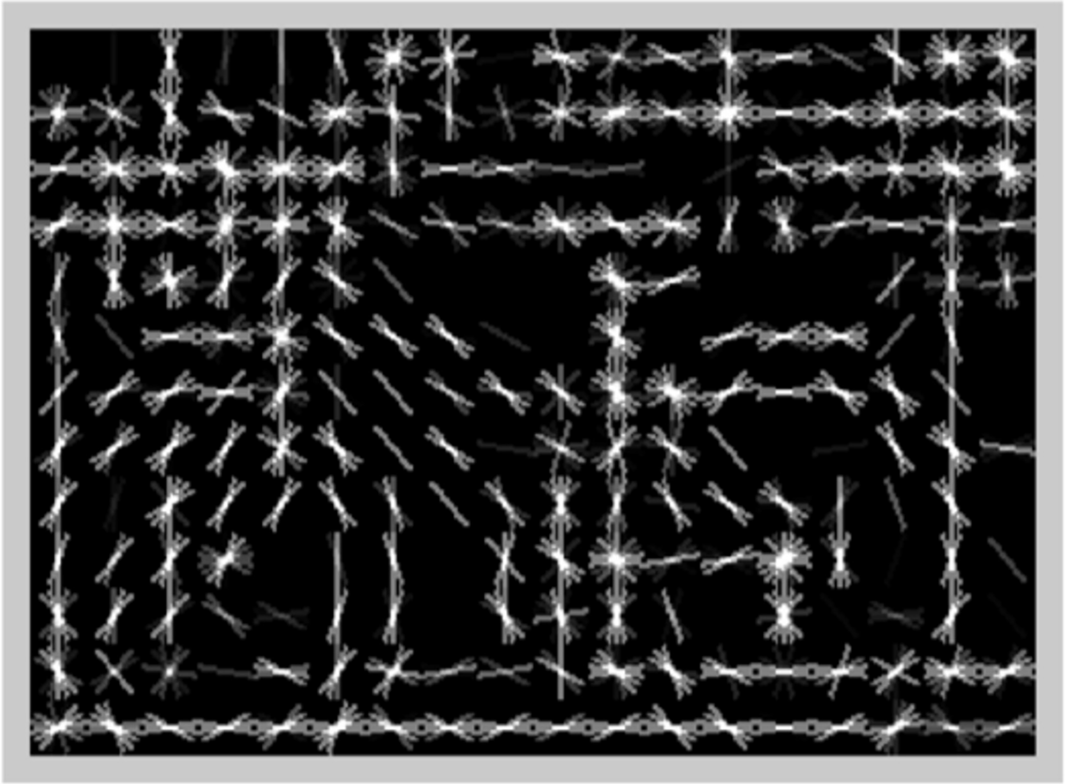
Object Detection



Manual Feature Design



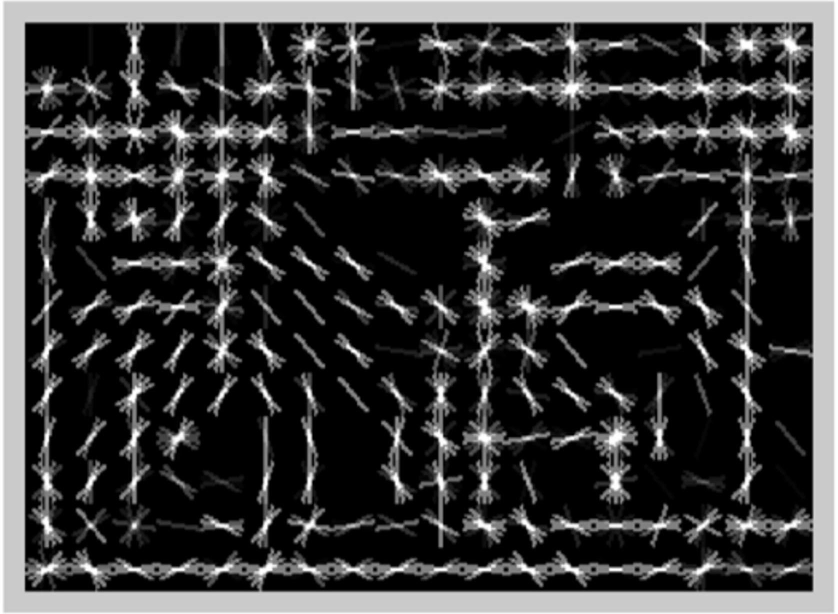
Features and Generalization



[HoG: Dalal and Triggs, 2005]

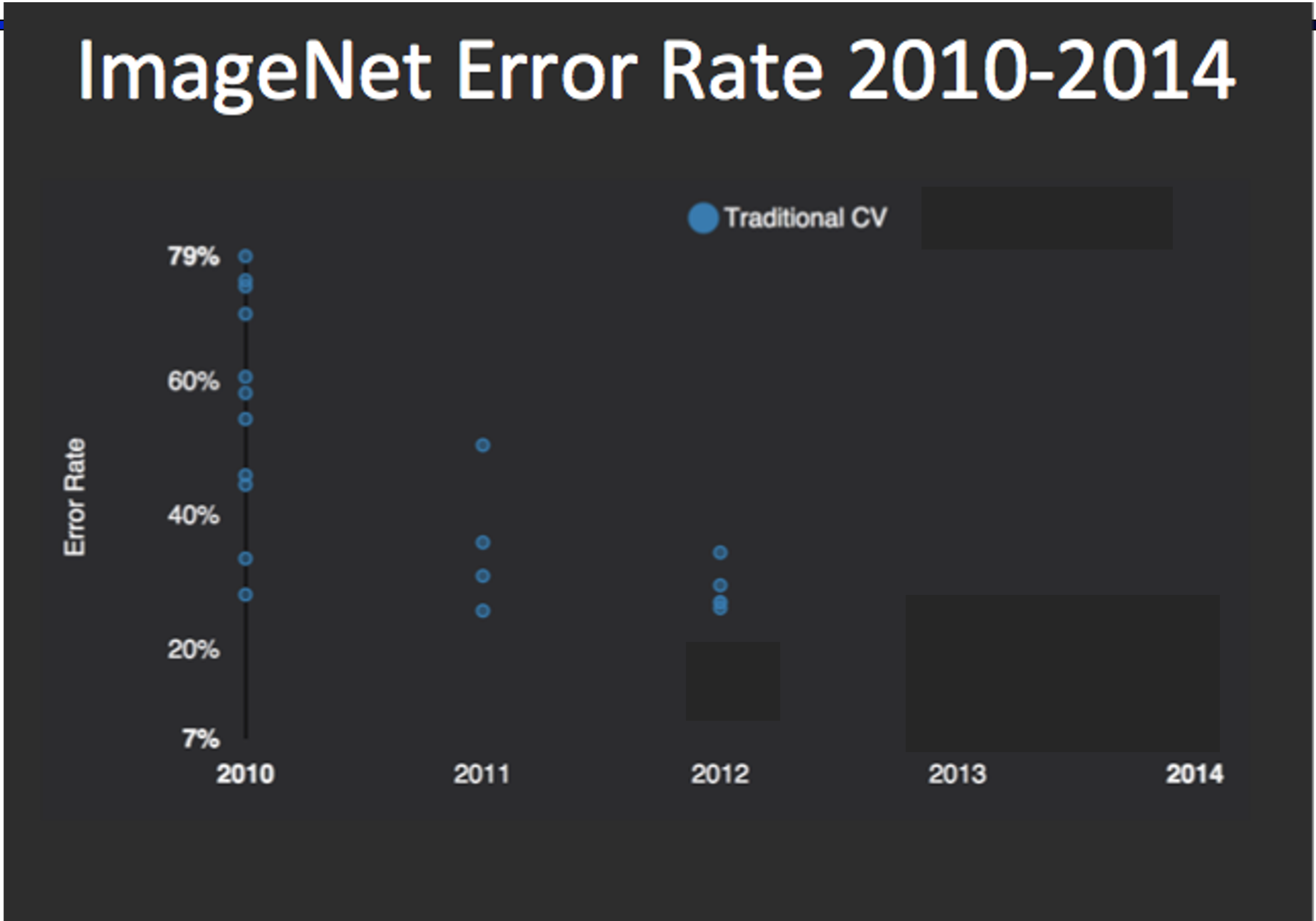
Features and Generalization

Image



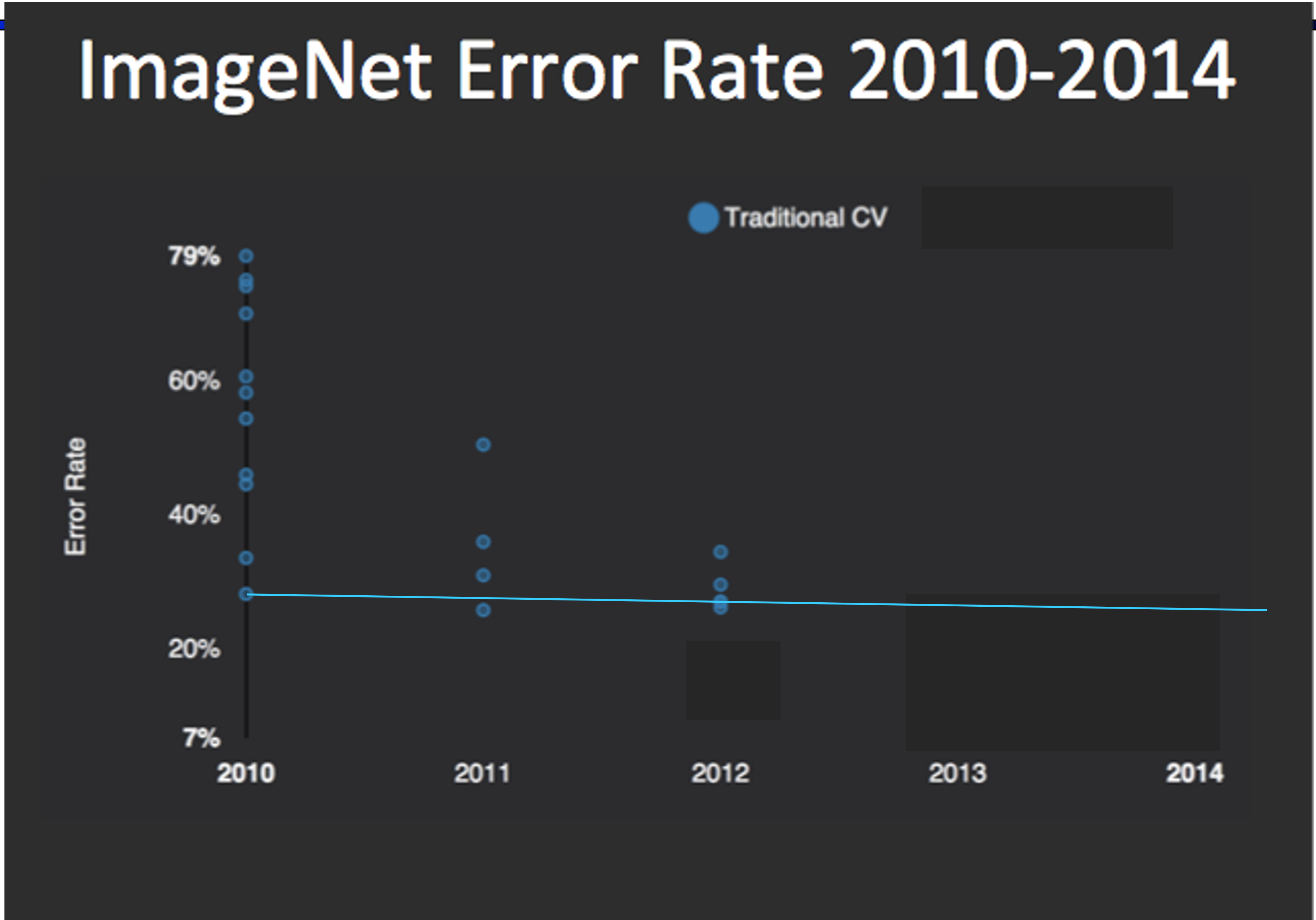
HoG

Performance



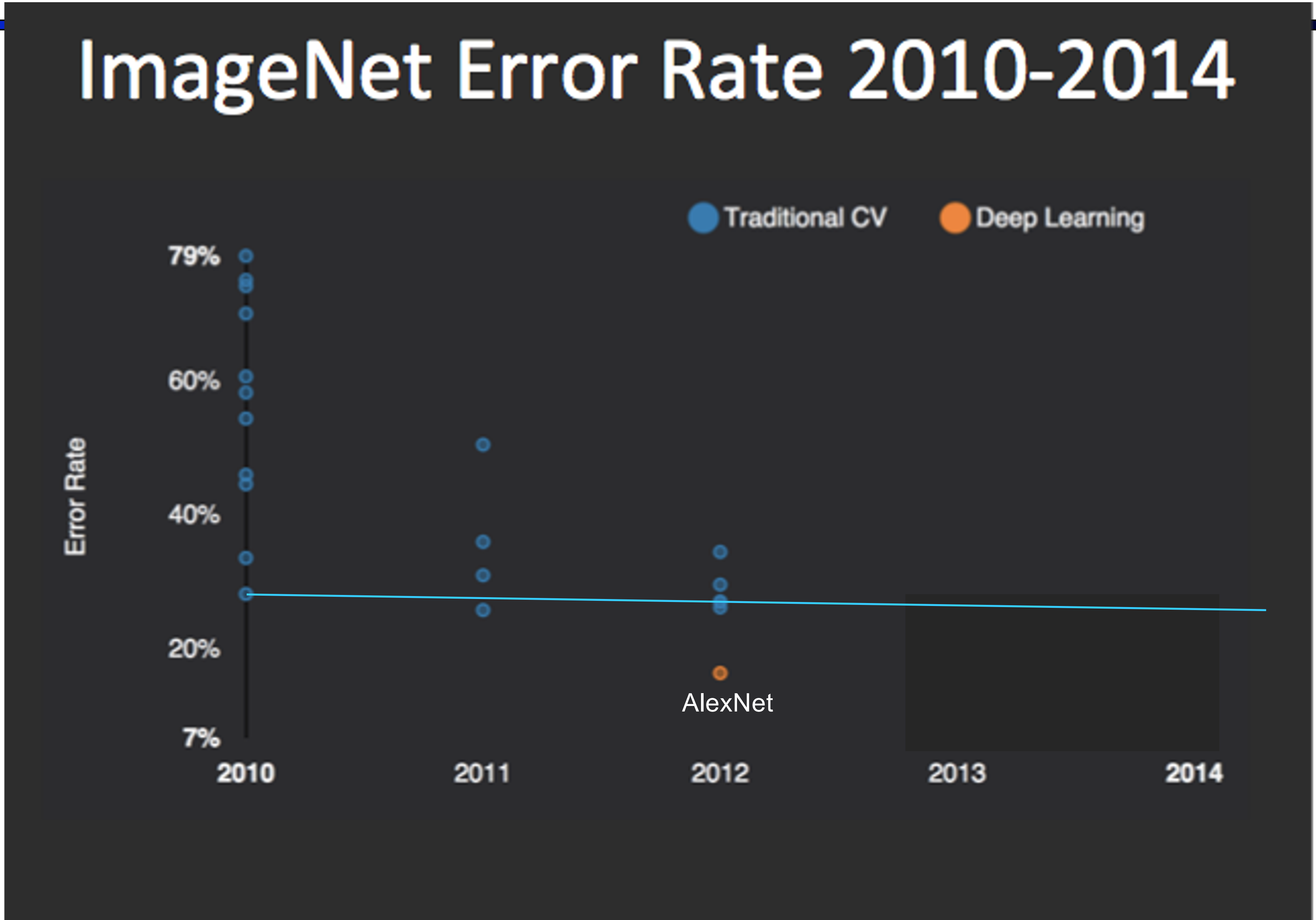
graph credit Matt Zeiler, Clarifai

Performance



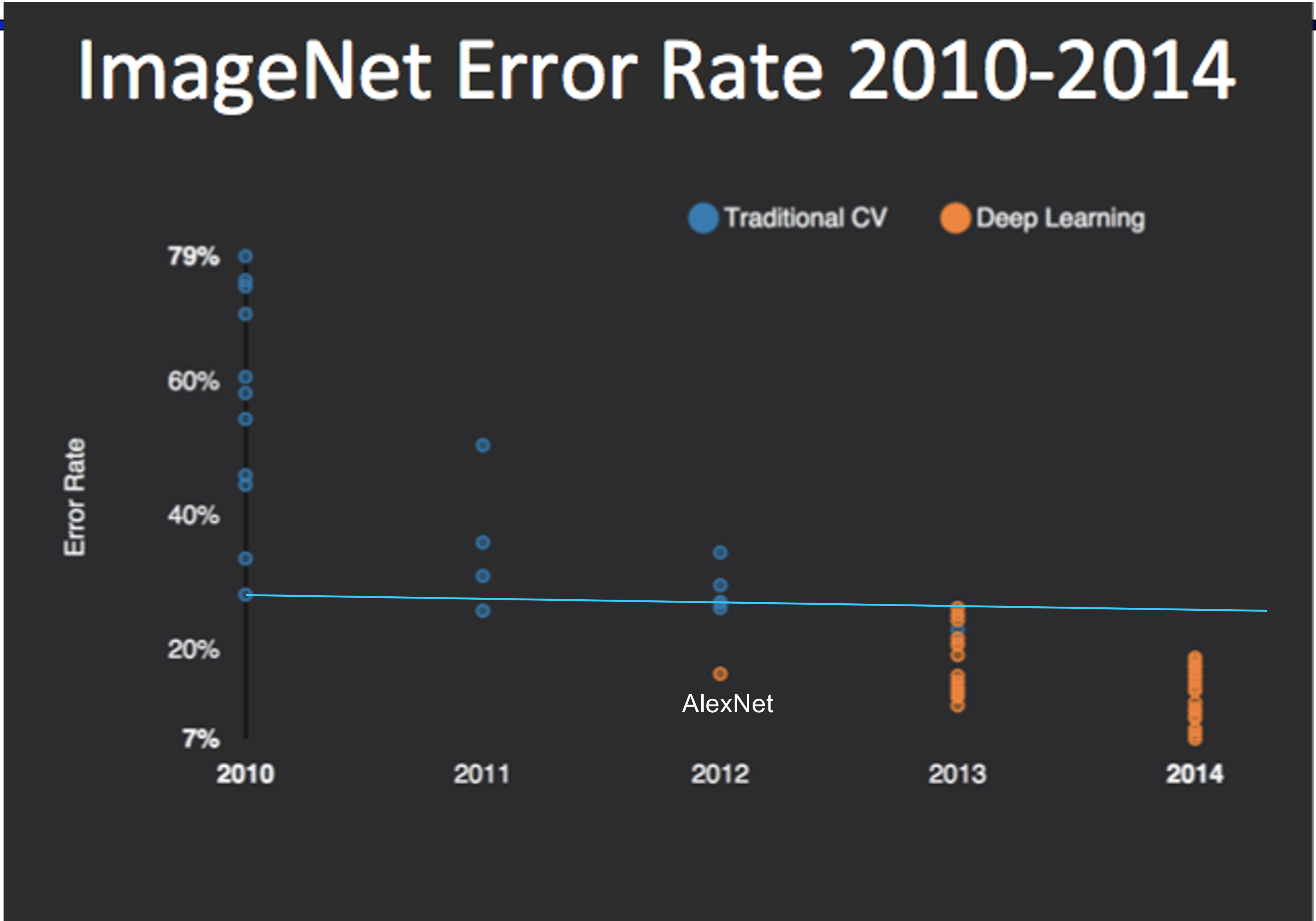
graph credit Matt Zeiler, Clarifai

Performance



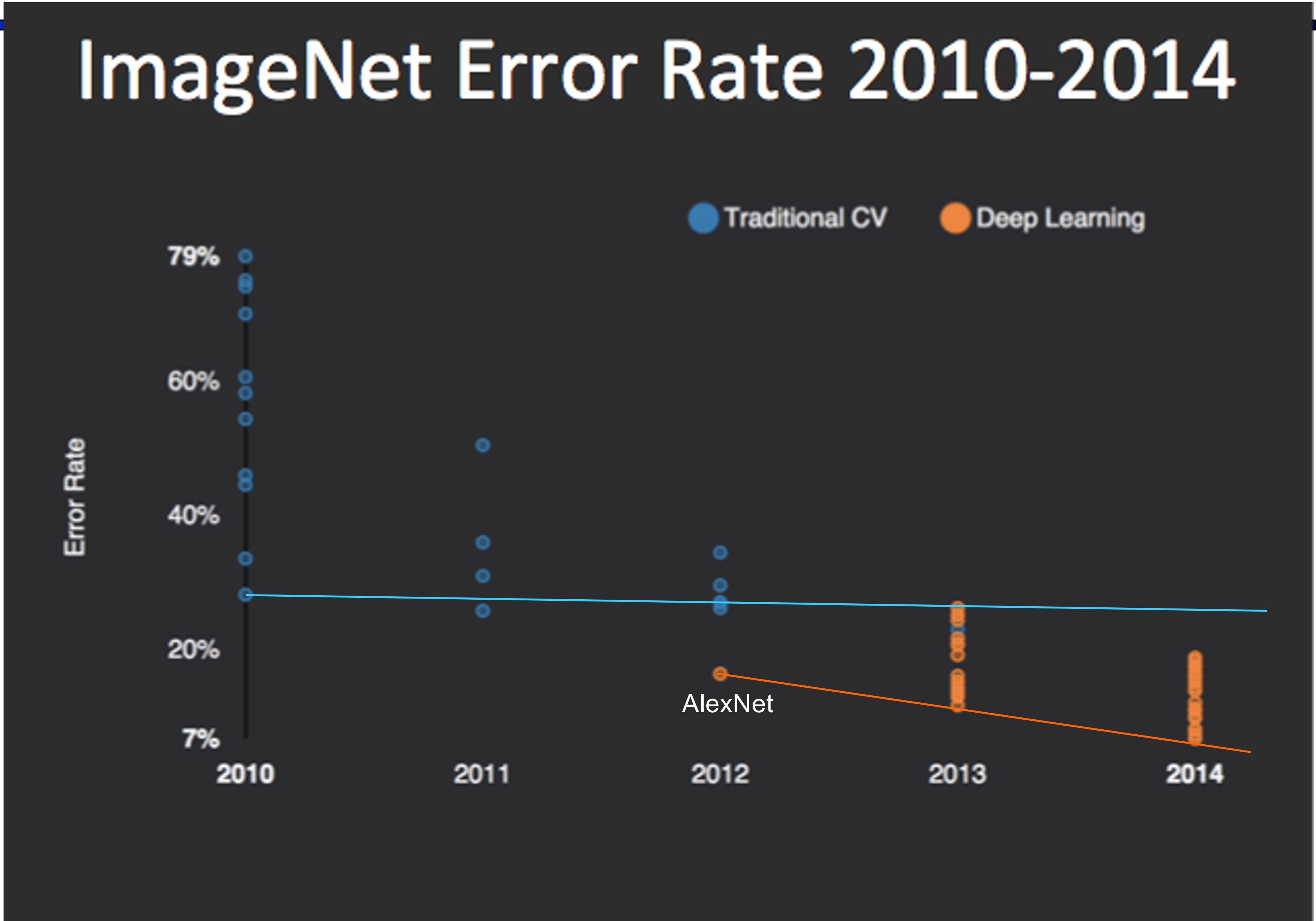
graph credit Matt Zeiler, Clarifai

Performance



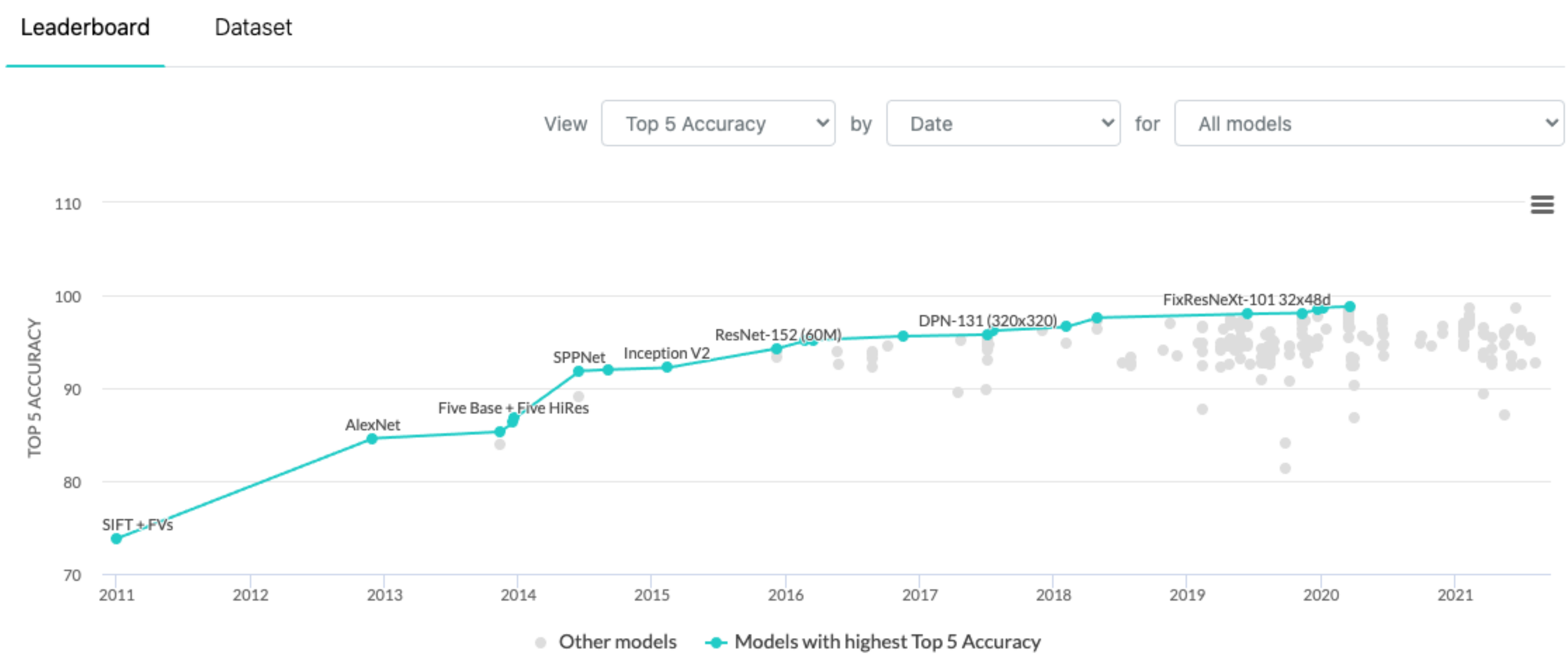
graph credit Matt Zeiler, Clarifai

Performance



graph credit Matt Zeiler, Clarifai

Papers With Code: ImageNet



MS COCO Image Captioning Challenge

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

"girl in pink dress is jumping in air."

"black and white dog jumps over bar."

"young girl in pink shirt is swinging on swing."

"man in blue wetsuit is surfing on wave."

Karpathy & Fei-Fei, 2015; Donahue et al., 2015; Xu et al, 2015; many more

Visual QA Challenge

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, Devi Parikh

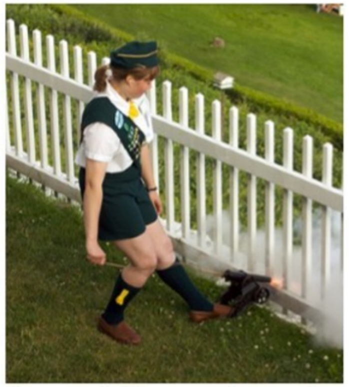
What vegetable is on the plate?
Neural Net: broccoli
Ground Truth: broccoli

What color are the shoes on the person's feet ?
Neural Net: brown
Ground Truth: brown

How many school busses are there?
Neural Net: 2
Ground Truth: 2

What sport is this?
Neural Net: baseball
Ground Truth: baseball

What is on top of the refrigerator?
Neural Net: magnets
Ground Truth: cereal



What uniform is she wearing?
Neural Net: shorts
Ground Truth: girl scout

What is the table number?
Neural Net: 4
Ground Truth:40

What are people sitting under in the back?
Neural Net: bench
Ground Truth: tent

Visual Dialogue

Alayrac et al, 2022

This is a picture of two teddy bears on the moon.

What are they doing?

They are having a conversation.

What object are they using?

It looks like a computer.

Is this surprising?

Yes, it is surprising.

Why is this picture surprising to you?

I think it is surprising because teddy bears are not usually found on the moon.

What is the common thing about these three images?

They are all flamingos.

What is the difference between these three images?

The first one is a cartoon, the second one is a real flamingo, and the third one is a 3D model of a flamingo.

This is an apple with a sticker on it.

What does the sticker say?

The sticker says "iPod".

Where is the photo taken?

It looks like it's taken in a backyard.

Do you think it is printed or handwritten?

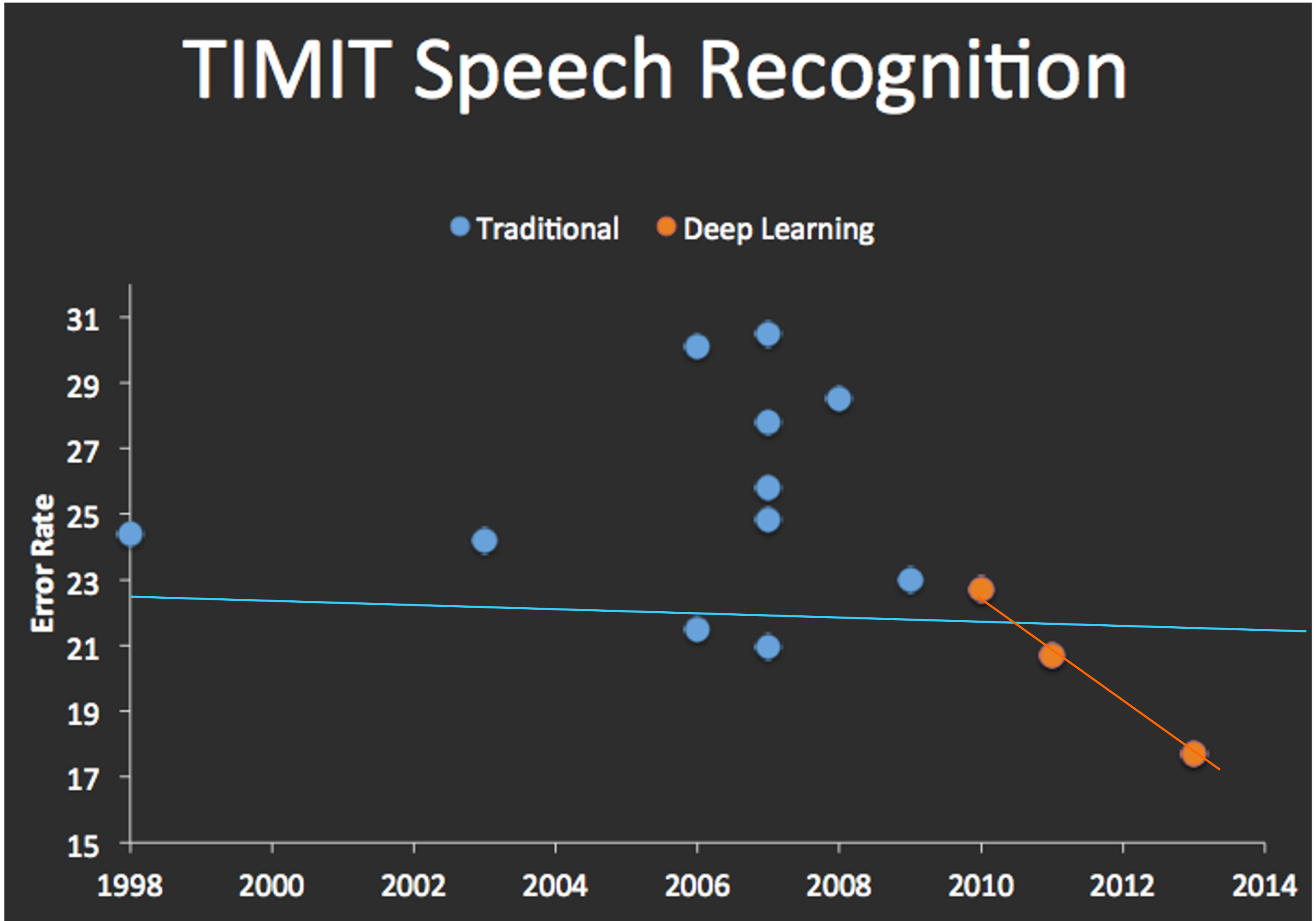
It looks like it's handwritten.

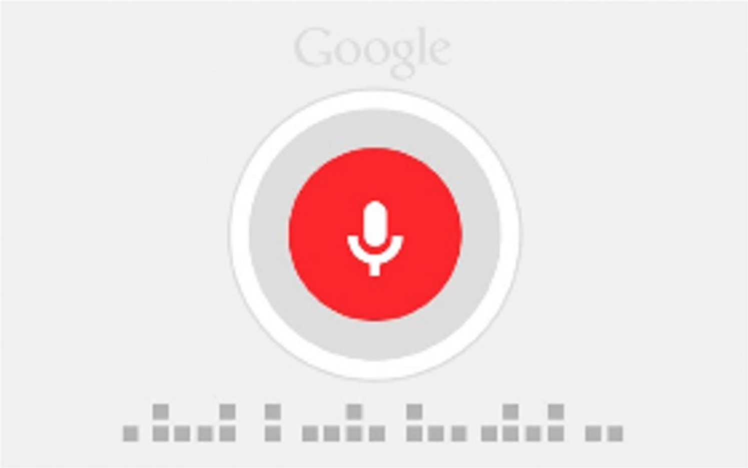
What color is the sticker?

It's white.

Image Segmentation

Speech Recognition

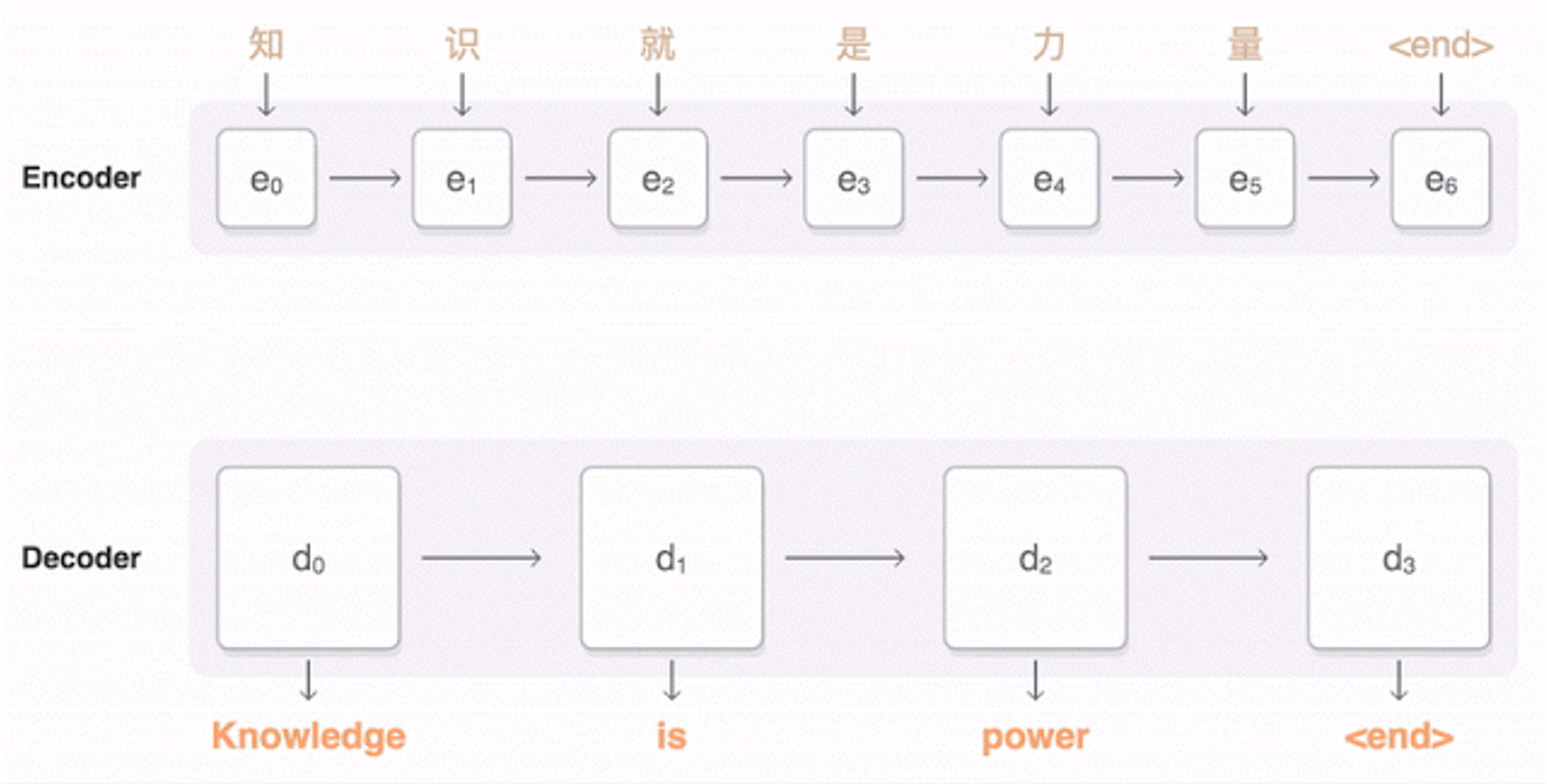




graph credit Matt Zeiler, Clarifai

Machine Translation

Google Neural Machine Translation (in production)



Google and DeepMind are using AI to predict the energy output of wind farms

To help make that energy more valuable to the power grid

By Nick Statt | @nickstatt | Feb 26, 2019, 2:42pm EST

SHARE

Google announced today that it has made energy produced by wind farms more viable using the artificial intelligence software of its London-based subsidiary DeepMind. By using DeepMind's machine learning algorithms to predict the wind output from the farms Google uses for its green energy initiatives, the company says it can now schedule set deliveries of energy output, which are more valuable to the grid than standard, non-time-based deliveries.

Change in Programming Paradigm?

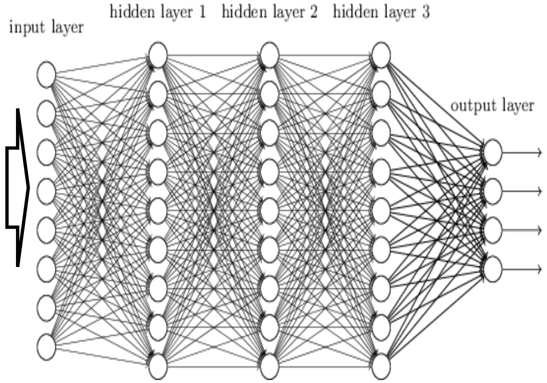
Traditional Programming:
program by writing lines of code

```
If Number = 1 Then
    Count1 = Count1 + 1
Else
    If Number = 2 Then
        Count2 = Count2 + 1
    Else
        If Number = 3 then
            Count3 = Count3 + 1
        Else
            CountX = CountX + 1
        End if
    End If
End If
```

?

Poor performance on AI problems

Deep Learning ("Software 2.0"):
program by providing data



?

Success!