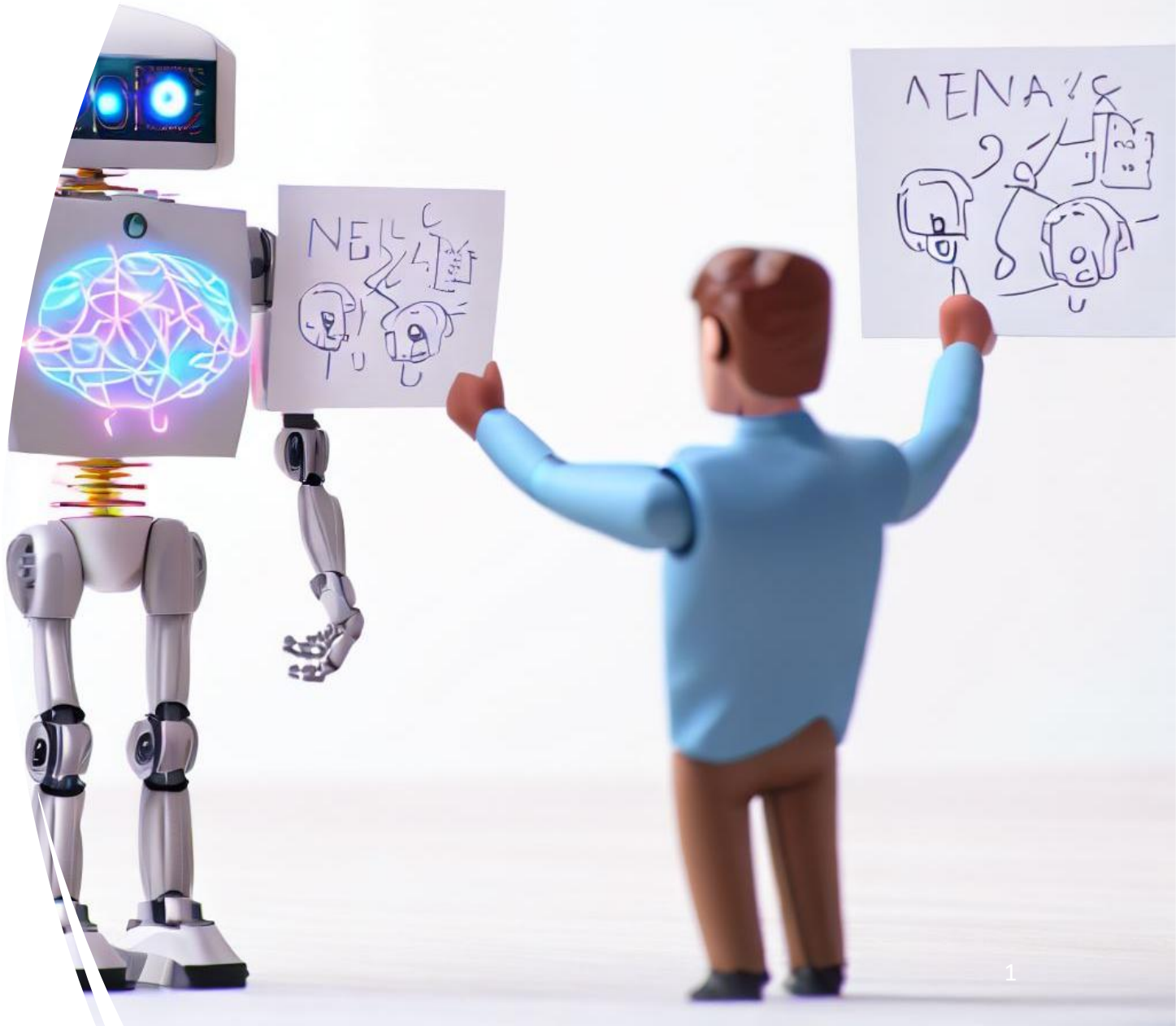
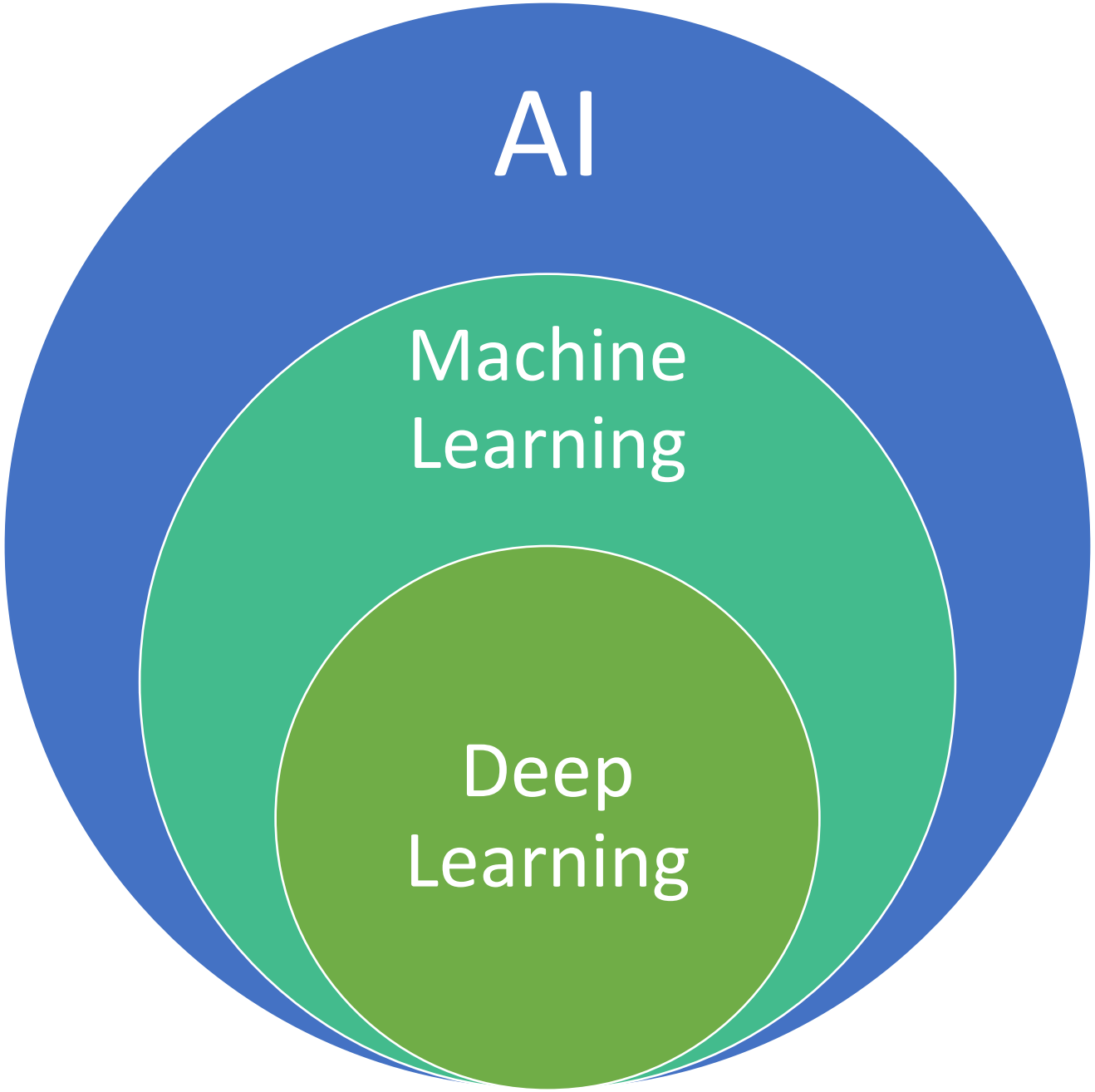
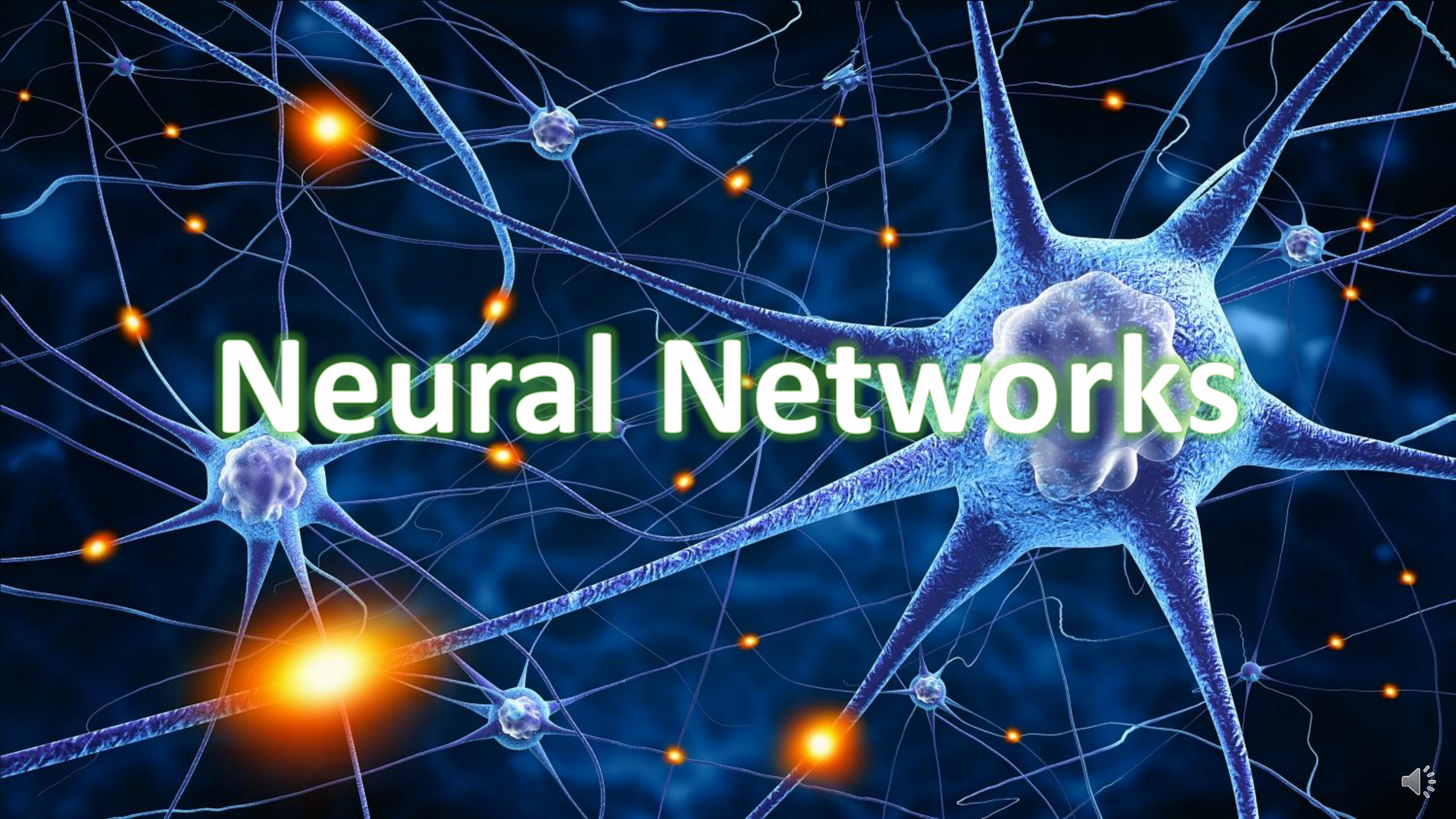
Introduction to Deep Learning

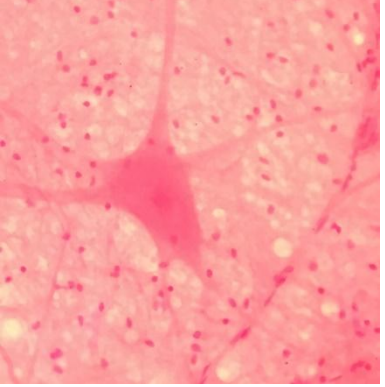
Prof. Kuan-Ting Lai

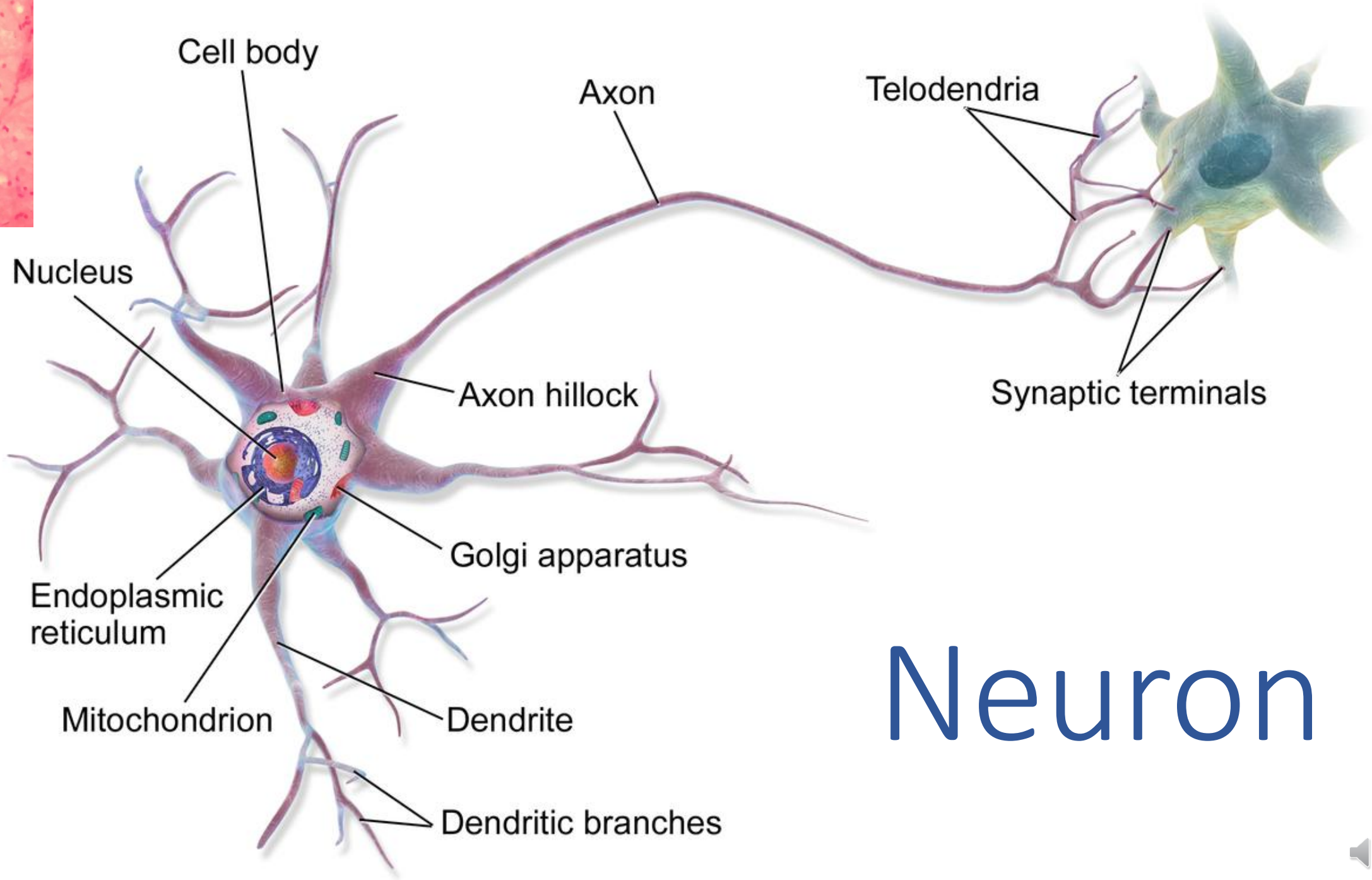
2023/9/19



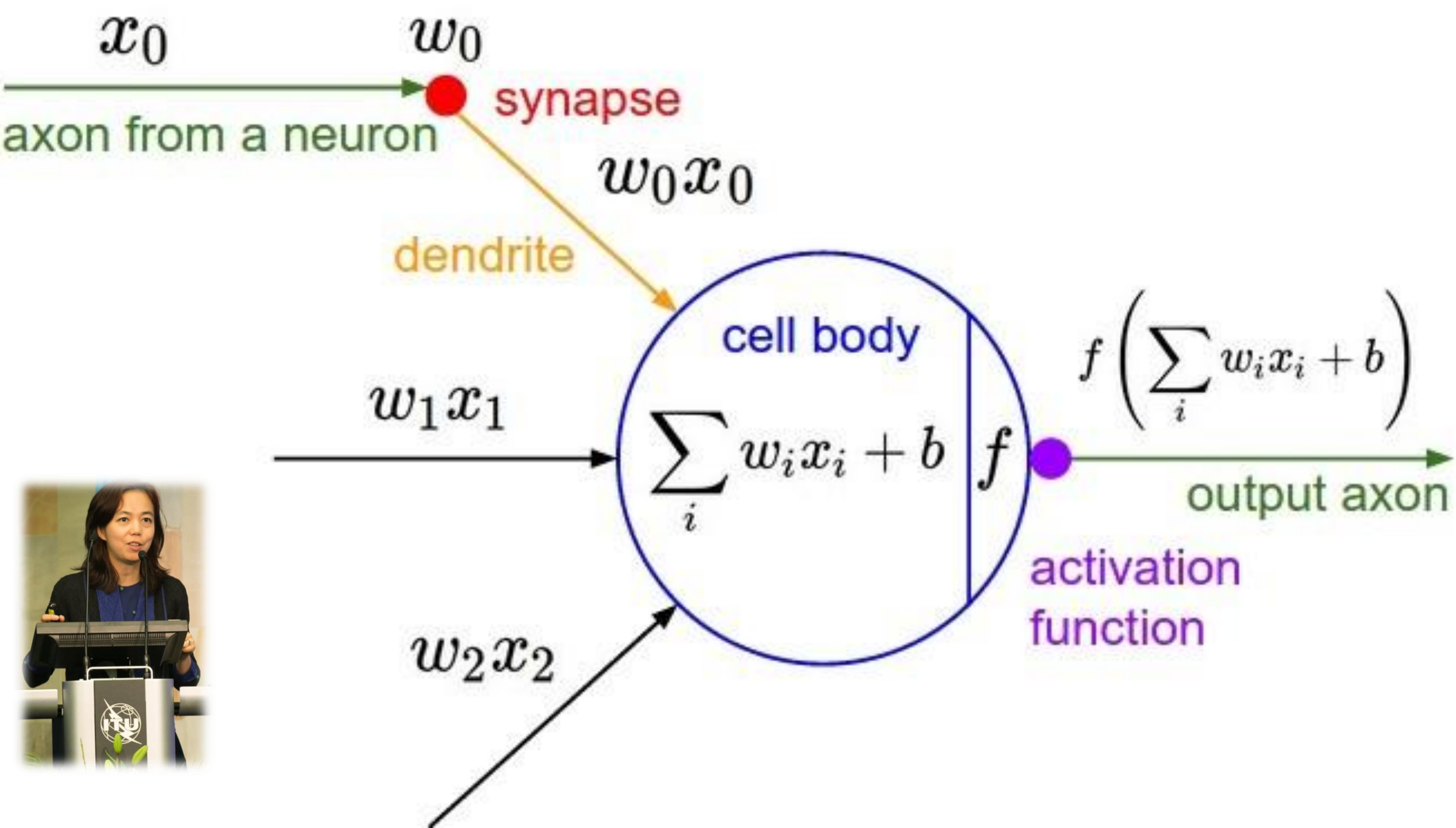

AI
Machine Learning
Deep Learning


Neural Networks



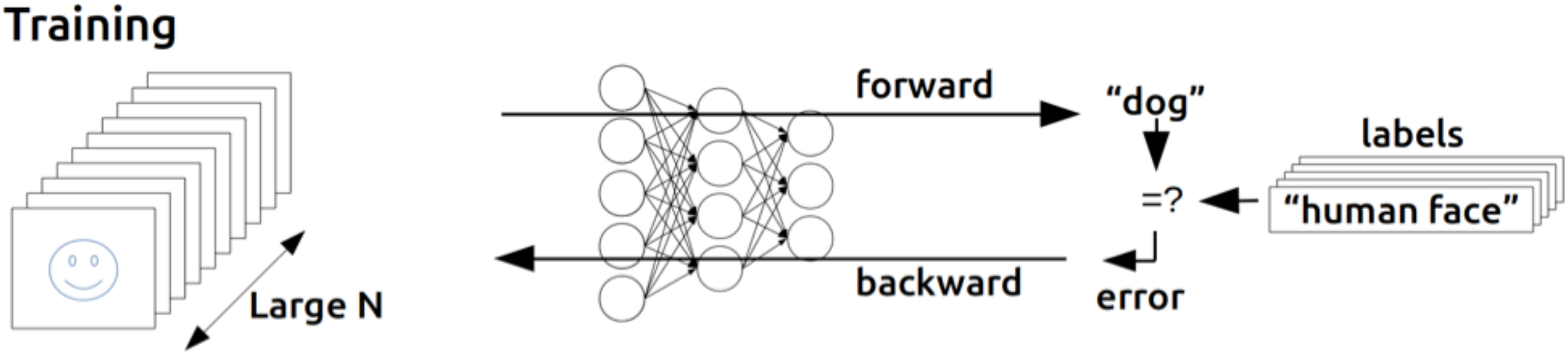

Cell body
Axon
Telodendria
Nucleus
Axon hillock
Synaptic terminals
Golgi apparatus
Endoplasmic reticulum
Mitochondrion
Dendrite
Dendritic branches

Neuron


x_0
w_0
synapse
axon from a neuron
$w_0 x_0$
dendrite
cell body
$w_1 x_1$
$\sum_i w_i x_i + b$
f
$f\left(\sum_i w_i x_i + b\right)$
output axon
activation function
$w_2 x_2$

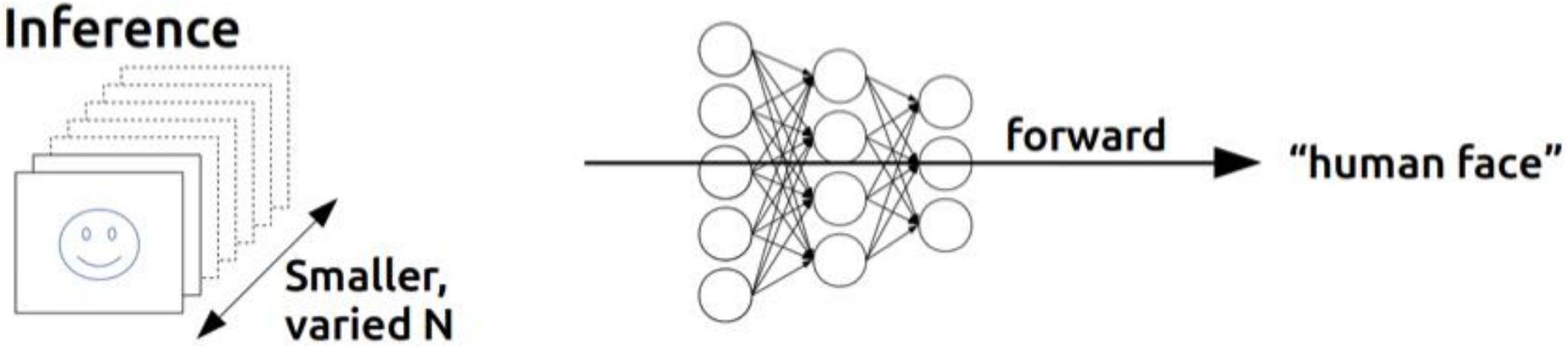
Learning (Training)

- Forward calculation + Backpropagation



Inference

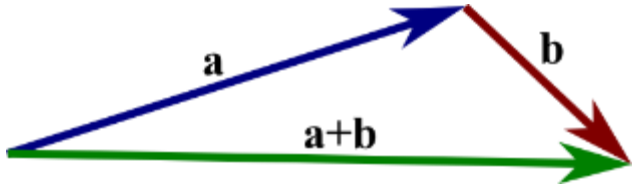
- Forward calculation



Single Variable vs. Multiple Variables

Linear Algebra

- Scalar
  - real numbers
- Vector (1D)
  - Has a magnitude & a direction



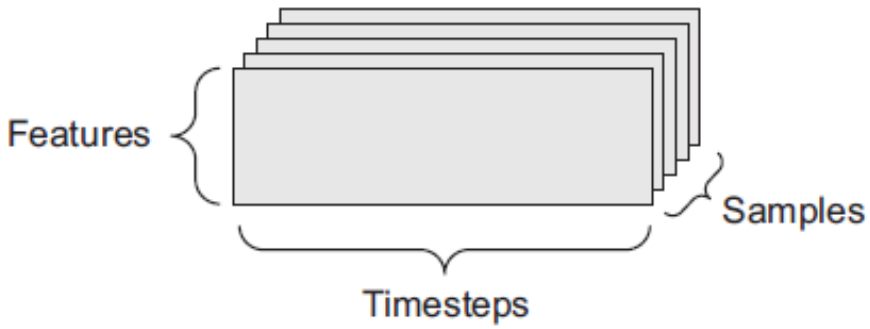
- Matrix (2D)
  - An array of numbers arranges in rows & columns

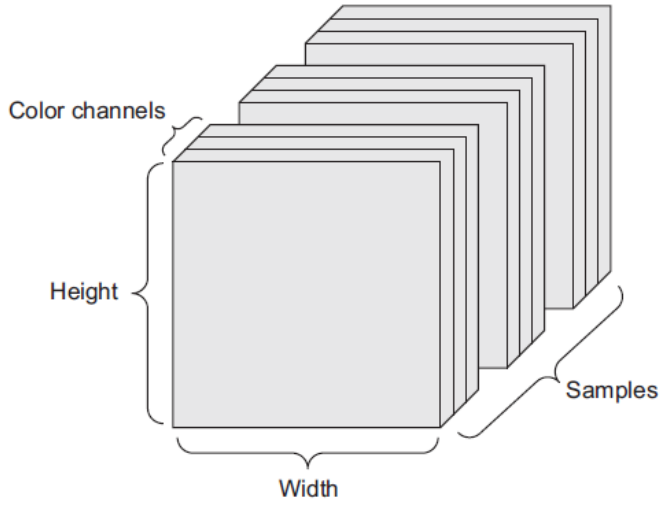
$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

- Tensor (>=3D)
  - Multi-dimensional arrays of numbers

Real-world examples of Data Tensors

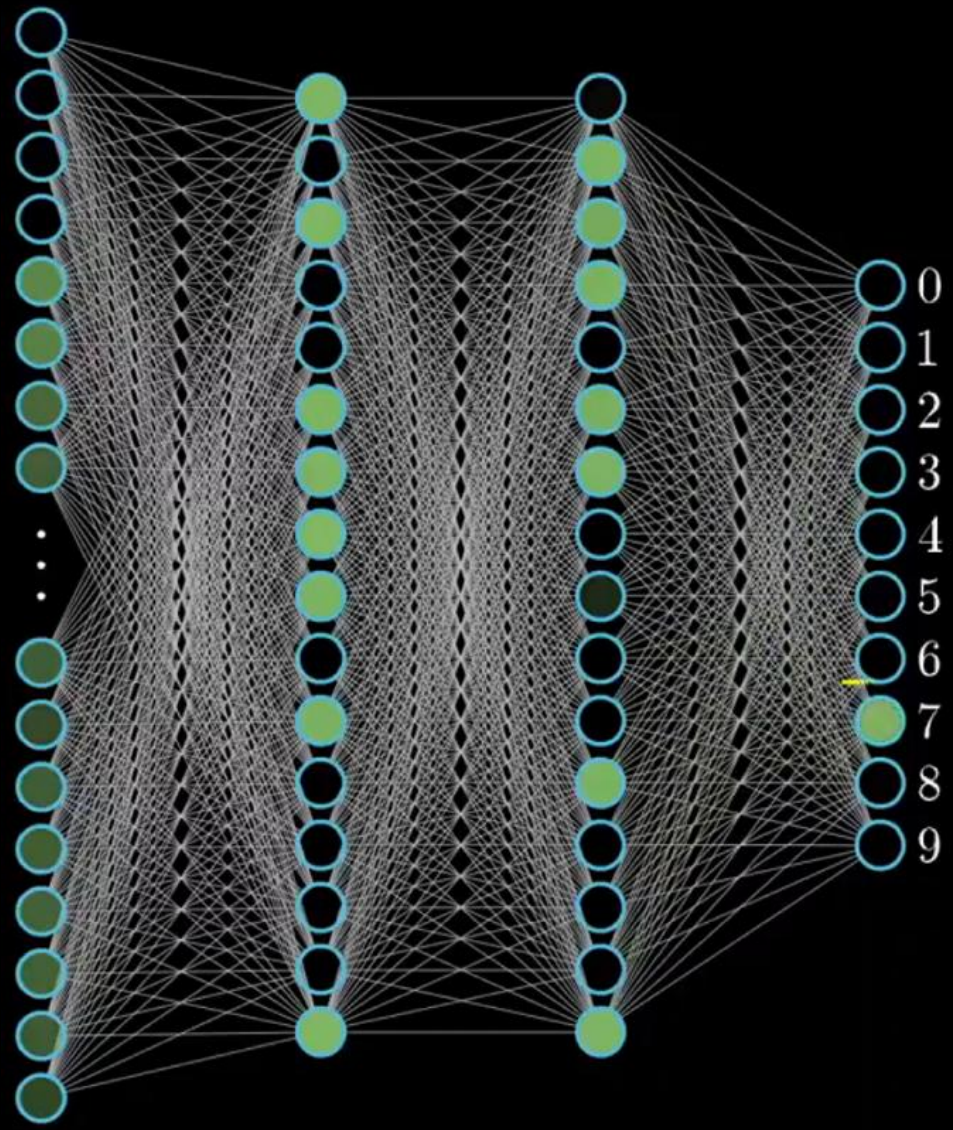
- Timeseries Data – 3D (samples, timesteps, features)
- Images – 4D (samples, height, width, channels)
- Video – 5D (samples, frames, height, width, channels)





Fully Connected Network

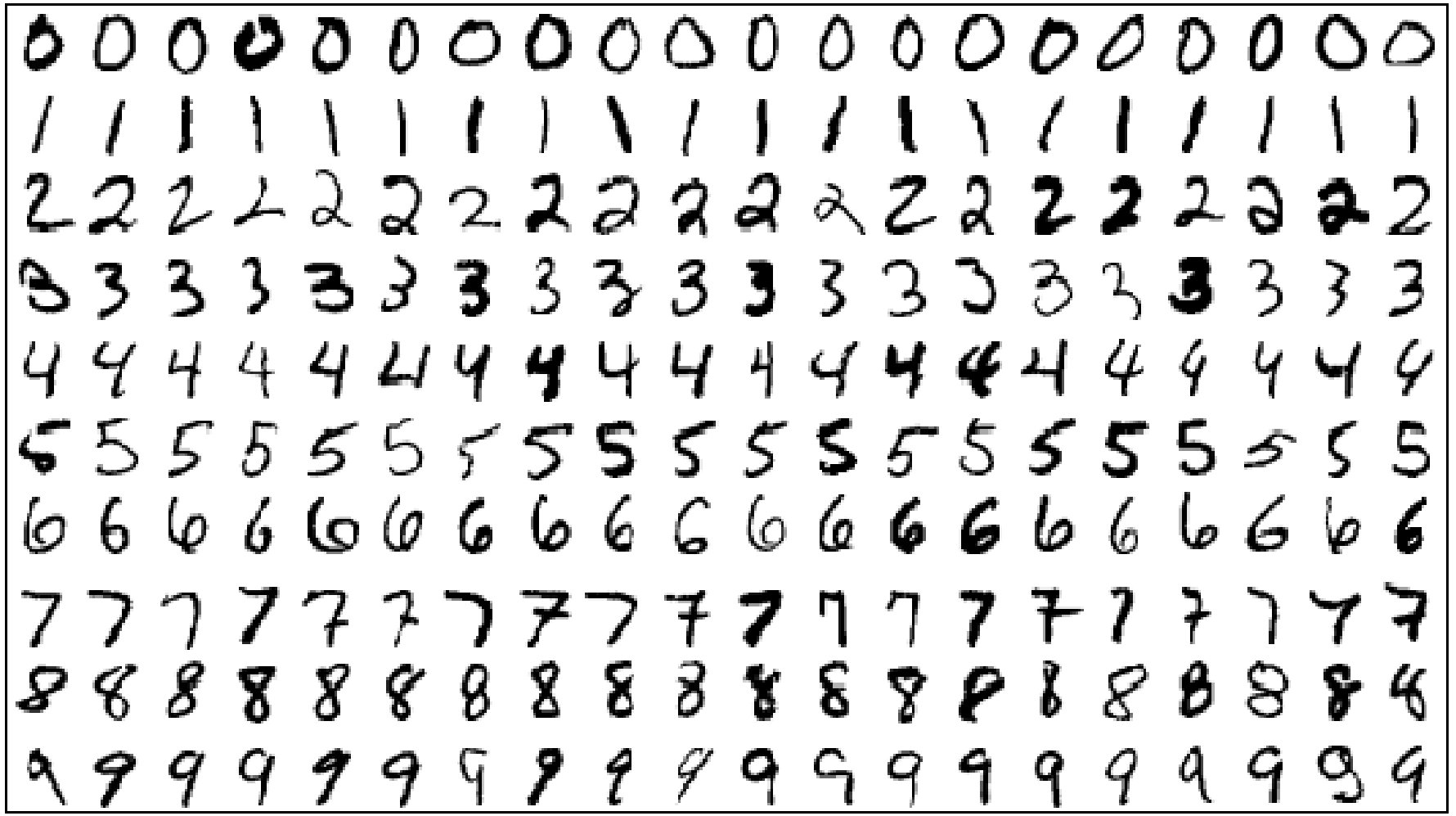
- Each neuron is connected to every neuron in the next layer

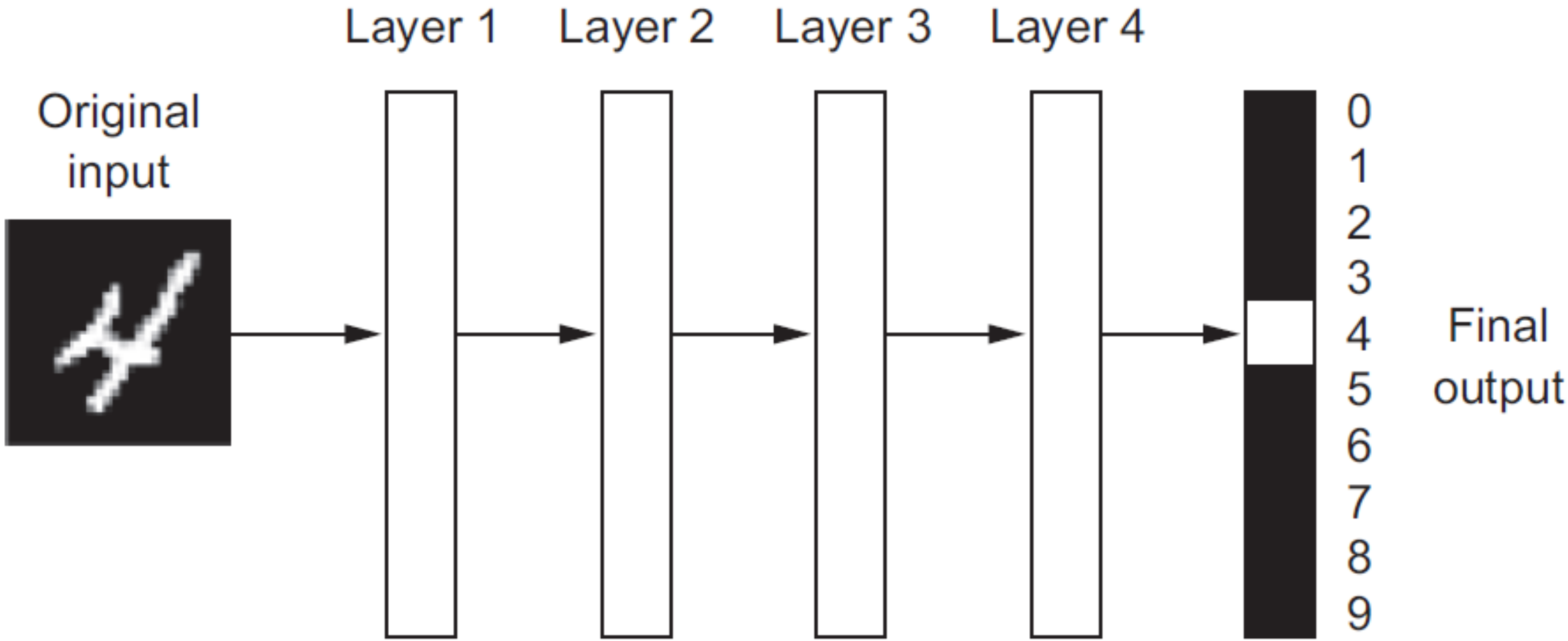


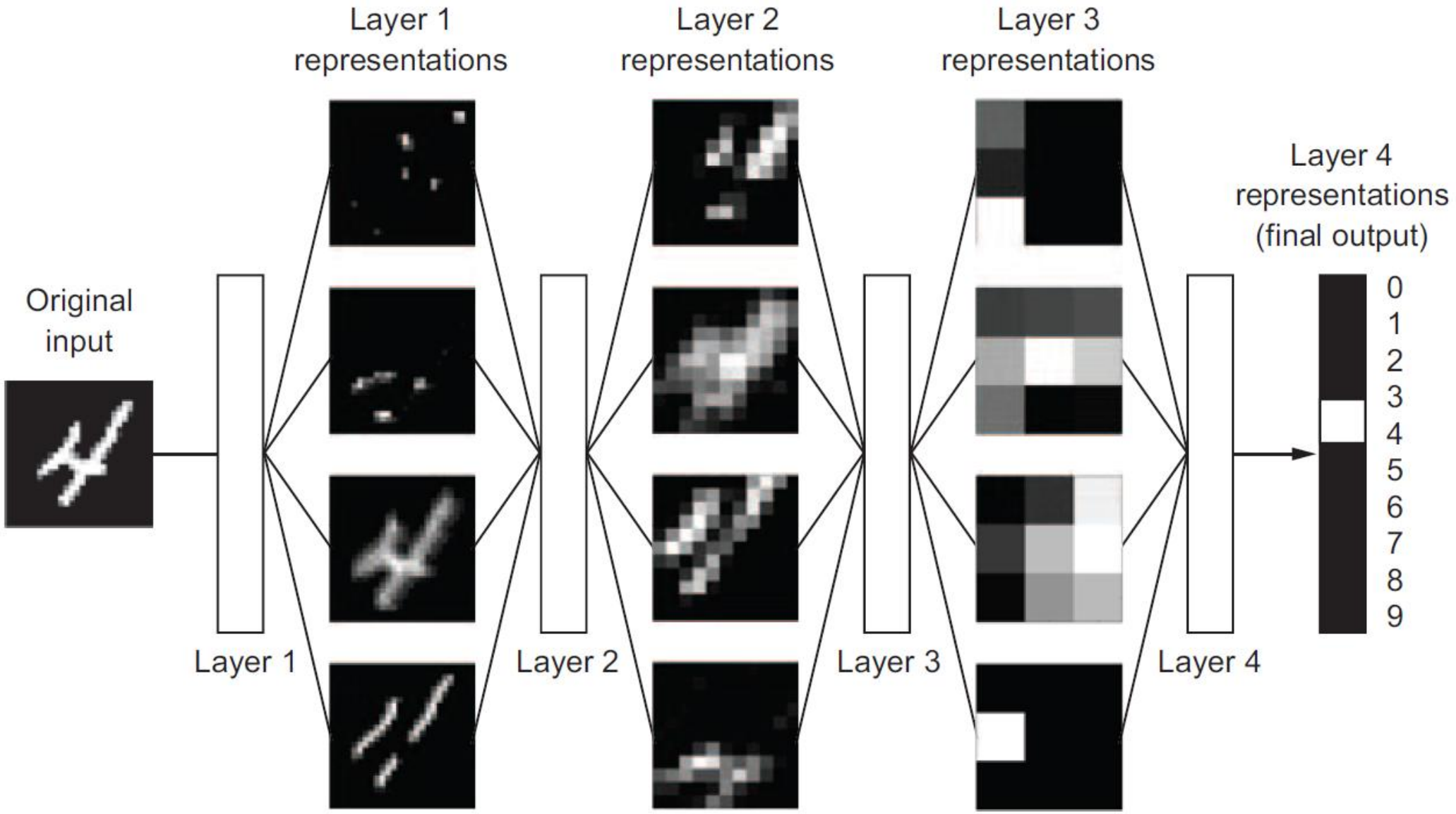
3Blue1Brown

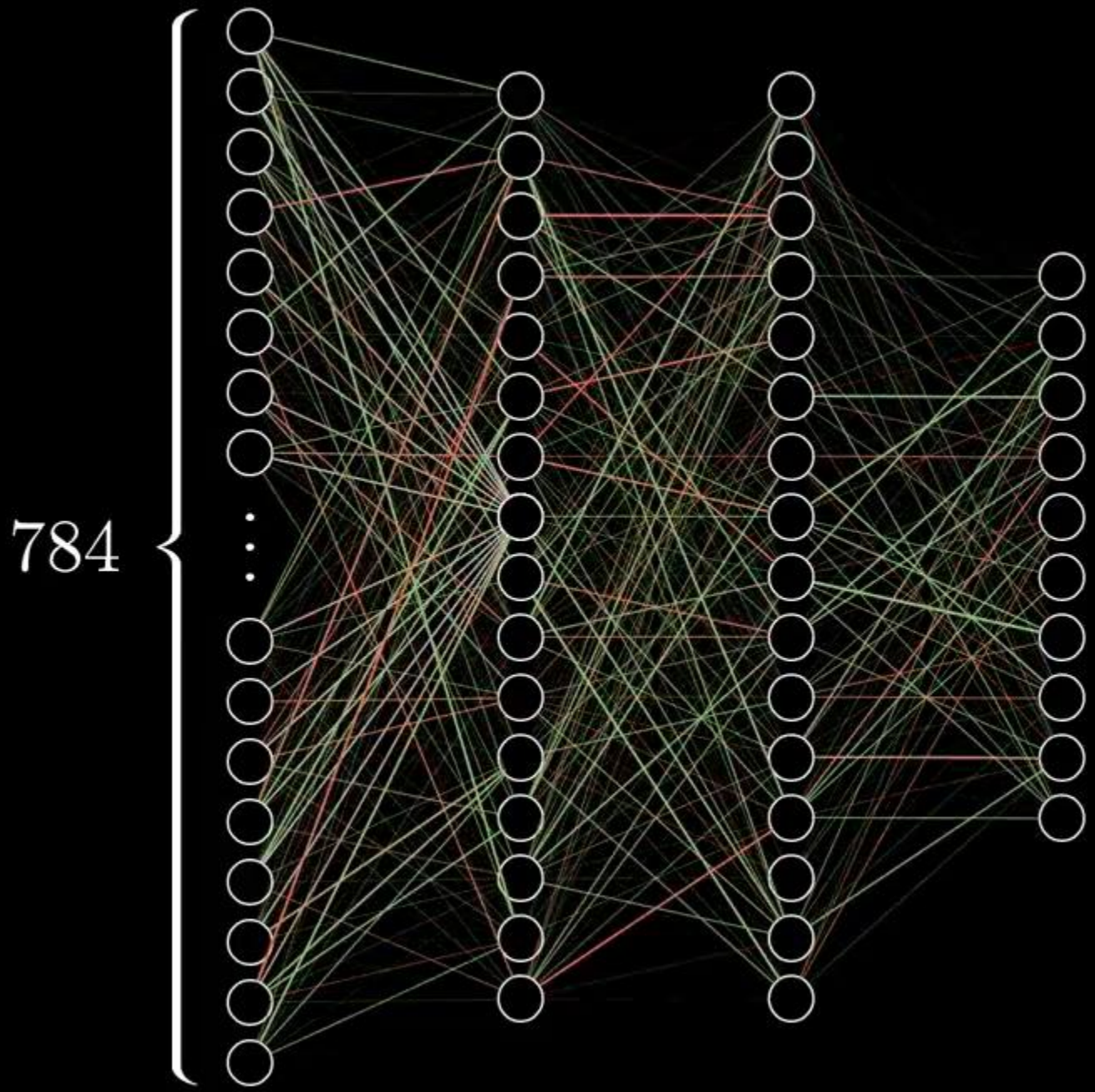
Example: Recognizing Handwritten Digits

- MNIST dataset




Layer 1
Layer 2
Layer 3
Layer 4
Original input
0
1
2
3
4
5
6
7
8
9
Final output


Layer 1 representations
Layer 2 representations
Layer 3 representations
Layer 4 representations (final output)
Original input
Layer 1
Layer 2
Layer 3
Layer 4
0
1
2
3
4
5
6
7
8
9



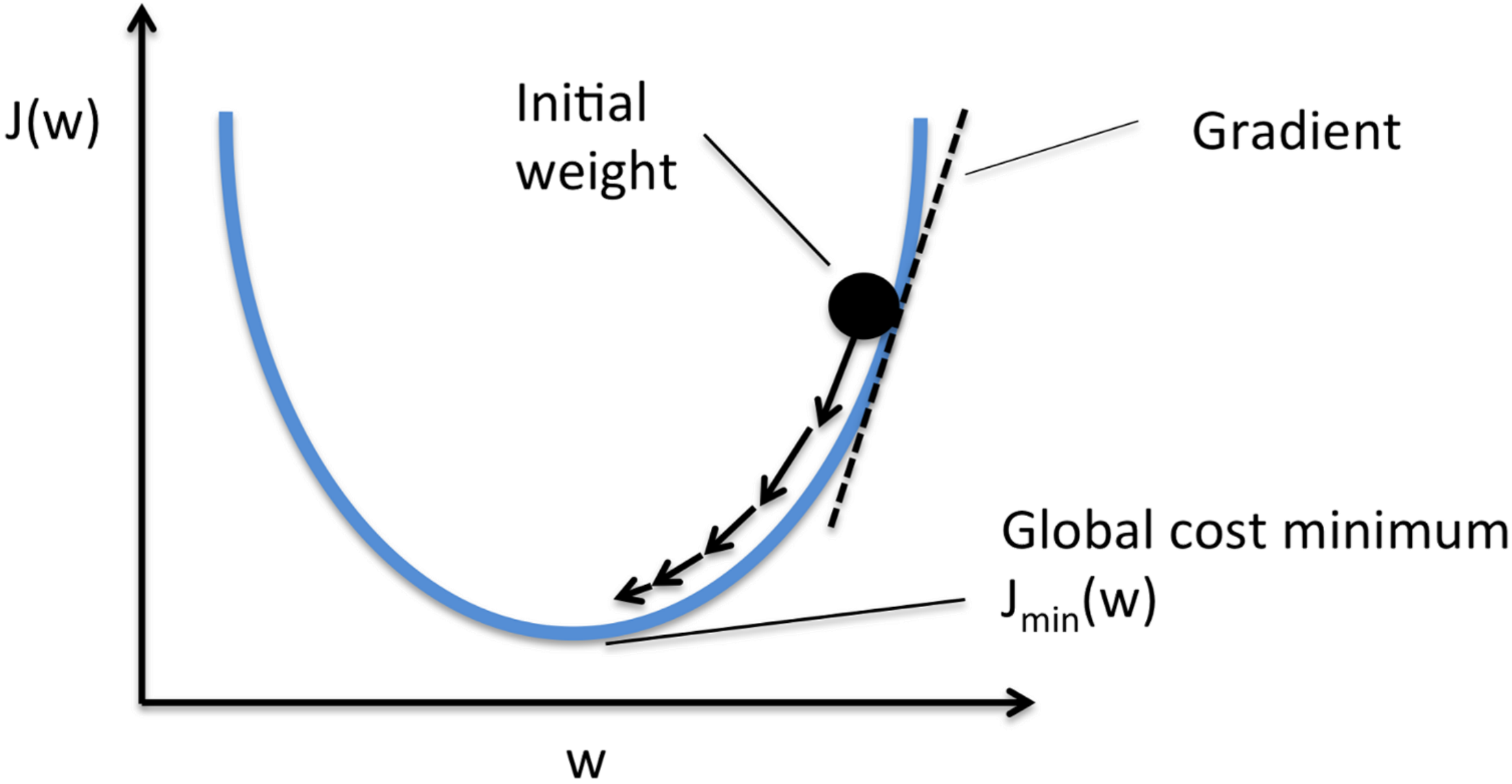
$784 \times 16 + 16 \times 16 + 16 \times 10$
weights

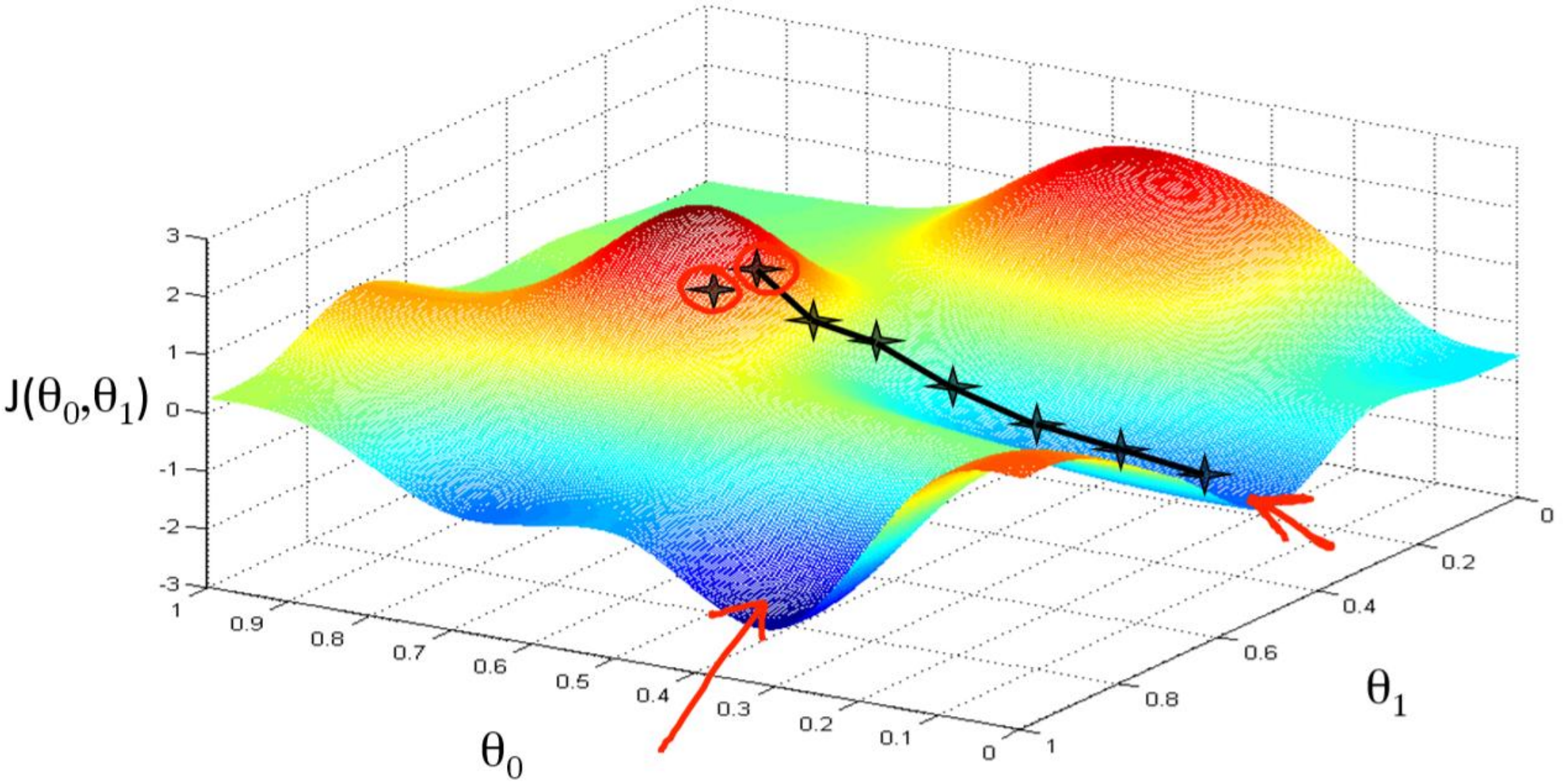
$16 + 16 + 10$
biases

$13{,}002$

Learning $\rightarrow$ Finding the right weights and biases

Gradient Descent





https://hackernoon.com/gradient-descent-aynk-7cbe95a778da

Cost Function

- Mean-Squared Error

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}(f_\theta(x_i) - y_i)^2$$

Gradient Descent of MSE

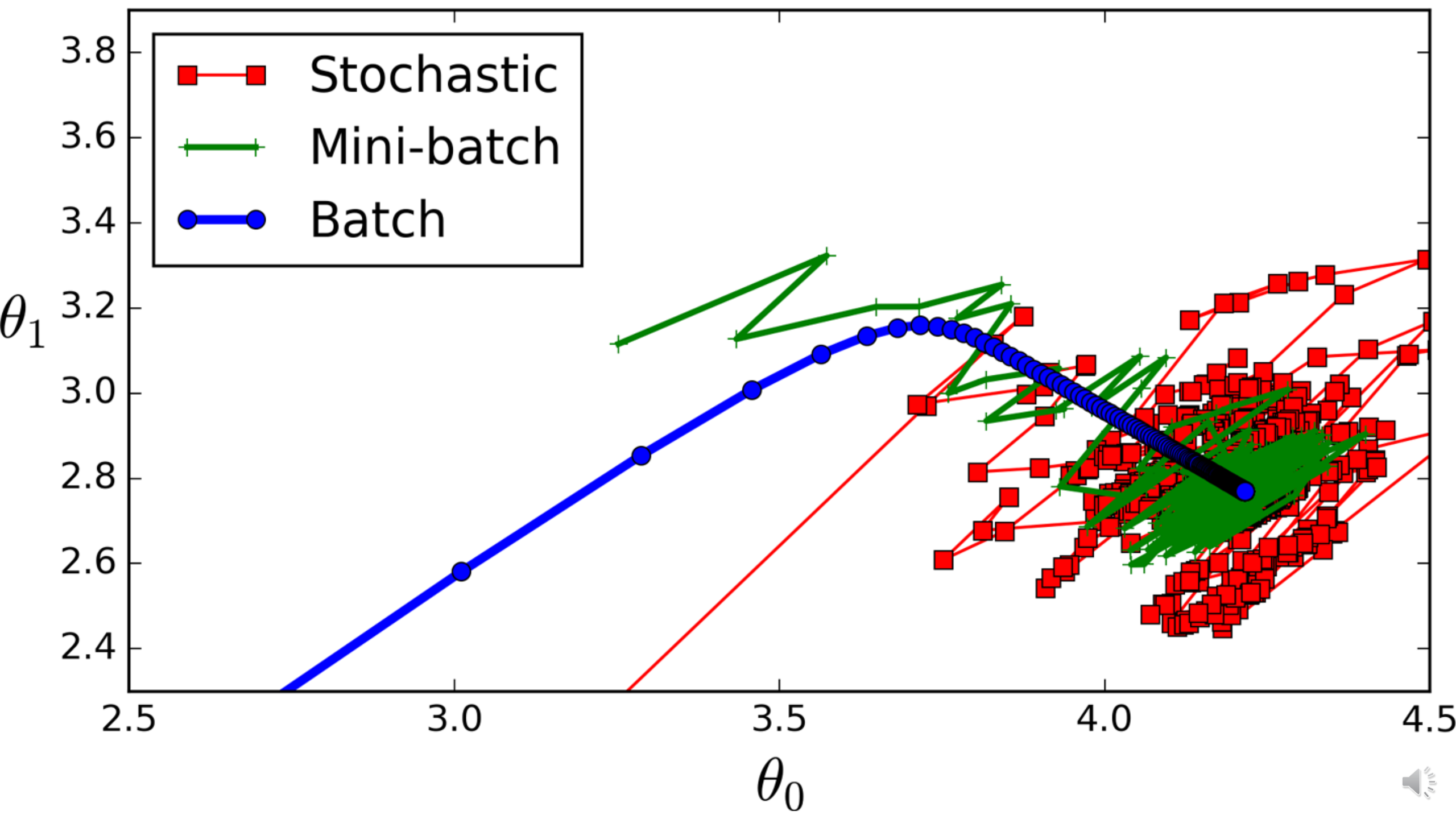
- Gradient of MSE

$$\frac{\partial J(\theta)}{\partial \theta} = \frac{2}{N}\sum_{i=1}^{N}(f_\theta(x_i) - y_i)f'_\theta(x_i)$$

- Update

$$\theta_j \leftarrow \theta_j - \alpha\frac{\partial J(\theta)}{\partial \theta_j}$$

- Repeat until Convergence


Stochastic
Mini-batch
Batch
3.8
3.6
3.4
3.2
3.0
2.8
2.6
2.4
θ_1
2.5
3.0
3.5
4.0
4.5
θ_0

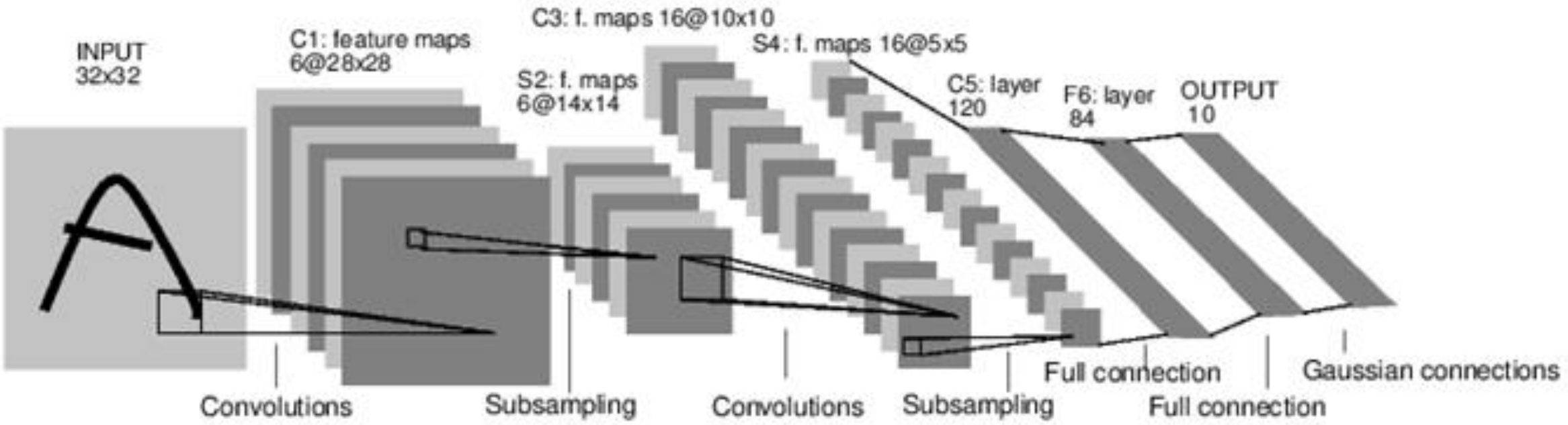
Cost function

$$C(\underbrace{w_1, w_2, \ldots, w_{13{,}002}}_{\text{Weights and biases}})$$

Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs)

- https://medium.com/@sh.tsang/paper-brief-review-of-lenet-1-lenet-4-lenet-5-boosted-lenet-4-image-classification-1f5f809dbf17



A Full Convolutional Neural Network (LeNet)

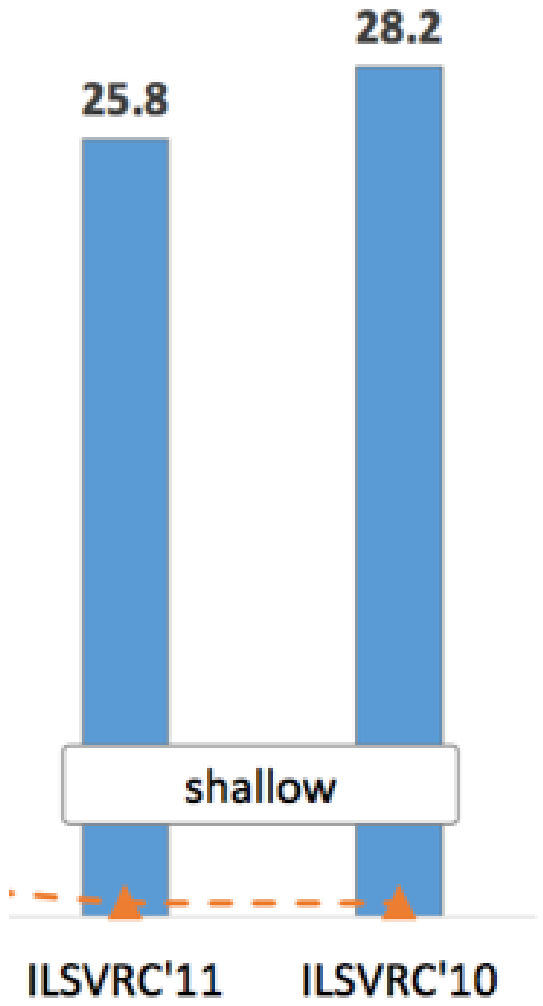
14,197,122 images, 21841 classes
(2021/9/21)

IMAGENET

ImageNet Large Scale Visual Object Recognition Challenge (ILSVRC)

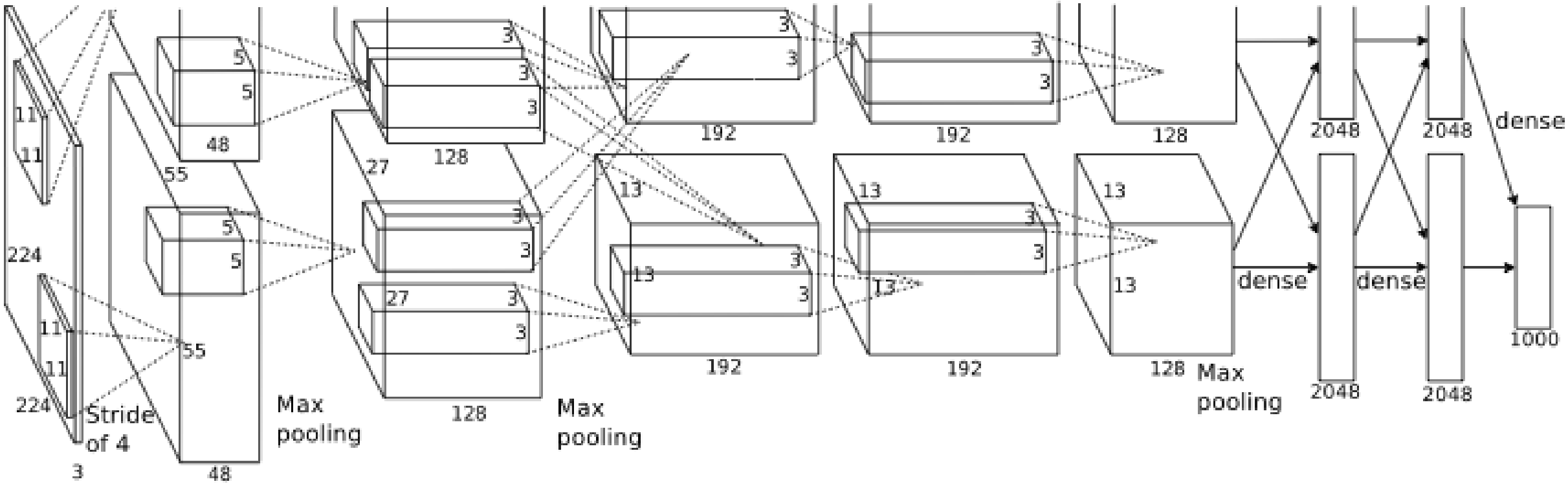
- 1000 categories
- For ILSVRC 2017
  - **Training images** for each category ranges from 732 to 1300
  - 50,000 validation **images** and 100,000 test **images**.
- Total number of images in ILSVRC 2017 is around 1,150,000

Winners’ Error Rates on ImageNet Challenge

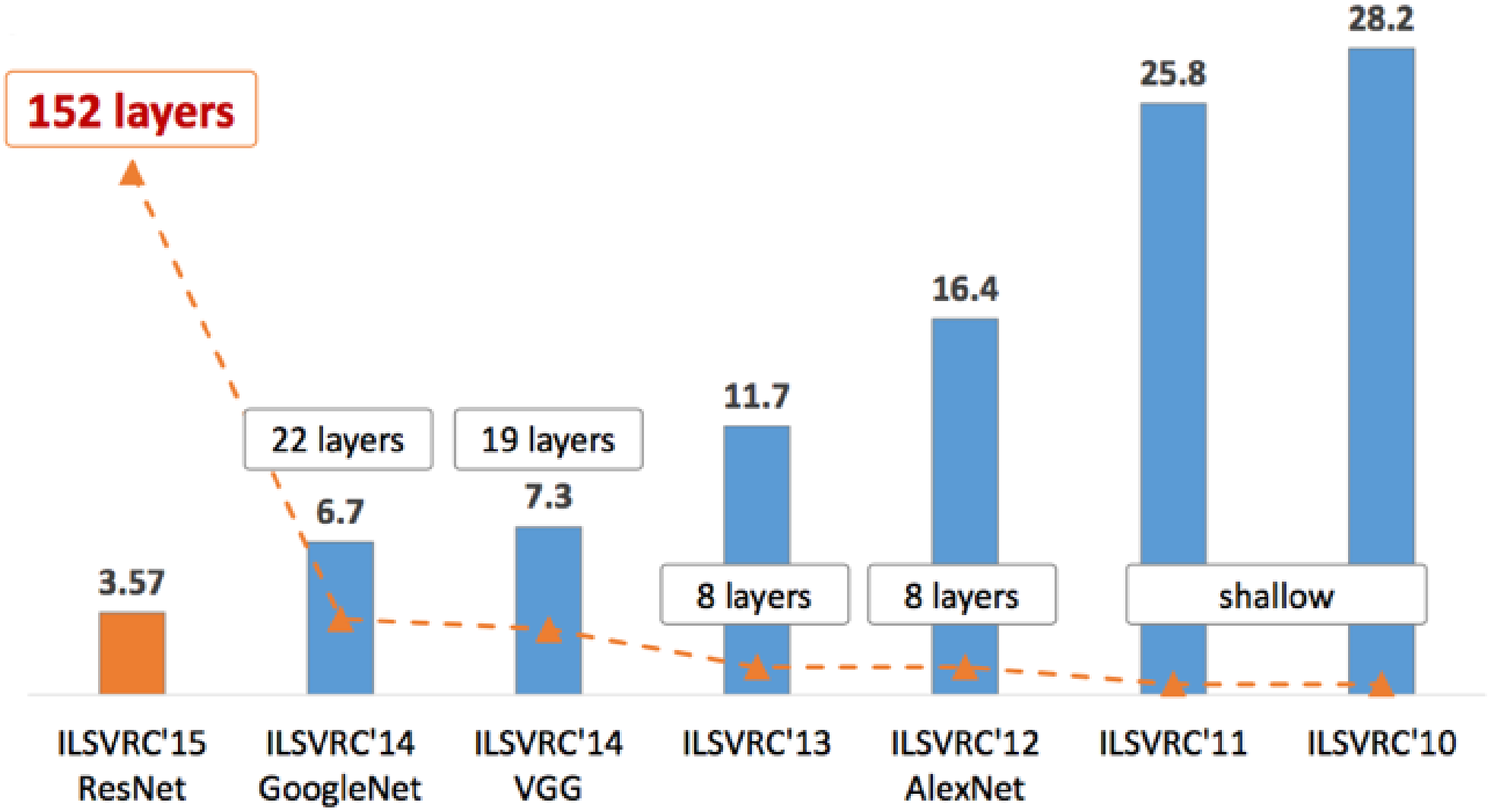


Convolutional Neural Network (AlexNet)

- Alex Krizhevsky, Geoffery Hinton et al., 2012



Winners' Error Rates on ImageNet Challenge



LeNet

AlexNet

VGG

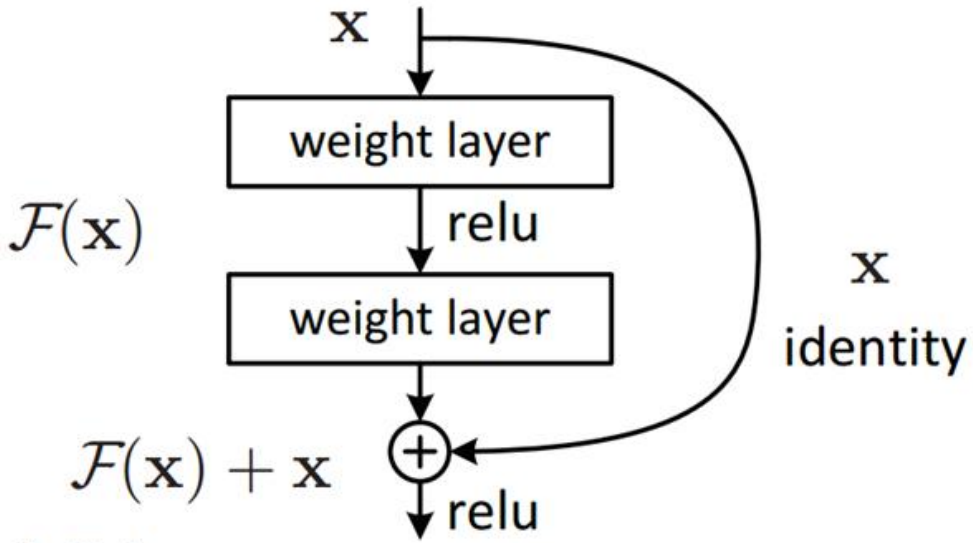
GoogLeNet

Inception V3

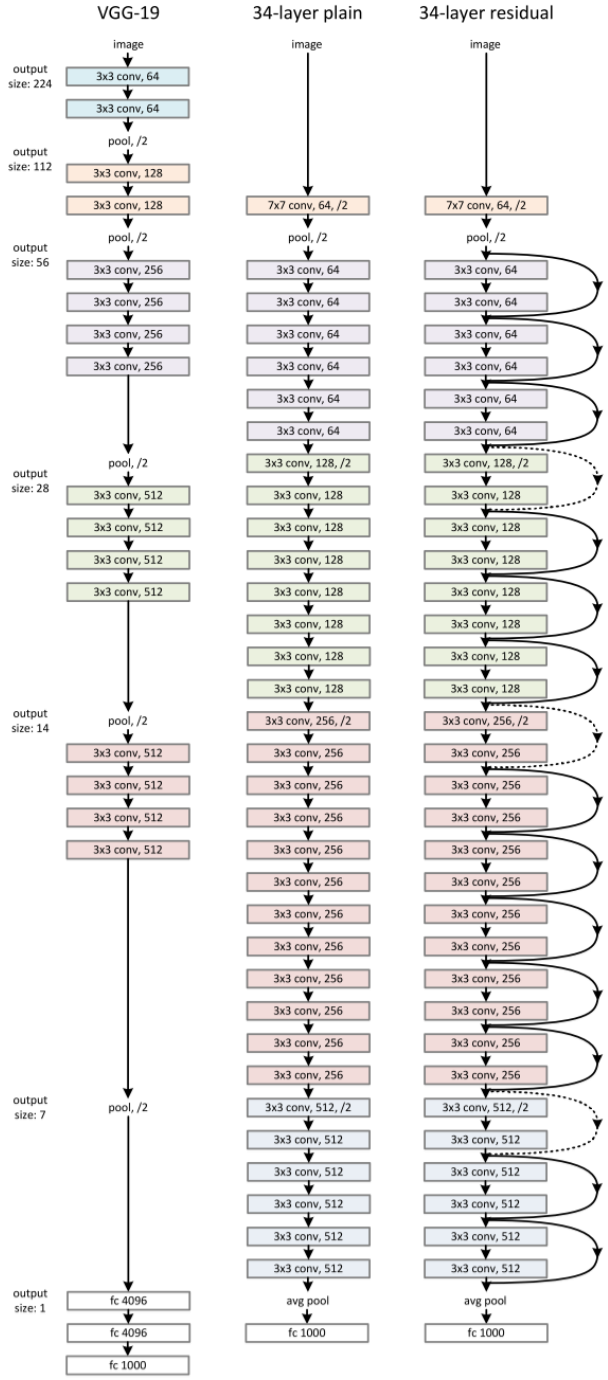
Inception BN

ResNet (2015)

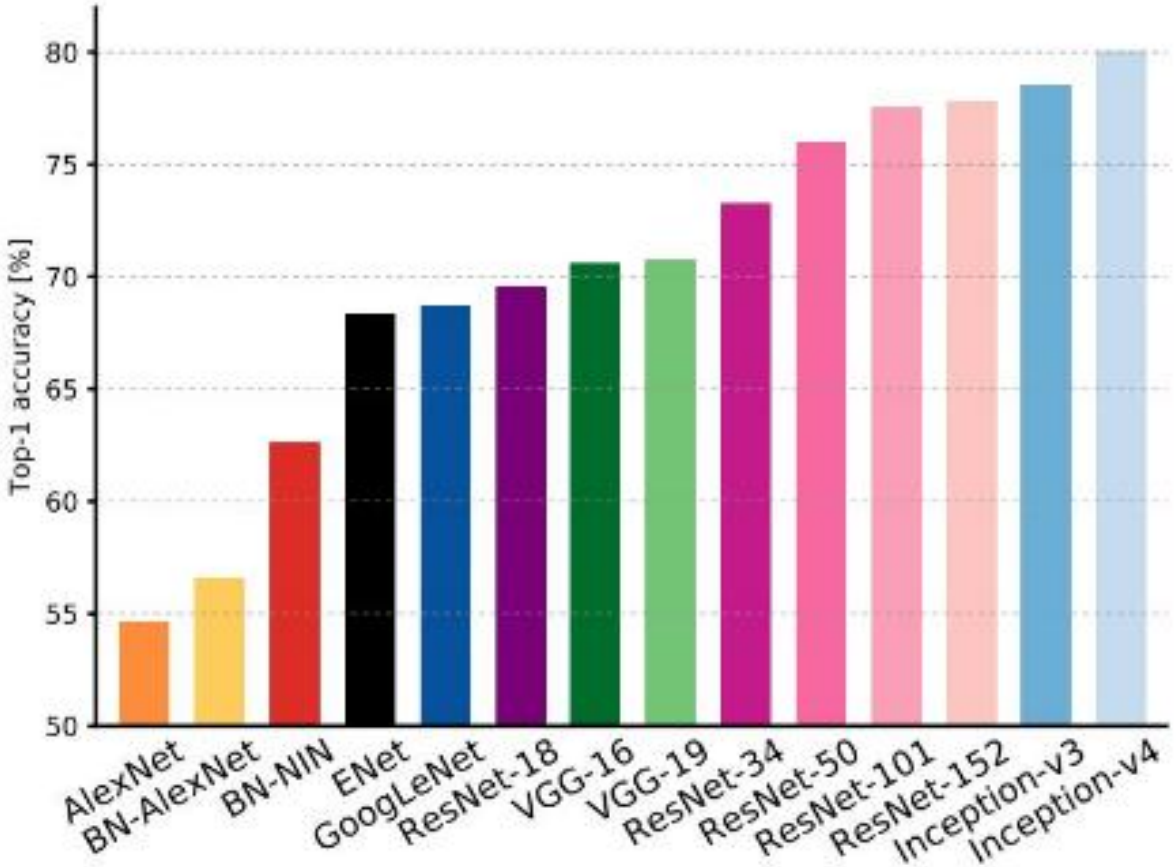
- Residual Neural Network
- Proposed “skip connection”
- 152-layer with 3.57% error rate

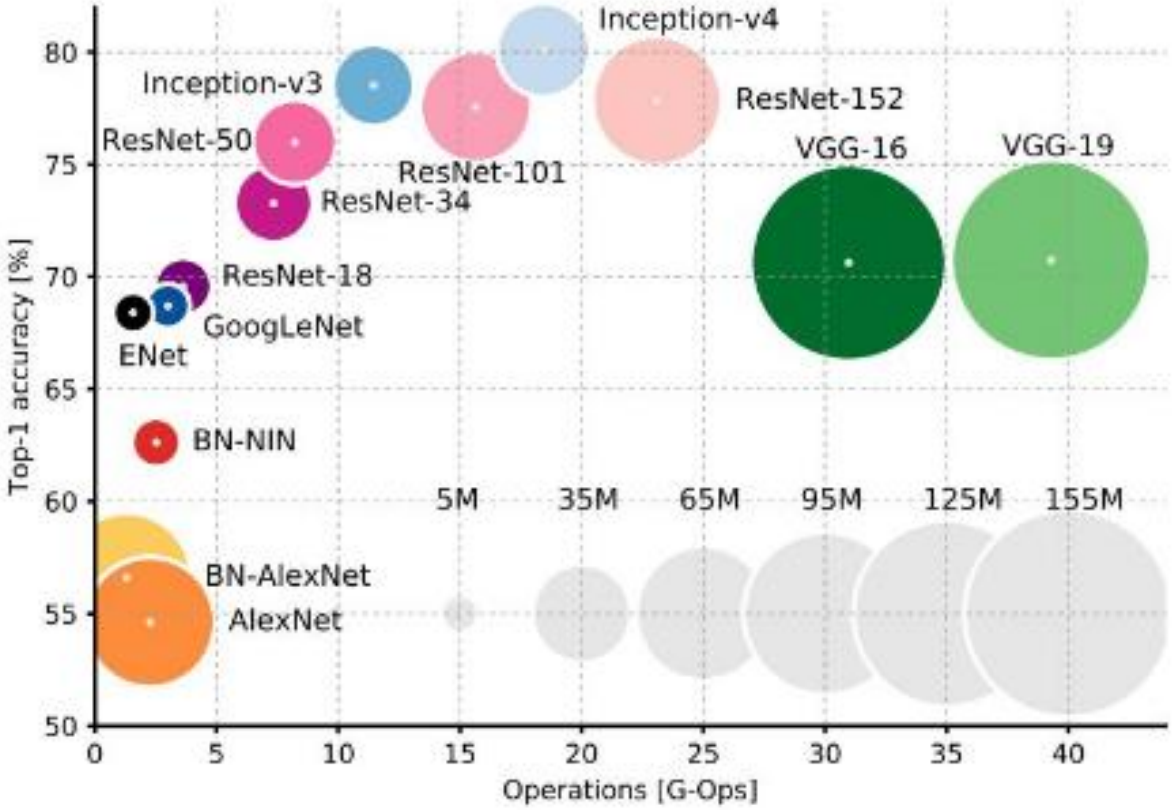


A residual block



CNN Comparison

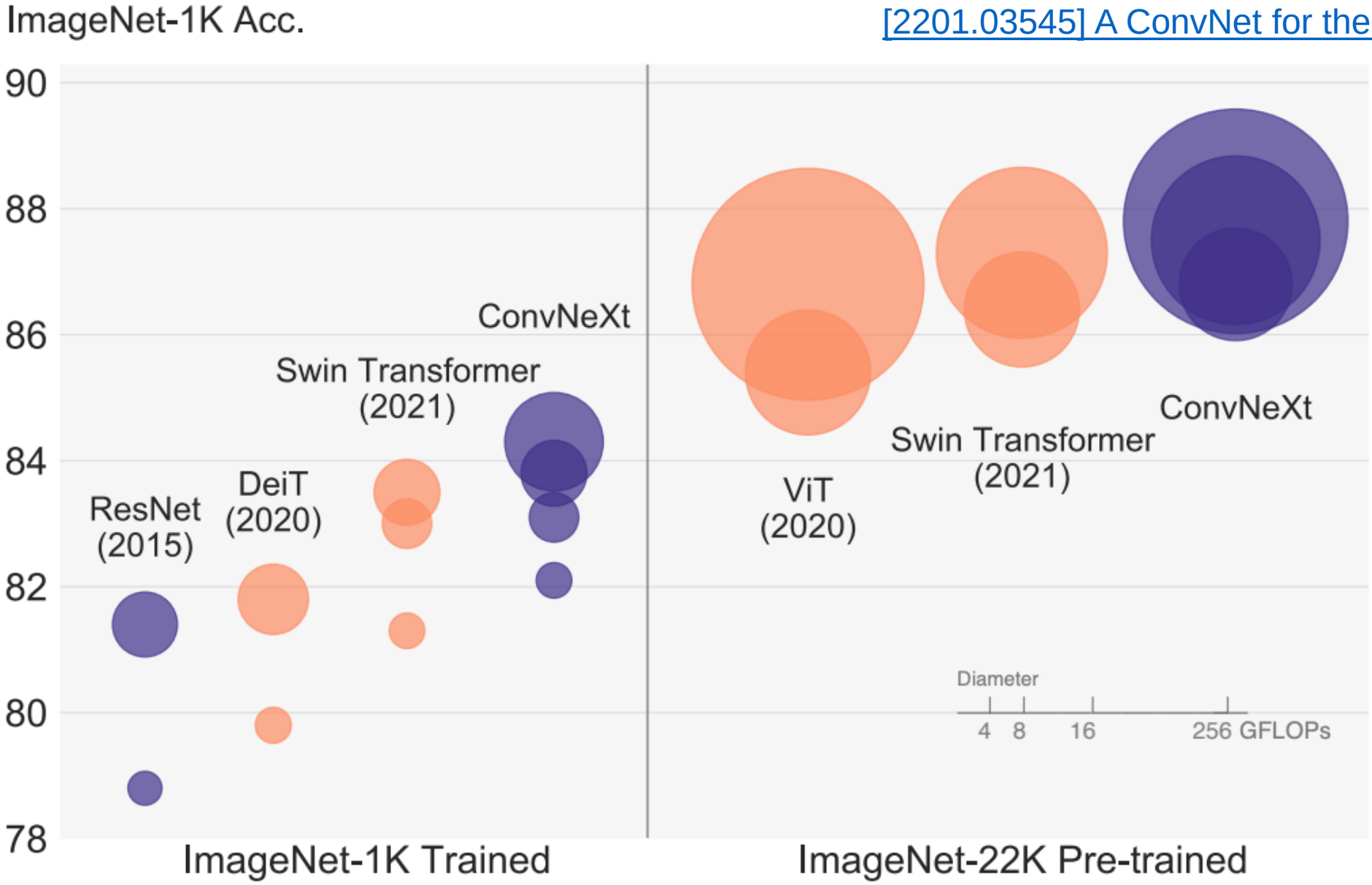




https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5

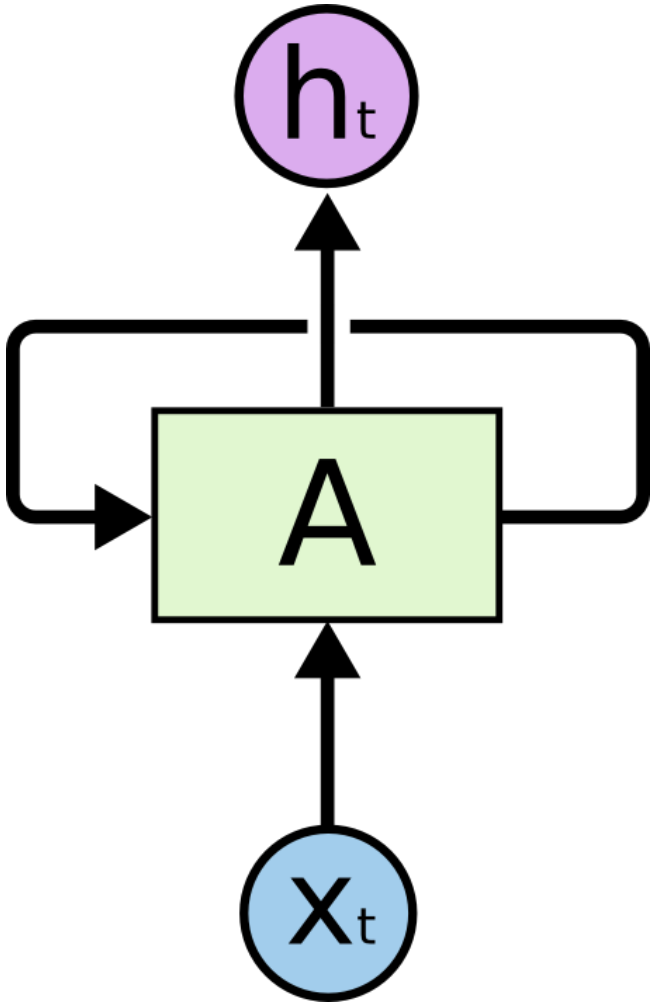
ConvNeXt (2022)

[2201.03545] A ConvNet for the 2020s

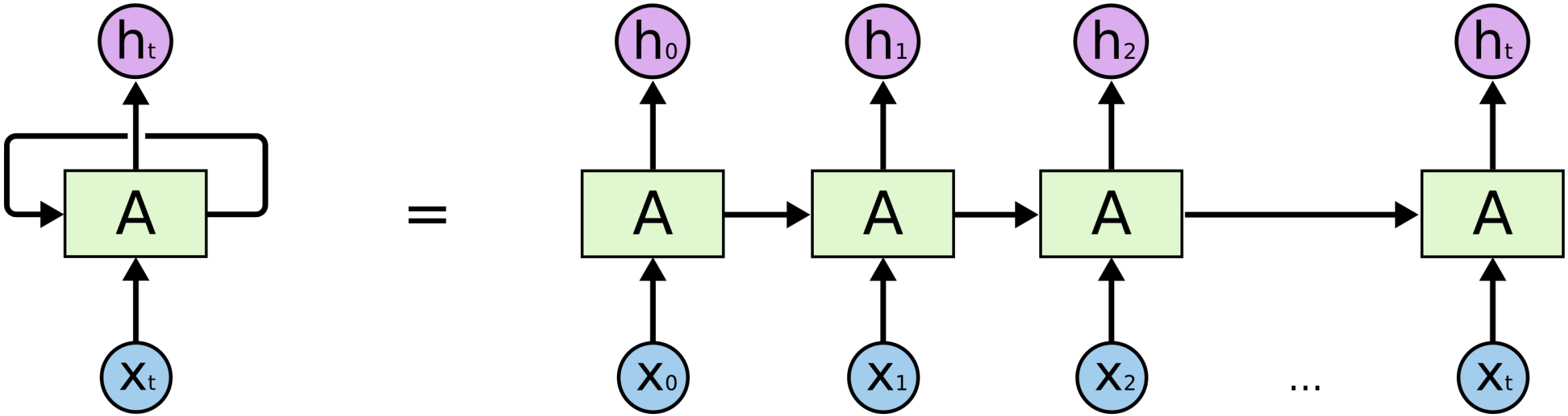


Recurrent Neural Networks (RNNs) and LSTM

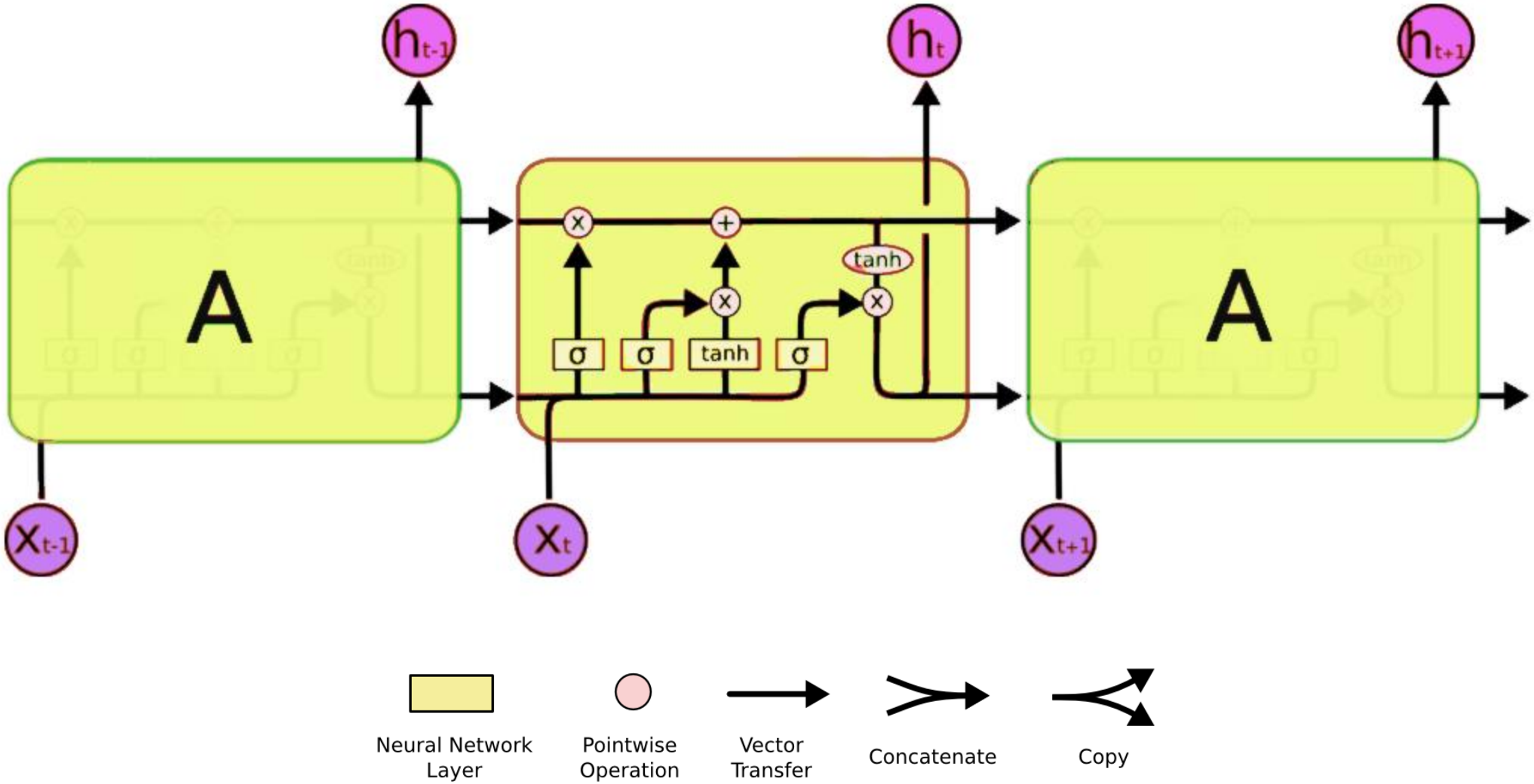
Recurrent Neural Networks (RNNs)



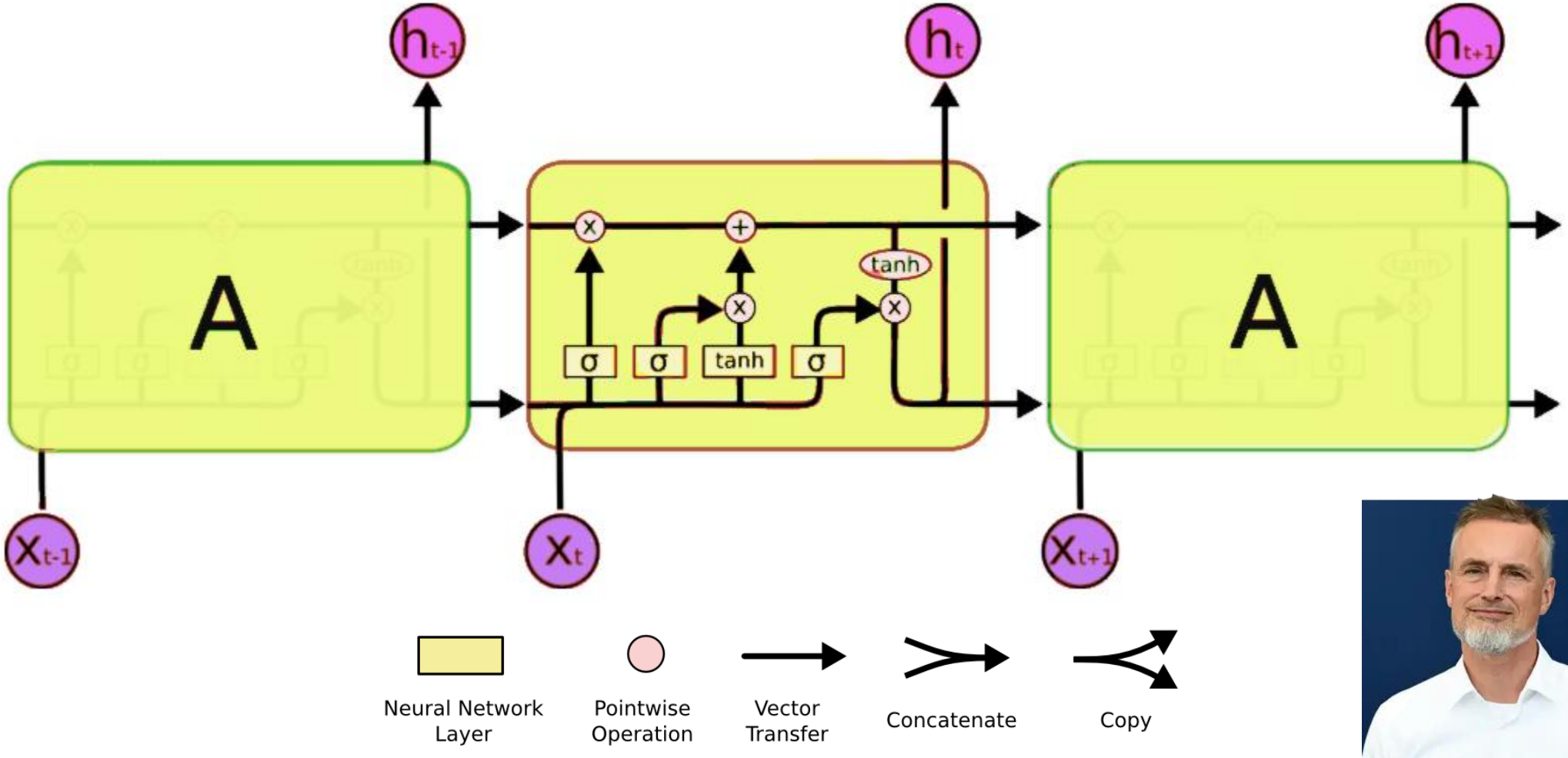
Unroll the RNN



Long Short-term Memory (LSTM)



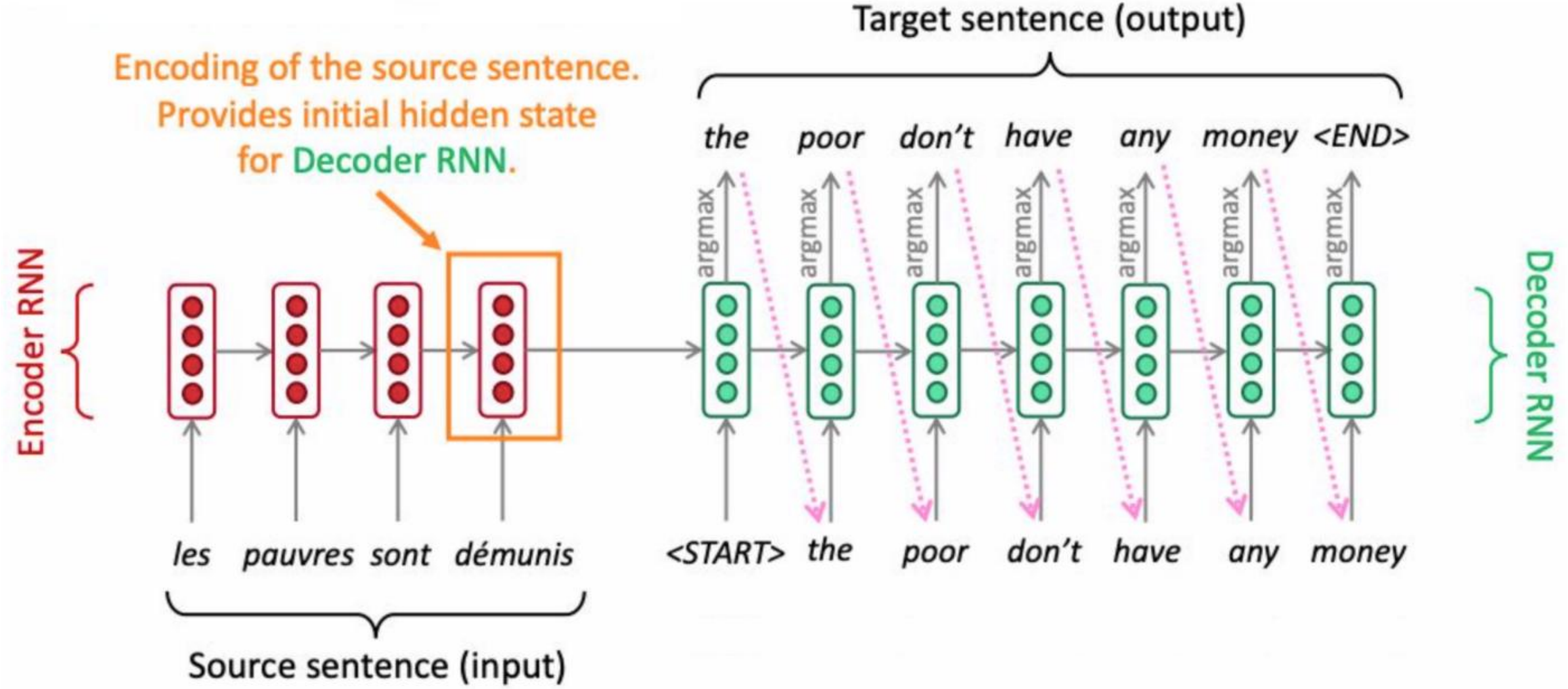
Long Short-term Memory (LSTM)



https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Jürgen Schmidhuber

Sequence-2-Sequence model (Language Translation)



https://towardsdatascience.com/introduction-to-rnns-sequence-to-sequence-language-translation-and-attention-fc43ef2cc3fd

Attention is All You Need!

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

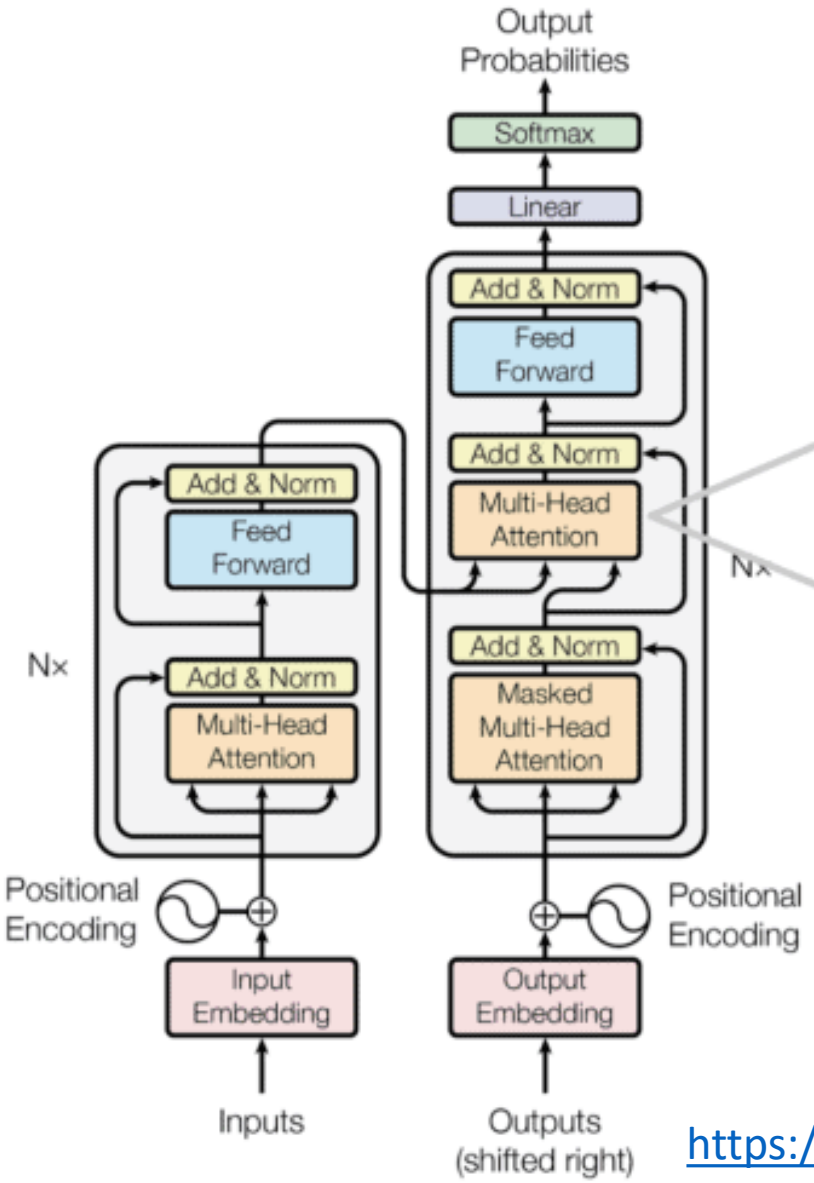
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

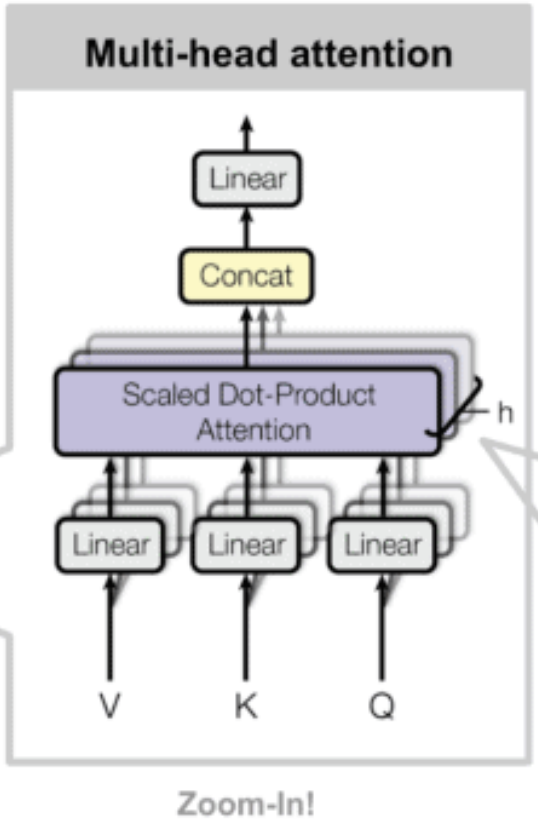
Łukasz Kaiser* Google Brain lukaszkaiser@google.com

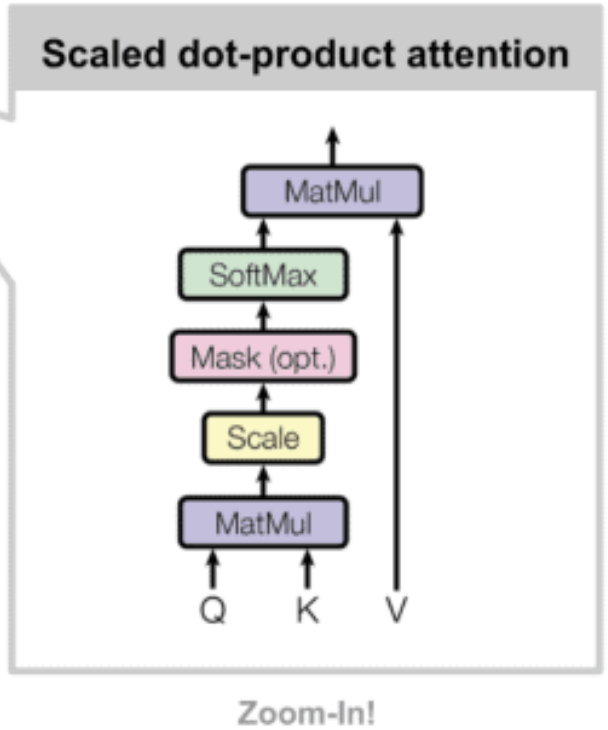
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Google Brain & University of Toronto, *NIPS*, 2017

Transformer







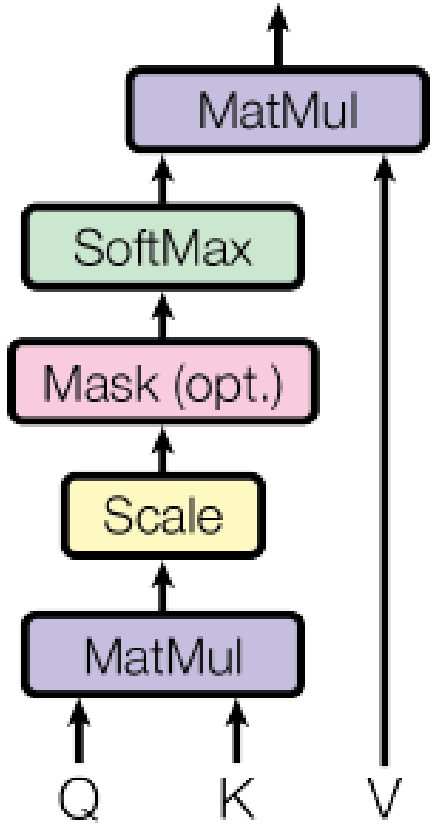
Attention is All You Need!

https://www.youtube.com/watch?v=SZorAJ4I-sA

Attention Module in Transformer

- Query (Q), Key (K), Value (V) attention

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$



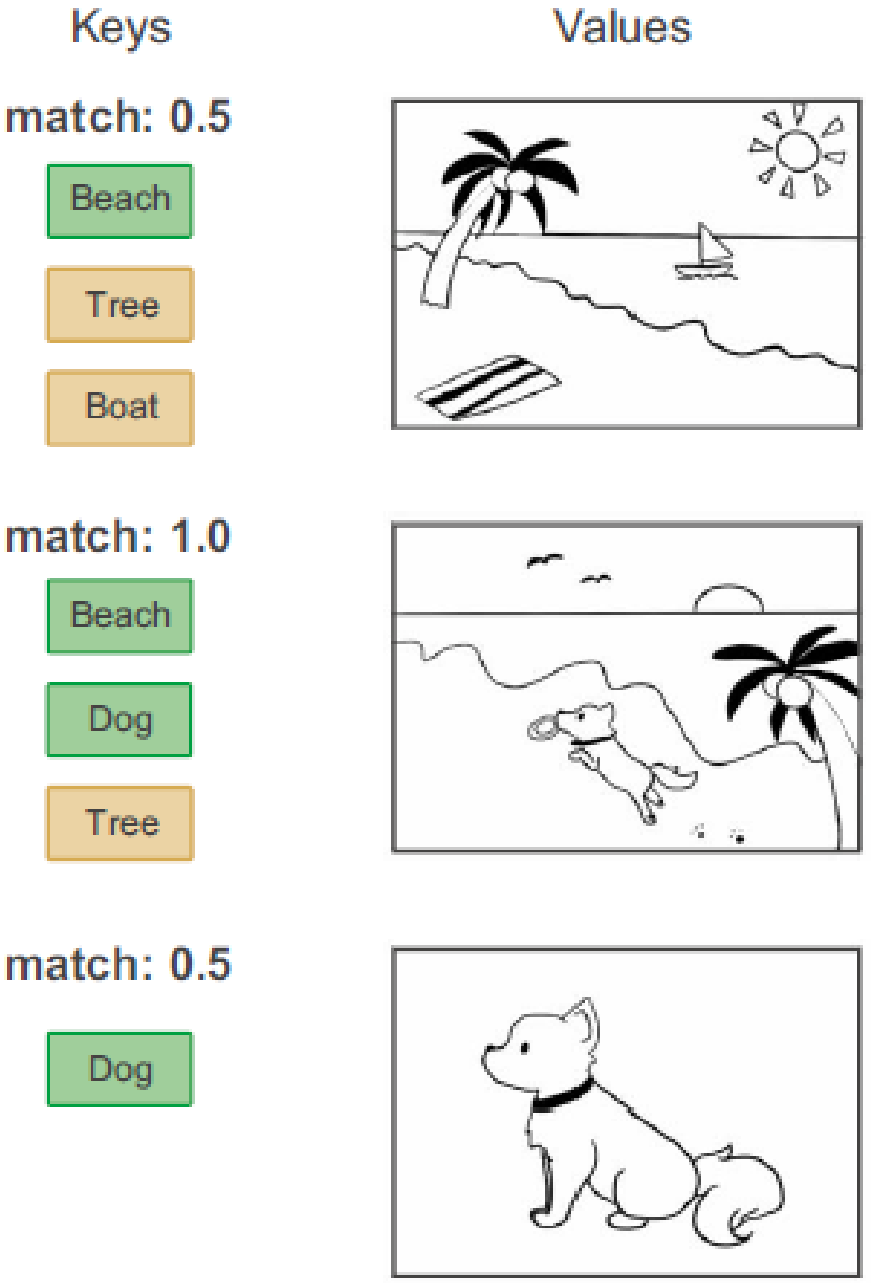
A. Waswani et al., "Attention is All You Need," NIPS, 2017

Query, Keys, Values

Query

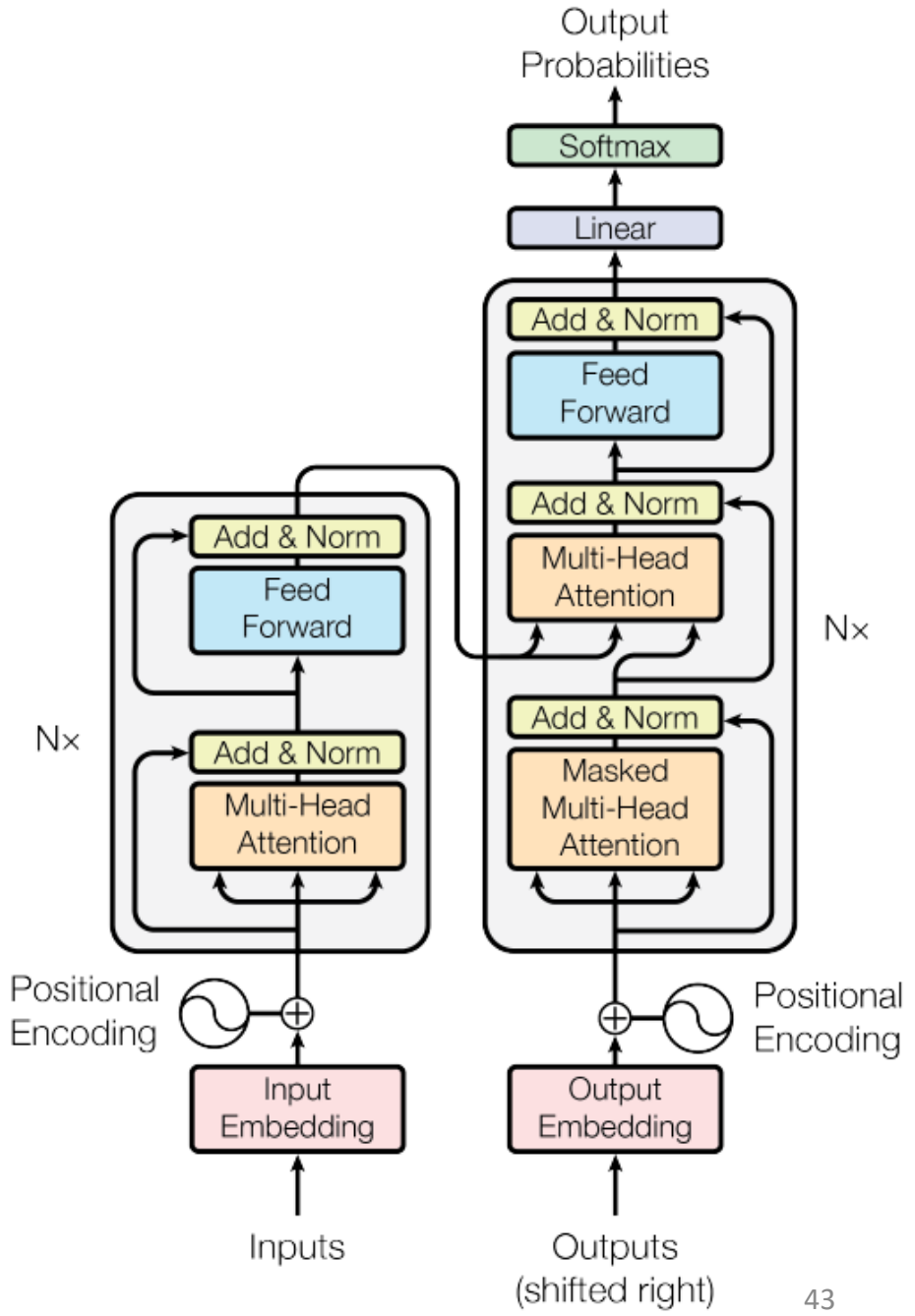
"dogs on the beach"

Retrieving images from a database



The Transformer Model

- Encoder-decoder architecture
- Multi-head attention
  - Self-attention in encoders
  - Masked Self-attention in decoders
  - Encoder-decoder attention
- Positional encoding

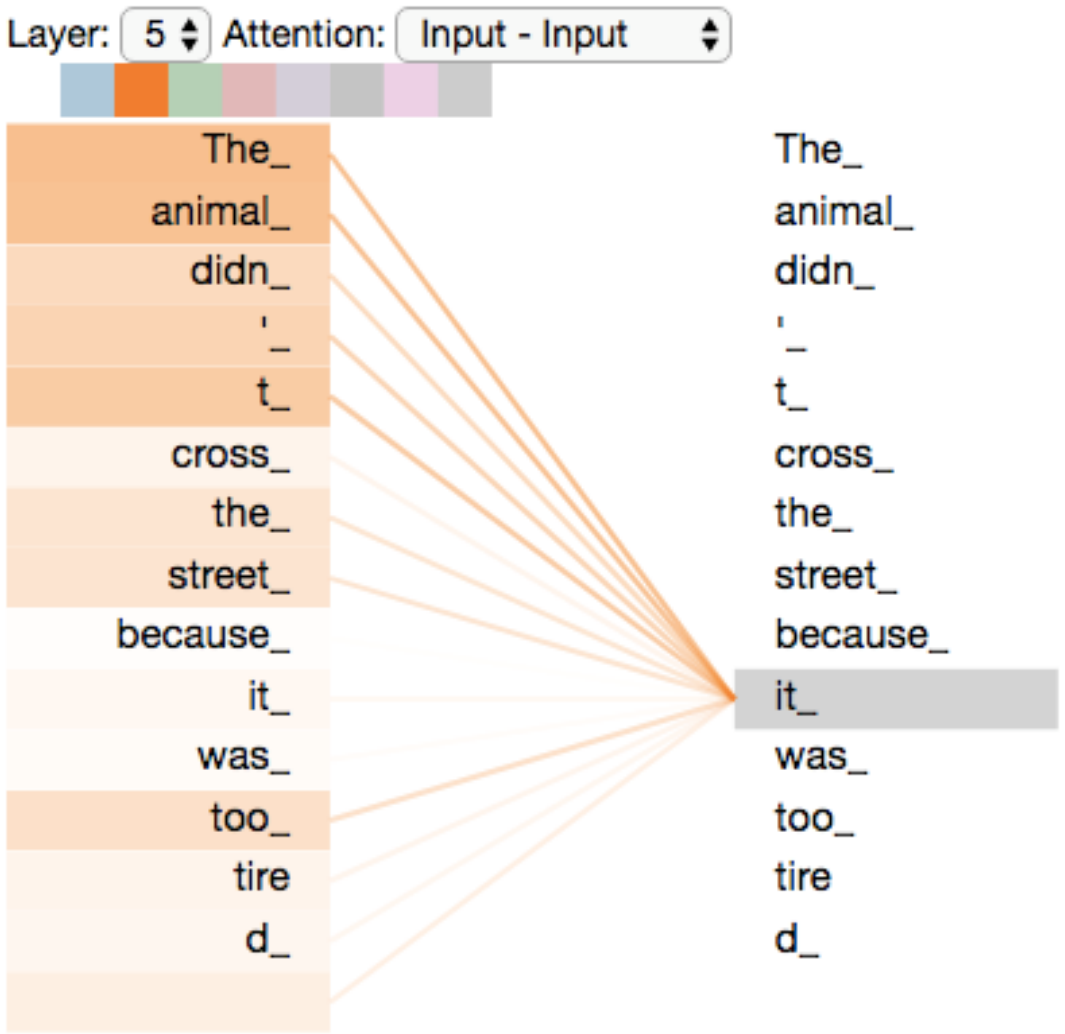


A. Waswani et al., "Attention is All You Need," NIPS, 2017

Visualizing Attention

- Tensor2Tensor Notebook

https://colab.research.google.com/github/tensorflow/tensor2tensor/blob/master/tensor2tensor/notebooks/hello_t2t.ipynb




```
Inputs: The animal didn't cross the
street because it was too tired
```

```
Outputs: Das Tier überquerte die Straße
nicht, weil es zu müde war, weil es zu
müde war.
```

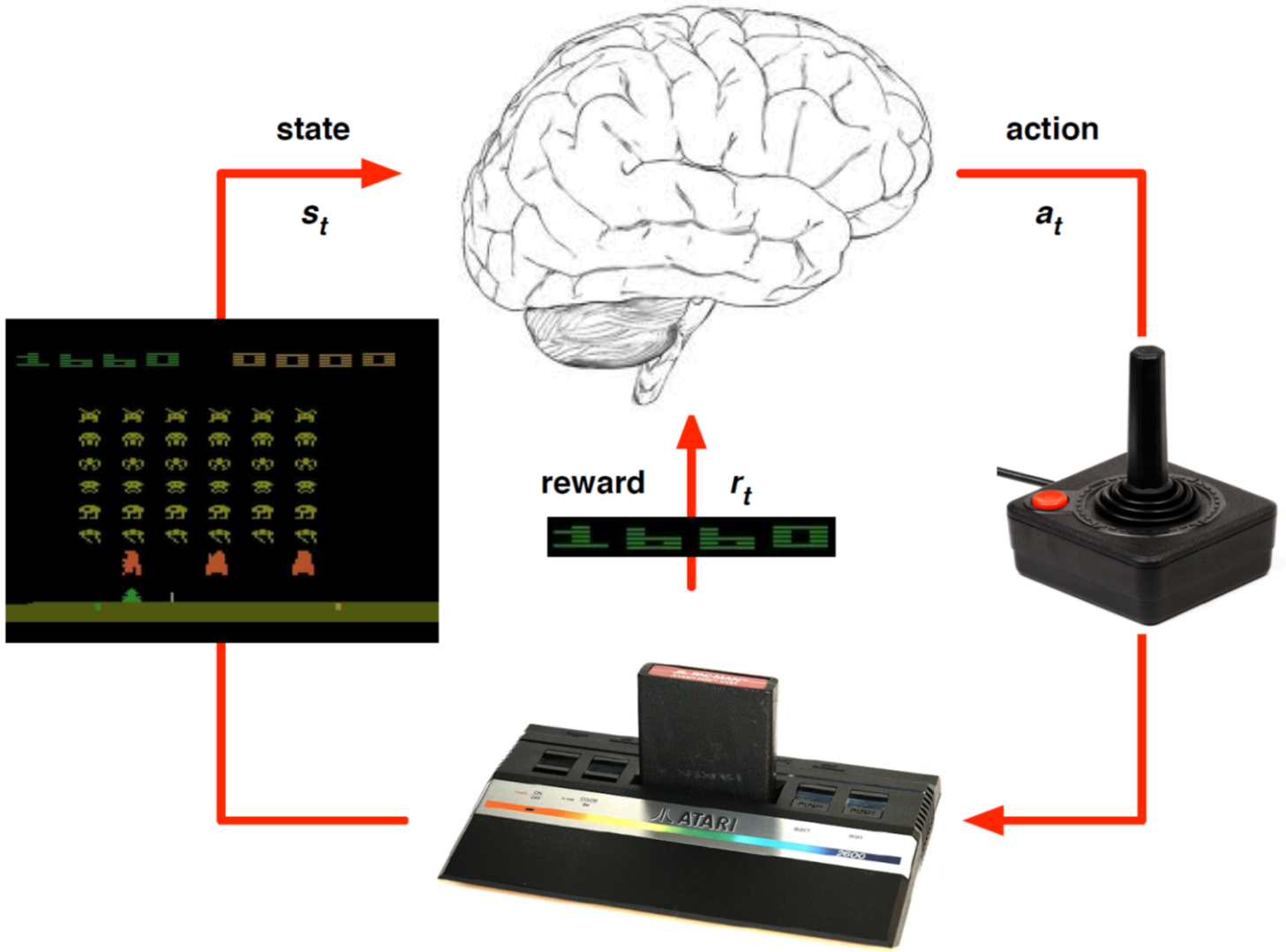
Deep Reinforcement Learning

DeepMind: DRL in Atari

DeepMind

Demis Hassabis

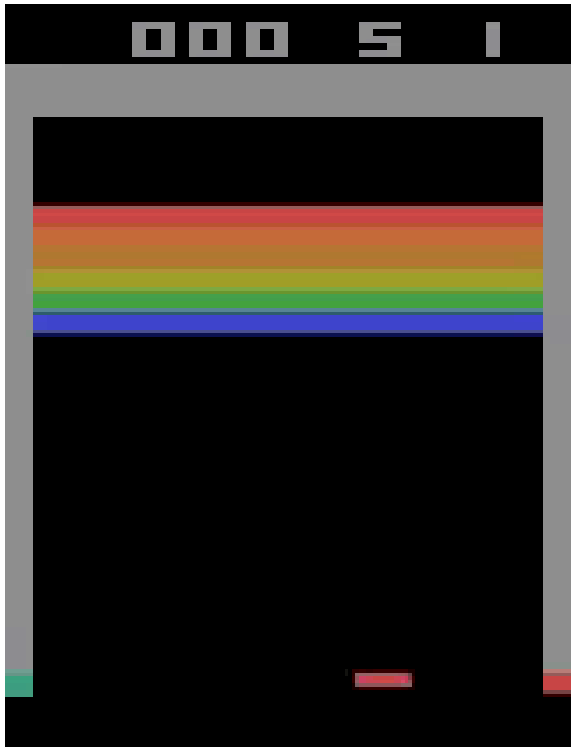
Mustafa Suleyman

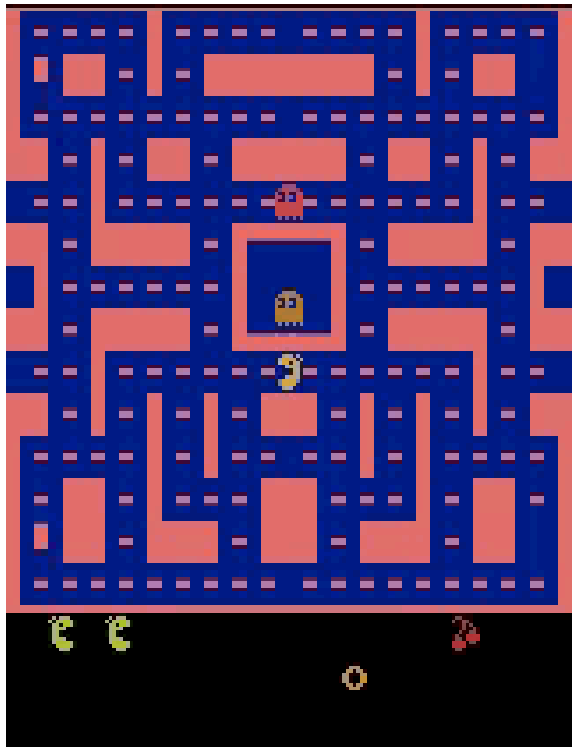


Mnih et al., "Human Level Control through Deep Reinforcement Learning," *Nature*, 2015

Learning to Play Atari Games

David Silver





AlphaGo vs. Lee Sedol

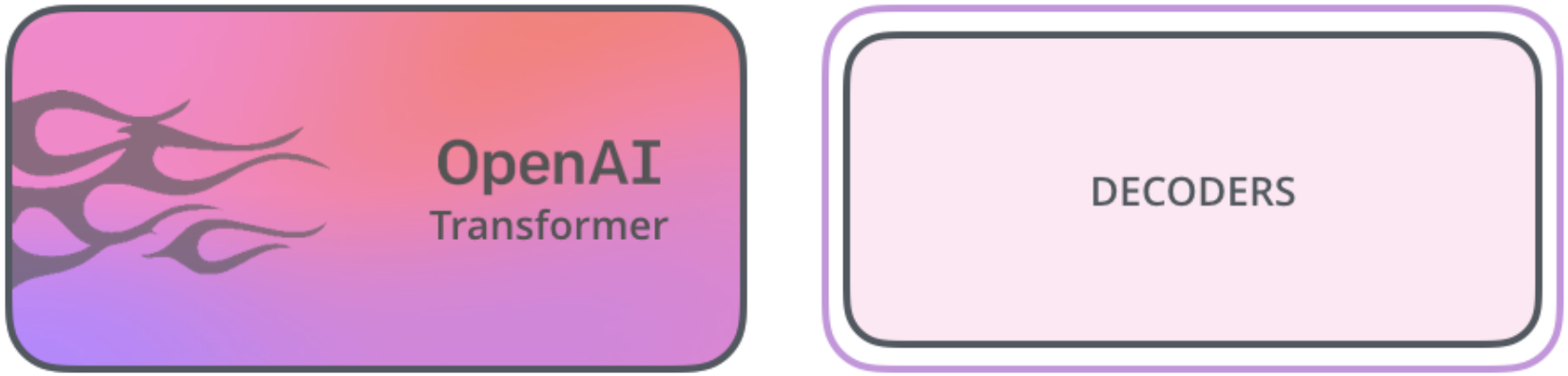
Stable-Baselines 3

- Stable Baselines3 (SB3) is a set of reliable implementations of reinforcement learning algorithms in PyTorch

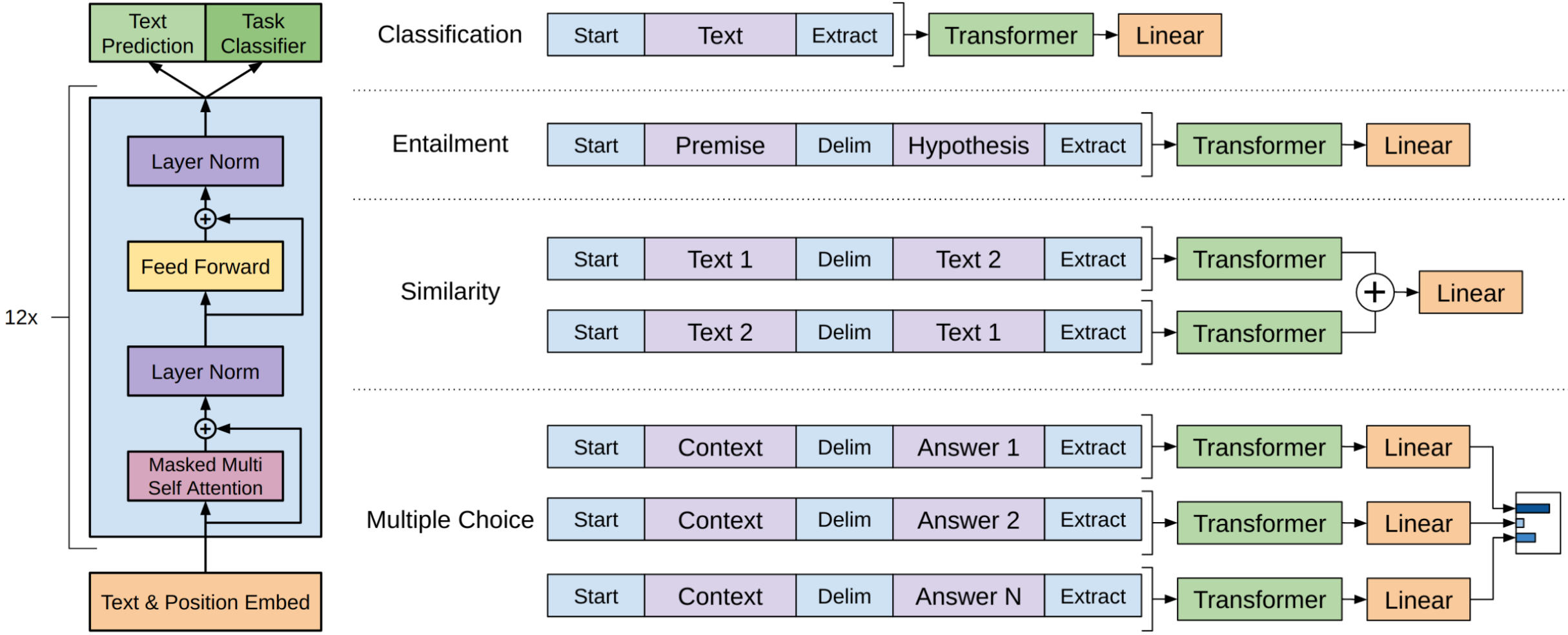

Generative AI

OpenAI GPT: Pre-training Transformer Decoders

- Unsupervised pre-train transform decoders for predicting the next word (GPT: Generative Pre-Training)
- Use 12 Transformer decoders in GPT-1
  - GPT-1: Improving Language Understanding with Unsupervised Learning (2018)
  - GPT-2: Better Language Models and Their Implications (2019)
  - GPT-3: Language Models are Few-Shot Learners (2020)



OpenAI GPT for Different Tasks



https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

OpenAI GPT-2

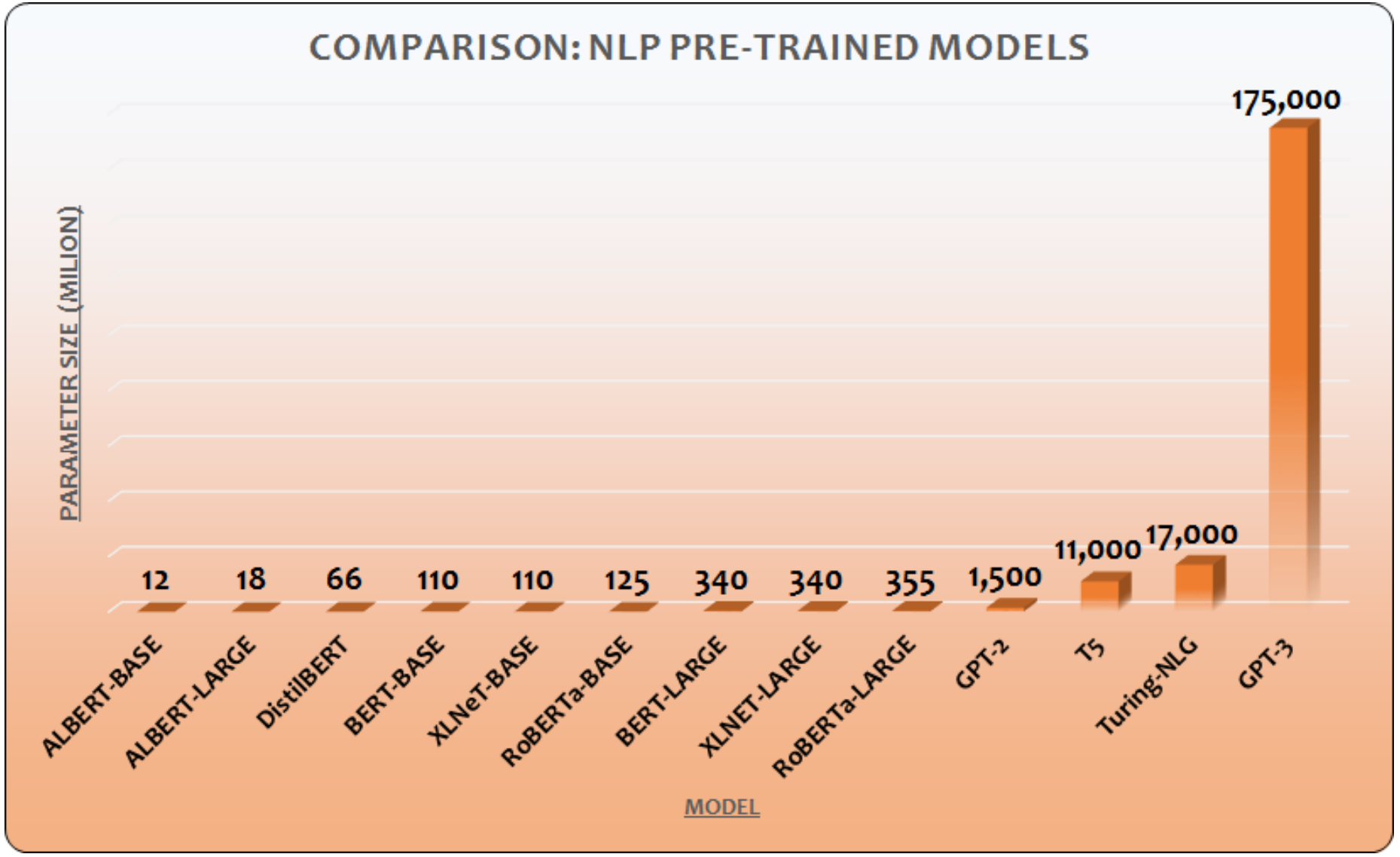
- Pre-trained using 40GB of Internet text
- Scale-up of GPT with 10X parameters trained with 10X data
- Other tricks
  - Layer normalization was moved to the input of each sub-block
  - An additional layer normalization was added after the final self-attention block

Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

https://openai.com/blog/better-language-models/

Size does Matter! GPT-3

- 175 Billion Parameters!
- 175×4=700GB
- 55 years and $4,600,000 to train - even with the lowest priced GPU cloud on the market.



https://medium.com/analytics-vidhya/openai-gpt-3-language-models-are-few-shot-learners-82531b3d3122

~~Open~~AI ChatGPT

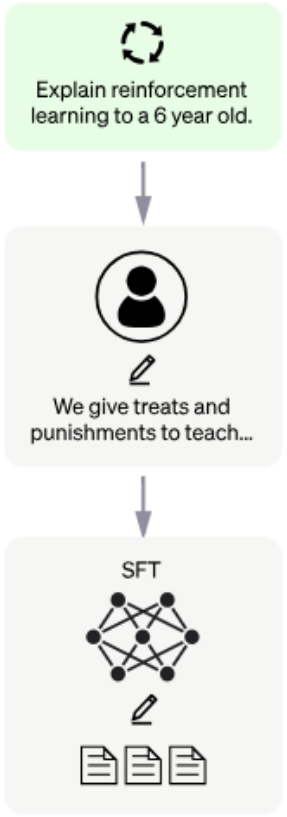
Step 1

Collect demonstration data and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3.5 with supervised learning.



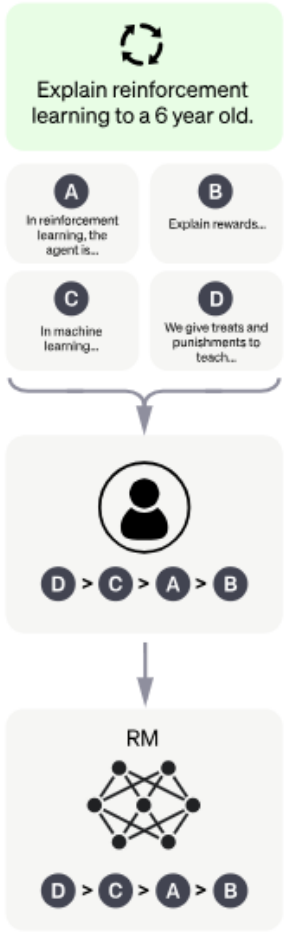
Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.



Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

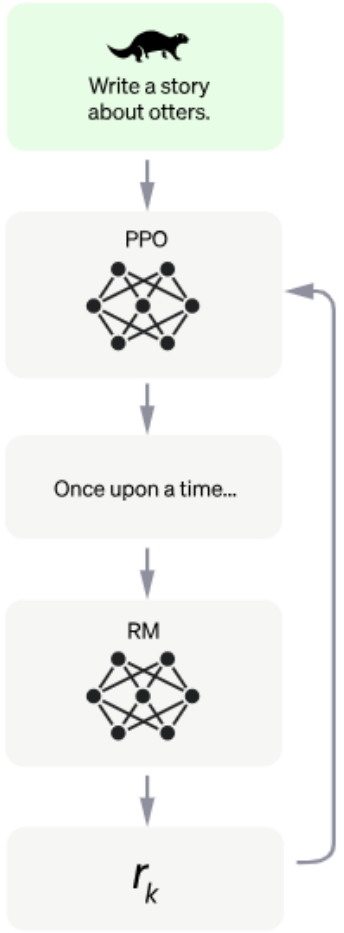
A new prompt is sampled from the dataset.

The PPO model is initialized from the supervised policy.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

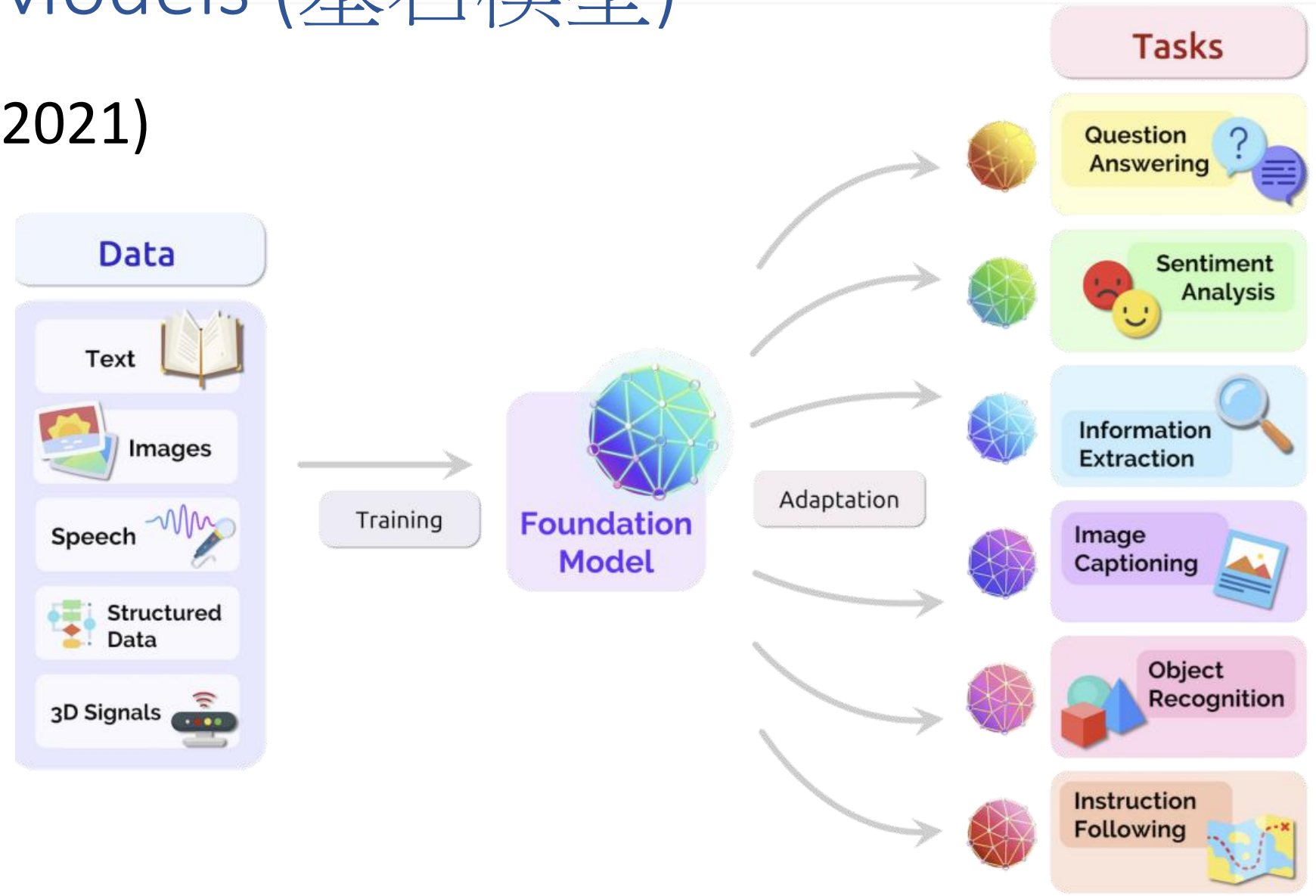


Foundation Models (基石模型)

- One model for All (2021)

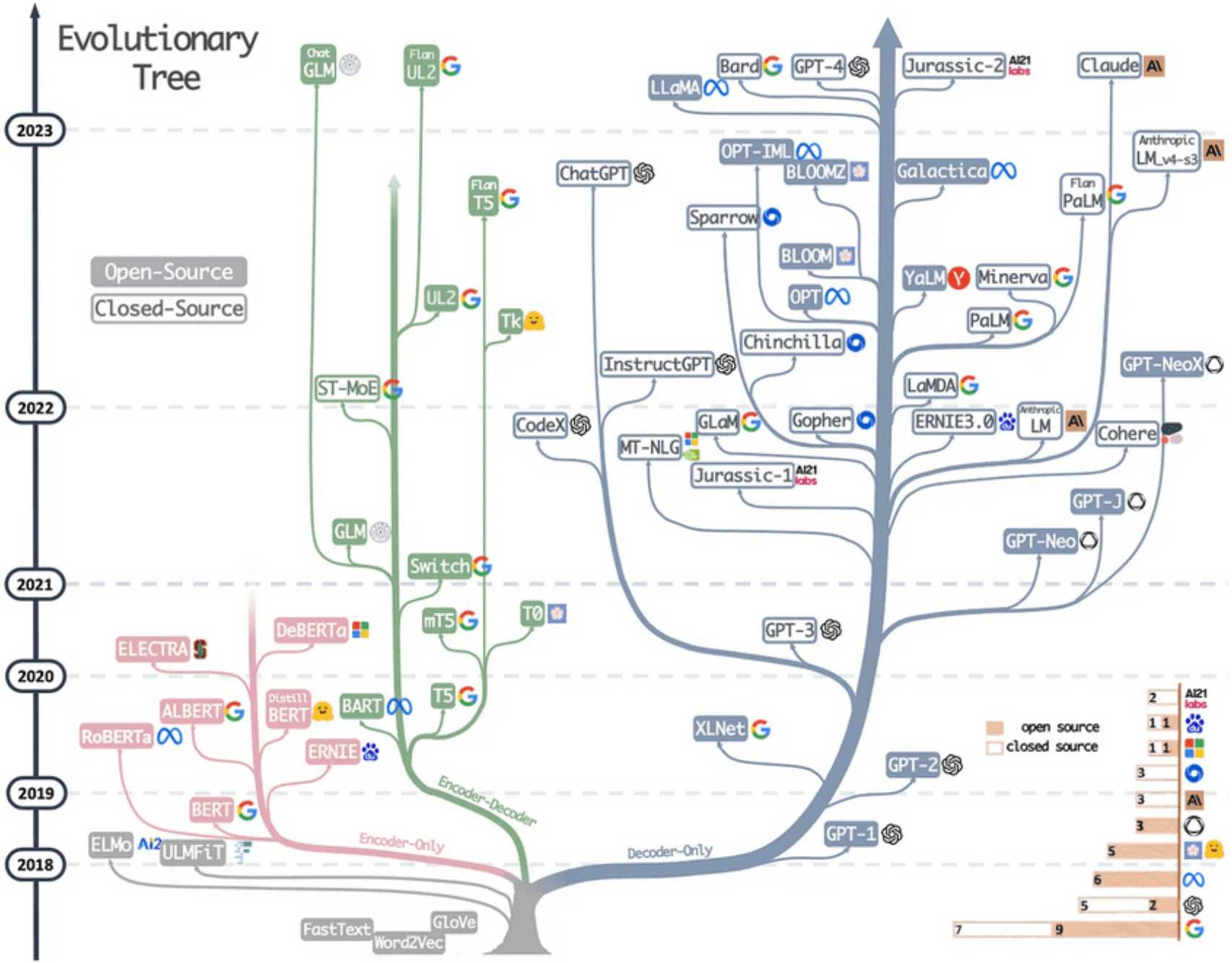
Stanford University
Human-Centered
Artificial Intelligence

Center for
Research on
Foundation
Models



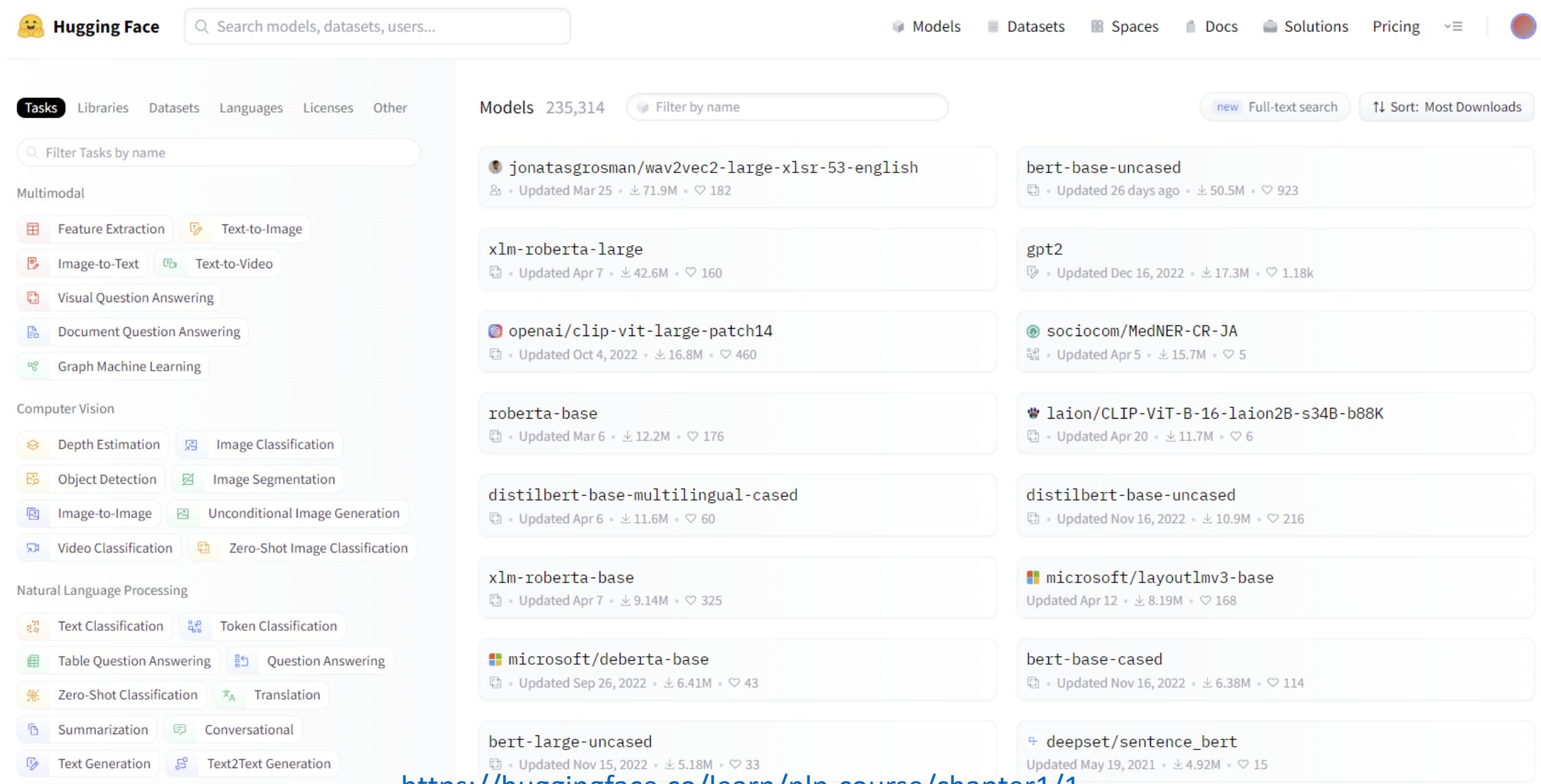
https://crfm.stanford.edu/assets/report.pdf

Large Language Model (LLM) Practical Guide



https://github.com/Mooler0410/LLMsPracticalGuide

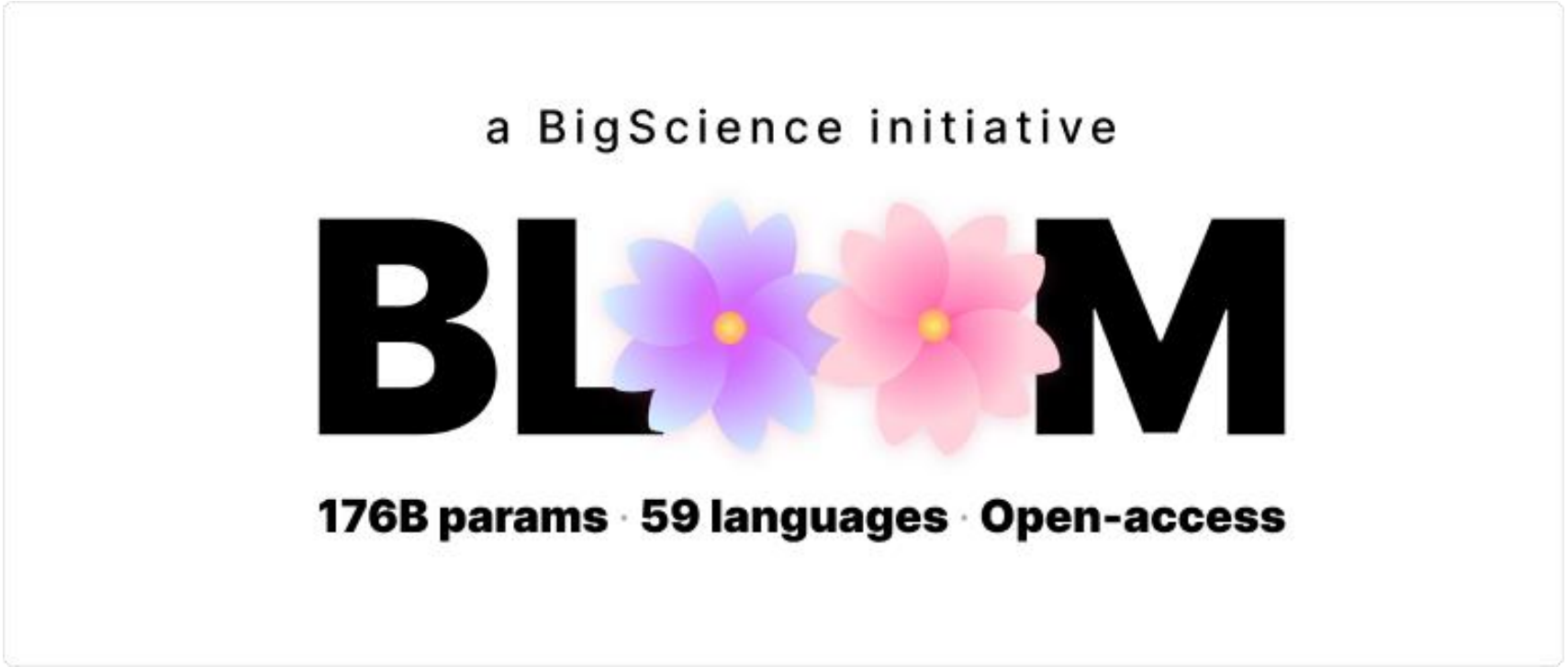
Hugging Face: Free LLM models



https://huggingface.co/learn/nlp-course/chapter1/1

BigScience Large Open-science Open-access Multilingual Language Model (BLOOM)

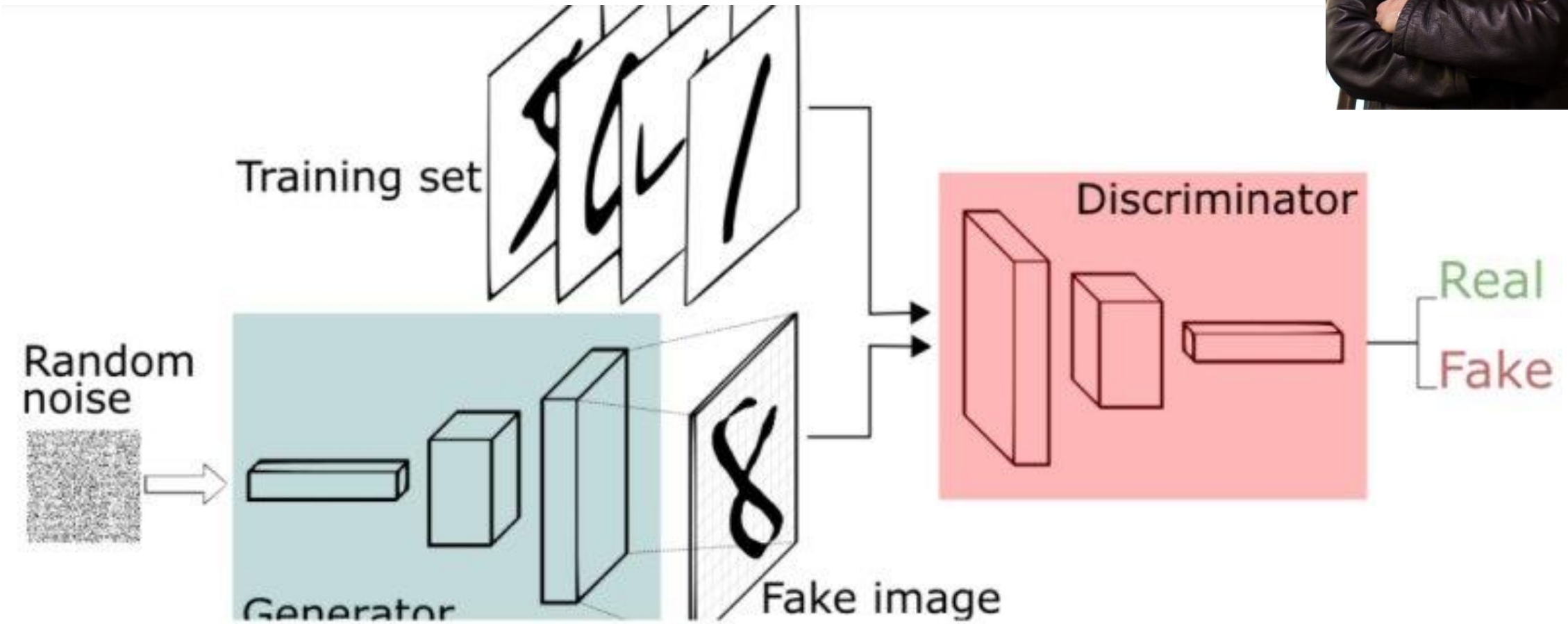
- With its 176 billion parameters, BLOOM is able to generate text in 46 natural languages and 13 programming languages.



https://huggingface.co/bigscience/bloom

Generative Adversarial Networks (GAN)

- Ian Goodfellow



Super Resolution

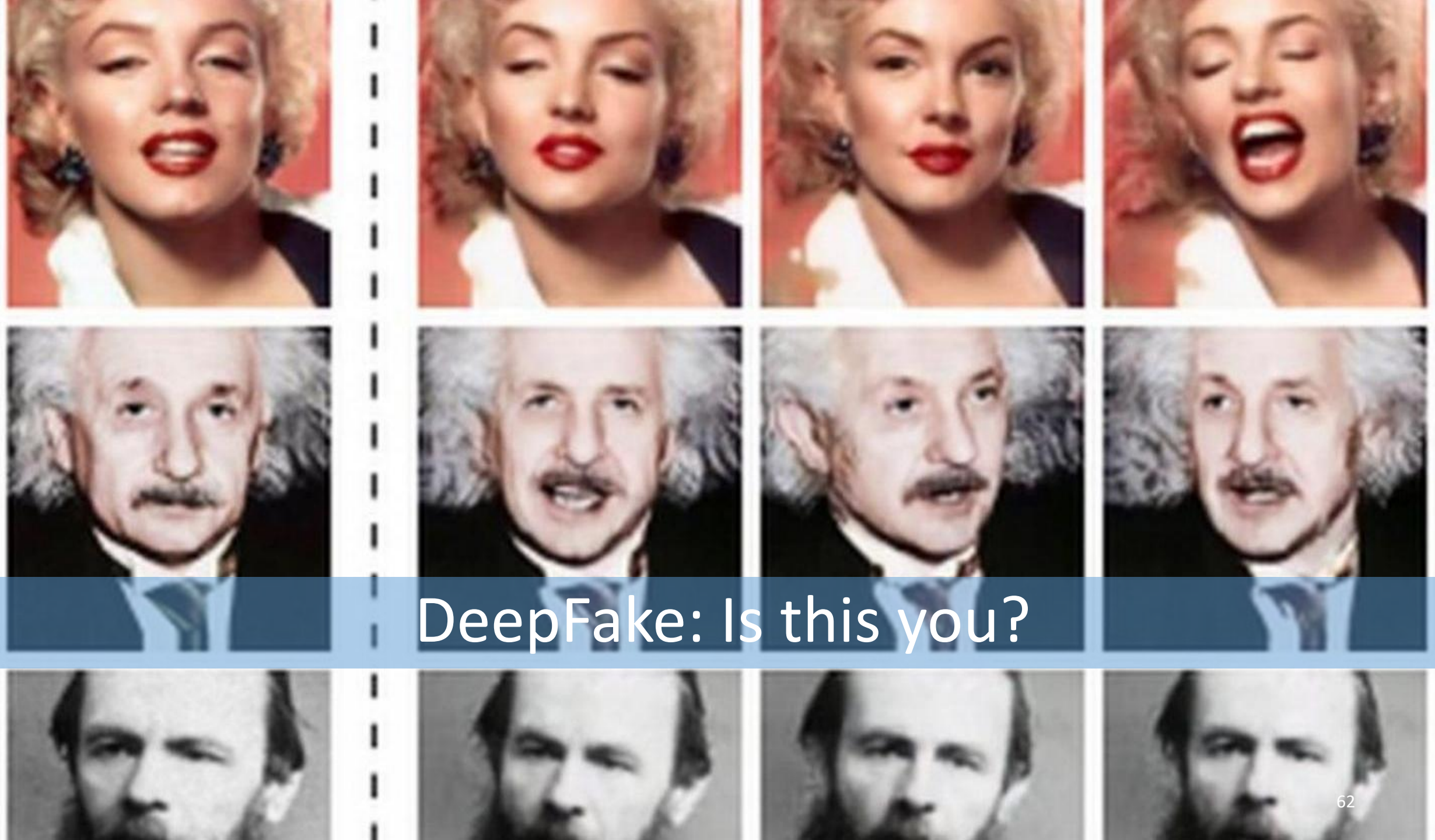
bicubic
(21.59dB/0.6423)

SRResNet
(23.53dB/0.7832)

SRGAN
(21.15dB/0.6868)

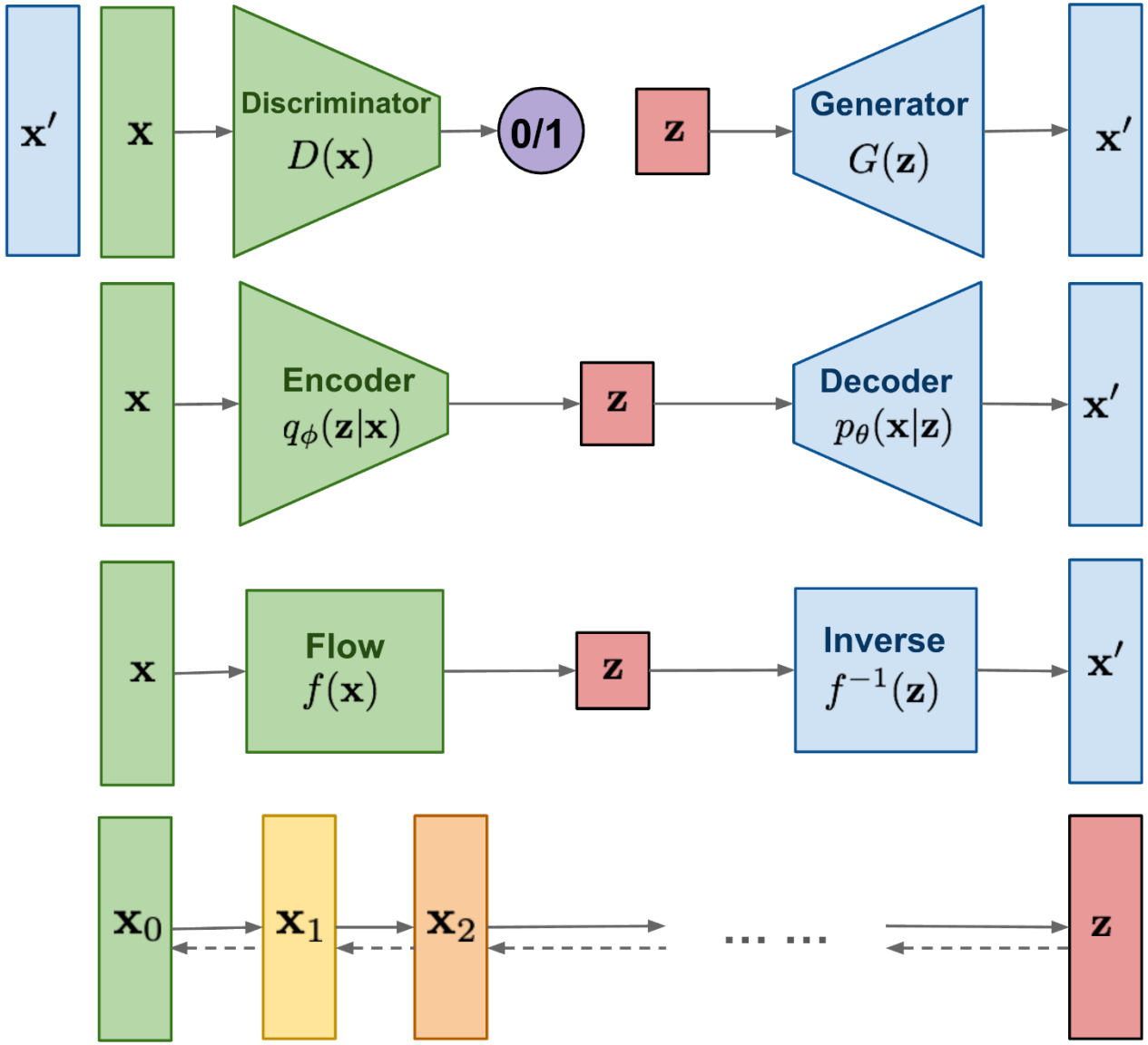
original

Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]



DeepFake: Is this you?

Overview of Different Generative Models



https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Diffusion is All You Need!

- Reverse diffusion process
- Flexible and tracible

18 Nov 2015

Deep Unsupervised Learning using Nonequilibrium Thermodynamics

Jascha Sohl-Dickstein JASCHA@STANFORD.EDU
Stanford University

Eric A. Weiss EAWEISS@BERKELEY.EDU
University of California, Berkeley

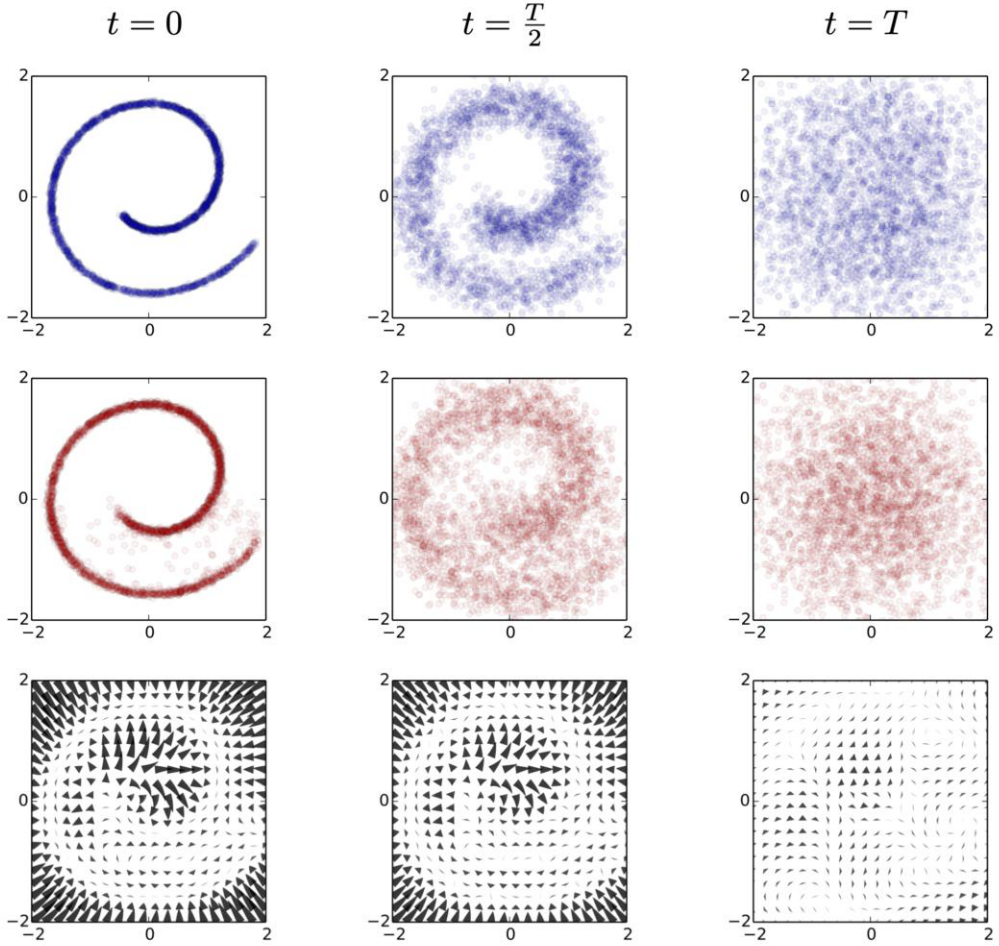
Niru Maheswaranathan NIRUM@STANFORD.EDU
Stanford University

Surya Ganguli SGANGULI@STANFORD.EDU
Stanford University

The forward trajectory $q(\mathbf{x}_{0:T})$

The reverse trajectory $p_\theta(\mathbf{x}_{0:T})$

The drifting term $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) - \mathbf{x}_t$



https://arxiv.org/pdf/1503.03585.pdf

Key Takeaways

1. Deep learning is a branch of Machine Learning, which is a sub-field of Artificial Intelligence.
2. There are two stages in machine learning: training (learning) and testing (inference).
3. Gradient Descent is used to train NN models by updating weights to minimize the prediction errors.
4. Convolutional Neural Networks (CNN) are used to recognize images.
5. RNN and LSTM are used to recognize sequential data such as text or speech.
6. Transformer told us that attention is all you need!
7. Generative Adversarial Networks (GANs) can be used to generate fake data, but now maybe diffusion is all you need.