Introduction to Deep Learning Methods and Their Scientific Applications

Yihui "Ray" Ren (yren@bnl.gov)
Computational Science Initiative

08/06/2021

BNL Physics Department Summer Lecture series

- Memorable Moments of AI
  - Computer Vision
  - Natural Language Processing (NLP)
  - Control and Optimization
- What is Deep Learning
- Some examples

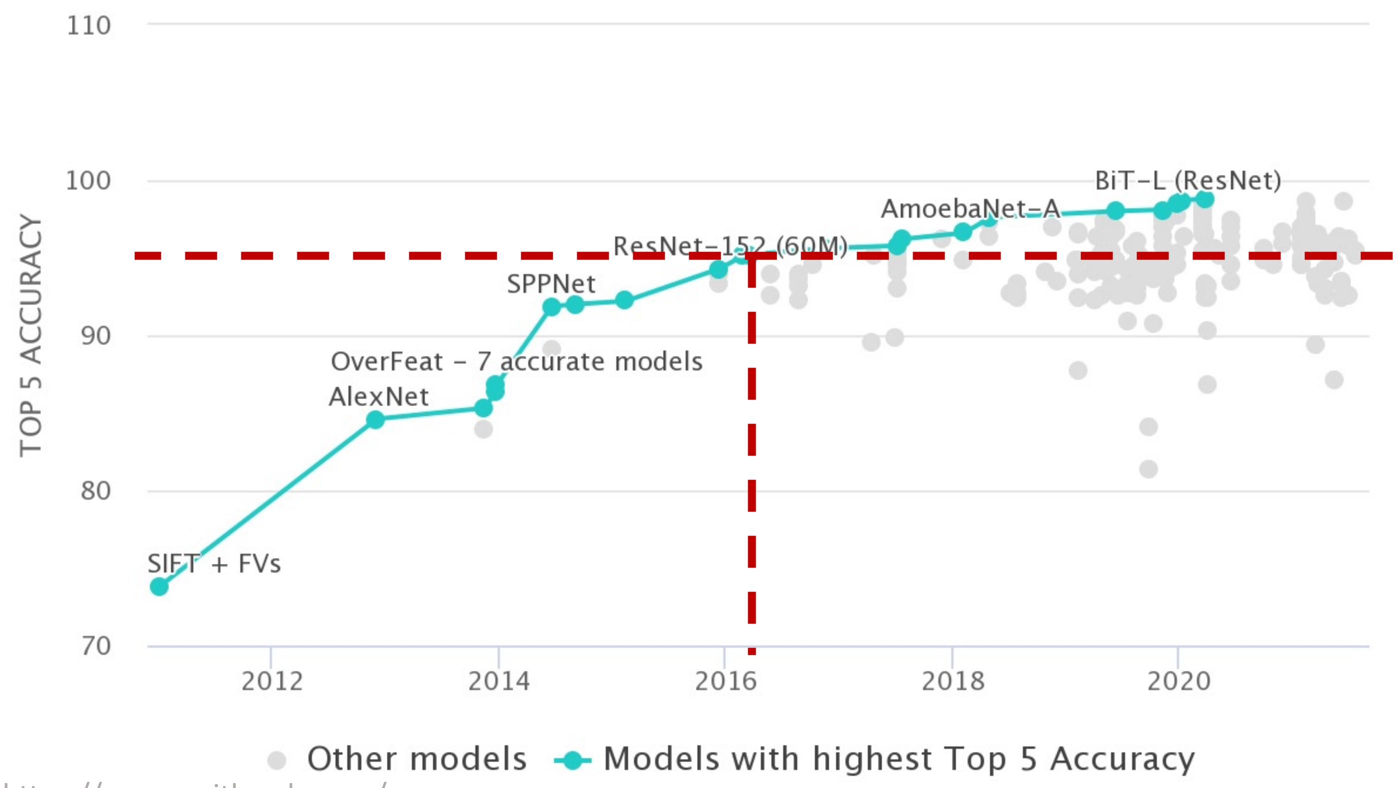
Feel free to ask questions at any moments!

ImageNet Large Scale Visual Recognition Challenge

Show affiliations

https://arxiv.org/pdf/1409.0575.pdf

Russakovsky, Olga ; Deng, Jia ; Su, Hao ; Krause, Jonathan ; Satheesh, Sanjeev ; Ma, Sean ; Huang, Zhiheng ; Karpathy, Andrej ; Khosla, Aditya ; Bernstein, Michael ; Berg, Alexander C. ; Fei-Fei, Li



https://paperswithcode.com/

Classify each image into one of 1000 different classes!

“The human error was estimated to be 5.1%”

Other popular computer vision tasks

- Image Segmentation
- Instance Segmentation
- Object Detection
- Pose Estimation
- Face Recognition

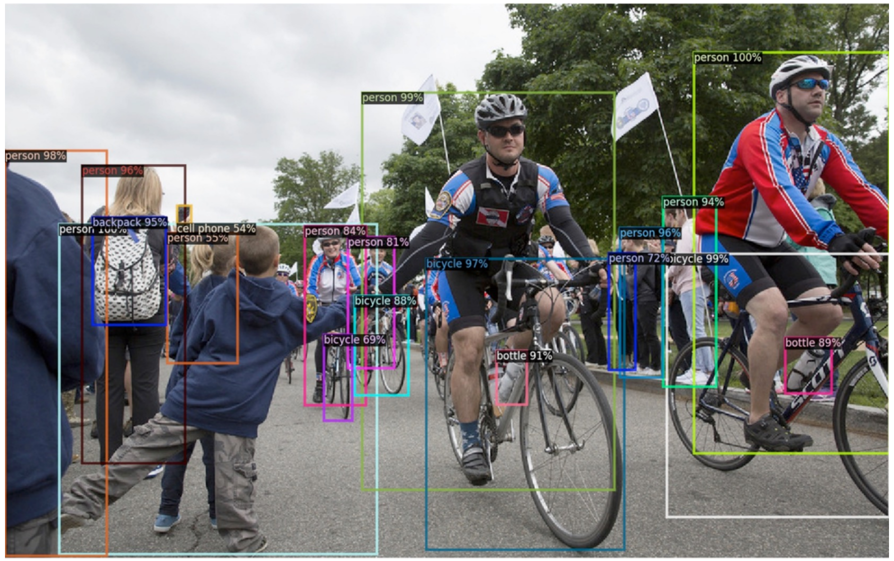


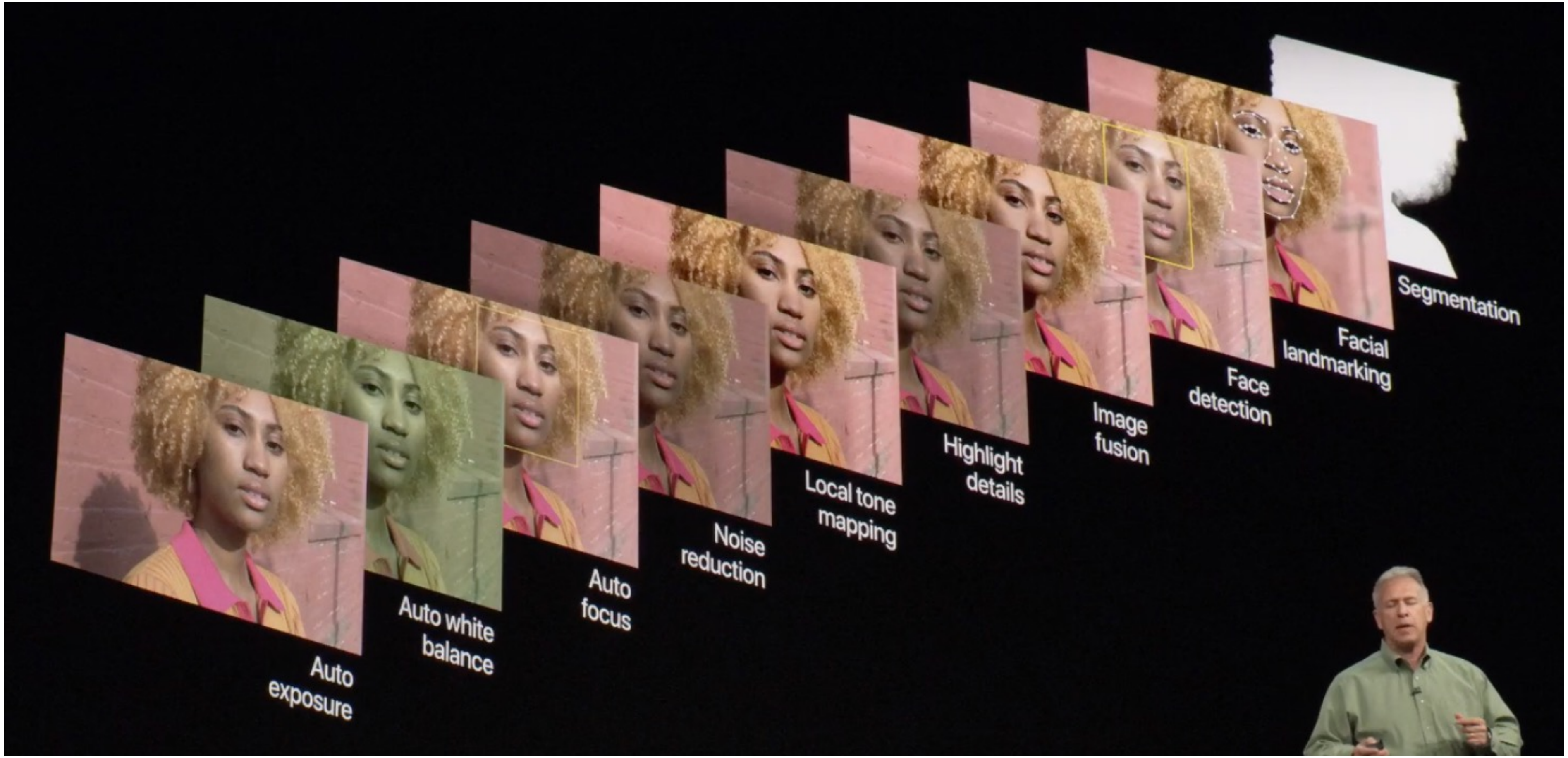
Image credit: https://ai.facebook.com/tools/detectron2/

How AI is changing photography

Cameras' biggest recent advancements have come from AI, not sensors and lenses

By Sam Byford | @345triangle | Jan 31, 2019, 8:00am EST

https://www.theverge.com/2019/1/31/18203363/ai-artificial-intelligence-photography-google-photos-apple-huawei



End-to-end computation Photography.

Self-driving Cars

- Level 1 “hands-on”
- <u>Level 2 “hands-off”</u>
- Level 3 “eyes-off”.
- Level 4 “minds-off”

https://www.tesla.com/videos/autopilot-self-driving-hardware-neighborhood-long

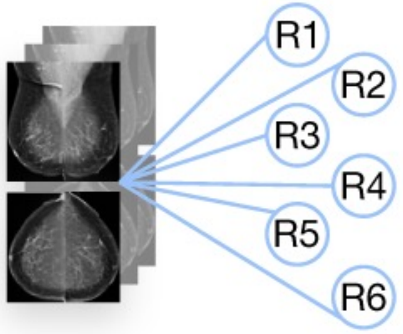
Article | Published: 01 January 2020

International evaluation of an AI system for breast cancer screening

Scott Mayer McKinney, Marcin Sieniek, […]Shravya Shetty

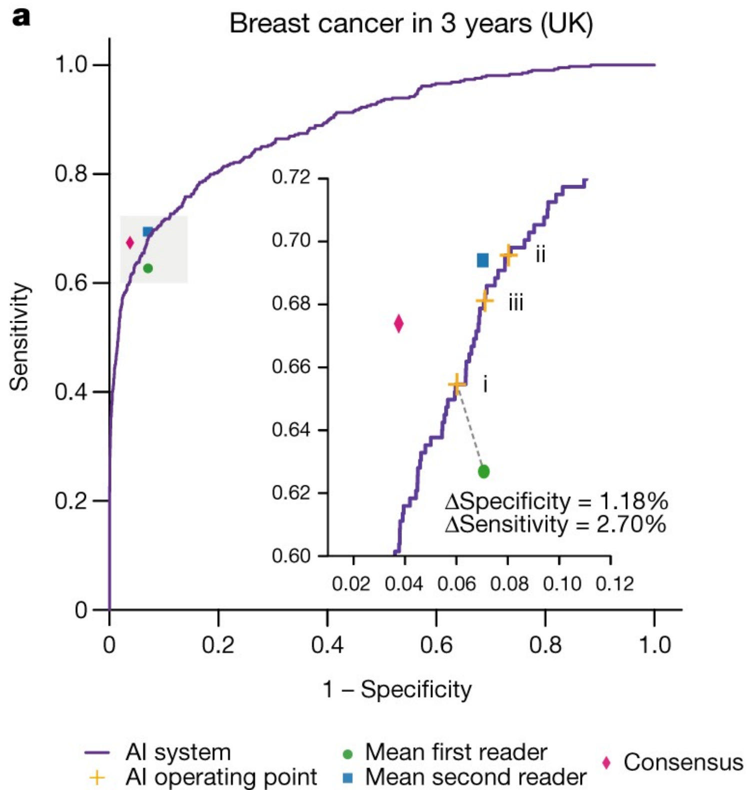
Nature **577**, 89–94 (2020) | Cite this article

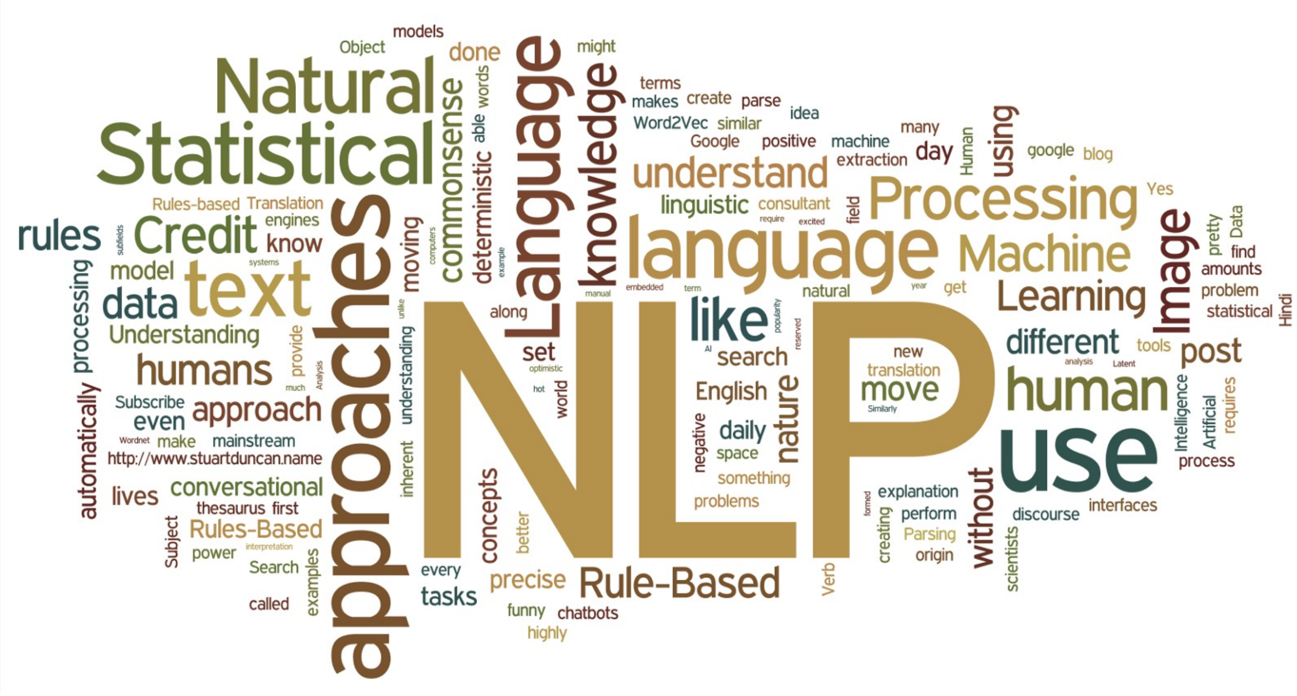
Independently conducted reader study



6 radiologists read 500 cases from US test set

"In an independent study of six radiologists, **<u>the AI system outperformed all of the human readers</u>**: the area under the receiver operating characteristic curve (AUC-ROC) for the AI system was greater than the AUC-ROC for the average radiologist by an absolute margin of 11.5%."

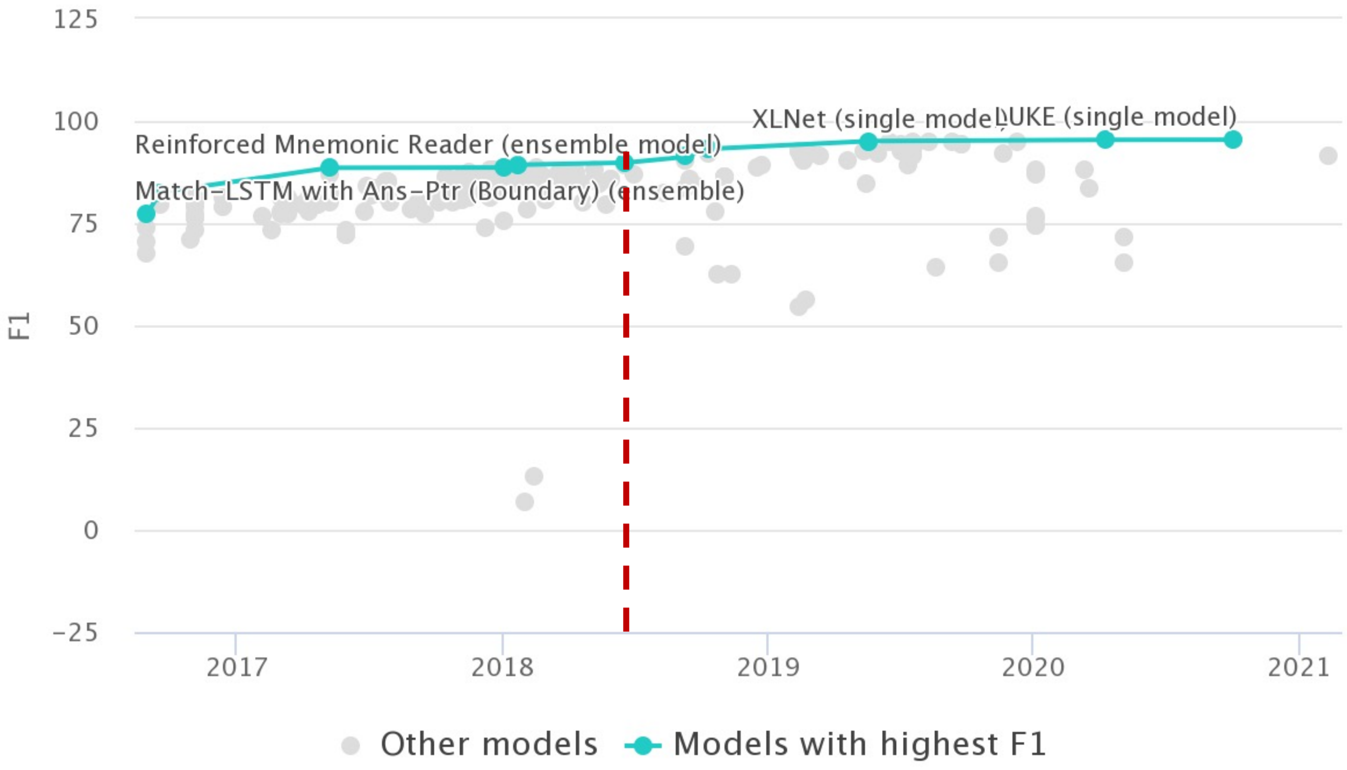




Common NLP tasks:

- Machine Translation
- Question Answering
- Text Generation
- Named Entity Recognition
- Sentiment Analysis
- Text Summarization
- Speech Recognition

Question Answering



The resulting human performance score on the test set **is 91.2% F1**.

Passage Sentence

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under gravity.

Question

What causes precipitation to fall?

Answer Candidate

gravity

The Stanford Question Answering Dataset (SQuAD)
https://rajpurkar.github.io/mlx/qa-and-squad/

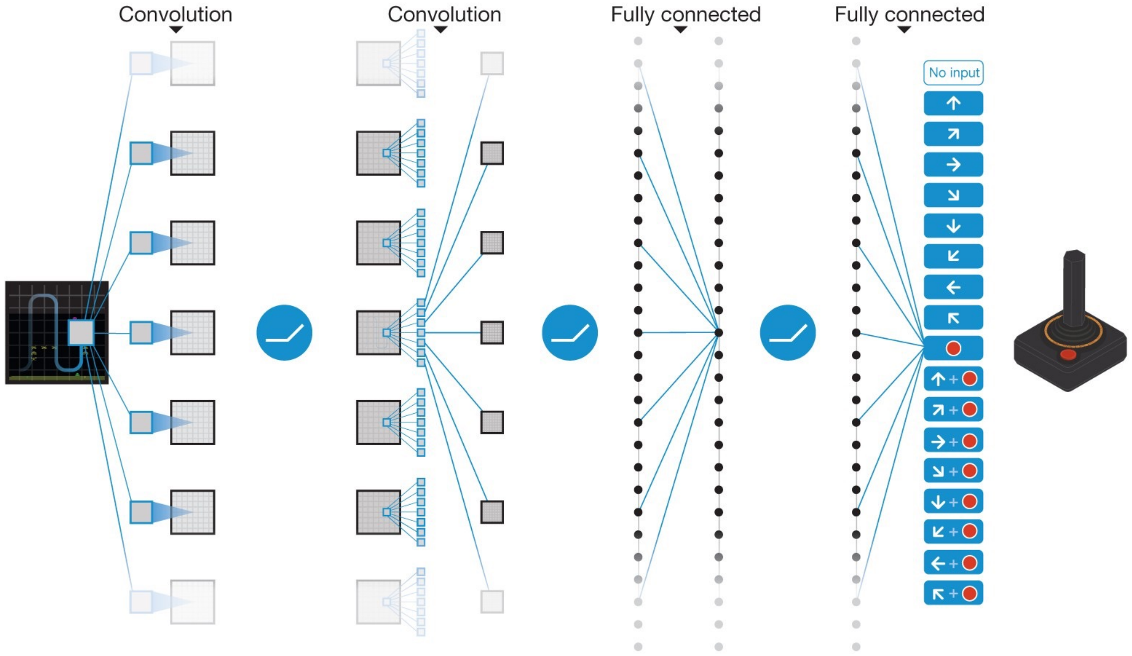
Published: 25 February 2015

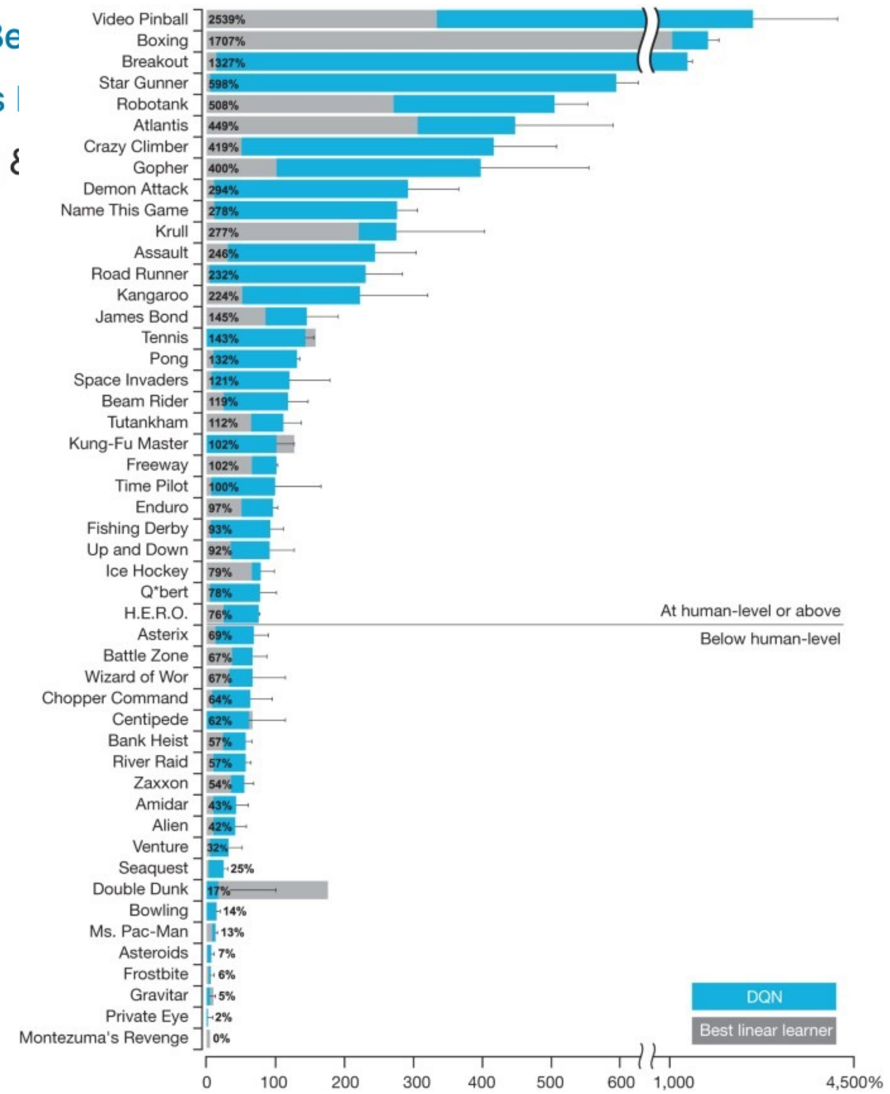
DeepMind

Human-level control through deep reinforcement learning

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Be
Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles
Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg &
Hassabis

Nature **518**, 529–533 (2015) | Cite this article

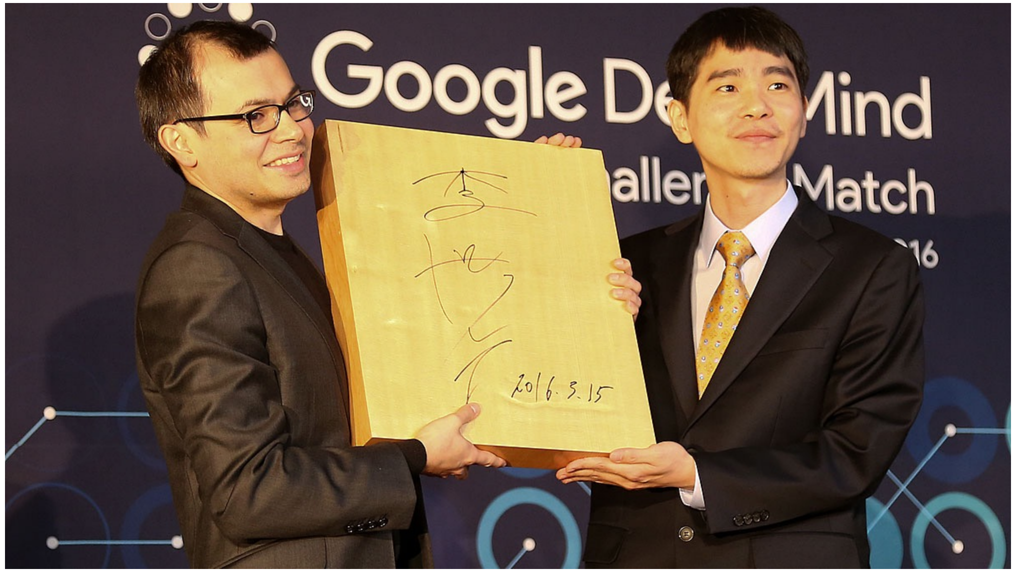


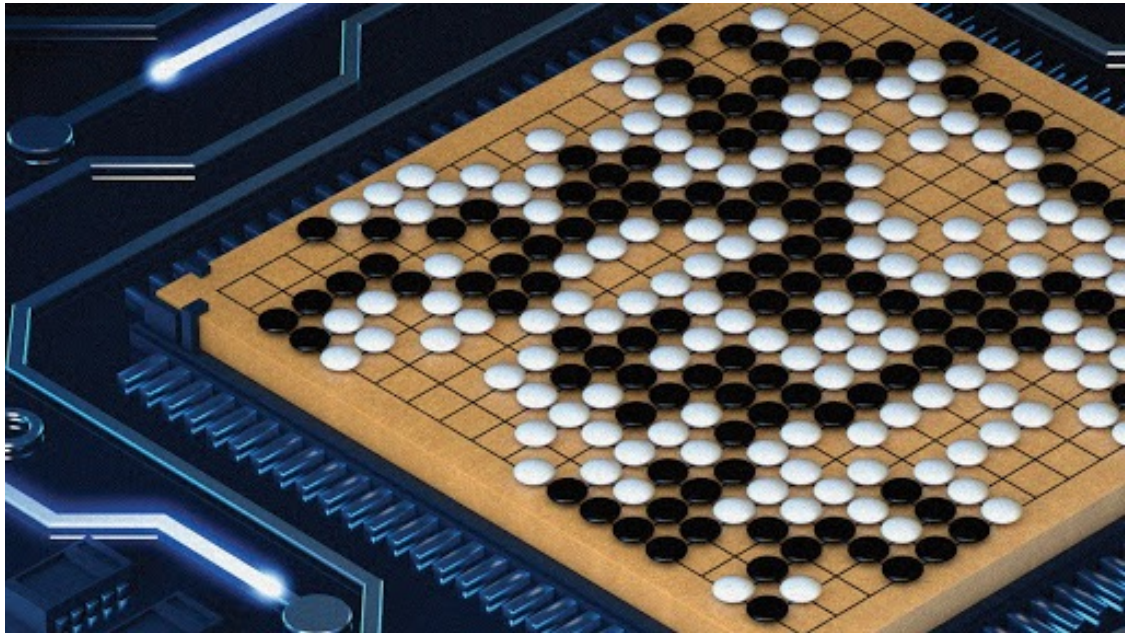


Published: 27 January 2016

Mastering the game of Go with deep neural networks and tree search

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche,





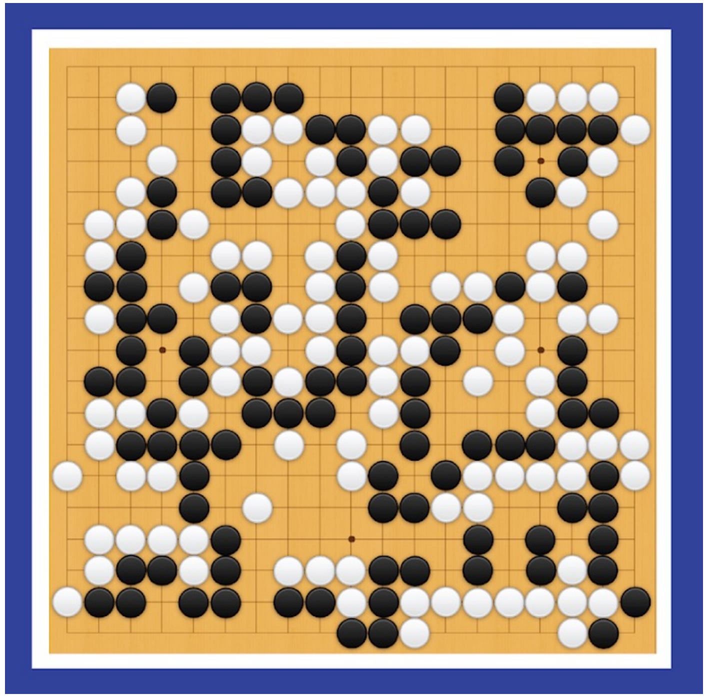
"AlphaGo's 4-1 victory in Seoul, South Korea, on March 2016"

Published: 19 October 2017

DeepMind

Mastering the game of Go without human knowledge

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez,



AlphaGo 3-0 Ke Jie world No. 1 ranking player Go player in a three-game Go match in May 2017.

NEWS | 30 November 2020

‘It will change everything’: DeepMind’s AI makes gigantic leap in solving protein structures

Google’s deep-learning program for determining the 3D shapes of proteins stands to transform biology, say scientists.

Article | Published: 15 July 2021

DeepMind

This is an unedited manuscript that has been accepted for publication. Nature Research are providing this early version of the manuscript as a service to our authors and readers. The manuscript will undergo copyediting, typesetting and a proof review before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers apply.

Highly accurate protein structure prediction with AlphaFold

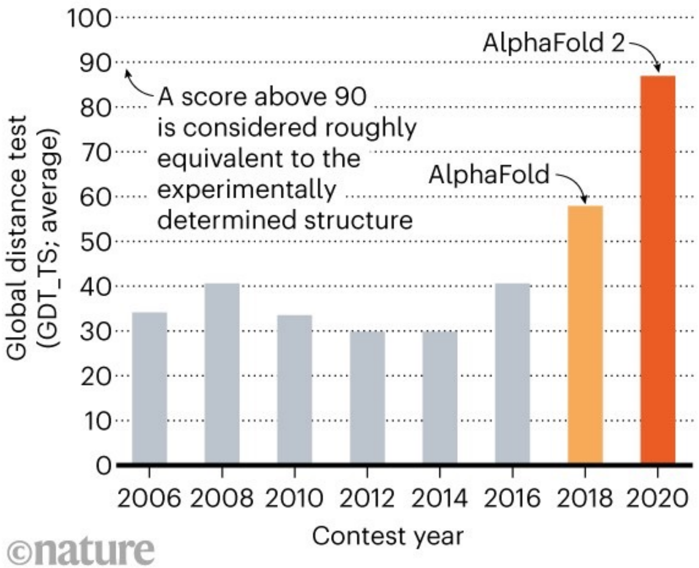
John Jumper, Richard Evans, [...]Demis Hassabis

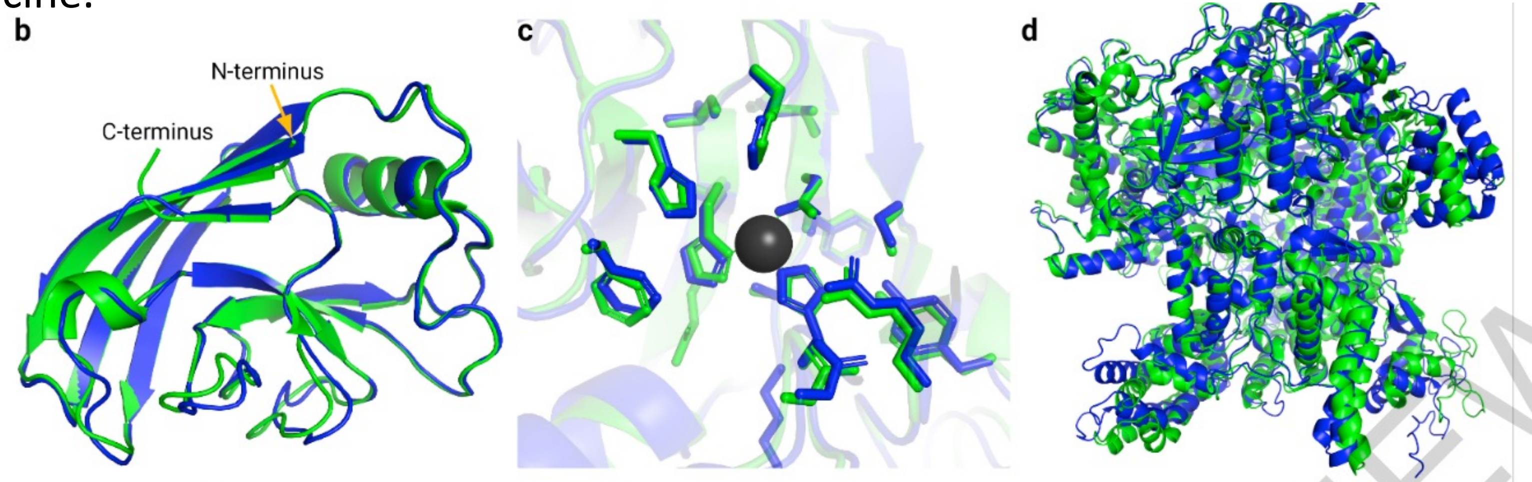
Nature (2021) | Cite this article

“The ability to accurately predict protein structures from their amino-acid sequence would be a huge boon to life sciences and medicine.”

STRUCTURE SOLVER

DeepMind’s AlphaFold 2 algorithm significantly outperformed other teams at the CASP14 protein-folding contest — and its previous version’s performance at the last CASP.





T1049 – AlphaFold / experiment
RMSD$_{95}$: 0.8 Å, TM-score: 0.93

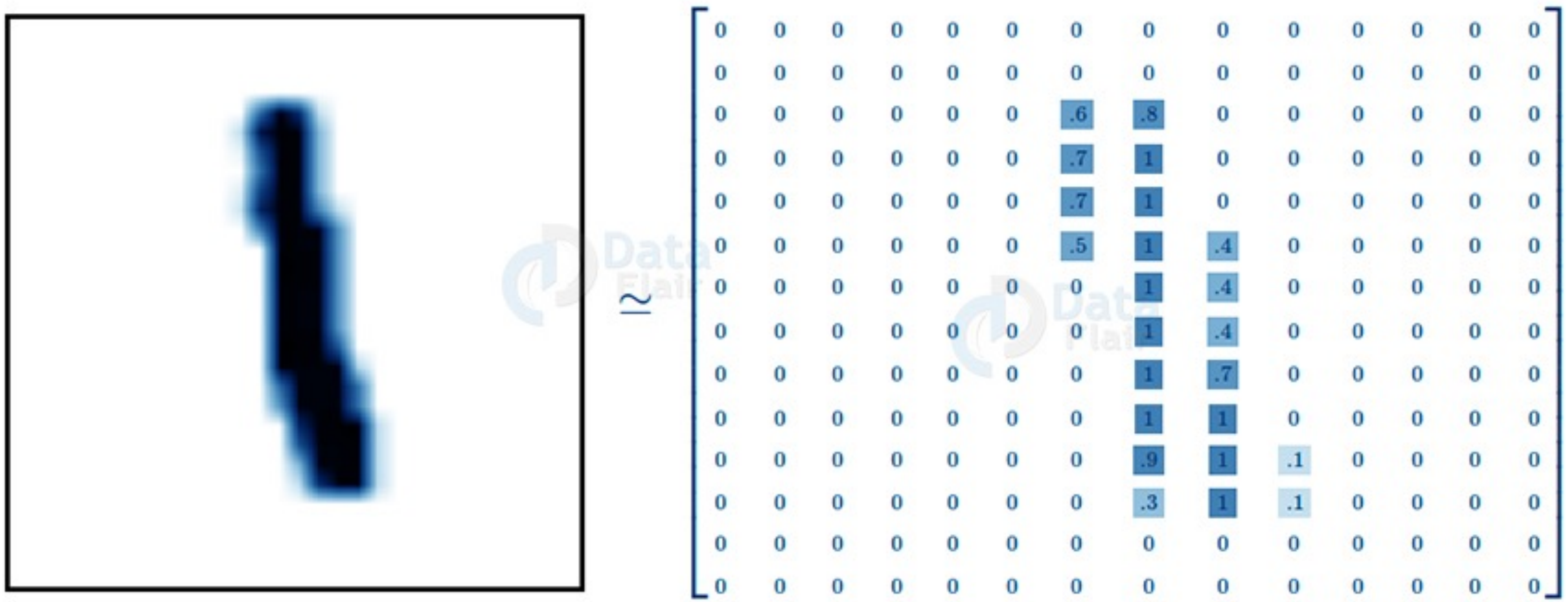
T1056 (6yj1) – AlphaFold / experiment
RMSD: 0.59 Å within 8 Å of Zn

T1044 – AlphaFold / experiment
RMSD$_{95}$: 2.2 Å, TM-score: 0.96

https://www.nature.com/articles/d41586-020-03348-4
https://www.nature.com/articles/s41586-021-03819-2

- Memorable Moments of AI
  - Computer Vision
  - Natural Language Processing (NLP)
  - Control and Optimization
- What is Deep Learning
  - Stochastic Gradient Descent & Backpropagation (90s)
  - Fast enough hardware & Big Data (recently)
- Some examples

Remixed from: http://introtodeeplearning.com/



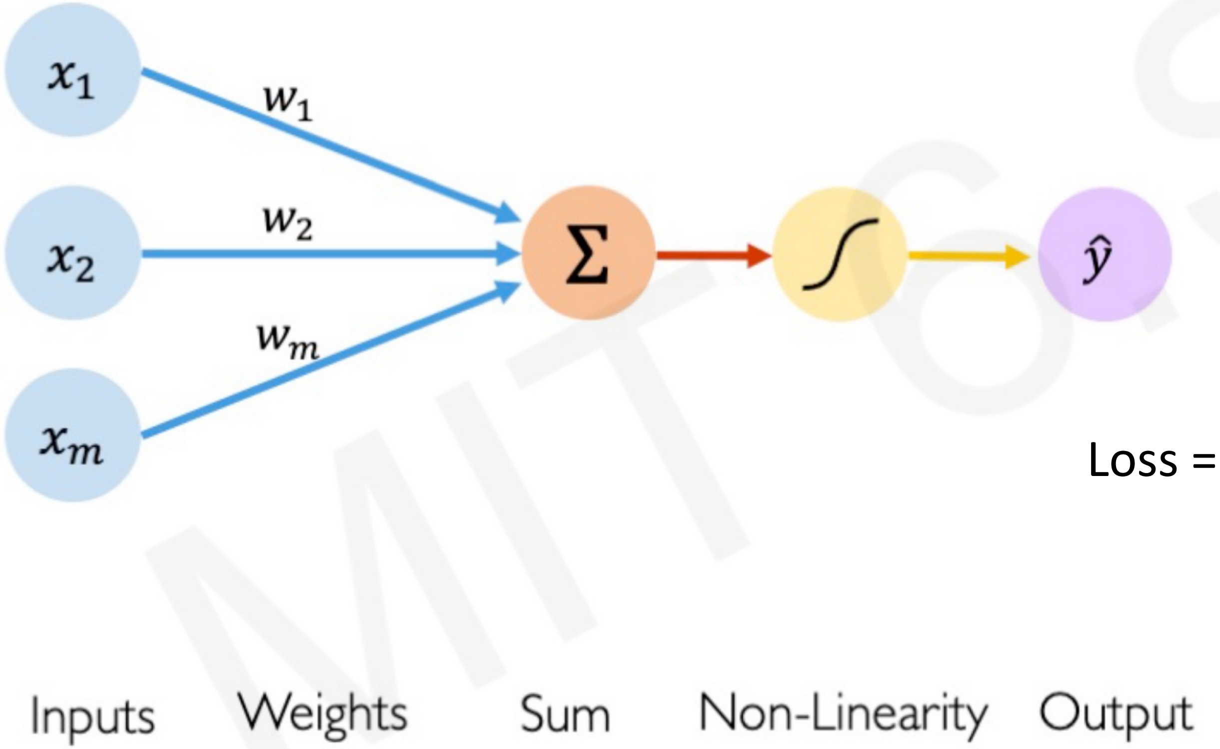
How can we teach a computer this is a “one”?

Image credit: https://data-flair.training/blogs/tensorflow-mnist-dataset/

How about cat vs dog?

Image credit: https://www.kaggle.com/c/dogs-vs-cats

The Perceptron: Forward Propagation



Output

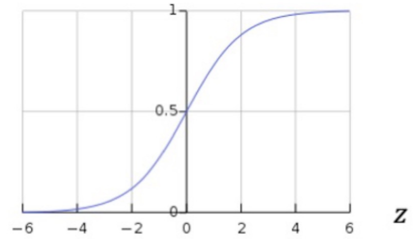
Linear combination of inputs

$$\hat{y} = g\left(\sum_{i=1}^{m} x_i \, w_i\right)$$

Non-linear activation function

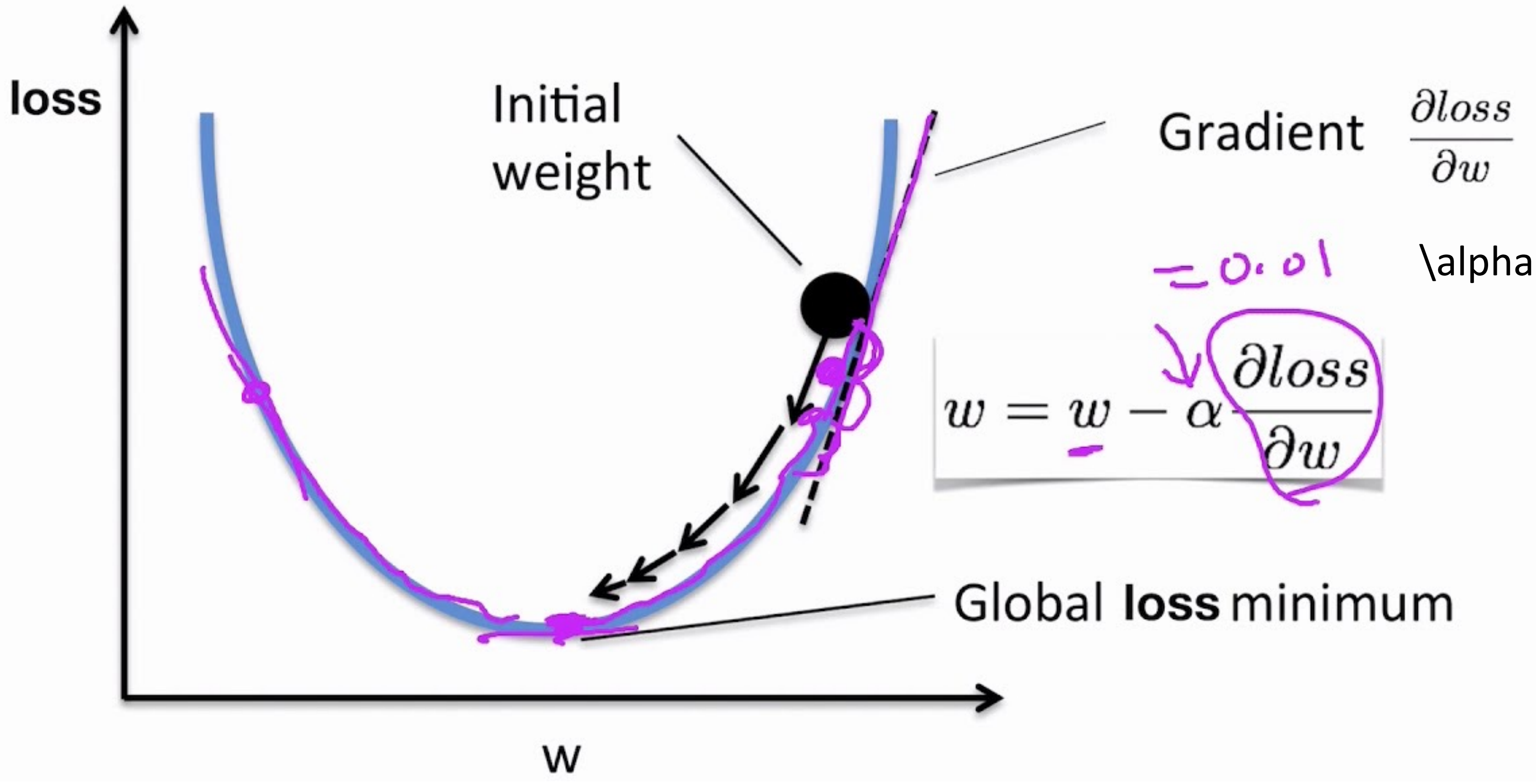
- Example: sigmoid function

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



Loss = L(y, y')

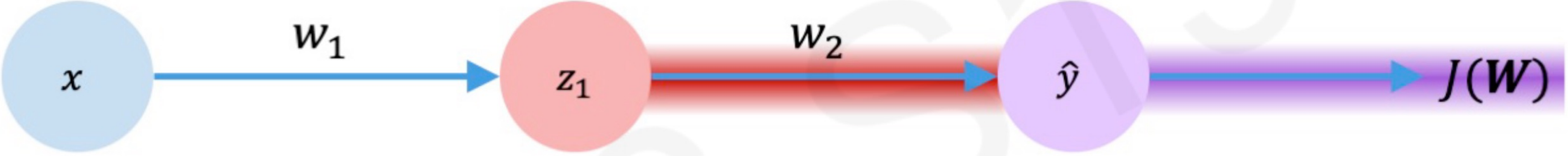
Gradient descent algorithm



\alpha = Learning Rate.

HKUSTCode: https://github.com/hunkim/PyTorchZeroToAll
Slides: http://bit.ly/PyTorchZeroAll

Computing Gradients: Backpropagation

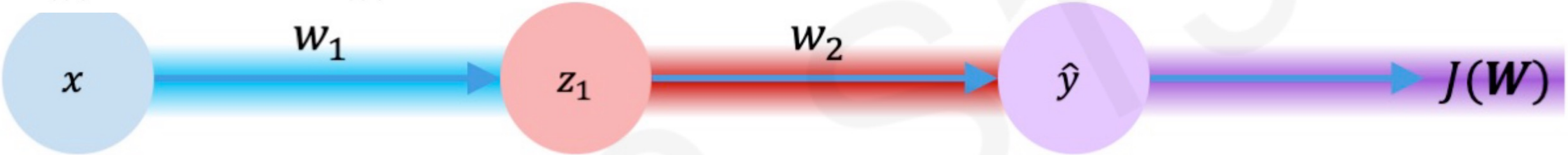


$$\frac{\partial J(\boldsymbol{W})}{\partial w_2} = \frac{\partial J(\boldsymbol{W})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial w_2}$$

Computing Gradients: Backpropagation

$$\frac{\partial J(\boldsymbol{W})}{\partial w_1} = \frac{\partial J(\boldsymbol{W})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial w_1}$$

Apply chain rule! Apply chain rule!

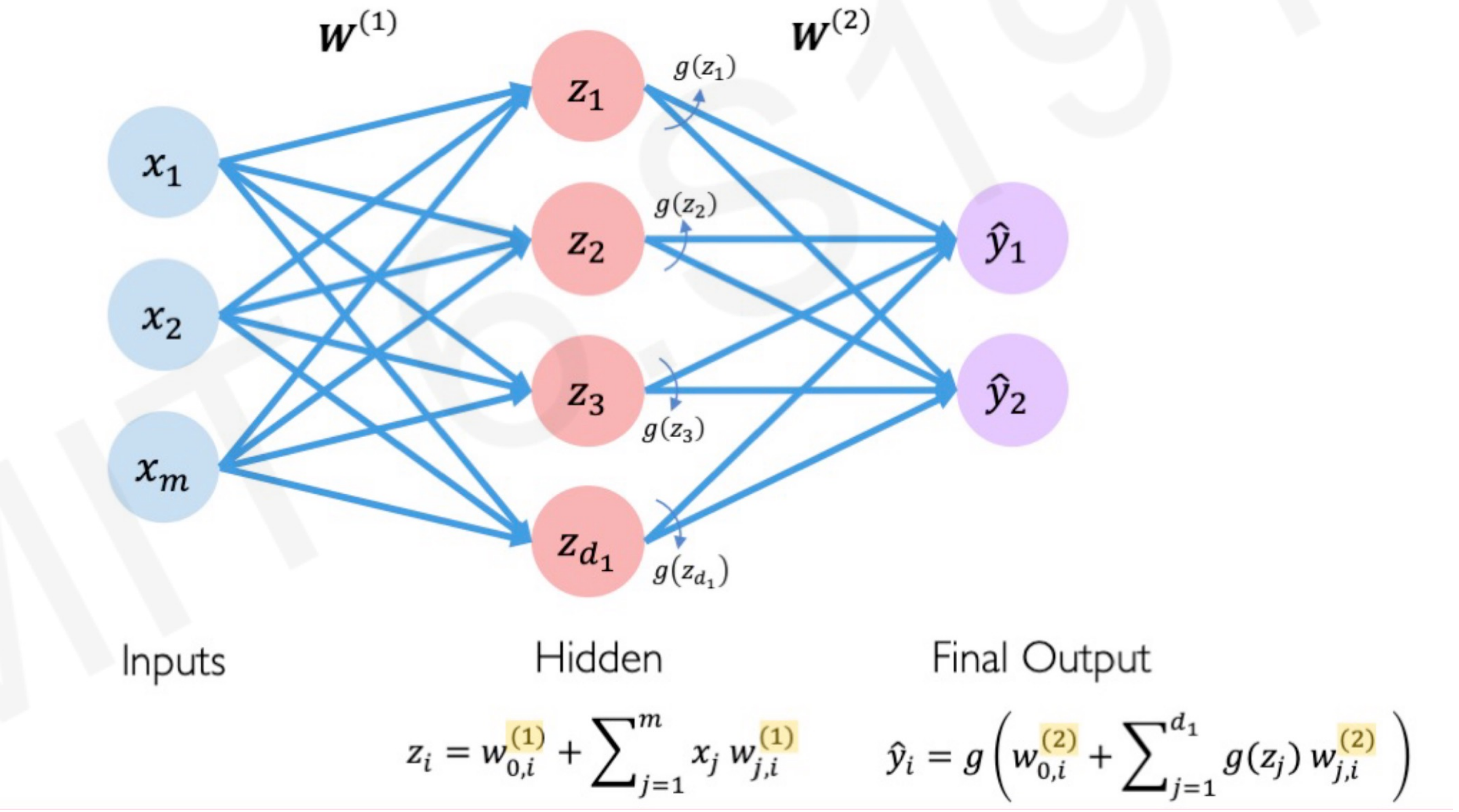


$$\frac{\partial J(\boldsymbol{W})}{\partial w_1} = \frac{\partial J(\boldsymbol{W})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial z_1} * \frac{\partial z_1}{\partial w_1}$$

We need to do this <u>for ALL weights</u> in the network.

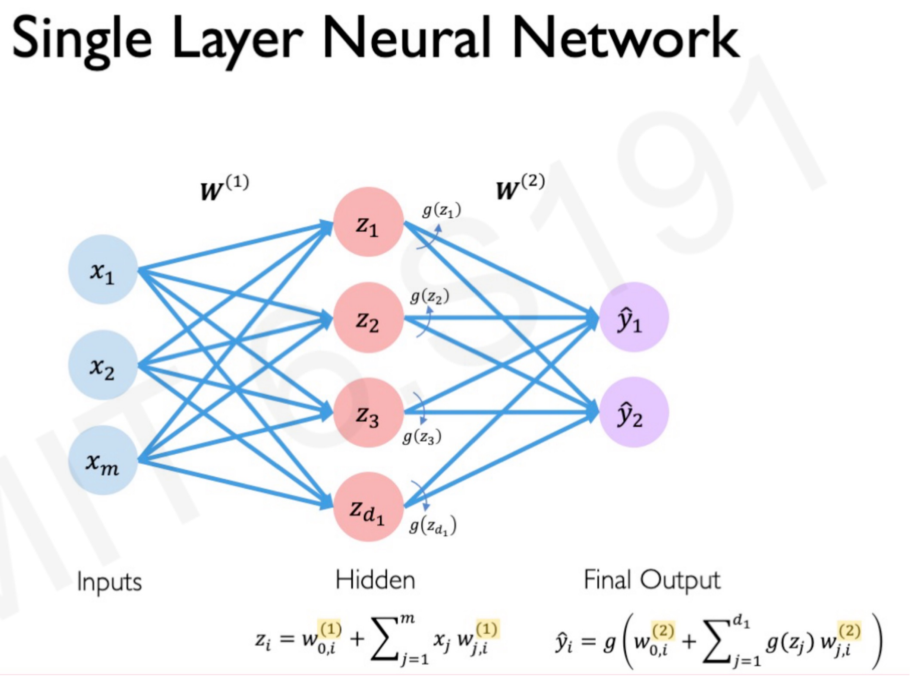
Single Layer Neural Network

Make it wider



Make it wider and deeper

Multilayer Perceptron (MLP)



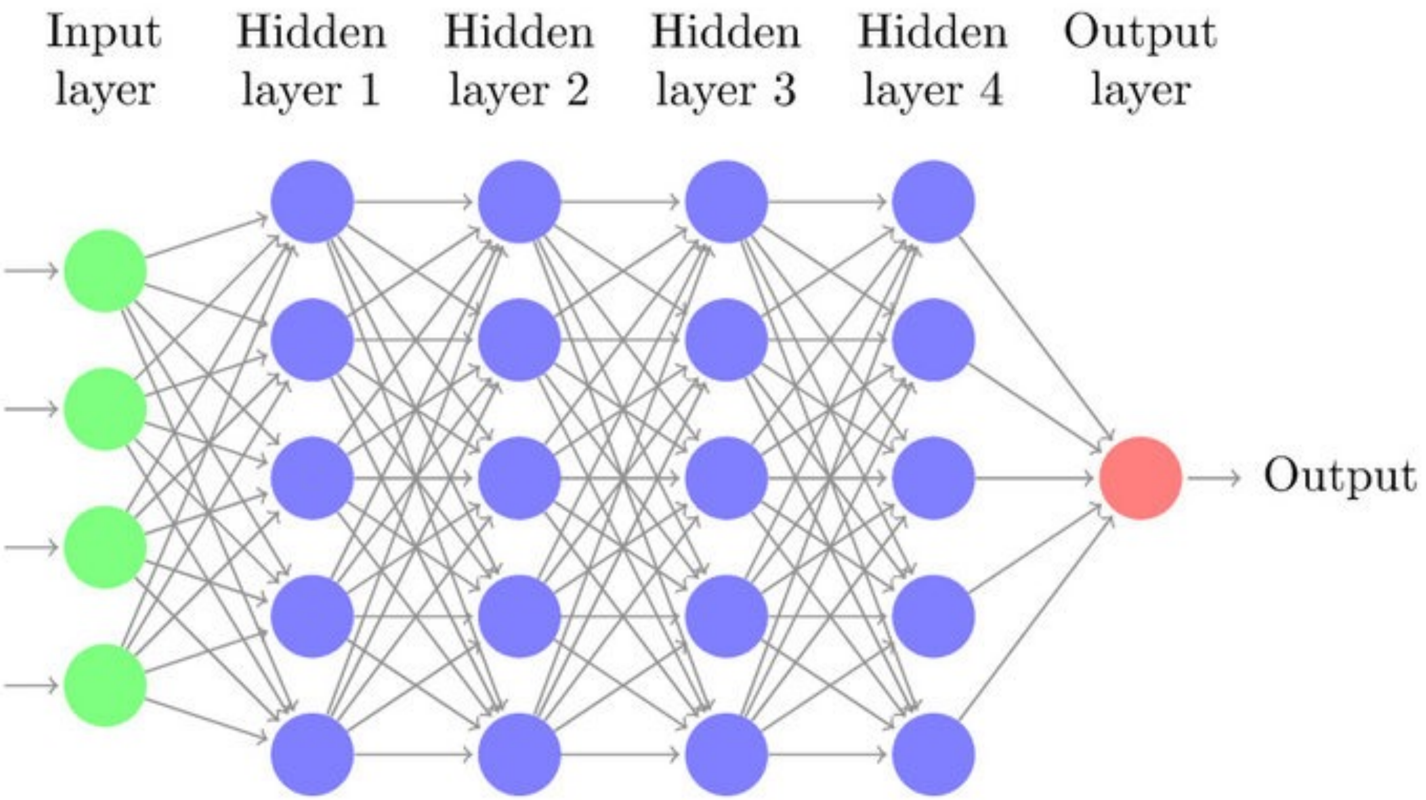
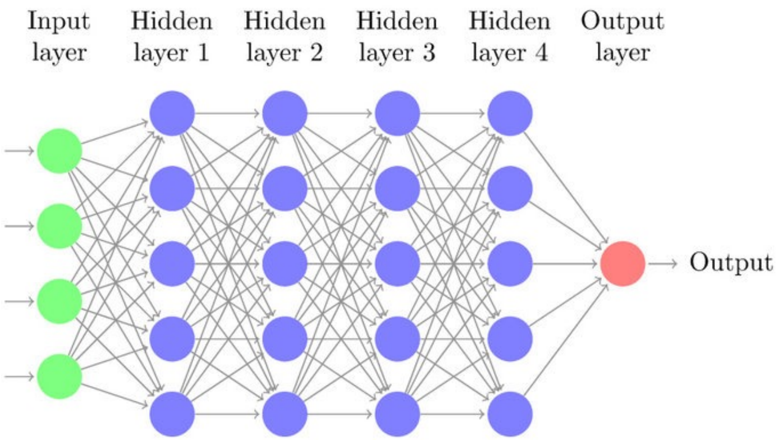
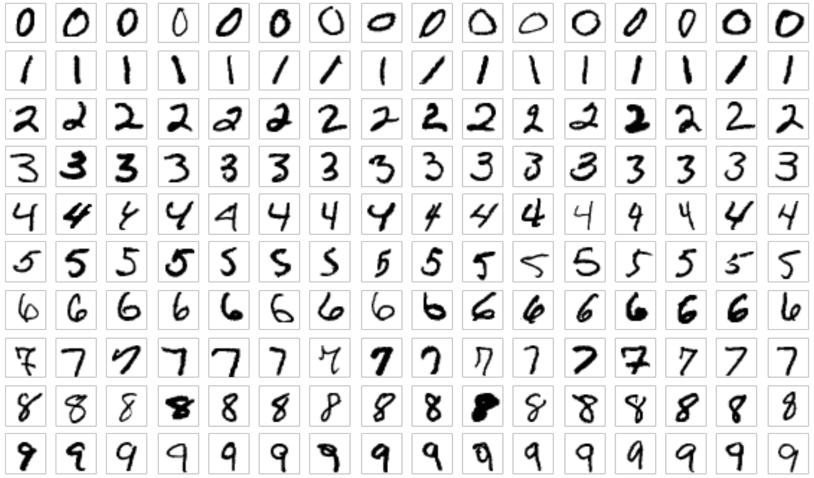


Image credit: https://www.researchgate.net/figure/Multi-Layer-Perceptron-MLP-diagram-with-four-hidden-layers-and-a-collection-of-single_fig1_334609713

However, there are two major problems of MLP when handling image data.

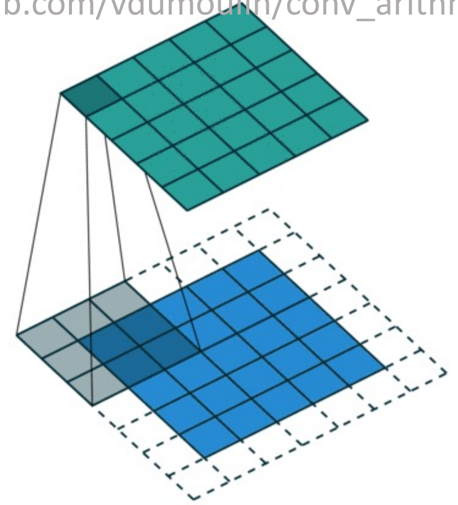
1. Parameter efficiency. Each layer is a large matrix of dimension `input_dim X output_dim`. (i.e. think about image size of 1000x1000)
2. Does not handle “shift-invariance” of images. If we shift the digital “1” several pixels away, the model has to learn it as a “new” instance.

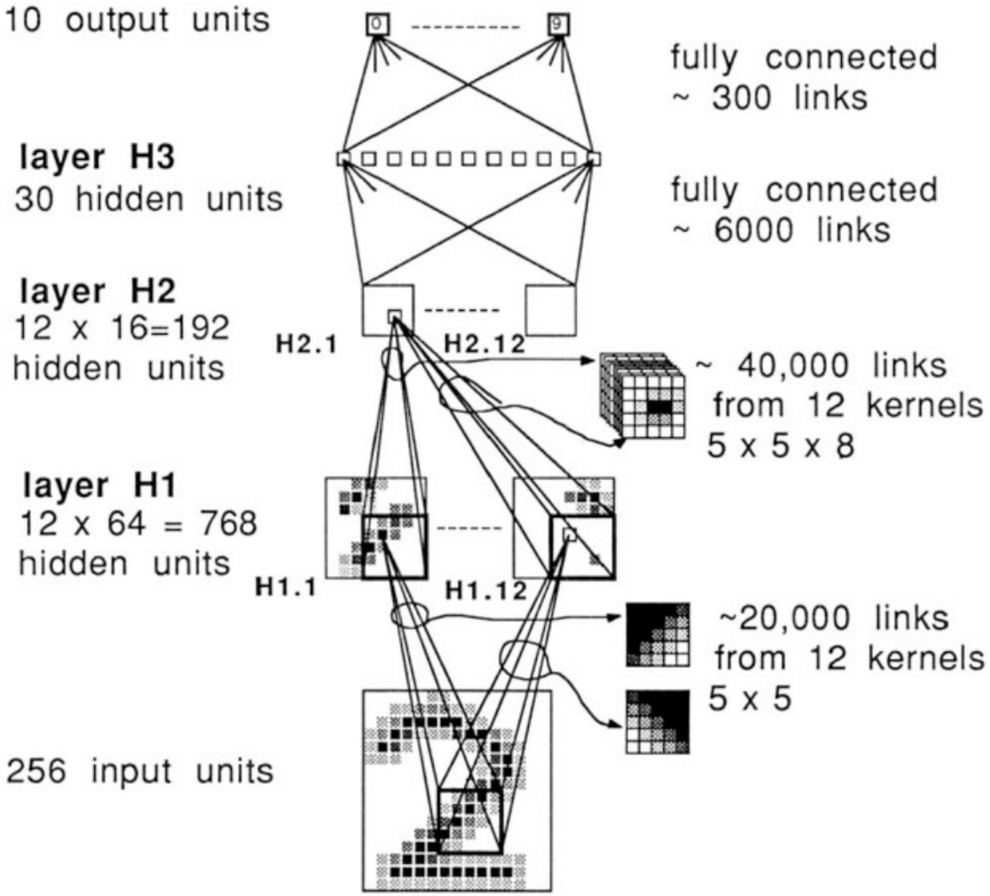




https://github.com/vdumoulin/conv_arithmetic

Convolutional Neural Networks (CNN)





Backpropagation Applied to Handwritten Zip Code Recognition

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel

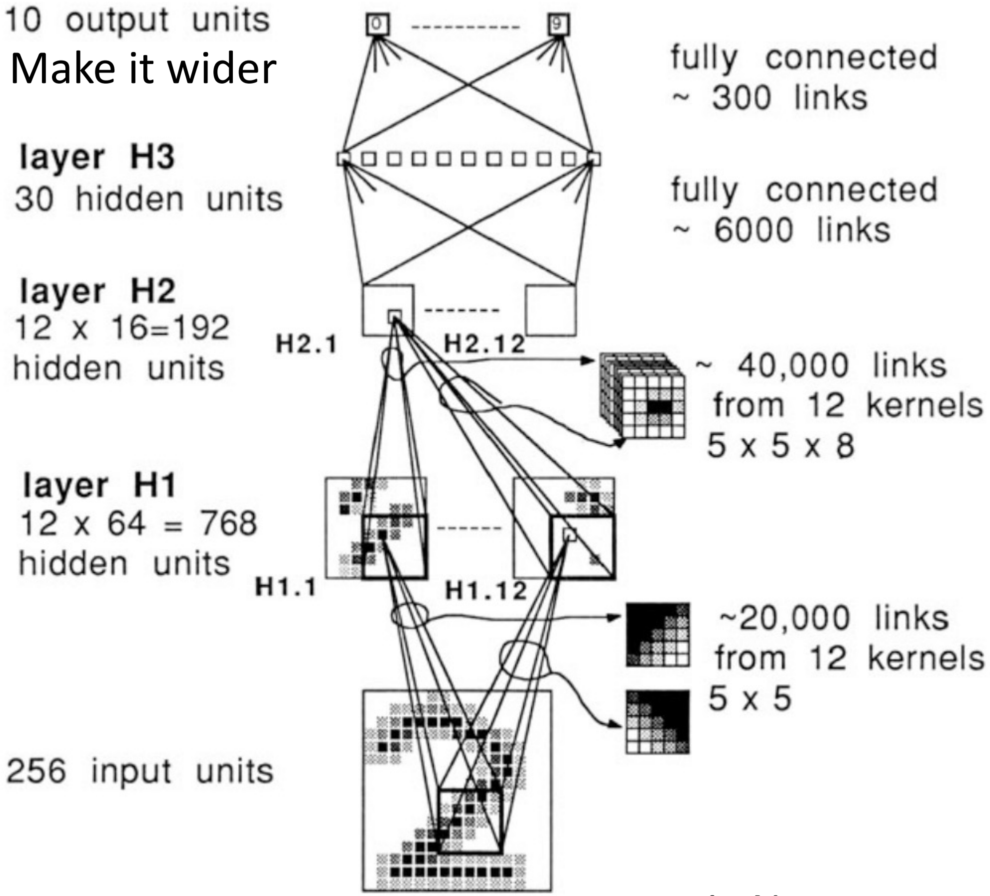
> Author and Article Information

Neural Computation (1989) 1 (4): 541–551.

https://doi.org/10.1162/neco.1989.1.4.541 **Article history**

1. <u>Parameter efficiency.</u> "Kernels" are reused at all locations. Parameters won't grow with input image dimension.
2. <u>Handles "shift-invariance" of images.</u> Any location containing "signal" will activate the kernel.

Convolutional Neural Networks (CNN)



LeNet

Backpropagation Applied to Handwritten Zip Code Recognition

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel

> Author and Article Information

Neural Computation (1989) 1 (4): 541–551.

https://doi.org/10.1162/neco.1989.1.4.541 **Article history**

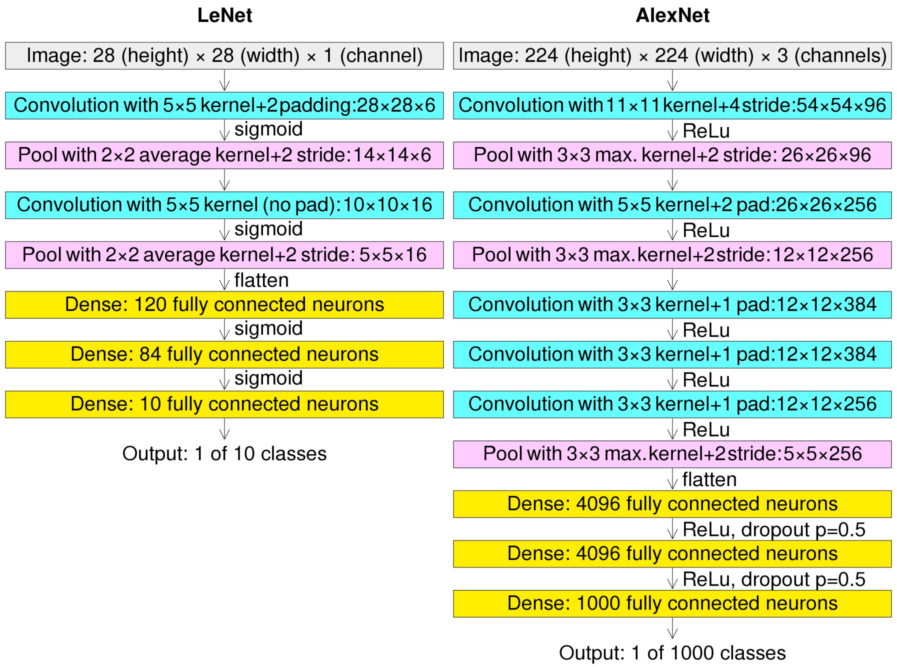
1. Parameter efficiency. “Kernels” are reused at all locations. Parameters won’t grow with input image dimension.
2. Handles “shift-invariance” of images. Any location containing “signal” will activate the kernel.

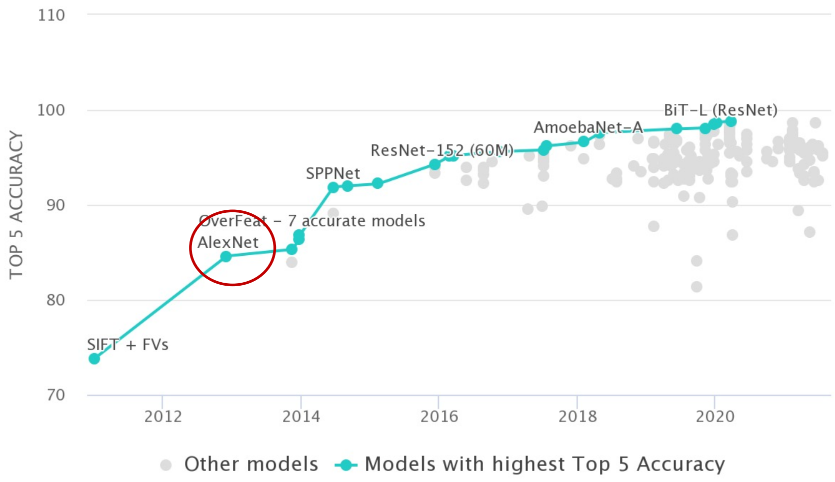
for. We used an off-the-shelf board that contains 256 kbytes of local memory and an AT&T DSP-32C general purpose DSP with a peak performance of 12.5 million multiply add operations per second on 32 bit floating point numbers (25 MFLOPS). The DSP operates as a coproces-

Big Data for benchmarking

AlexNet: implemented in CUDA and utilized GPU.

60K parameters **61M parameters**





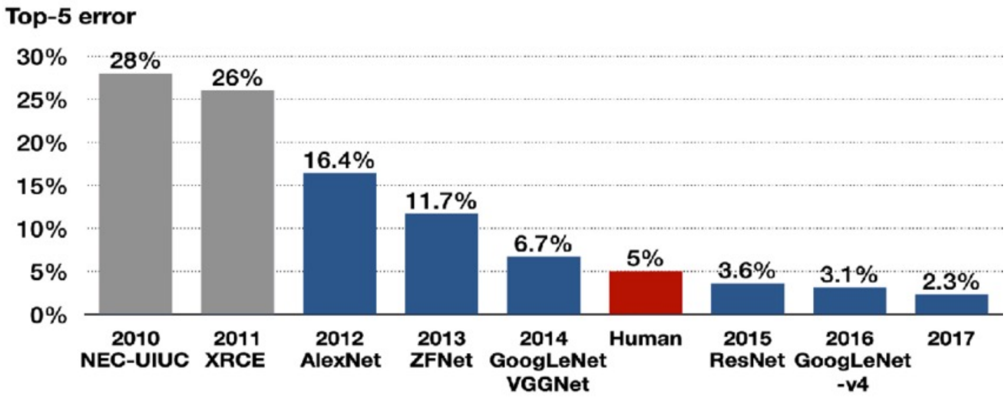


Image credit:
https://en.wikipedia.org/wiki/AlexNet#/media/File:Comparison_image_neural_networks.svg

Big Data for benchmarking

AlexNet: implemented in CUDA and utilized GPU.

Backpropagation Applied to Handwritten Zip Code Recognition

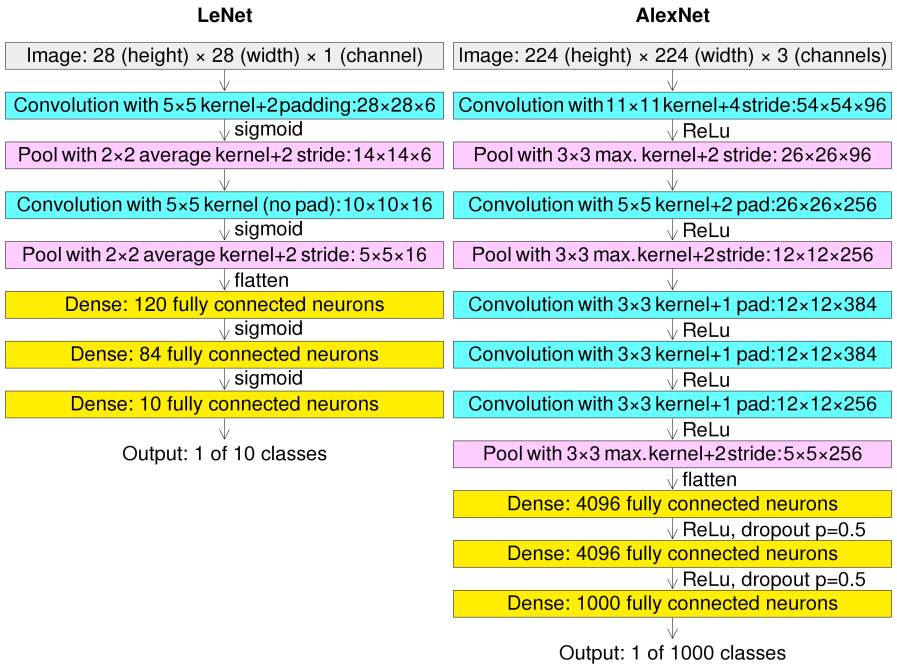
Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel

Author and Article Information

Neural Computation (1989) 1 (4): 541–551.

https://doi.org/10.1162/neco.1989.1.4.541 **Article history**

60K parameters **61M parameters**



for. We used an off-the-shelf board that contains 256 kbytes of local memory and an AT&T DSP-32C general purpose DSP with a peak performance of 12.5 million multiply add operations per second on 32 bit floating point numbers (25 MFLOPS). The DSP operates as a coproces-

"Our network takes between five and six days to train on two GTX 580 3GB GPUs"

GTX 580: 1.581 TFLOPS Each.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems* 25 (2012): 1097-1105.

Image credit: https://en.wikipedia.org/wiki/AlexNet#/media/File:Comparison_image_neural_networks.svg

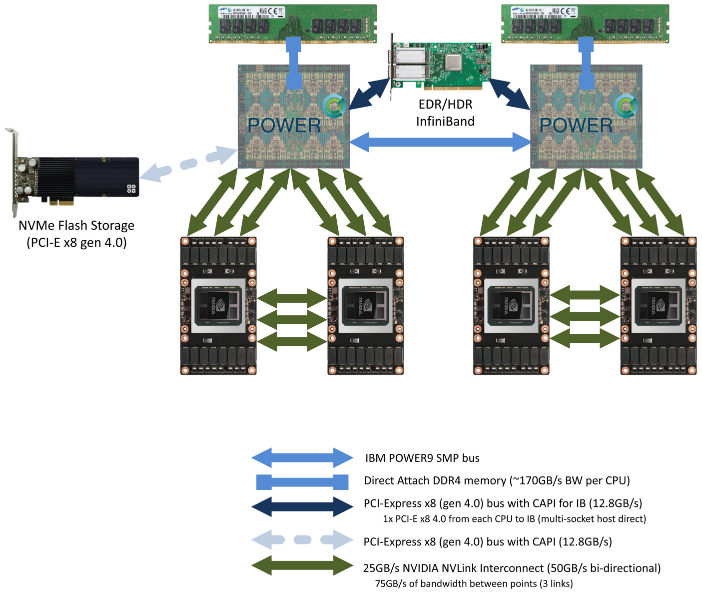
Most modern High-performance computing (HPC) are powered by GPUs.

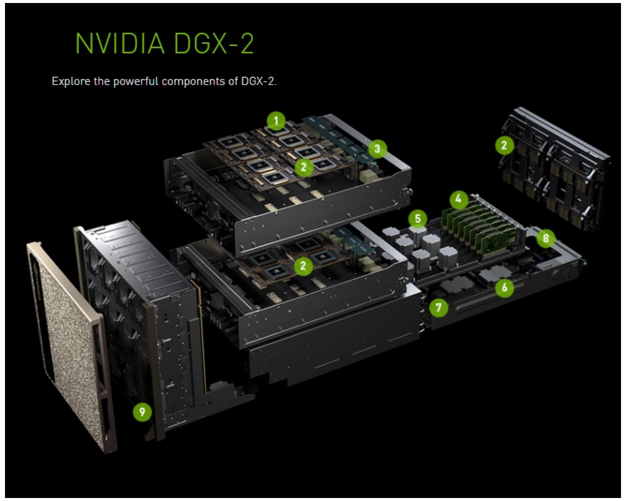
https://www.top500.org/

IBM AC922

Server Block Diagram

Power Systems AC922 with NVIDIA Tesla V100 with Enhanced NVLink GPUs





“the world’s first 2 petaFLOPS system integrating 16 NVIDIA V100 Tensor Core GPUs for large-scale AI projects” -- NVIDIA

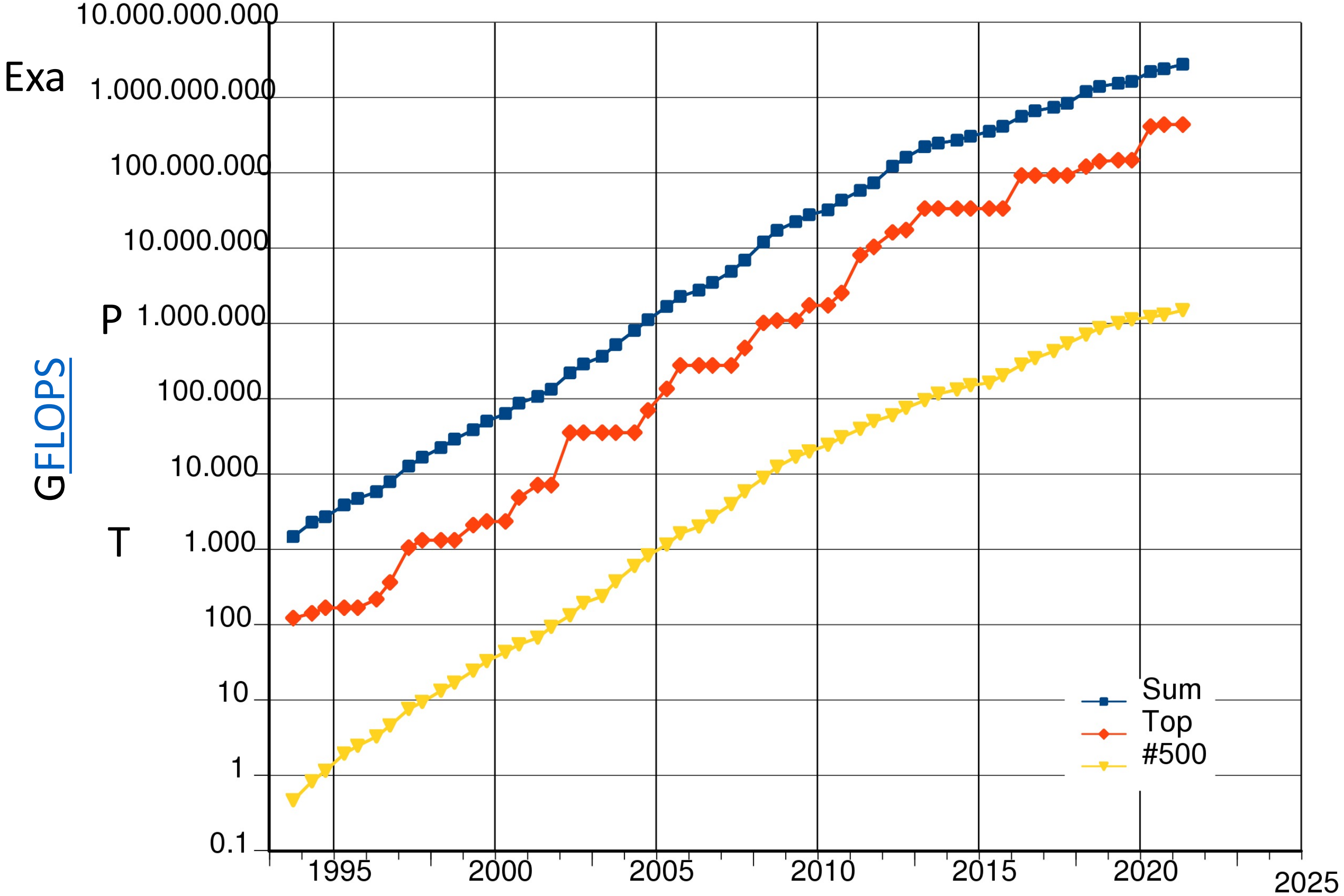
	Accelerator/Co-Processor	Count	System Share (%)
1	NVIDIA Tesla V100	80	16
2	NVIDIA A100	15	3
3	NVIDIA Tesla V100 SXM2	12	2.4
4	NVIDIA Tesla P100	8	1.6
5	NVIDIA A100 SXM4 40 GB	5	1
6	NVIDIA A100 40GB	4	0.8
7	NVIDIA Volta GV100	4	0.8
8	NVIDIA Tesla K40	3	0.6
9	NVIDIA A100 80GB	2	0.4
10	Matrix-2000	1	0.2
11	NVIDIA 2050	1	0.2
12	NVIDIA Tesla K40m	1	0.2
13	NVIDIA Tesla K40/Intel Xeon Phi 7120P	1	0.2
14	NVIDIA Tesla P100 NVLink	1	0.2
15	Preferred Networks MN-Core	1	0.2
16	Nvidia Volta V100	1	0.2
17	NVIDIA Tesla K80	1	0.2
18	Intel Xeon Phi 31S1P	1	0.2
19	Deep Computing Processor	1	0.2
20	AMD Vega 20	1	0.2
21	Intel Xeon Phi 5110P	1	0.2
22	NVIDIA Tesla K20x	1	0.2

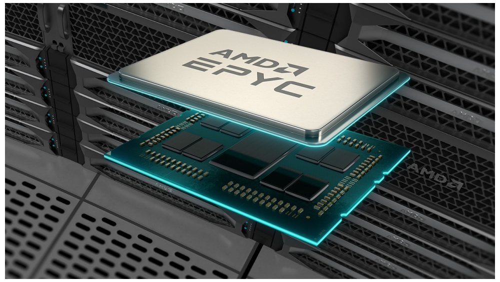
For example, IBM AC922 has 4 (or 6 GPUs) per node.
NVIDIA DGX2 has 16 GPUs per node, and DGX1 has 8 GPUs per node.

“Performance Analysis of Deep Learning Workloads on Leading-edge Systems” (2019)
https://ieeexplore.ieee.org/document/9059262

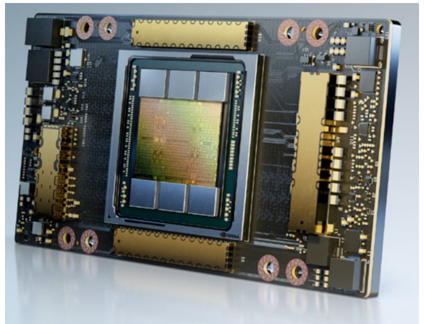
Comparing most advanced CPU and GPU

supercomputer performance





AMD EPYC 7763 (64-cores)
FP32: 3.58 TFLOPS



NVIDIA A100 GPU (6912 CUDA cores)
FP32: 19.5 TFLOPS
Tensor Float 32 (TF32): 156 TFLOPS

Image credit:
https://en.wikipedia.org/wiki/TOP500#/media/File:Supercomputers-history.svg
https://www.amd.com/en/products/cpu/amd-epyc-7763
https://www.nvidia.com/en-us/data-center/a100/

However…

- Coding GPU in C/C++ with CUDA is difficult.
- Manually code forward and backward passes, and differentiation using chain rule is tedious.

- Coding GPU used to be difficult (C/C++ with CUDA).
- Manually code forward and backward pass, and differentiation using chain rule is tedious.

Modern Deep Learning libraries made these very easy!
Mostly Python-based: PyTorch, TensorFlow, JAX and so on.

```python
class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()
        self.conv1 = nn.Conv2d(3, 6, 5)
        self.pool = nn.MaxPool2d(2, 2)
        self.conv2 = nn.Conv2d(6, 16, 5)
        self.fc1 = nn.Linear(16 * 5 * 5, 120)
        self.fc2 = nn.Linear(120, 84)
        self.fc3 = nn.Linear(84, 10)

    def forward(self, x):
        x = self.pool(F.relu(self.conv1(x)))
        x = self.pool(F.relu(self.conv2(x)))
        x = x.view(-1, 16 * 5 * 5)
        x = F.relu(self.fc1(x))
        x = F.relu(self.fc2(x))
        x = self.fc3(x)
        return x

net = Net()
print(net)
```

Algorithm

1. Initialize weights randomly $\sim\mathcal{N}(0, \sigma^2)$
2. Loop until convergence:
3. Pick batch of B data points
4. Compute gradient, $\frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}} = \frac{1}{B}\sum_{k=1}^{B} \frac{\partial J_k(\boldsymbol{W})}{\partial \boldsymbol{W}}$
5. Update weights, $\boldsymbol{W} \leftarrow \boldsymbol{W} - \eta \frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}}$
6. Return weights


```
# Initialize the loss function
loss_fn = nn.CrossEntropyLoss()
```

#Initialize optimizer

```
optimizer = torch.optim.SGD(model.parameters(), lr=learning_rate)
```

```
for batch, (X, y) in enumerate(dataloader):
    # Compute prediction and loss
    pred = model(X)
    loss = loss_fn(pred, y)

    # Backpropagation
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
```

To move model or data to GPU, simply
`model = model.cuda()`

https://pytorch.org/tutorials/beginner/basics/optimization_tutorial.html

- Coding GPU used to be difficult (C/C++ with CUDA).
- Manually code forward and backward pass, and differentiation using chain rule is tedious.

Modern Deep Learning libraries made these very easy!
Mostly Python-based: PyTorch, TensorFlow, JAX and so on.

🤗 **Hugging Face**

https://huggingface.co/

Some higher-level deep learning libraries can get an inference job done in three lines of code!

```
>>> from transformers import pipeline
>>> classifier = pipeline('sentiment-analysis')
```

```
>>> classifier('We are very happy to show you the 🤗 Transformers library.')
[{'label': 'POSITIVE', 'score': 0.9997795224189758}]
```

https://huggingface.co/transformers/quicktour.html

- Coding GPU used to be difficult (C/C++ with CUDA).
- Manually code forward and backward pass, and differentiation using chain rule is tedious.

Modern Deep Learning libraries made these very easy!
Mostly Python-based: PyTorch, TensorFlow, JAX and so on.

🤗 **Hugging Face**

https://huggingface.co/

Some higher-level deep learning libraries can get an inference job done in three lines of code!

```
>>> from transformers import pipeline
>>> classifier = pipeline('sentiment-analysis')
```

```
>>> classifier('We are very happy to show you the 🤗 Transformers library.')
[{'label': 'POSITIVE', 'score': 0.9997795224189758}]
```

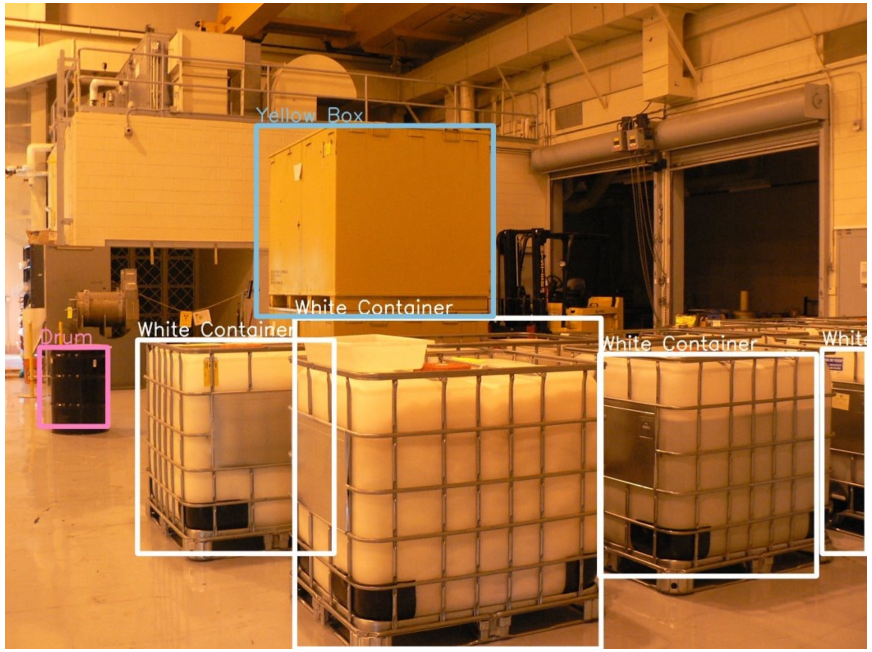
It's the best time to learn and practice deep learning!

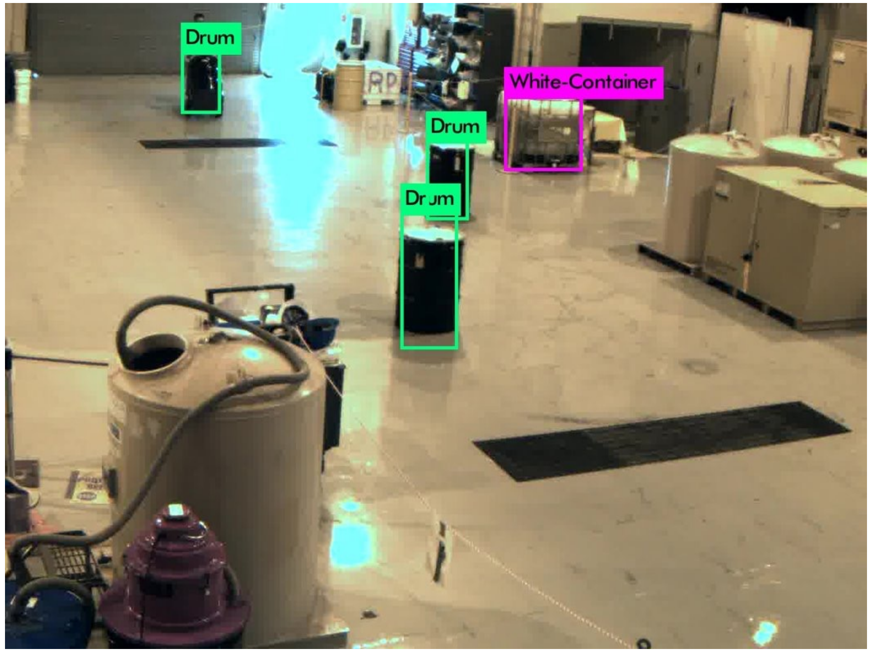
https://huggingface.co/transformers/quicktour.html

- Memorable Moments
- What is Deep Learning
- Some examples
  - Object Detection for NNSA / IAEA
  - Neural Fingerprints for COVID drug screening
  - Bayesian Neural Nets for climate prediction
  - Auto-encoder for sPHENIX Data Compression
  - Bridge the gap between simulation and experiments, GAN!
  - Software hardware Co-design: deploying AI at edge.

Supervised Deep Learning Software for Surveillance Cameras

- Use computer vision (deep learning) algorithms to assist IAEA inspectors reviewing surveillance footage more efficiently.





Jihwan Park, Yuewei Lin, Shinjae Yoo, Yonggang Cui

2019 Joule Awards by NA-241 SG Tech

National Nuclear Security Administration (NNSA)

Thank you!

- Please Contact me if you are interested in AI/ML SULI projects!
- yren@bnl.gov