Carnegie Mellon University

18-649 Guest Lecture:

Machine Learning

Tianshu Huang

(1) When should I use ML / DL?
(2) How do I develop and deploy ML to the edge?

This lecture brought to you by:

RadarML

Interested in machine learning research and working with **real systems**, not just canned datasets?

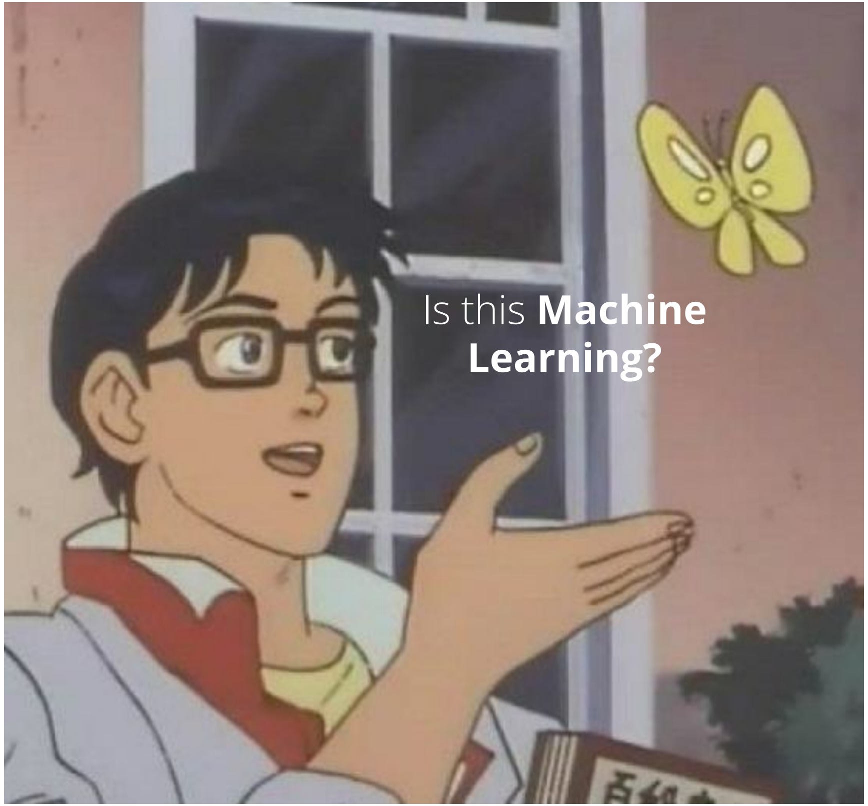
- **Embedded HW/SW** development for learning-enabled radar applications
- **Machine learning** for radar and radar+X fusion

Contact: tianshu2@andrew.cmu.edu

“AI”

Machine Learning

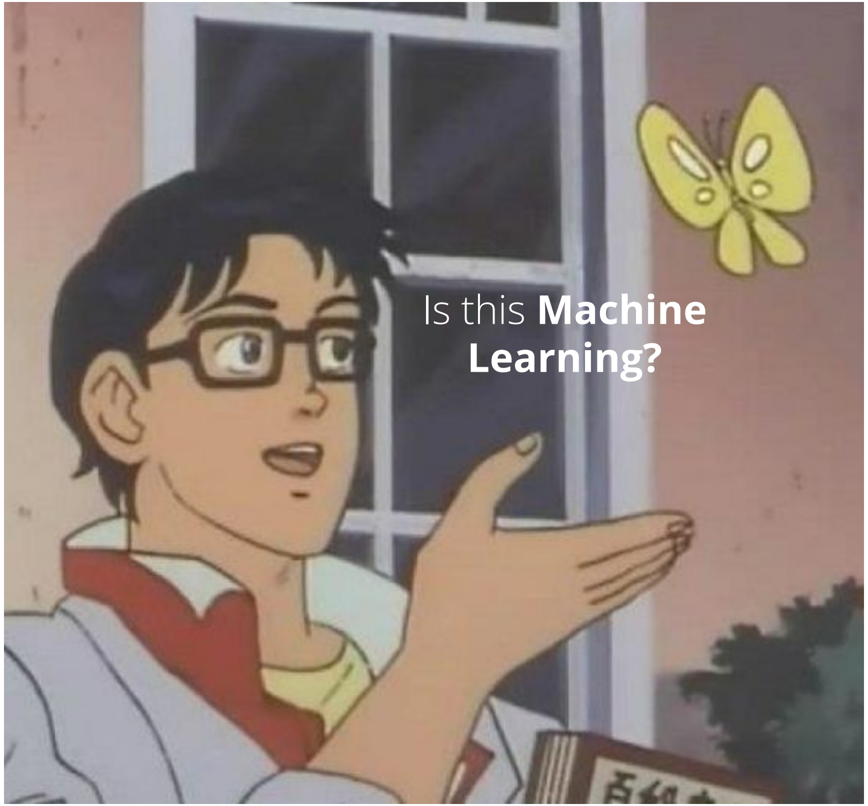
- $$$
- model fitting



Machine Learning

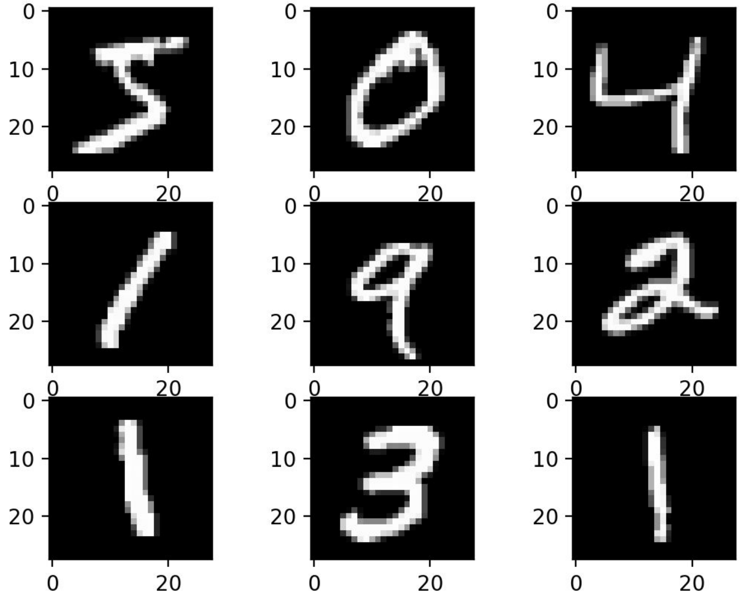
Empirical risk minimization:

1. Data
2. Risk function (e.g. accuracy, loss)
3. What model minimizes the risk function on the data?



Machine Learning

Example: digit classification



No data: manually specified features

Some data: Compare with data

manually extract features

→ model

A lot of data → deep learning

Machine Learning

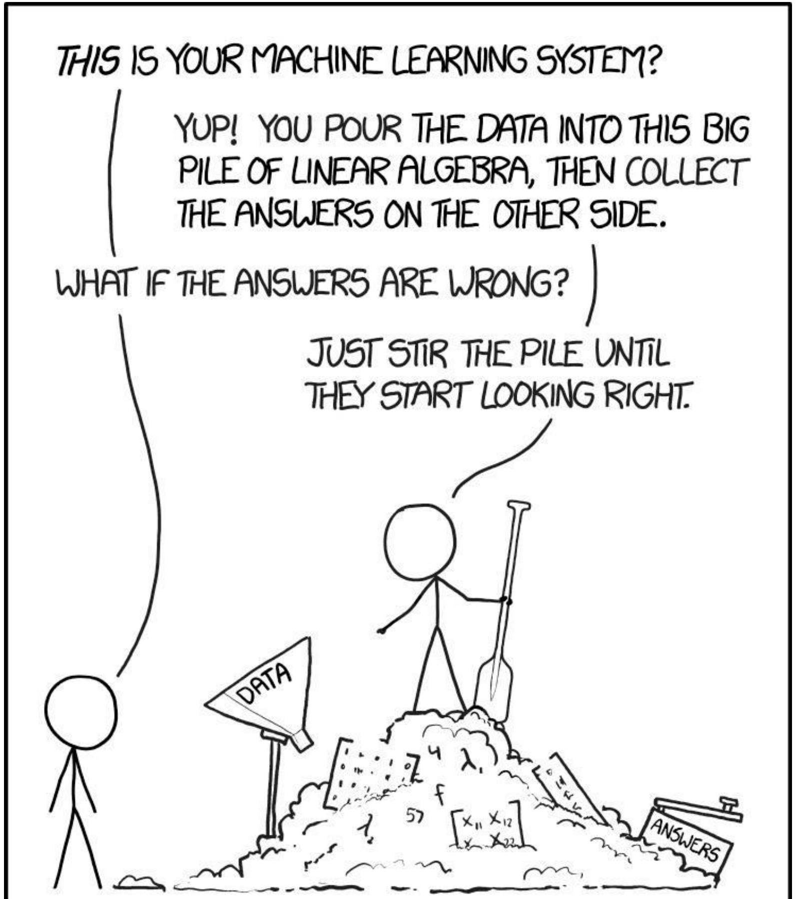
random forest

k-nearest neighbors

linear regression & variants

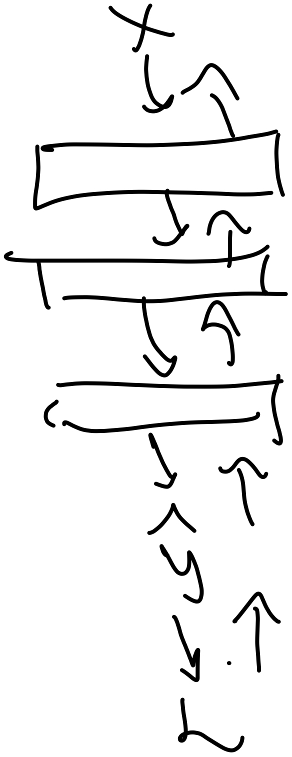
clustering, other unsupervised

deep learning



https://xkcd.com/1838/

Deep Learning

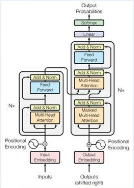


STOP DOING DEEP LEARNING

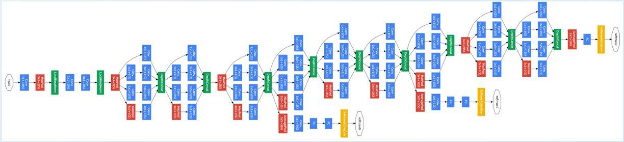
- PERCEPTRONS WERE ONLY EVER MEANT TO BE FULLY CONNECTED
- THOUSANDS OF PAPERS yet NO REAL-WORLD USE FOUND for going deeper than ONE LAYER
- Wanted to add more nonlinearity anyway for a laugh? We had a tool for that: It was called "KERNEL METHODS"
- "Yes please give me a network that can PAY ATTENTION TO ITSELF. Please give me PRETRAINED WEIGHTS for my YOLO-9000" - Statements dreamed up by the utterly Deranged

LOOK at what "Research Scientists" have been demanding your Respect for all this time, with the statistical methods & optimization algorithms we built for them

(This is REAL "Deep Learning", done by REAL "ML Engineers"):



?????



????????????????????

"Hello I would like to learn **1.6 TRILLION** parameters please"

They have played us for absolute fools

https://www.reddit.com/r/okbuddyphd/comments/n2m6vz/stop_doing_deep_learning/

Deep Learning

The Higher Mysteries

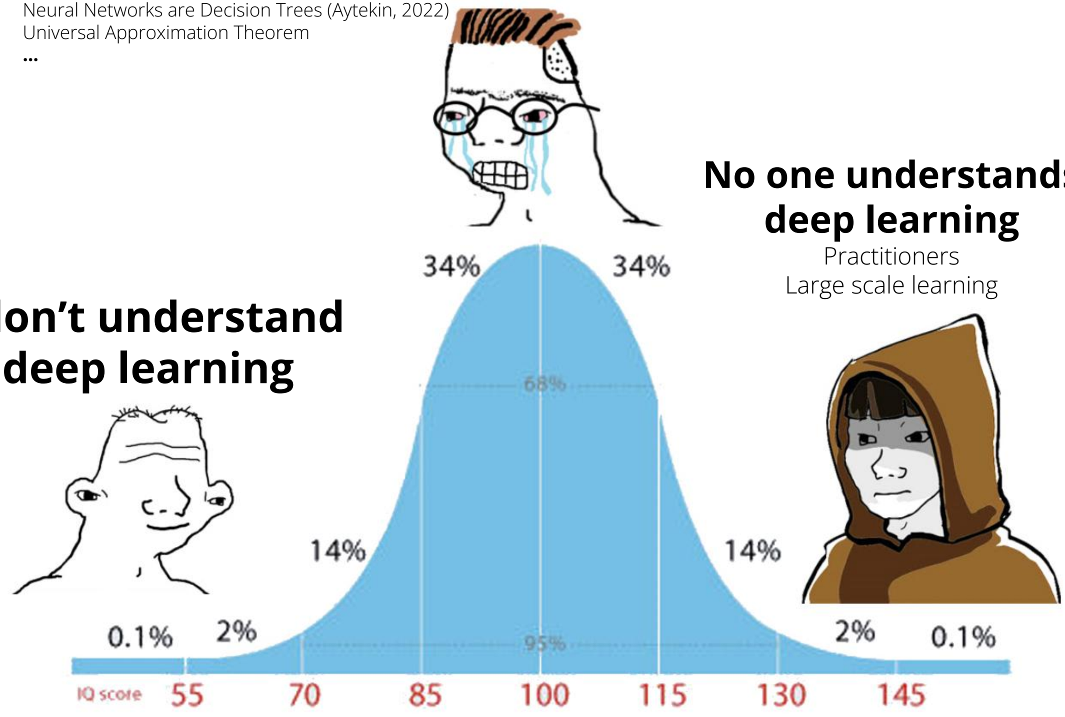
NOoO!! You have to understand Deep learning!!!

Merging Models modulo Permutation Symmetries (Ainsworth et al., 2022)
Neural Networks are Decision Trees (Aytekin, 2022)
Universal Approximation Theorem
...

I don’t understand deep learning

No one understands deep learning

Practitioners
Large scale learning



The Natural Image Manifold

The **natural image manifold** is a surface embedded in a **high dimensional space** that is:

(1) **low-dimensional**,
(2) **highly nonlinear**, and
(3) **locally smooth**.

100 x 100 pt greyscale

- 10,000 dims

- random image → noise

⟹ natural images are low dimensional

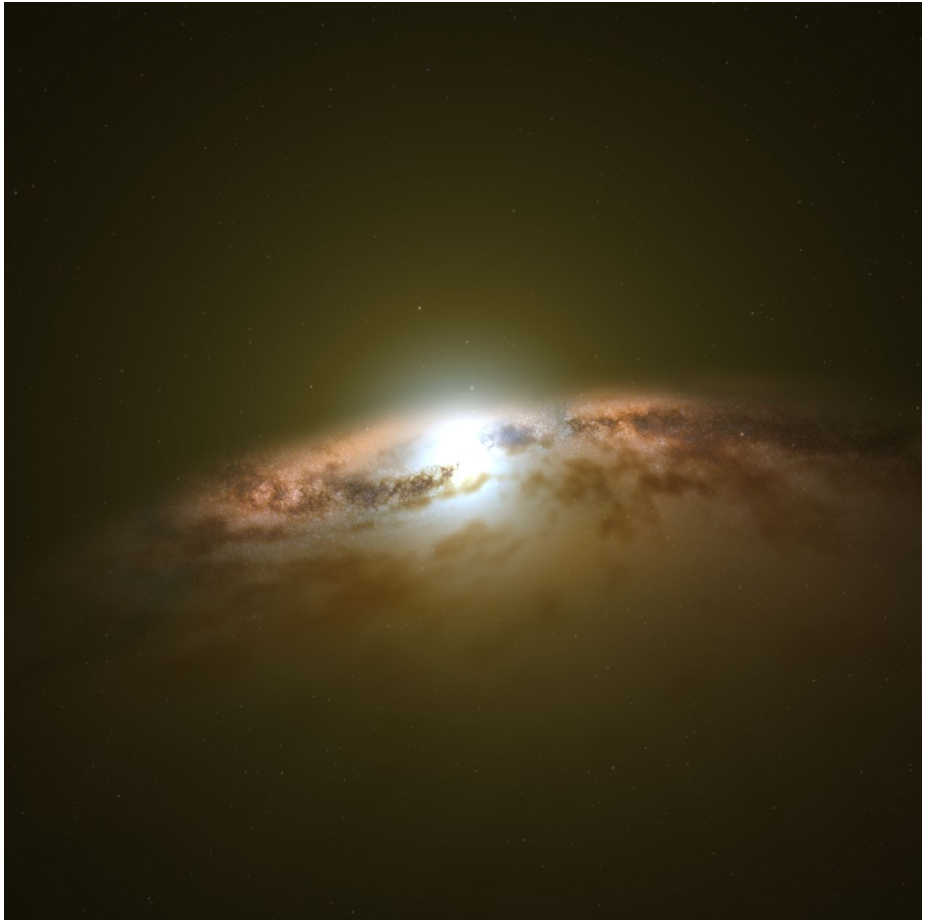
- image 1 + image 2 ≠ image ⟹ nonlinear

- image + noise = image ⟹ smooth

The Natural Image Manifold

The **natural image manifold** is a surface embedded in a **high dimensional space** that is:

(1) **low-dimensional**,
(2) **highly nonlinear**, and
(3) **locally smooth**.



The Natural Image Manifold

The **natural image manifold** is a surface embedded in a **high dimensional space** that is:

(1) **low-dimensional**,
(2) **highly nonlinear**, and
(3) **locally smooth**.

- images, video
- text
- audio

- health data* (except continuous monitoring)
- few sensors, few observations

object detection

power analysis → "compounded' time-series, maybe DL

→ not DL

Practical Deep Learning

See **Adversarial Robustness**.
Image from “Robust Physical-World Attacks on Deep Learning Models,” Eykholt et al, 2018.

Pre-Trained Models & Transfer Learning

[https://www.reddit.com/r/machinelearningmemes/comments/g1mxz8/transfer_learning_101/]

Practical Deep Learning

- Train from scratch?
- Transfer learn?
- Fully pre-trained model?

Deep Learning on the Edge

Please inspect your power cables if you own a RTX 4090, even if you don't use it for deep learning.

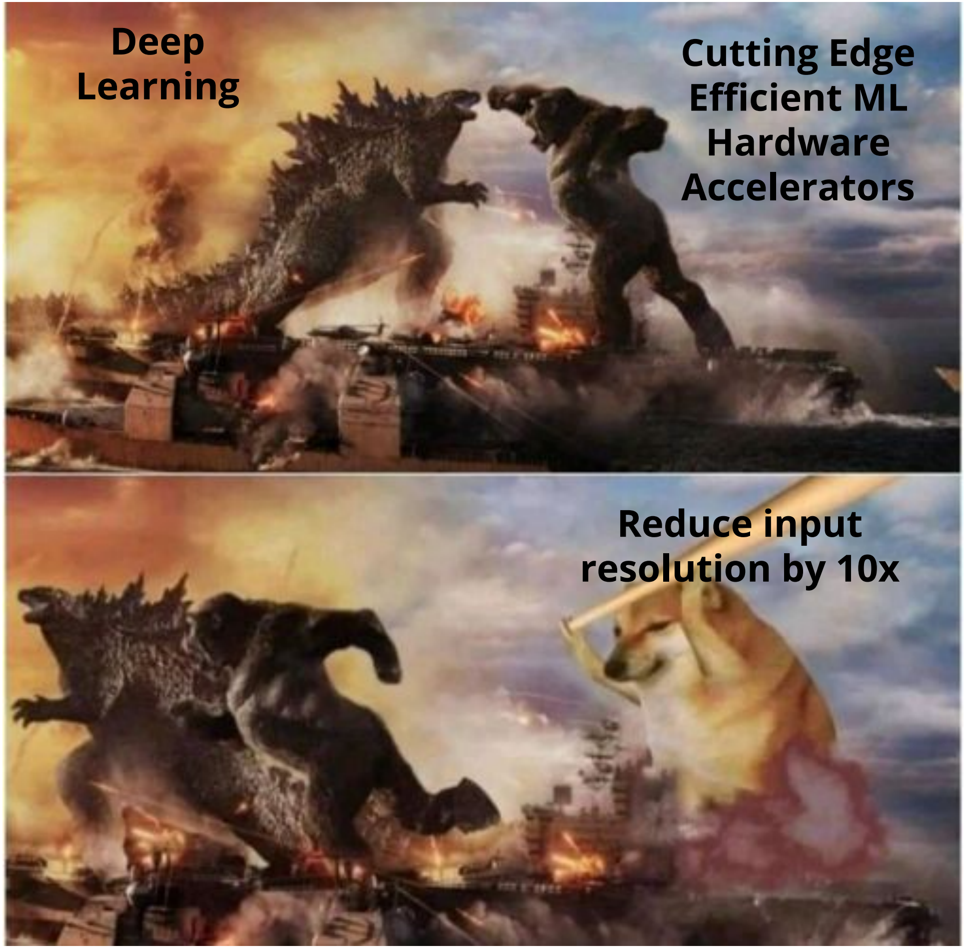
Deep Learning on the Edge

- Trained a neural network with a given architecture
- Weights are 32-bit float
- Fixed power budget

- offloading
- Specialized HW
- Shrink the model somehow (quantization)
- adaptive inference

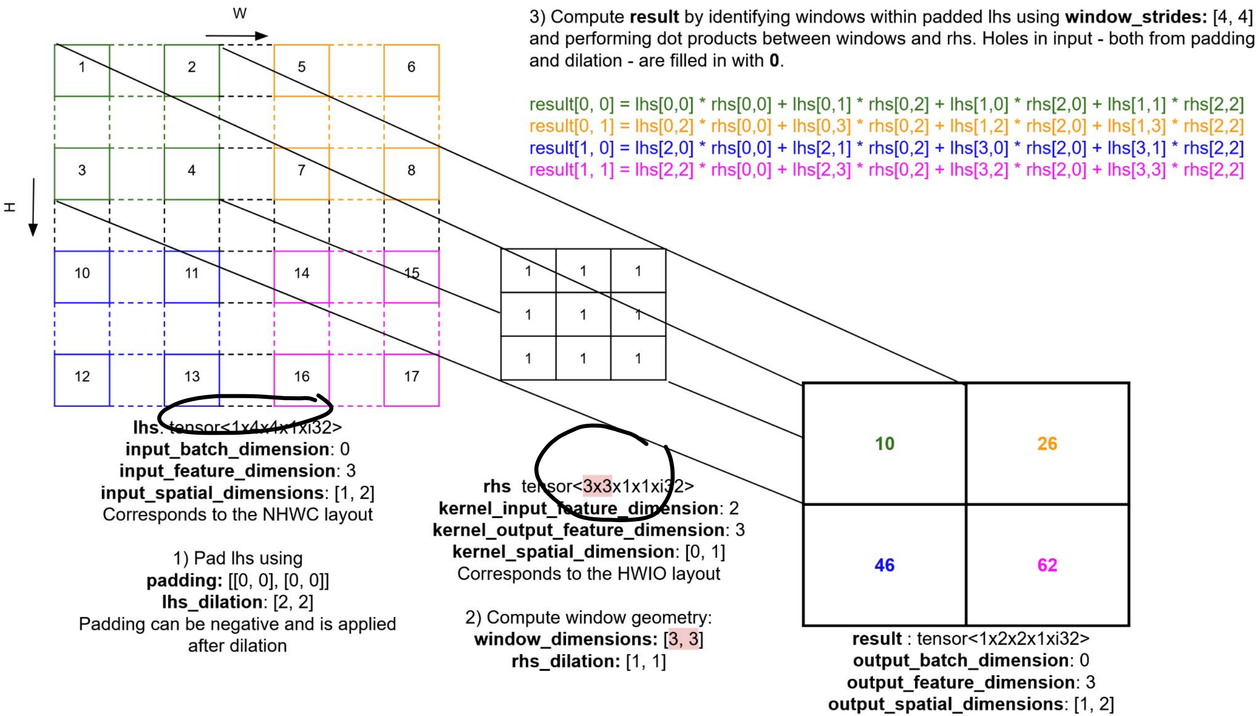
Simplest is Best

- Cloud offloading
- Reduce input resolution
- Don’t use deep learning



Use better software

- Compilers, e.g. XLA
- Custom CUDA / accelerator code



① a = b + c

② e = a + d

- Pipeline stall
- cache

“ImageNet Classification with Deep Convolutional Neural Networks,” Krizhenvsky et al., 2012 (AlexNet - First “true” ConvNet)

Quantization

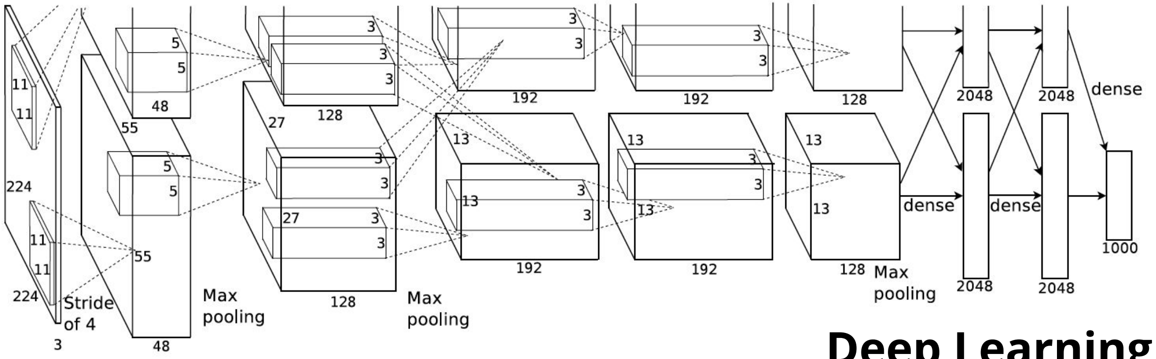
doable

- float32 → float16
- even better: int8
- even even better: int4
- binary, ternary is even possible!

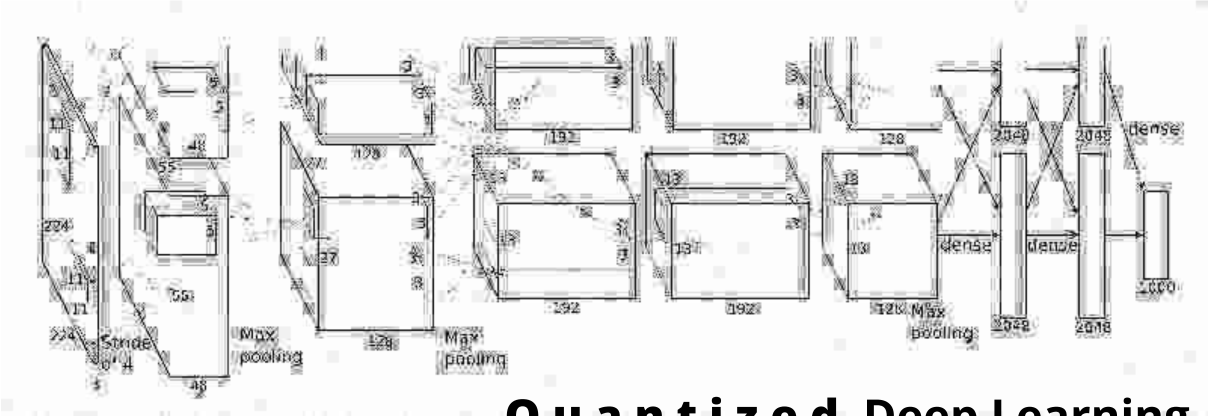
not doable

binary $\begin{cases} 0 & 1 \\ -1 & 1 \end{cases}$

ternary $-1 \quad 0 \quad 1$



Deep Learning



Quantized Deep Learning

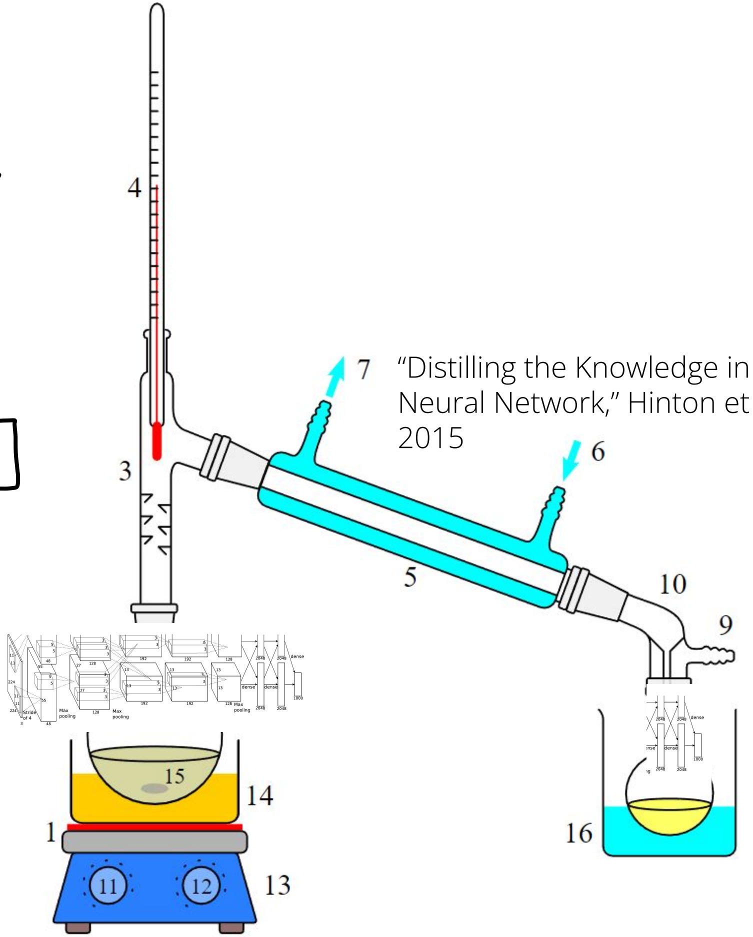
Distillation

$(x, y^*) \rightarrow$ model

GPT-4 / ChatGPT

$x, y^* \rightarrow$ model $\rightarrow \hat{y} \rightarrow$ model

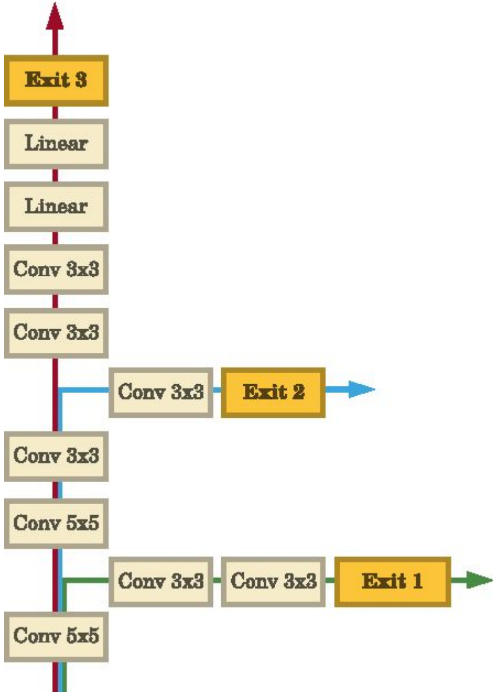
$\tilde{x}$



“Distilling the Knowledge in a Neural Network,” Hinton et al. 2015

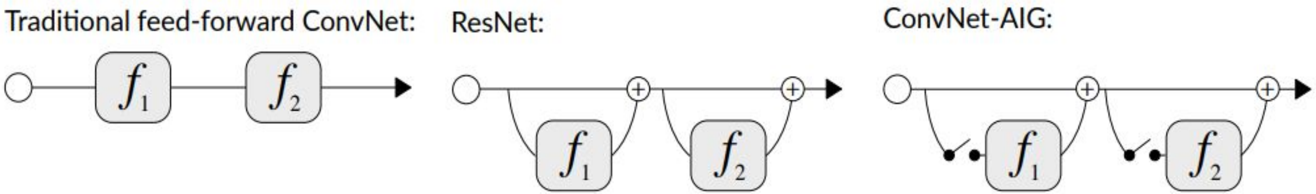
Adaptive Inference

"BranchyNet: Fast inference via early exiting from deep neural networks," Teerapittayanon, 2017



- Different numbers of layers
- Different input resolution
- ...

"Convolutional Networks with Adaptive Inference Graphs," Veit & Belongie, 2018



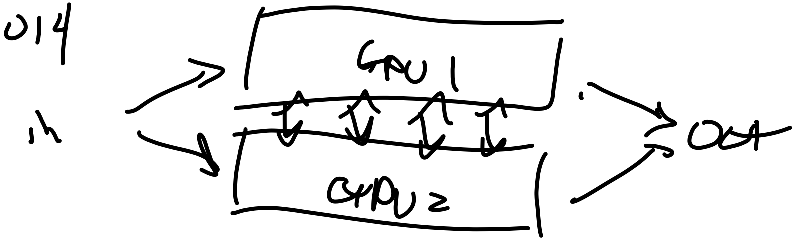
TL;DR

1. Don’t use machine learning
2. Don’t use deep learning
3. Don’t use deep learning on the edge (do it in the cloud)
4. Don’t train deep learning models (use a pre-trained model optimized for edge deployment)
5. Don’t train deep learning from scratch (use transfer learning)
6. Give up (or take a machine learning class)

Q&A

- Distributed/federated training
- Edge training
- GPU vs CPU vs TPU
- Deep learning frameworks
- ML accelerators
- ML Compilers
- Efficient Architectures
- Anything ML related

Alexnet, 2014



MoE