



NETWORK COMPRESSION

Smaller Model

Less parameters



Deploying ML models in resource-constrained environments



Lower latency, Privacy, etc.



Outline

- Network Pruning
- Knowledge Distillation
- Parameter Quantization
- Architecture Design
- Dynamic Computation

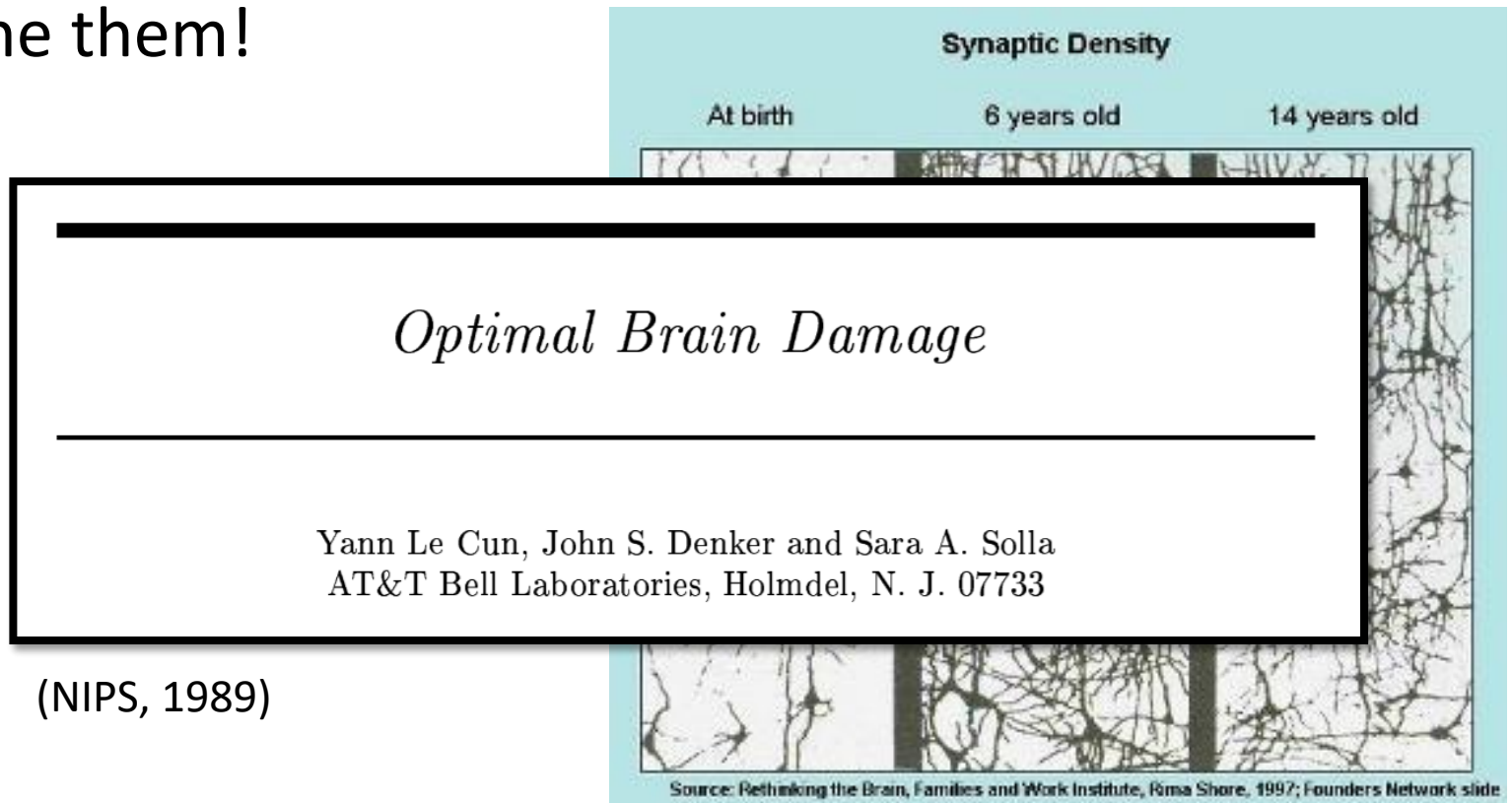
We will not talk about hard-ware solution today.

Network Pruning



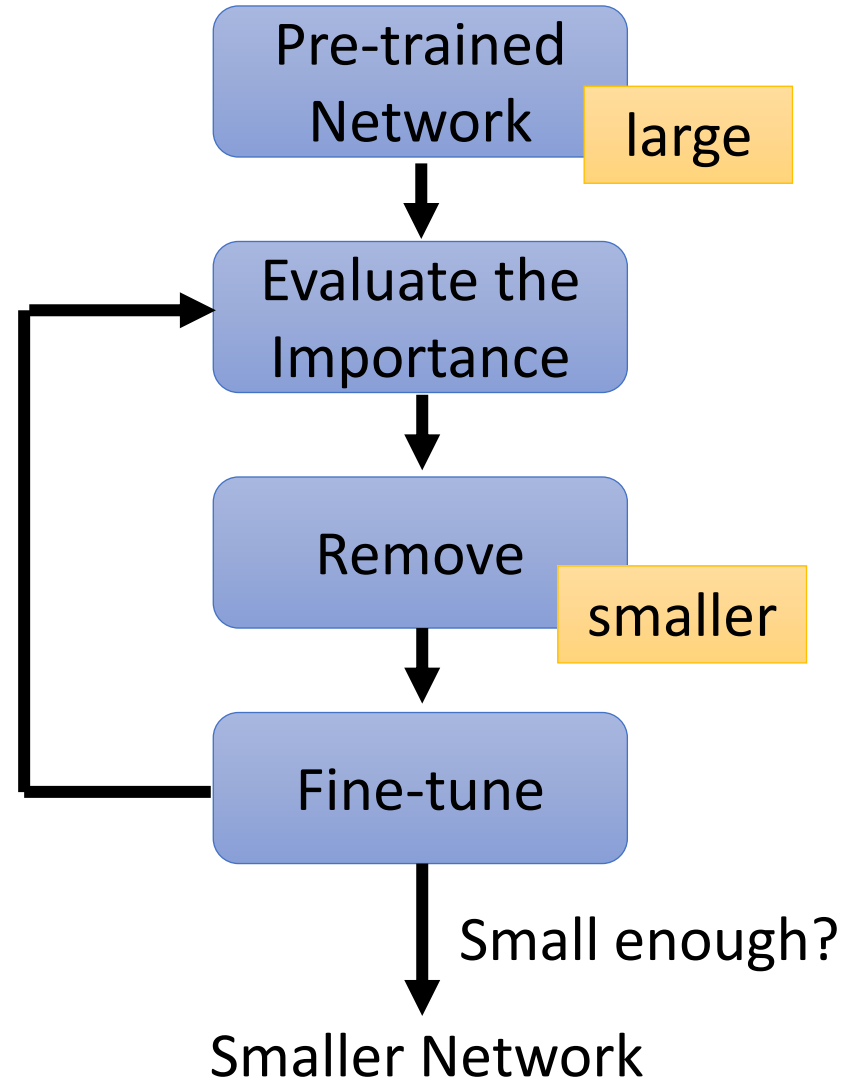
Network can be pruned

- Networks are typically over-parameterized (there is significant redundant weights or neurons)
- Prune them!



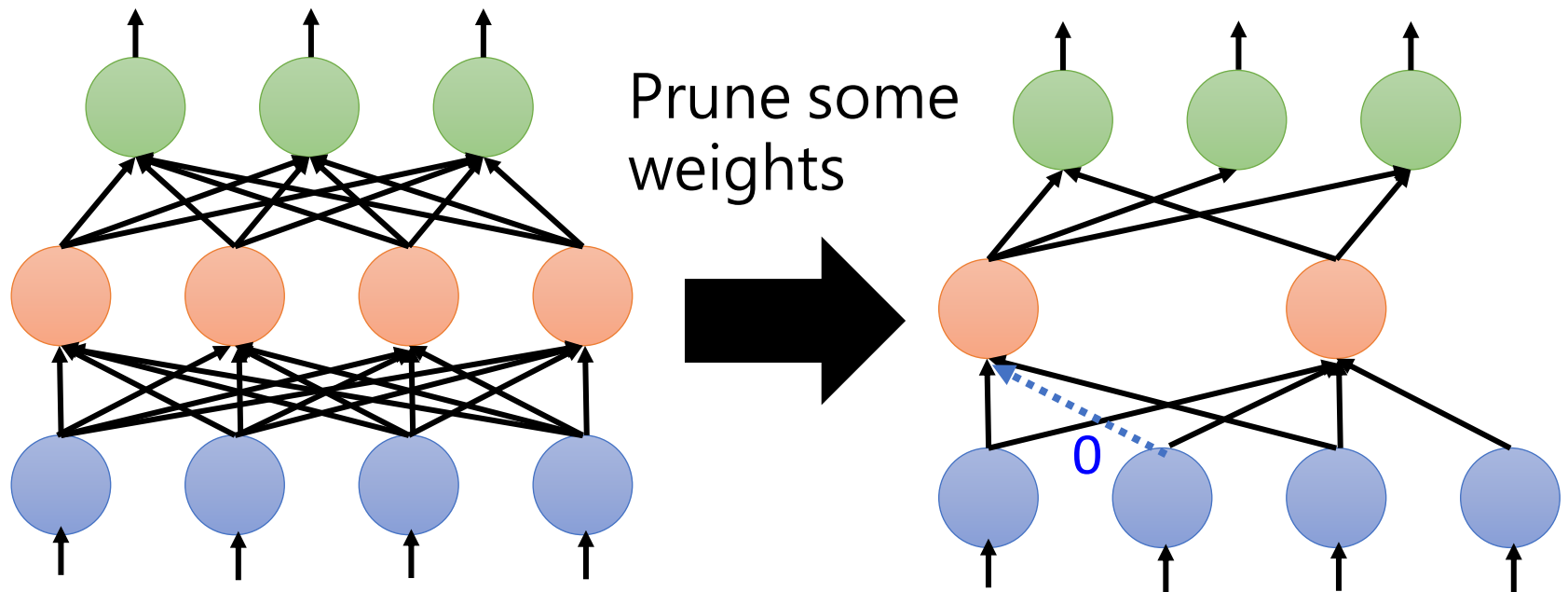
Network Pruning

- Importance of a weight:
absolute values, life long ...
- Importance of a neuron:
the number of times it wasn't zero on a given data set
- After pruning, the accuracy will drop (hopefully not too much)
- Fine-tuning on training data for recover
- Don't prune too much at once, or the network won't recover.



Network Pruning - Practical Issue

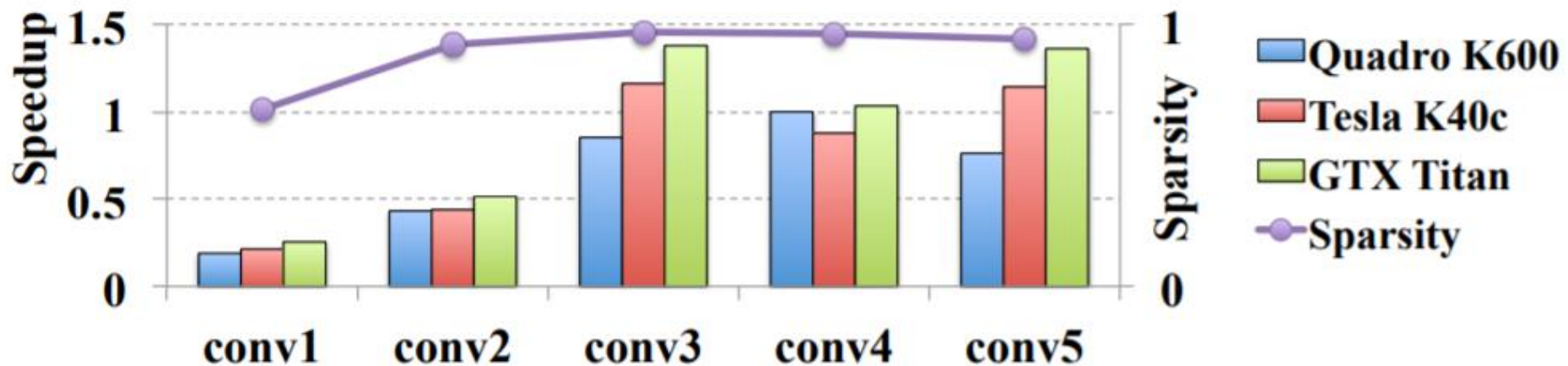
- Weight pruning



Hard to implement, hard to speedup

Network Pruning - Practical Issue

- Weight pruning

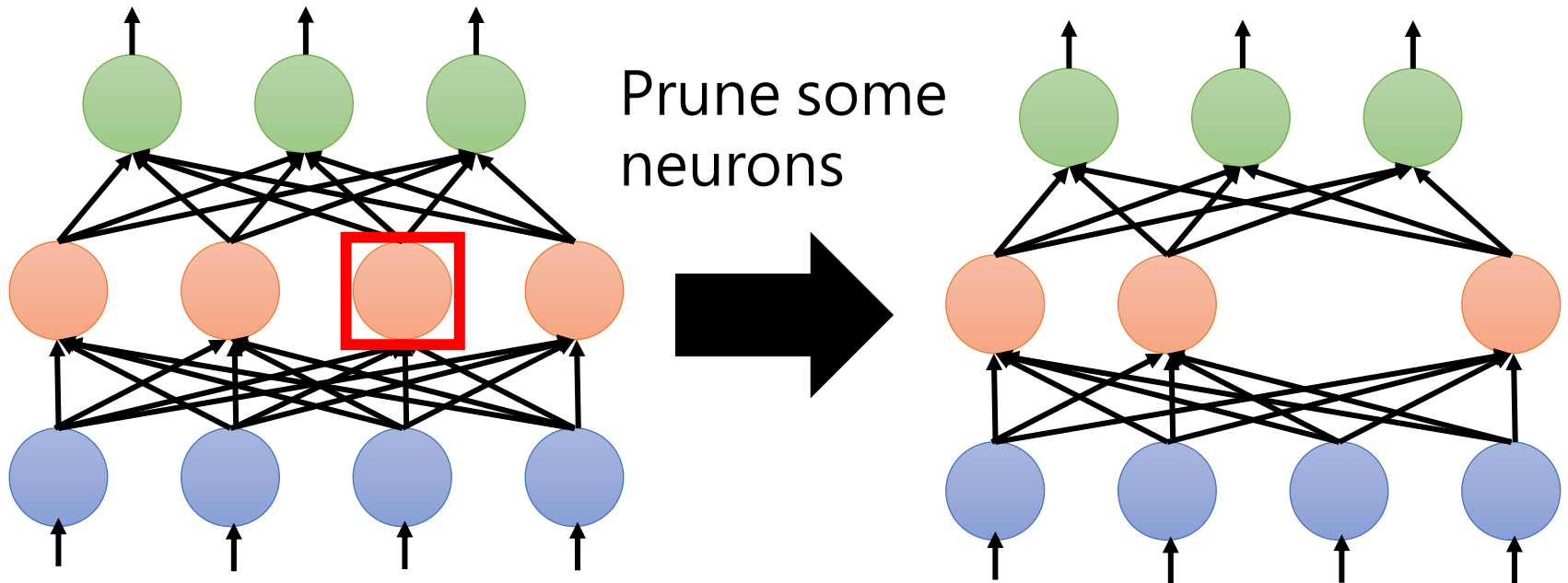


<https://arxiv.org/pdf/1608.03665.pdf>

Network Pruning - Practical Issue

- Neuron pruning

The network architecture is regular.



Easy to implement, easy to speedup

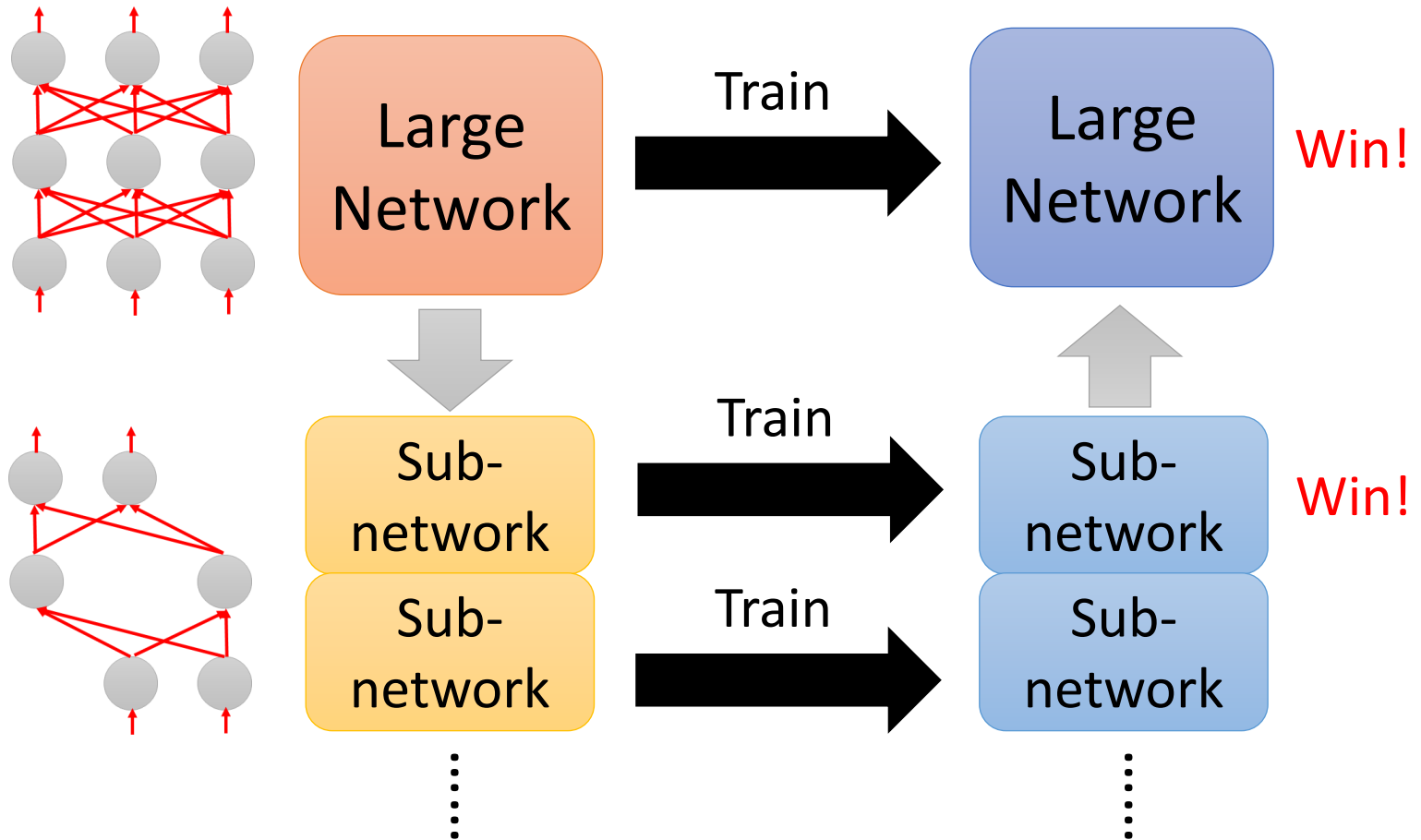
Why Pruning?

- How about simply train a smaller network?
- It is widely known that smaller network is more difficult to learn successfully.
 - Larger network is easier to optimize?
https://www.youtube.com/watch?v=_VuWvQUMQVk
- Lottery Ticket Hypothesis
<https://arxiv.org/abs/1803.03635>



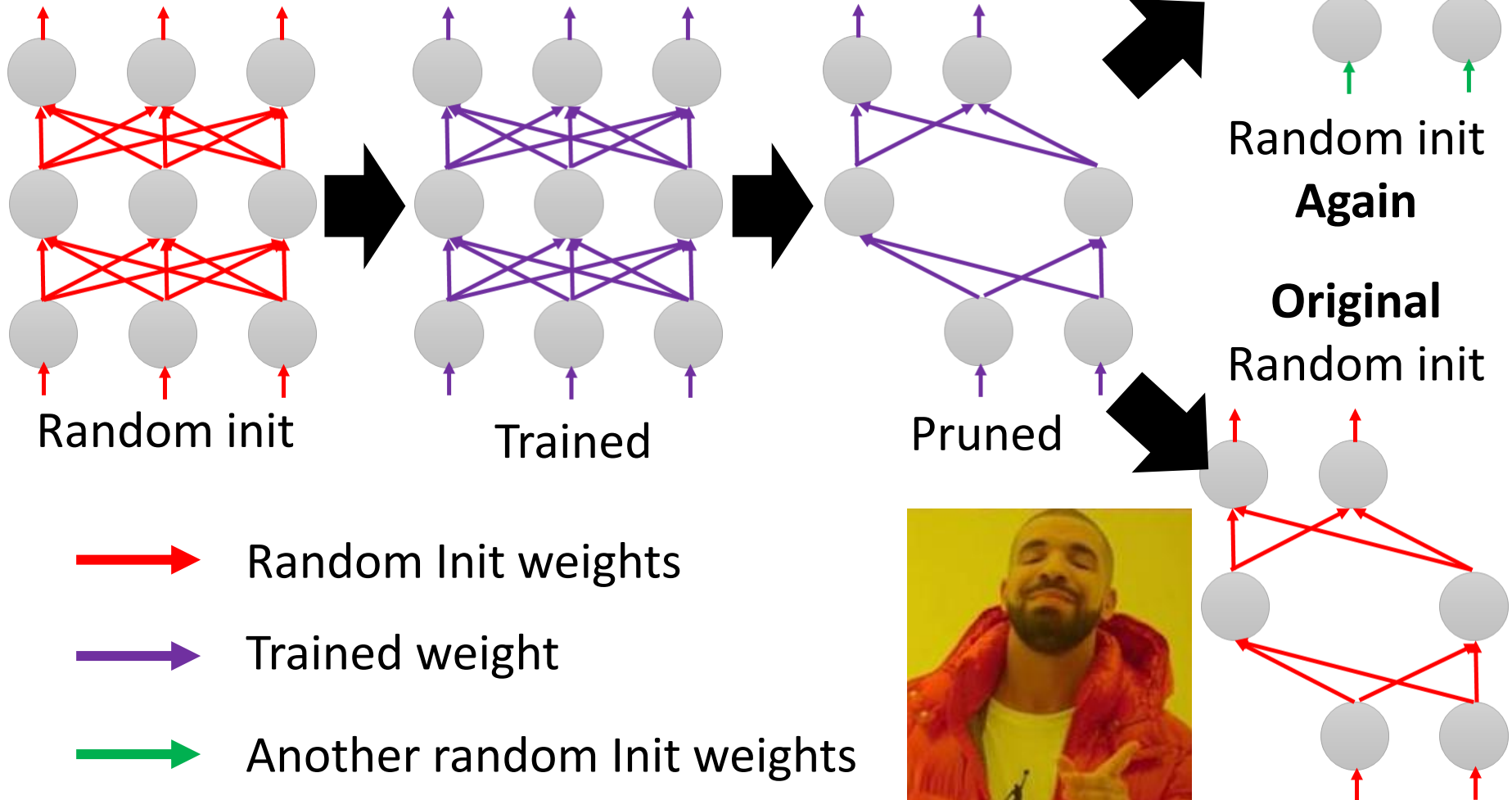
Why Pruning?

Lottery Ticket Hypothesis



Why Pruning?

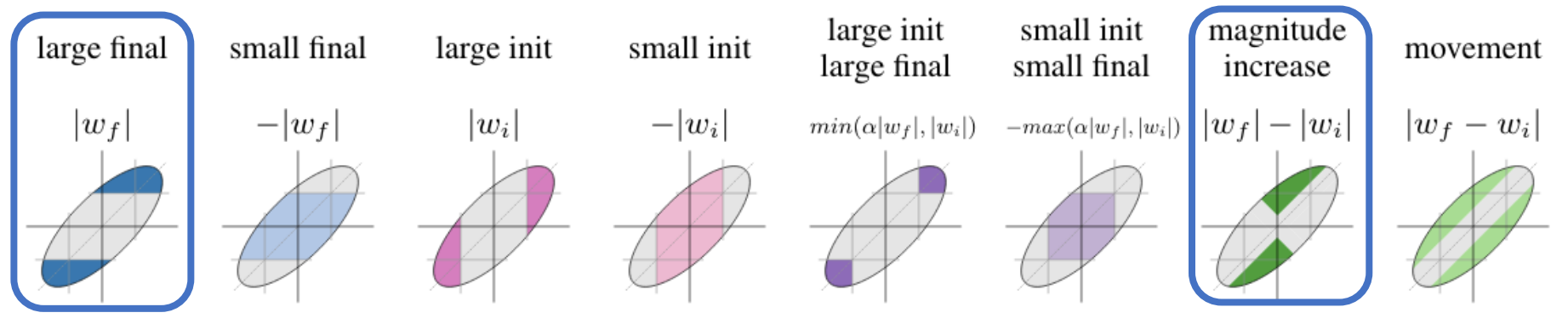
Lottery Ticket Hypothesis



Why Pruning?

Lottery Ticket Hypothesis

- Different pruning strategy



- “sign-ificance” of initial weights: Keeping the sign is critical

0.9, 3.1, -9.1, 8.5 \longrightarrow $+\alpha, +\alpha, -\alpha, +\alpha$

- Pruning weights from a network with random weights

Why Pruning?

<https://arxiv.org/abs/1810.05270>

- Rethinking the Value of Network Pruning

Dataset	Model	Unpruned	Pruned Model	Fine-tuned	Scratch-E	Scratch-B
CIFAR-10	VGG-16	93.63 (± 0.16)	VGG-16-A	93.41 (± 0.12)	93.62 (± 0.11)	93.78 (± 0.15)
	ResNet-56	93.14 (± 0.12)	ResNet-56-A	92.97 (± 0.17)	92.96 (± 0.26)	93.09 (± 0.14)
			ResNet-56-B	92.67 (± 0.14)	92.54 (± 0.19)	93.05 (± 0.18)
	ResNet-110	93.14 (± 0.24)	ResNet-110-A	93.14 (± 0.16)	93.25 (± 0.29)	93.22 (± 0.22)
			ResNet-110-B	92.69 (± 0.09)	92.89 (± 0.43)	93.60 (± 0.25)
ImageNet	ResNet-34	73.31	ResNet-34-A	72.56	72.77	73.03
			ResNet-34-B	72.29	72.55	72.91

- **New** random initialization, not **original** random initialization in “Lottery Ticket Hypothesis”
- Limitation of “Lottery Ticket Hypothesis” (unstructured)

Knowledge Distillation



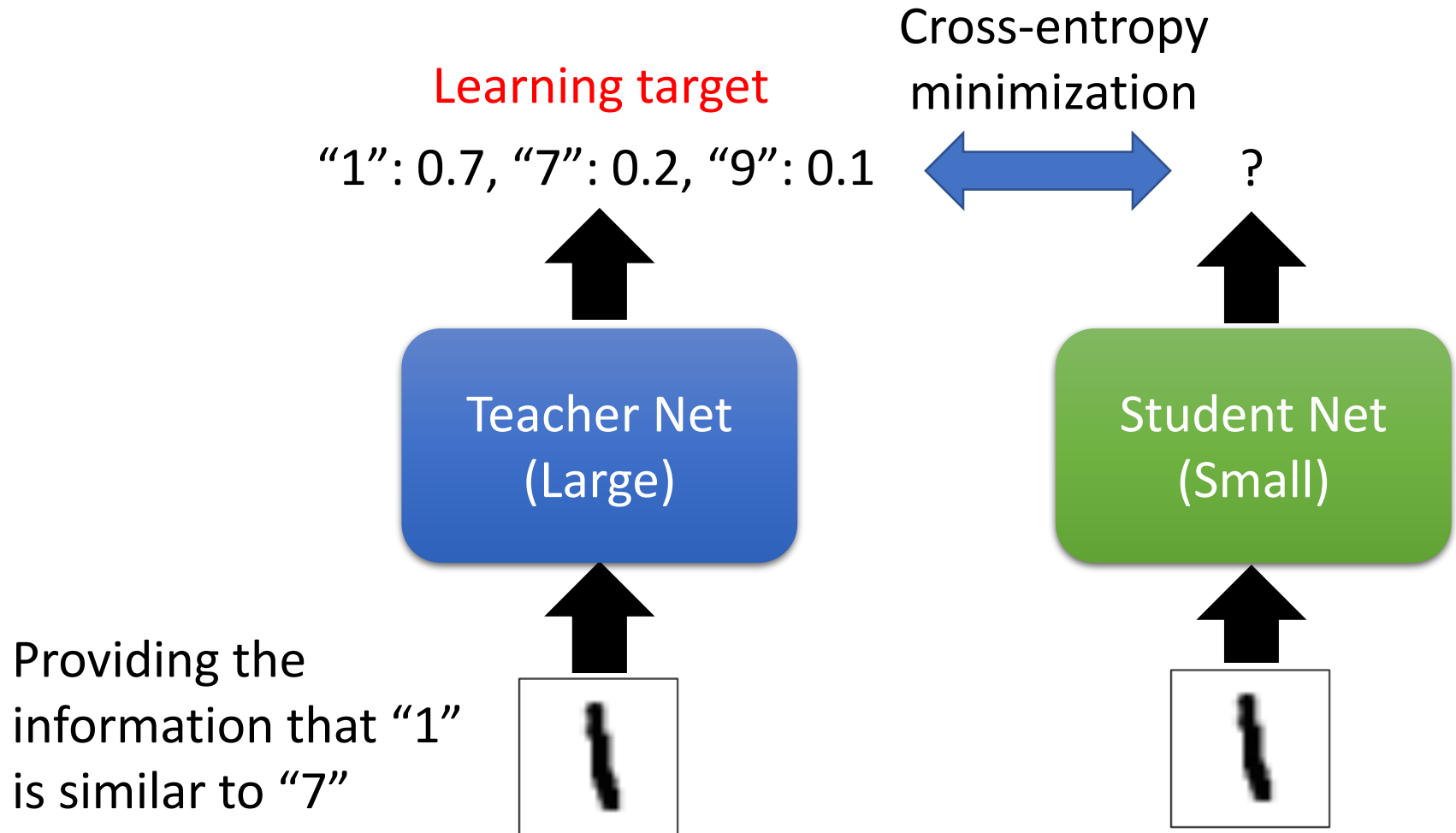
Knowledge Distillation

Knowledge Distillation

<https://arxiv.org/pdf/1503.02531.pdf>

Do Deep Nets Really Need to be Deep?

<https://arxiv.org/pdf/1312.6184.pdf>



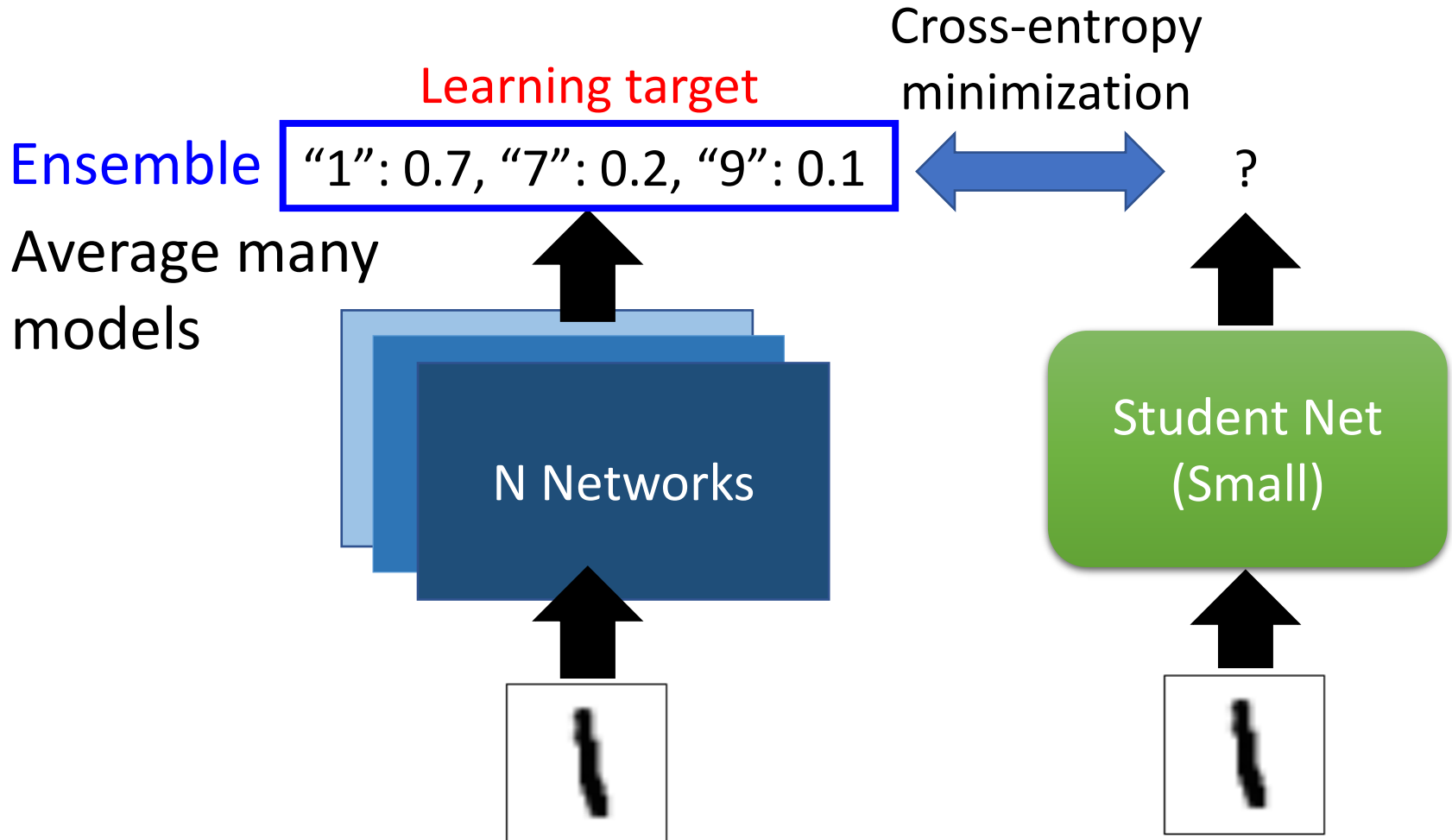
Knowledge Distillation

Knowledge Distillation

<https://arxiv.org/pdf/1503.02531.pdf>

Do Deep Nets Really Need to be Deep?

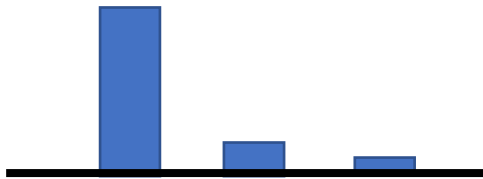
<https://arxiv.org/pdf/1312.6184.pdf>



Knowledge Distillation

- Temperature for softmax

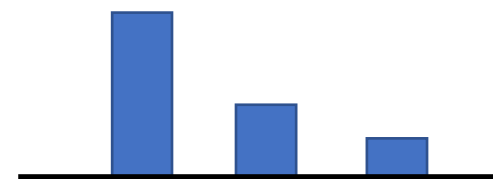
$$y'_i = \frac{\exp(y_i)}{\sum_j \exp(y_j)} \xrightarrow{T=100} y'_i = \frac{\exp(y_i/T)}{\sum_j \exp(y_j/T)}$$



$$y_1 = 100 \quad y'_1 = 1$$

$$y_2 = 10 \quad y'_2 \approx 0$$

$$y_3 = 1 \quad y'_3 \approx 0$$



$$y_1/T = 1 \quad y'_1 = 0.56$$

$$y_2/T = 0.1 \quad y'_2 = 0.23$$

$$y_3/T = 0.01 \quad y'_3 = 0.21$$

Parameter Quantization



Parameter Quantization

- 1. Using less bits to represent a value
- 2. Weight clustering

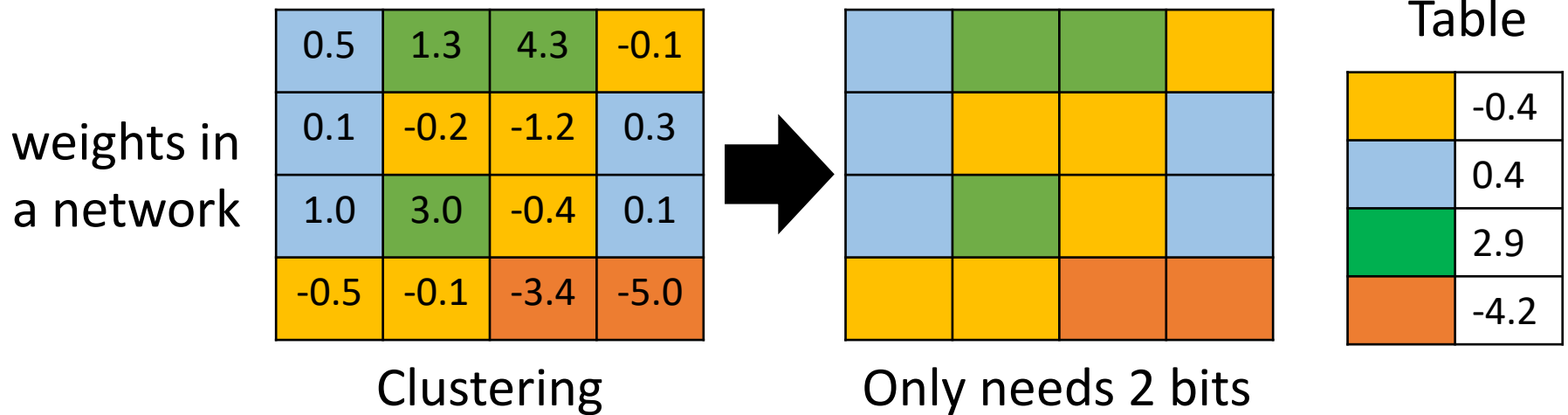
weights in
a network

0.5	1.3	4.3	-0.1
0.1	-0.2	-1.2	0.3
1.0	3.0	-0.4	0.1
-0.5	-0.1	-3.4	-5.0

Clustering

Parameter Quantization

- 1. Using less bits to represent a value
- 2. Weight clustering



- 3. Represent frequent clusters by less bits, represent rare clusters by more bits
 - e.g. Huffman encoding

Binary Weights

Your weights are always +1 or -1

- Binary Connect

Binary Connect:

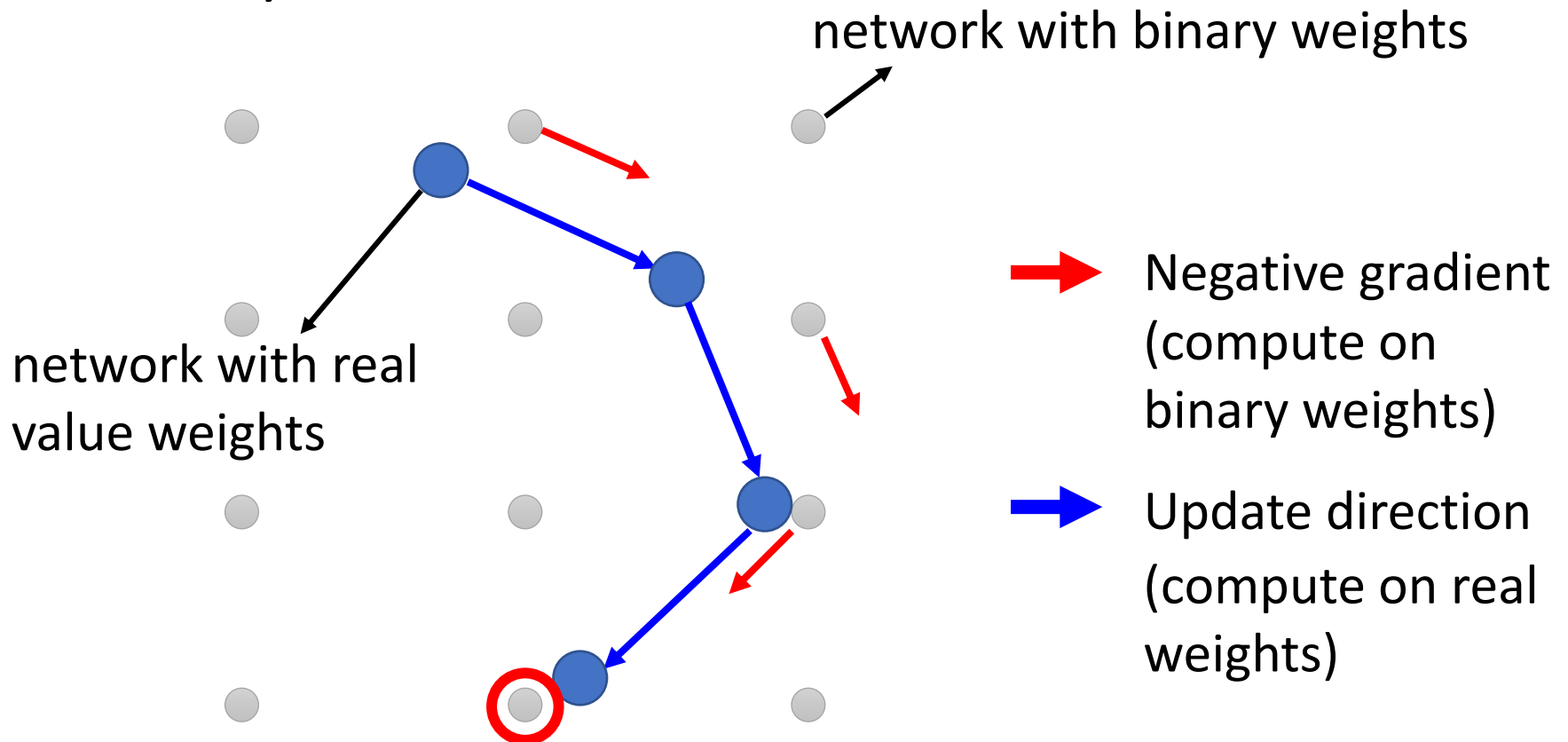
<https://arxiv.org/abs/1511.00363>

Binary Network:

<https://arxiv.org/abs/1602.02830>

XNOR-net:

<https://arxiv.org/abs/1603.05279>



Binary Weights

Method	MNIST	CIFAR-10	SVHN
No regularizer	$1.30 \pm 0.04\%$	10.64%	2.44%
BinaryConnect (det.)	$1.29 \pm 0.08\%$	9.90%	2.30%
BinaryConnect (stoch.)	$1.18 \pm 0.04\%$	8.27%	2.15%
50% Dropout	$1.01 \pm 0.04\%$		

