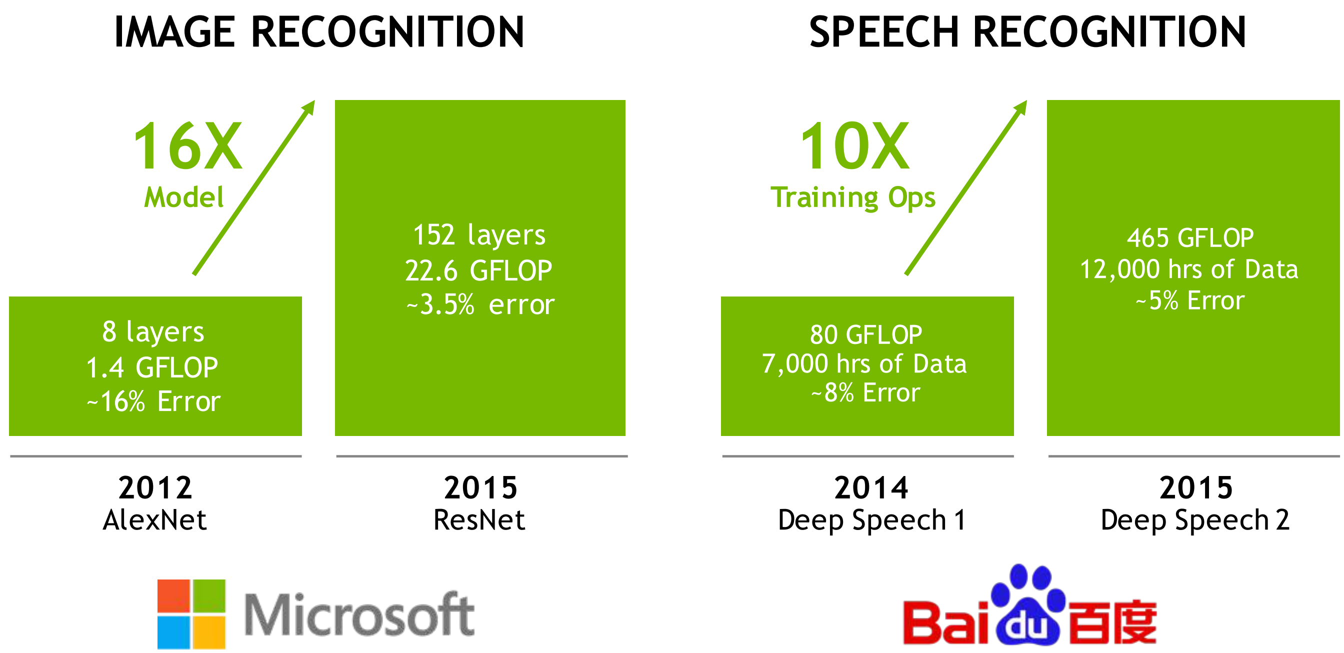
Efficient Methods and Hardware for Deep Learning

Song Han

Stanford University
DeePhi

Models are Getting Larger

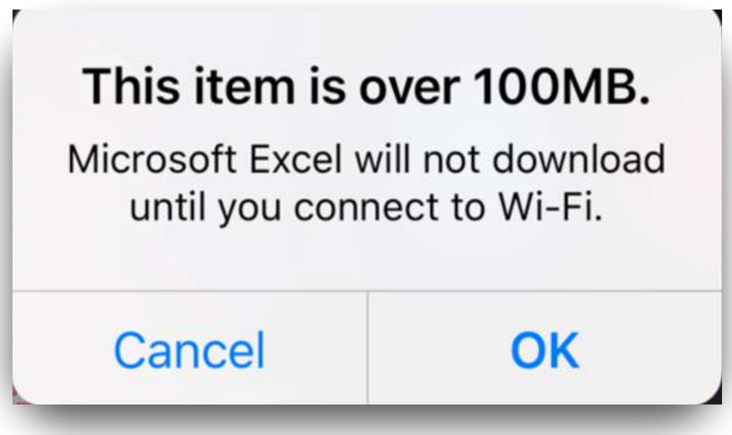


Dally, NIPS'2016 workshop on Efficient Methods for Deep Neural Networks

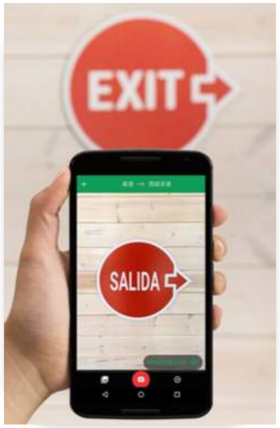
Problem of Large DNN Model: Difficult to Deploy

Large DNN Model: Difficult to Deploy

App developers suffers from the model size



Large DNN Model: Difficult to Deploy



Phones

Drones

Robots

Glasses

Self Driving Cars

- Limited Computation Resource
- Battery Constrained
- Cooling Constrained

Large DNN Model: Difficult to Deploy

Hardware engineer suffers from the model size

larger model => more memory reference => more energy

Operation	Energy [pJ]	Relative Cost
32 bit int ADD	0.1	1
32 bit float ADD	0.9	9
32 bit Register File	1	10
32 bit int MULT	3.1	31
32 bit float MULT	3.7	37
32 bit SRAM Cache	5	50
32 bit DRAM Memory	**640**	**6400**

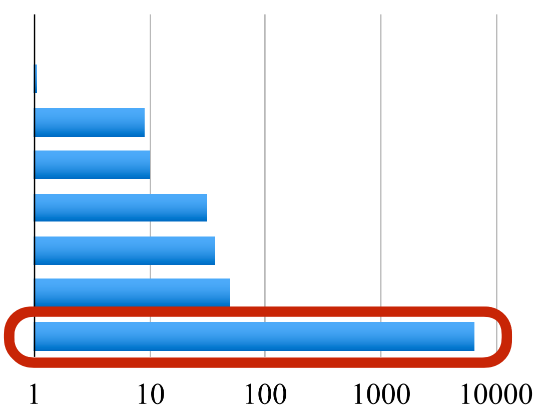
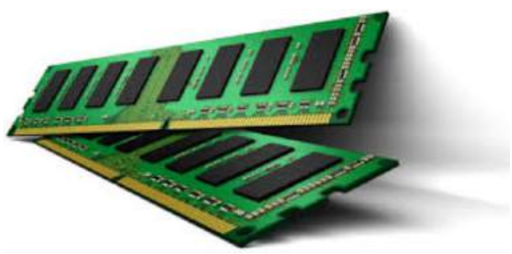
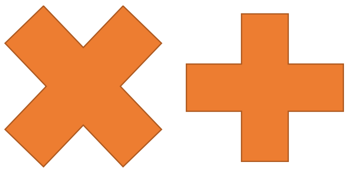


Figure 1: Energy table for 45nm CMOS process. Memory access is 2 orders of magnitude more energy expensive than arithmetic operations.

1  **= 100** 

Large DNN Model: Difficult to Deploy

Hardware engineer suffers from the model size

larger model => more memory reference => more energy

Operation	Energy [pJ]	Relative Cost
32 bit int ADD	0.1	1
32 bit float ADD	0.9	9
32 bit Register File	1	10
32 bit int MULT	3.1	31
32 bit float MULT	3.7	37
32 bit SRAM Cache	5	50
32 bit DRAM Memory	**640**	**6400**

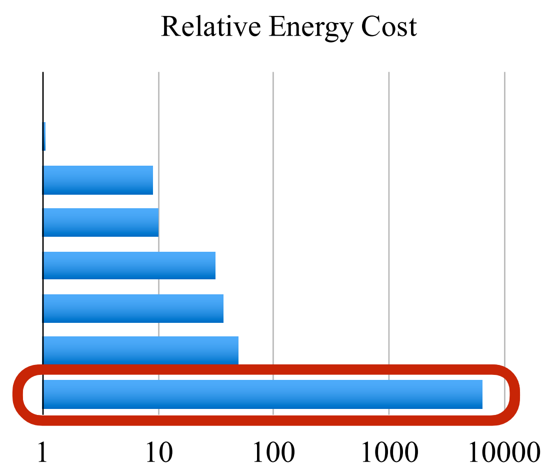
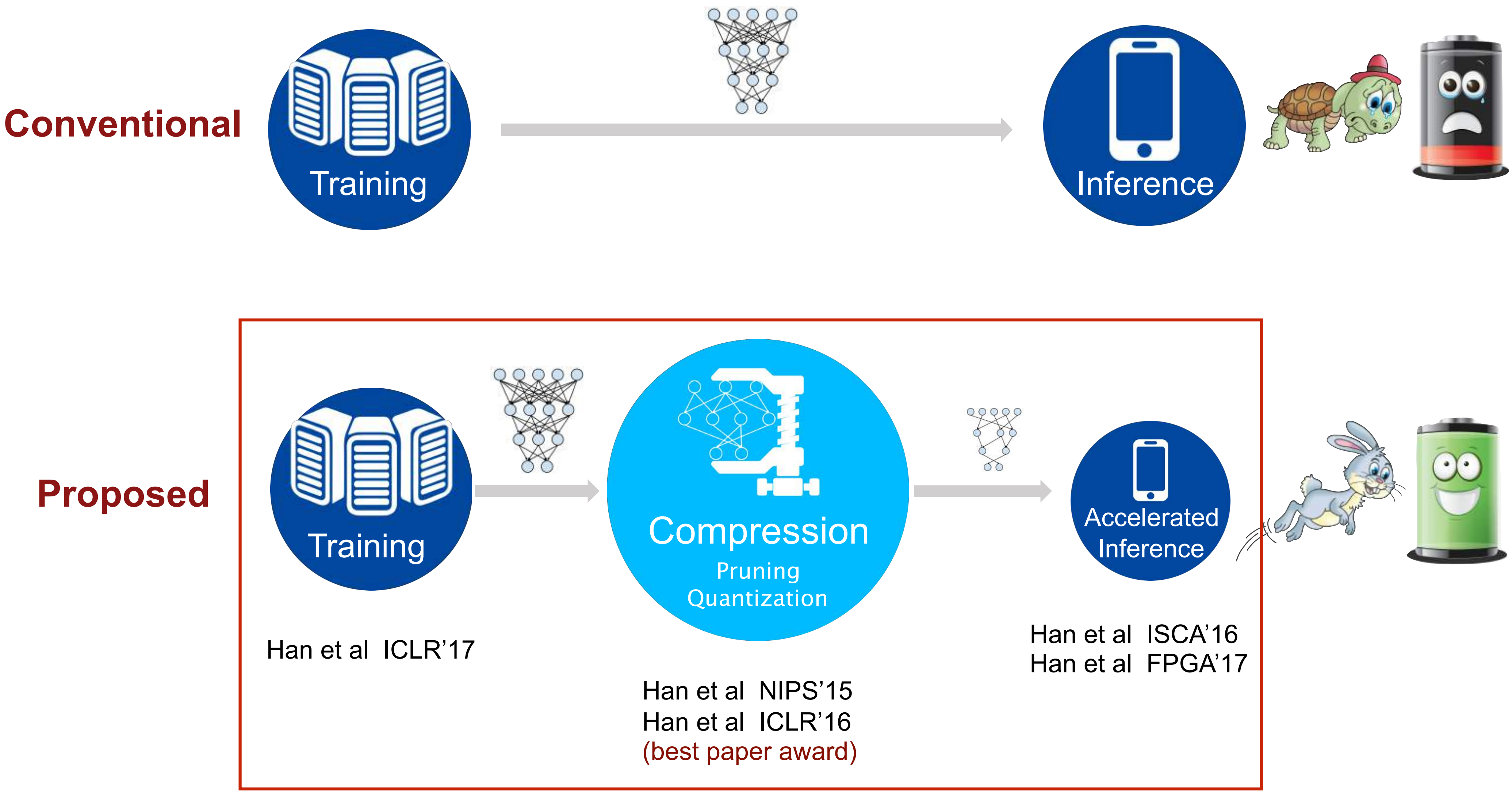


Figure 1: Energy table for 45nm CMOS process. Memory access is 2 orders of magnitude more energy expensive than arithmetic operations.

Given the power budget, Moore's law is no longer providing more computation

Improve the Efficiency of Deep Learning by Algorithm-Hardware Co-Design

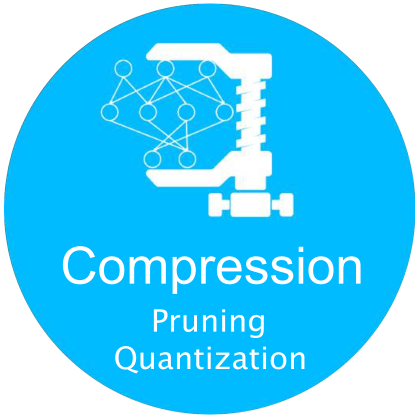
Proposed Paradigm



Agenda

✦ **Model Compression (<u>size</u>)**

- Pruning / Quantization
- Ternary Net



✦ **Hardware Acceleration (<u>speed, energy</u>)**

- EIE Accelerator (ASIC)
- ESE Accelerator (FPGA)

✦ **Efficient Training (<u>accuracy</u>)**

- Dense-Sparse-Dense Regularization

Agenda

✦ **Model Compression**
- Pruning / Quantization
- Ternary Net

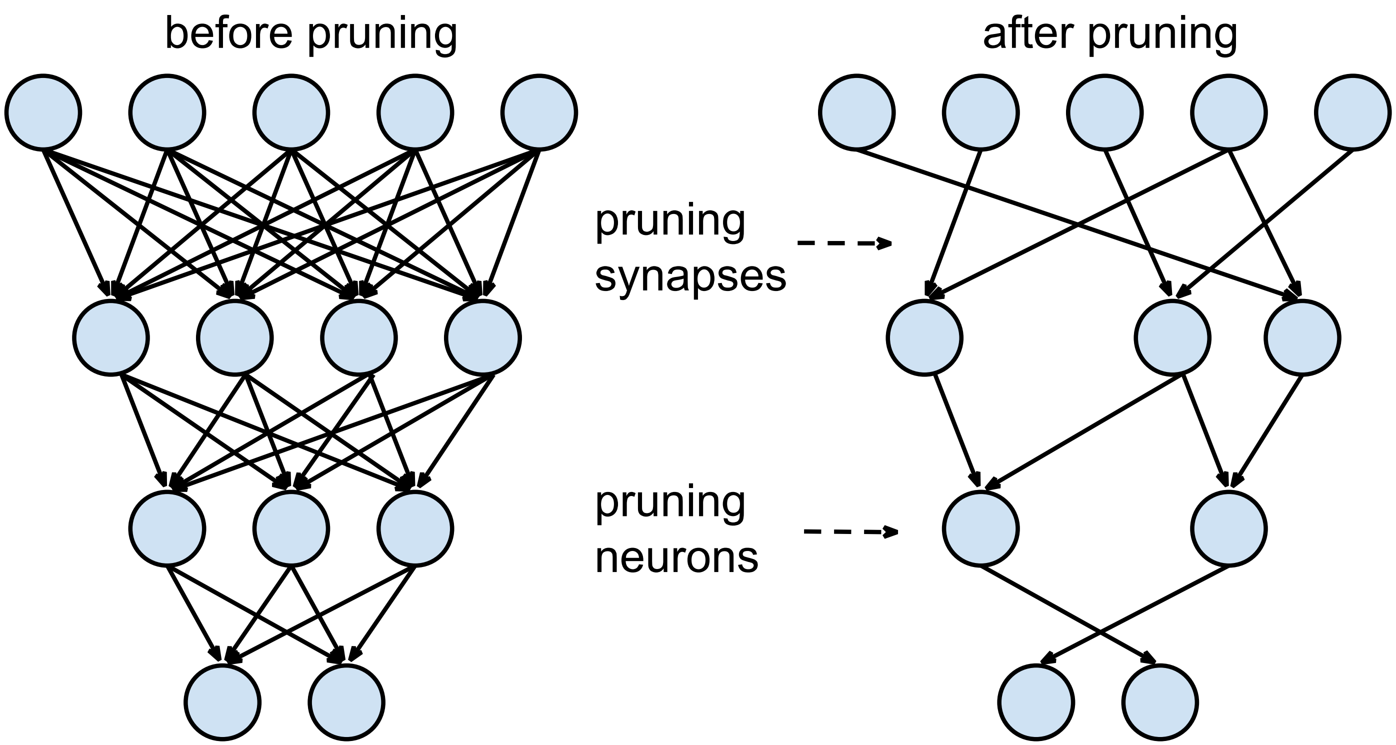
✦ **Hardware Acceleration**
- EIE Accelerator (ASIC)
- ESE Accelerator (FPGA)

✦ **Efficient Training**
- Dense-Sparse-Dense Regularization

Deep Compression Pipeline

- **Network Pruning:**
  Less Number of Weights
- **Trained Quantization:**
  Reduce Storage for Each Remaining Weight
- **Huffman Coding:**
  Entropy of the Remaining Weights

Pruning



Han et al. Learning both Weights and Connections for Efficient Neural Networks, NIPS'15

Pruning: Motivation

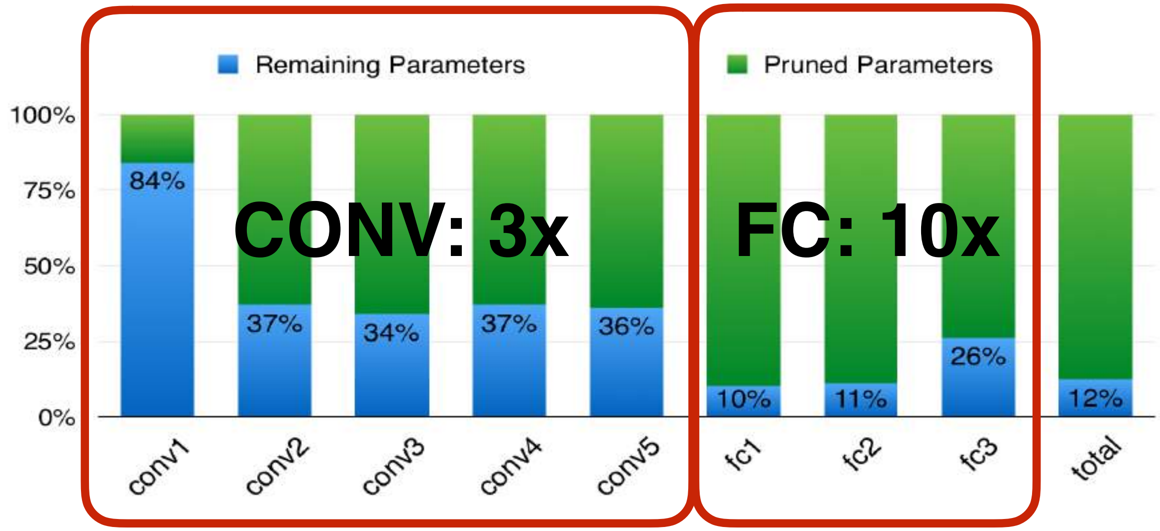
Age	Number of Connections	Stage
at birth	$\boxed{\text{50 Trillion}}$	newly formed
1 year old	$\boxed{\text{1000 Trillion}}$	peak
10 year old	$\boxed{\text{500 Trillion}}$	pruned and stabilized

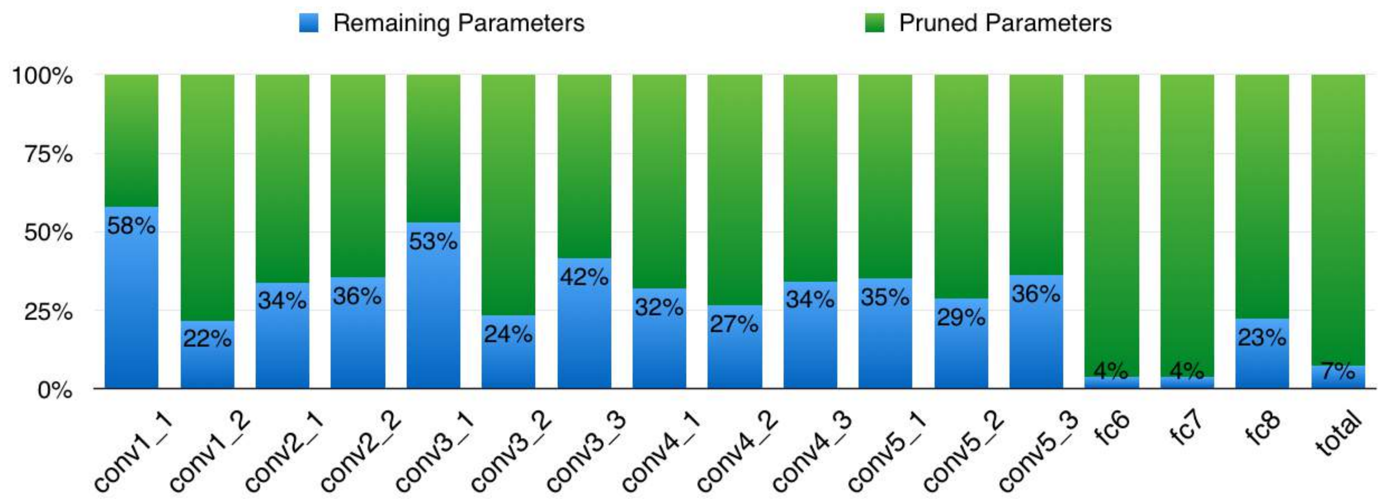
Table 1: The synapses pruning mechanism in human brain development

- **At birth,** Trillions of synapses
- **1 year old**, peaked at **1000 trillion**
- Pruning begins to occur.
- **10 years old**, pruned to nearly **500 trillion** synapses
- This "pruning" mechanism removes redundant connections in the brain.

[1] Christopher A Walsh. Peter Huttenlocher (1931-2013). Nature, 502(7470):172–172, 2013.

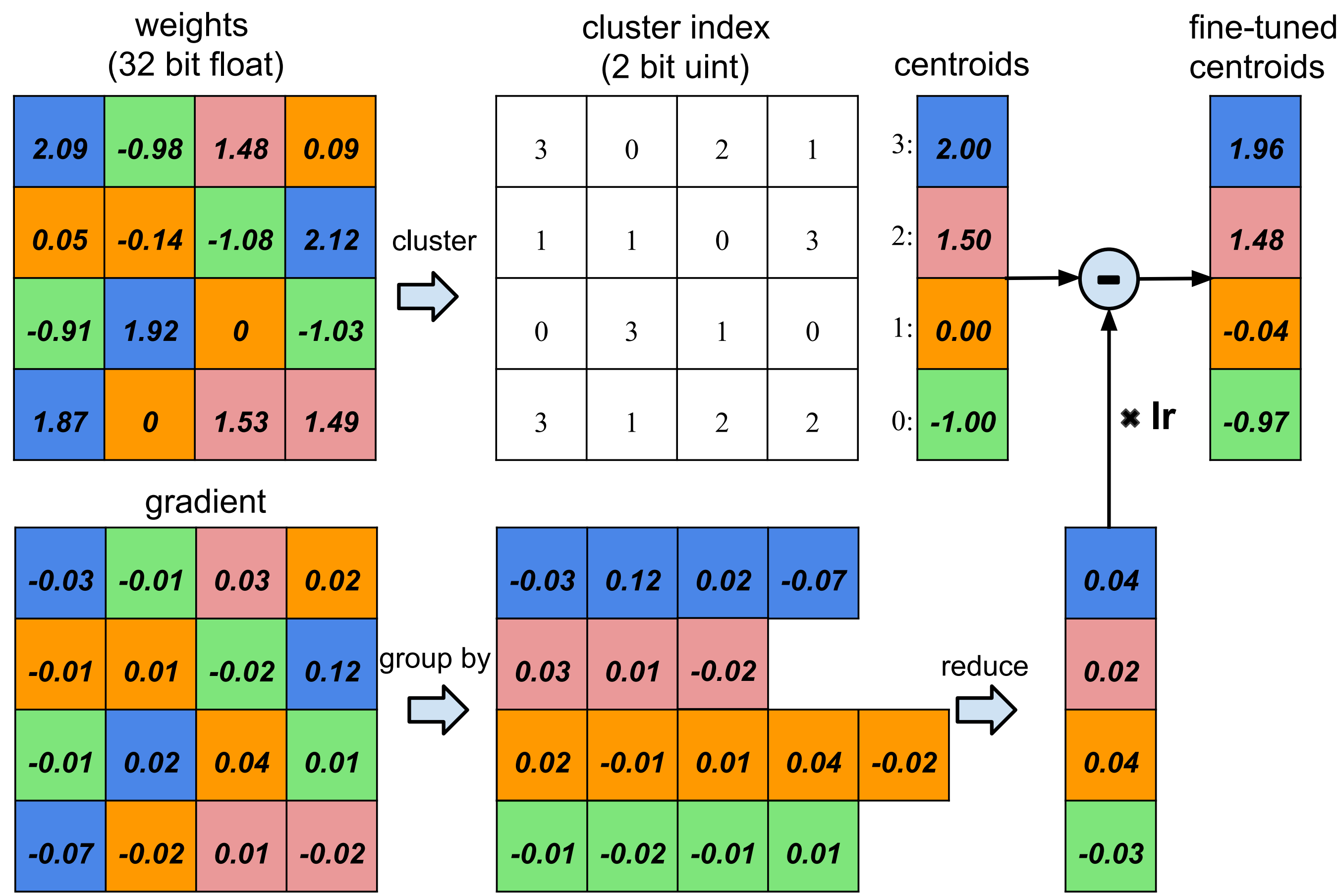
AlexNet & VGGNet





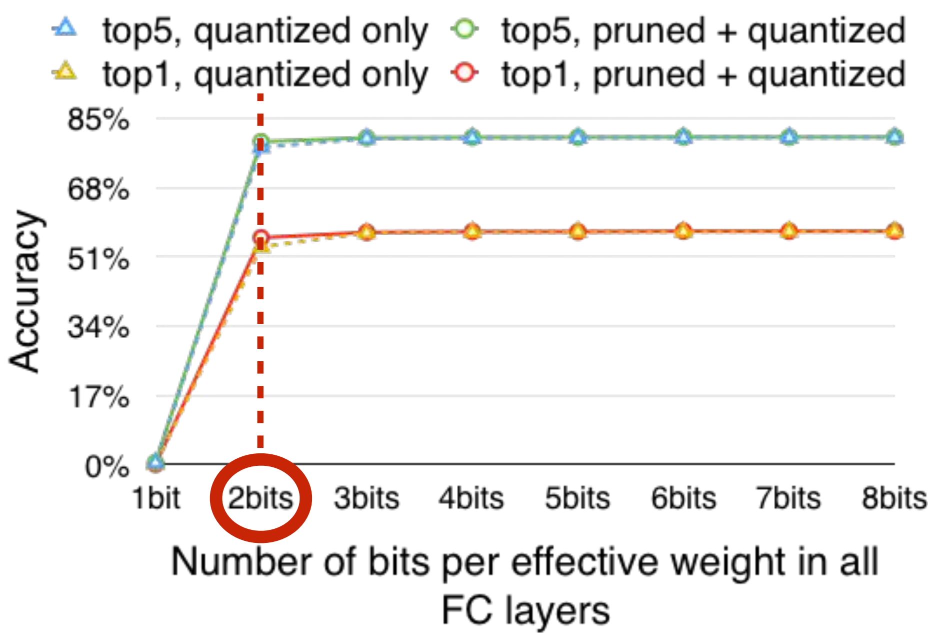
Han et al. Learning both Weights and Connections for Efficient Neural Networks, NIPS 2015

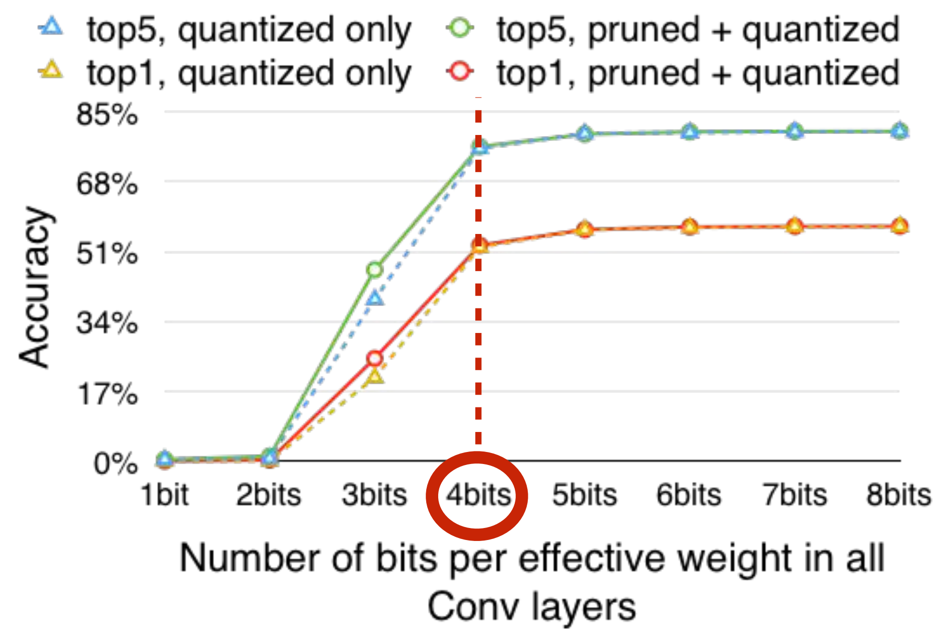
Trained Quantization



Han et al. Deep Compression, ICLR 2016 Best Paper Award

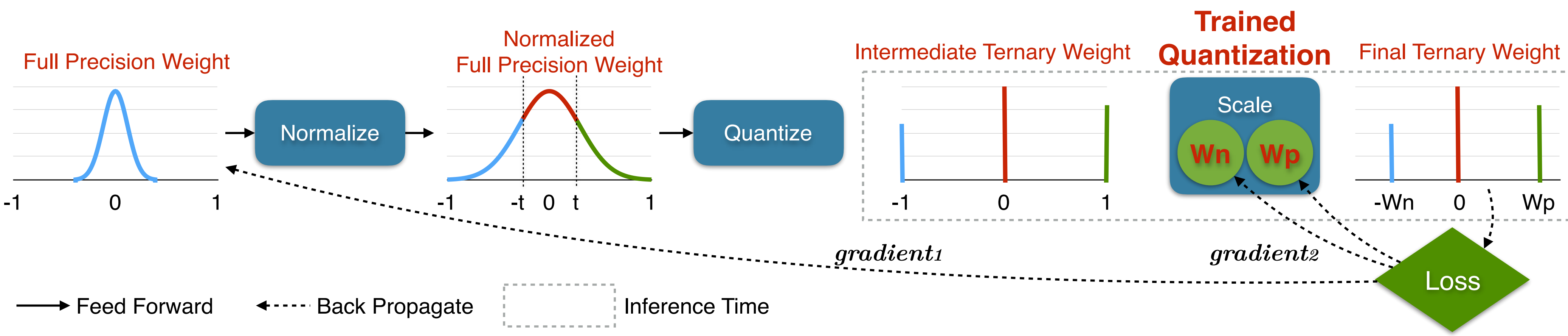
Bits Per Weight





Han et al. Deep Compression, ICLR 2016 Best Paper Award

Even Fewer Bits: Trained Ternary Quantization



Zhu, Han, Mao, Dally. Trained Ternary Quantization, ICLR'17

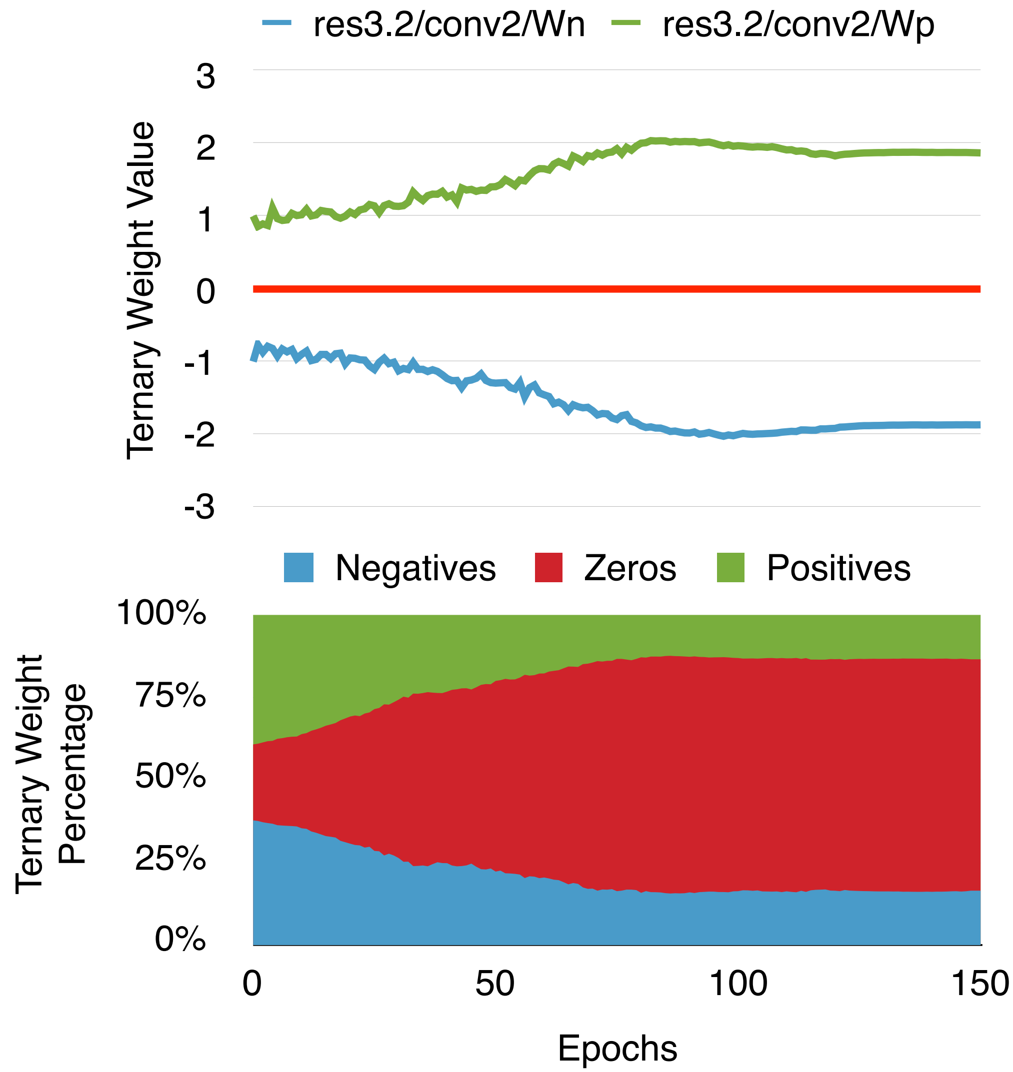
Trained Ternary Quantization — Learn both Centroid and Grouping

Learn Centroids:

0 stays 0, positive weight gets larger
negative weight gets smaller

Learn Grouping:

more weights grouped to zero (red)
less grouped to positive (green)
less grouped to negative (blue)
80% sparse in the end



Zhu, Han, Mao, Dally. Trained Ternary Quantization, ICLR'17

Ternary Net is Sparse

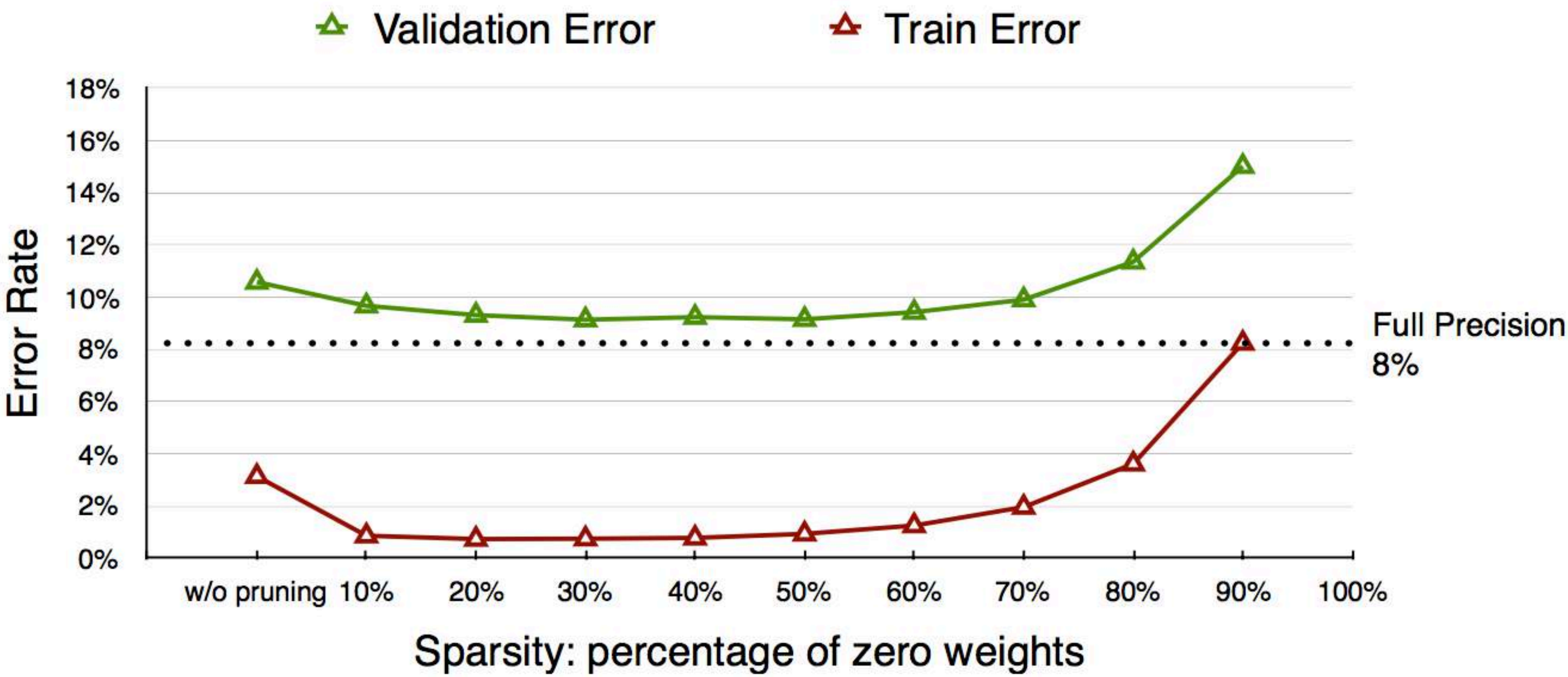
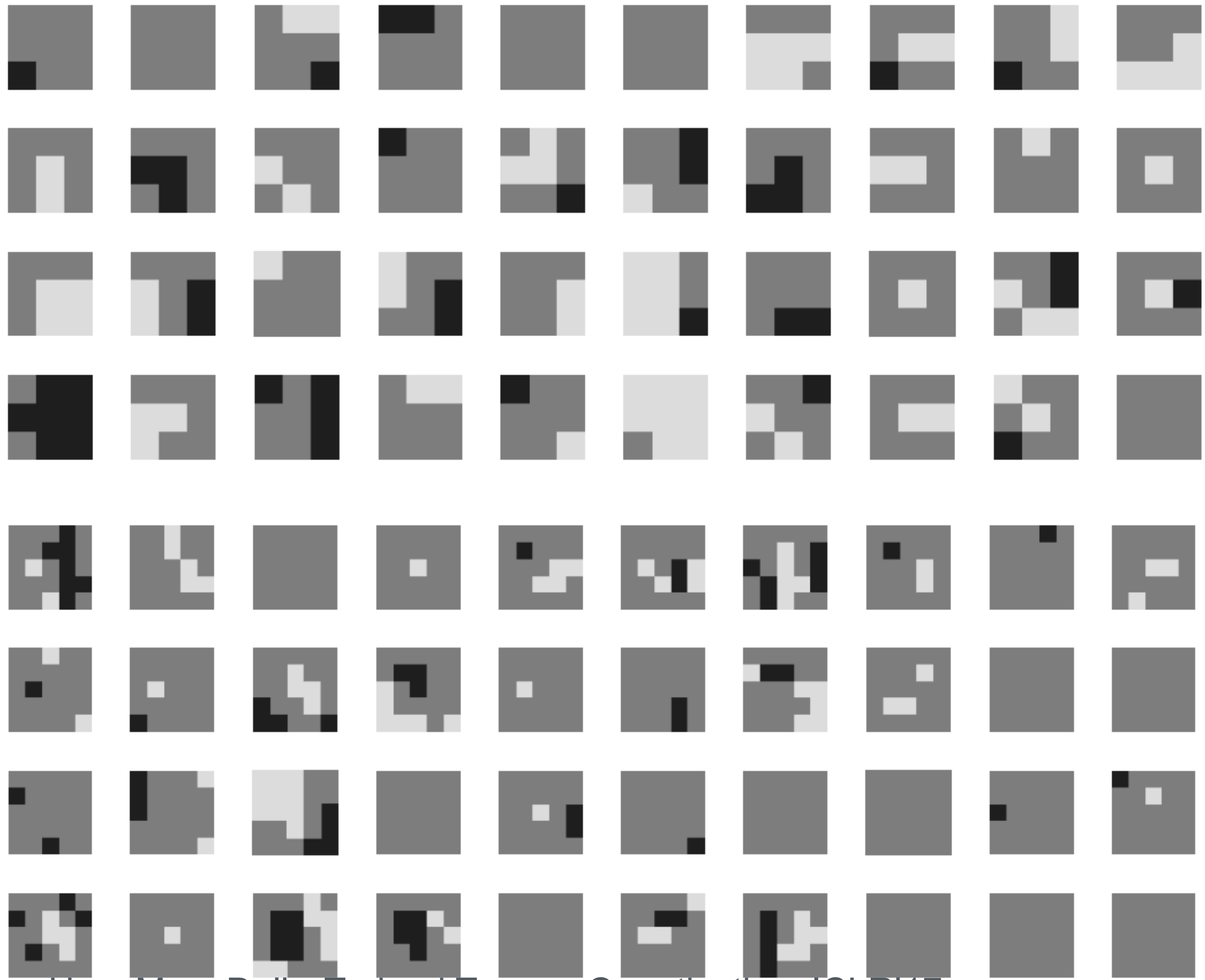


Figure 5: Accuracy v.s. Sparsity on ResNet-20

Zhu, Han, Mao, Dally. Trained Ternary Quantization, ICLR'17

Visualization of the TTQ Kernels



Zhu, Han, Mao, Dally. Trained Ternary Quantization, ICLR'17

TTQ: Accuracy

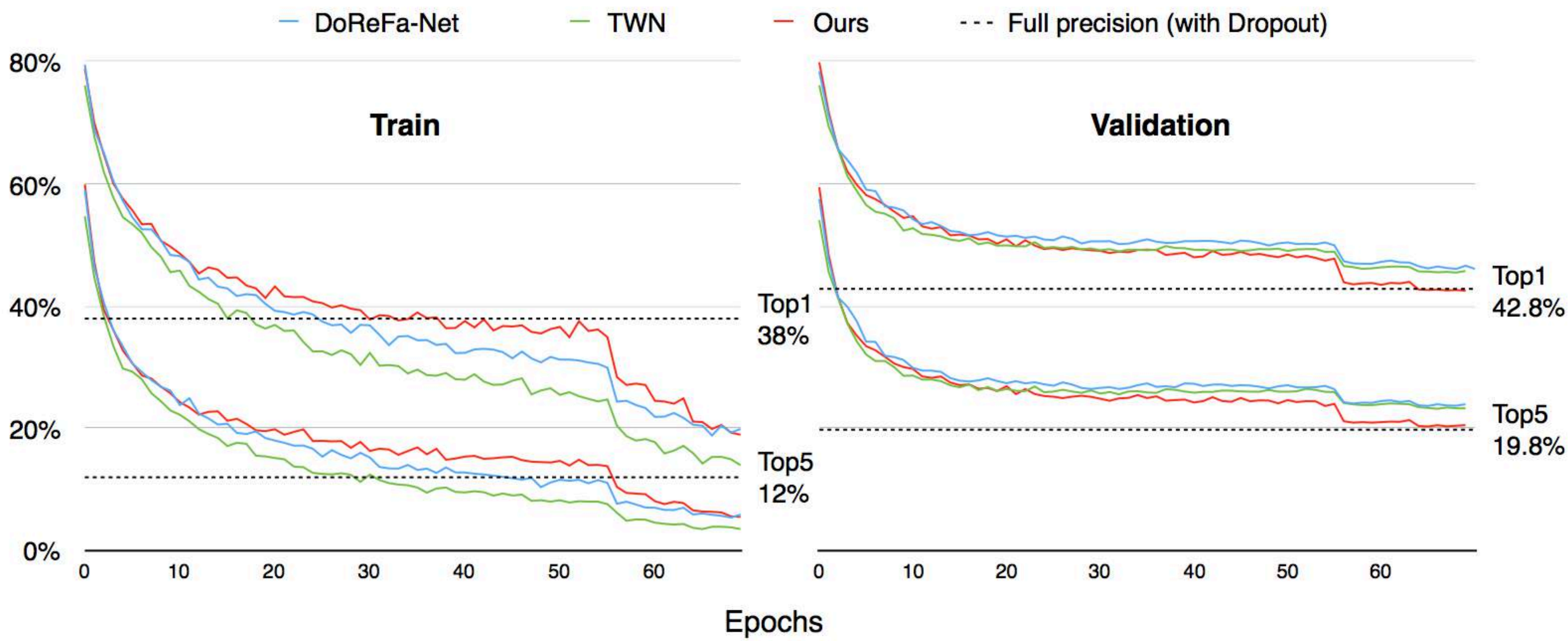


Figure 4: Training and validation accuracy of AlexNet on ImageNet

Zhu, Han, Mao, Dally. Trained Ternary Quantization, ICLR'17

Model Compression Means

- **Complex DNNs can be put in mobile applications (<10MB total)**
  - 500MB with-FC network (125M weights) becomes 10MB
  - 10MB all-CONV network (2.5M weights) becomes 1MB
- **Memory bandwidth reduced by 10-50x**
  - Particularly for FC layers in real-time applications with no reuse
  - Good for distributed training => less communication overhead
- **Memory working set fits in on-chip SRAM**
  - 5pJ/word access v.s. 640pJ/word

Challenges

- **Online de-compression while computing**
  - Special purpose logic
- **Computation becomes irregular**
  - Sparse weight
  - Sparse activation
  - Indirect lookup
- **Parallelization becomes challenging**
  - Synchronization overhead.
  - Load imbalance issue.
  - Scalability

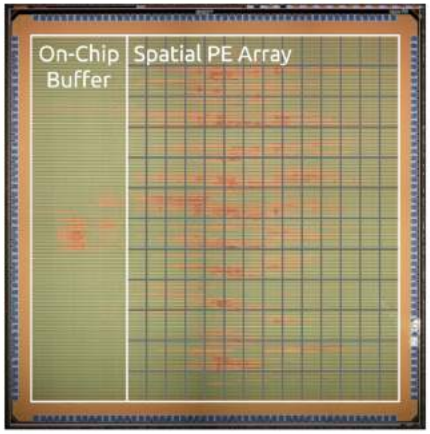
Agenda

✦ **Deep Compression (size)**
- Pruning
- Trained Quantization
- Huffman Coding

✦ **Hardware Acceleration (speed, energy)**
- EIE Accelerator (ASIC)
- ESE Accelerator (FPGA)

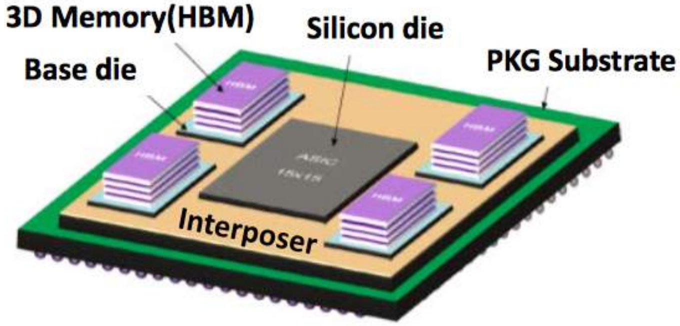
✦ **Efficient Training (accuracy)**
- Dense-Sparse-Dense Regularization

Related Work



Eyeriss, MIT

TPU, Google



Nervana

Agenda

- **Model Compression**
  - Pruning / Quantization
  - Ternary Net
- **Hardware Acceleration**
  - EIE Accelerator (ASIC)
  - ESE Accelerator (FPGA)
- **Efficient Training**
  - Dense-Sparse-Dense Regularization

EIE: Inference on Sparse, Compressed Model

logically

$$\vec{a} \quad \begin{pmatrix} 0 & \boldsymbol{a_1} & 0 & a_3 \end{pmatrix}$$

$$\times$$

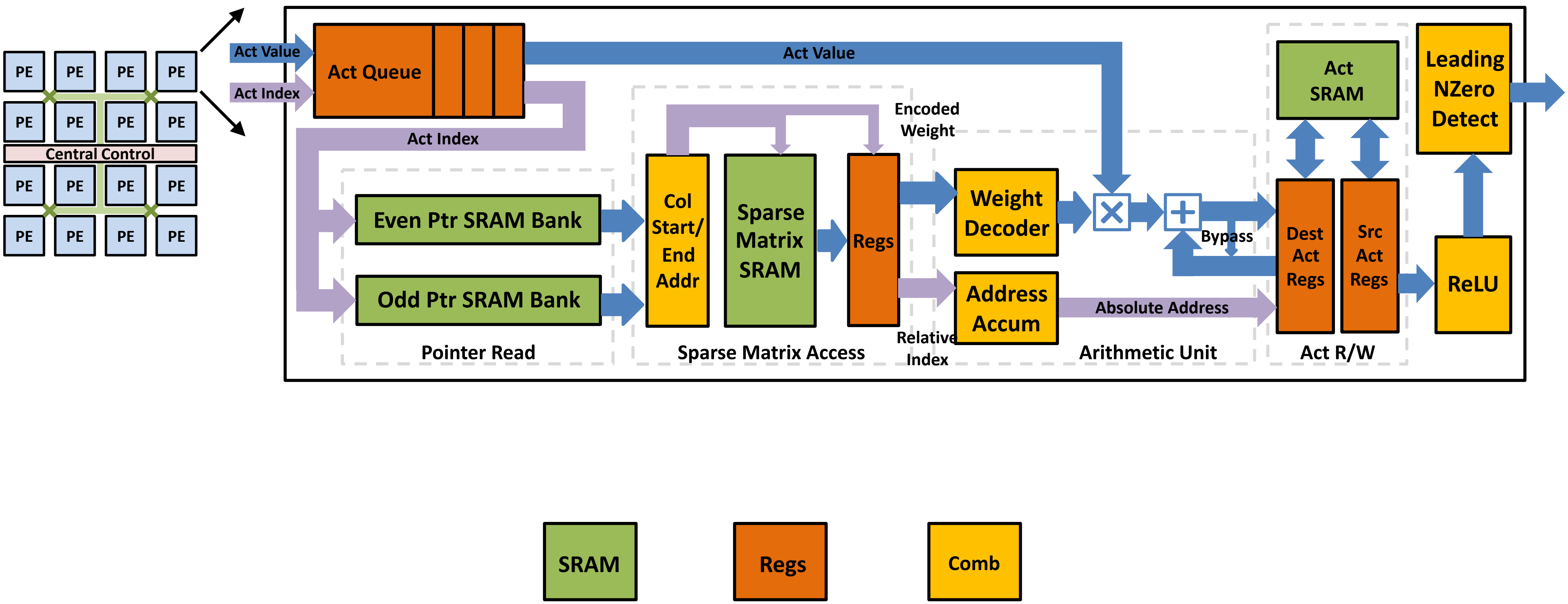
$$\begin{array}{c} PE0 \\ PE1 \\ PE2 \\ PE3 \\ \\ \\ \\ \\ \end{array} \begin{pmatrix} w_{0,0} & \boldsymbol{w_{0,1}} & 0 & w_{0,3} \\ 0 & \boldsymbol{0} & w_{1,2} & 0 \\ 0 & \boldsymbol{w_{2,1}} & 0 & w_{2,3} \\ 0 & \boldsymbol{0} & 0 & 0 \\ 0 & \boldsymbol{0} & w_{4,2} & w_{4,3} \\ w_{5,0} & \boldsymbol{0} & 0 & 0 \\ 0 & \boldsymbol{0} & 0 & w_{6,3} \\ 0 & \boldsymbol{w_{7,1}} & 0 & 0 \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ -b_2 \\ b_3 \\ -b_4 \\ b_5 \\ b_6 \\ -b_7 \end{pmatrix} \overset{ReLU}{\Rightarrow} \overset{\vec{b}}{\begin{pmatrix} b_0 \\ b_1 \\ 0 \\ b_3 \\ 0 \\ b_5 \\ b_6 \\ 0 \end{pmatrix}}$$

physically

Virtual Weight	$W_{0,0}$	$W_{0,1}$	$W_{4,2}$	$W_{0,3}$	$W_{4,3}$
Relative Index	0	1	2	0	0
Column Pointer	0	1	2	3	

Han et al. “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016, Hotchips 2016

PE Architecture



Han et al. “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016, Hotchips 2016

Where are the savings from?

Sparse Matrix

90% *static* sparsity
in the weights,
10x less computation,
5x less memory footprint

Han et al. “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016, Hotchips 2016

Where are the savings from?

Sparse Matrix

90% *static* sparsity
in the weights,
10x less computation,
5x less memory footprint

Sparse Vector

70% *dynamic* sparsity
in the activation
3x less computation

Han et al. "EIE: Efficient Inference Engine on Compressed Deep Neural Network", ISCA 2016, Hotchips 2016

Where are the savings from?

Sparse Matrix

90% *static* sparsity
in the weights,
10x less computation,
5x less memory footprint

Sparse Vector

70% *dynamic* sparsity
in the activation
3x less computation

Weight Sharing

4bits weights
8x less memory
footprint

Han et al. "EIE: Efficient Inference Engine on Compressed Deep Neural Network", ISCA 2016, Hotchips 2016

Where are the savings from?

Sparse Matrix

90% *static* sparsity
in the weights,
10x less computation,
5x less memory footprint

Sparse Vector

70% *dynamic* sparsity
in the activation
3x less computation

Weight Sharing

4bits weights
8x less memory
footprint

Fully fits in SRAM

120x less energy than DRAM

Han et al. "EIE: Efficient Inference Engine on Compressed Deep Neural Network", ISCA 2016, Hotchips 2016

Where are the savings from?

Sparse Matrix

90% *static* sparsity in the weights, **10x less** computation, **5x less** memory footprint

Sparse Vector

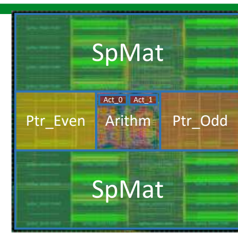
70% *dynamic* sparsity in the activation **3x less** computation

Weight Sharing

4bits weights **8x less** memory footprint

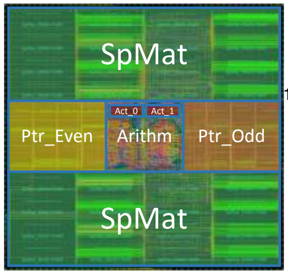
Fully fits in SRAM

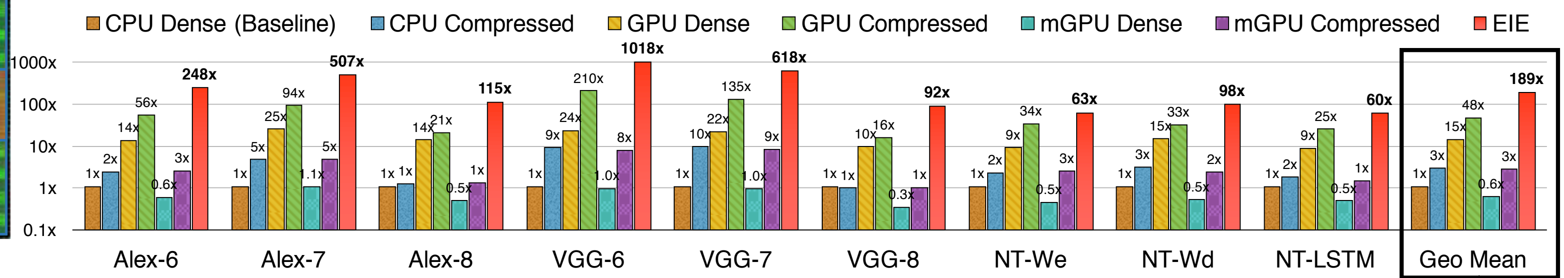
120x less energy than DRAM

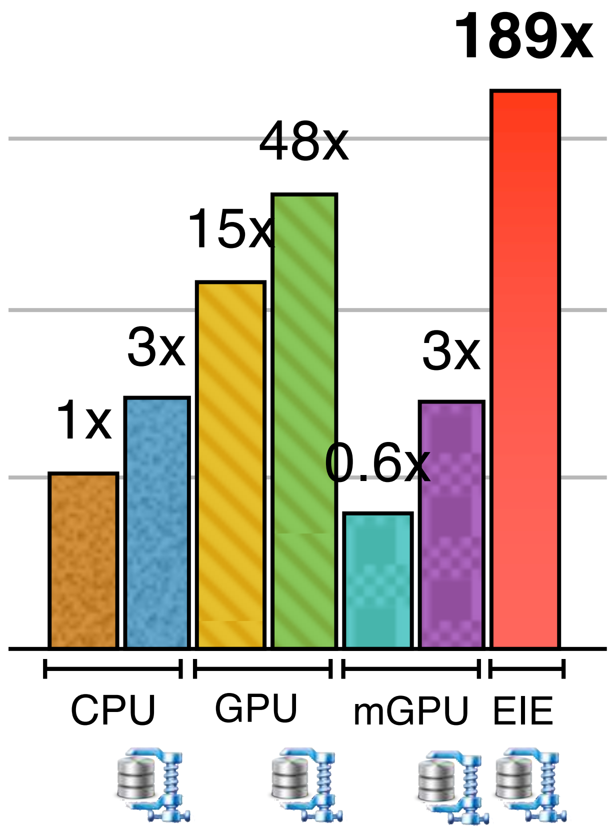


Han et al. “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016, Hotchips 2016

Speedup of EIE



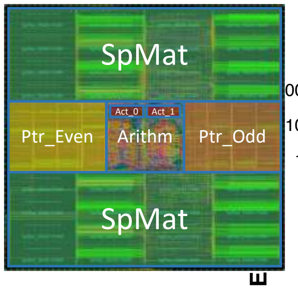


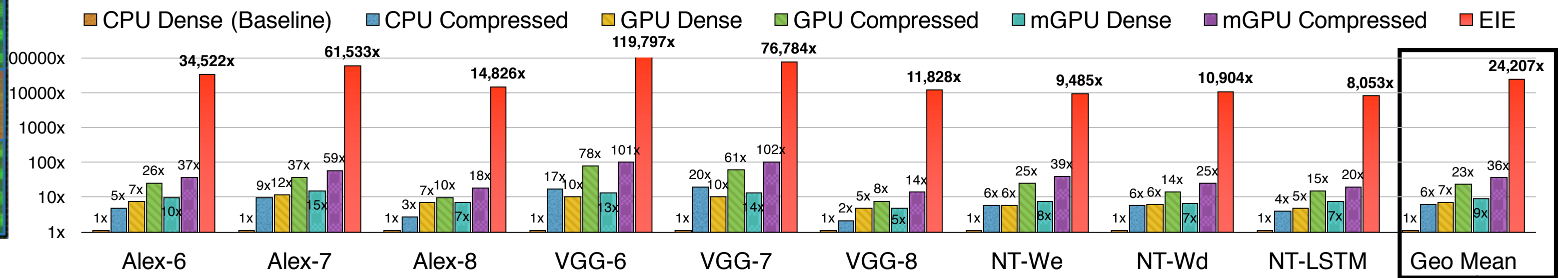


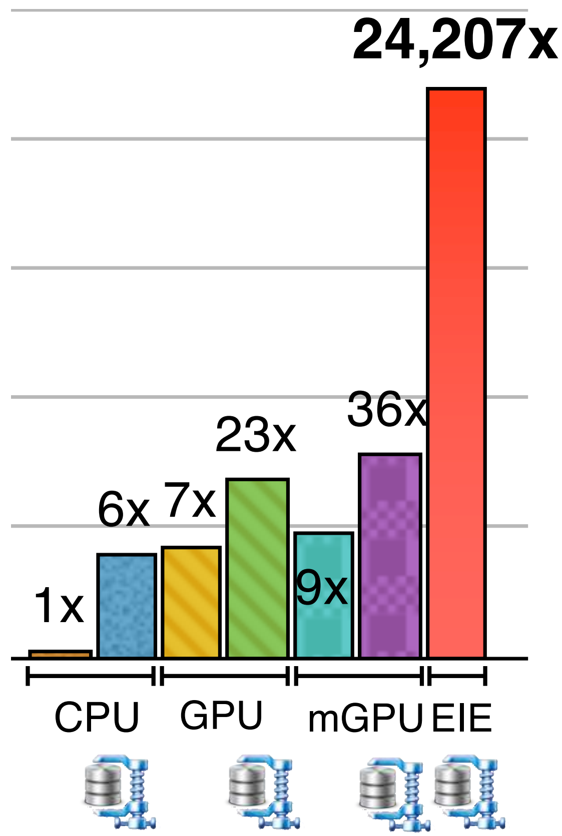
Baseline:

- Intel Core i7 5930K: MKL CBLAS GEMV, MKL SPBLAS CSRMV
- NVIDIA GeForce GTX Titan X: cuBLAS GEMV, cuSPARSE CSRMV
- NVIDIA Tegra K1: cuBLAS GEMV, cuSPARSE CSRMV

Energy Efficiency of EIE



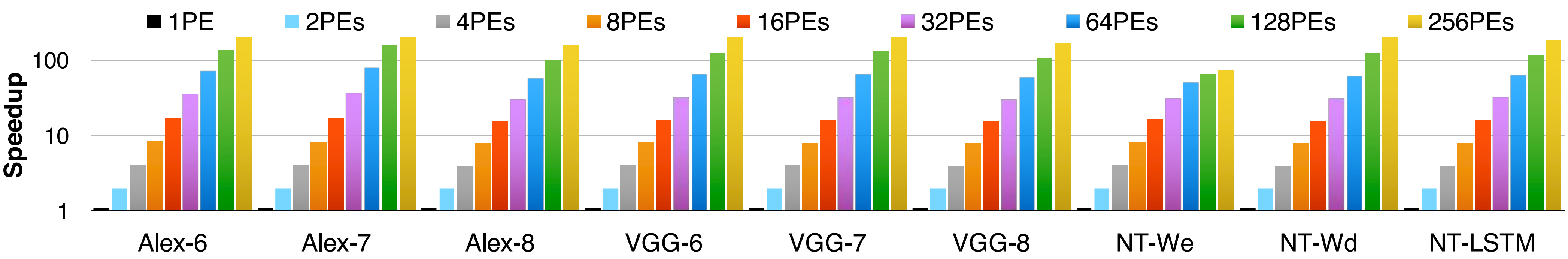




Baseline:

- Intel Core i7 5930K: MKL CBLAS GEMV, MKL SPBLAS CSRMV
- NVIDIA GeForce GTX Titan X: cuBLAS GEMV, cuSPARSE CSRMV
- NVIDIA Tegra K1: cuBLAS GEMV, cuSPARSE CSRMV

Scalability

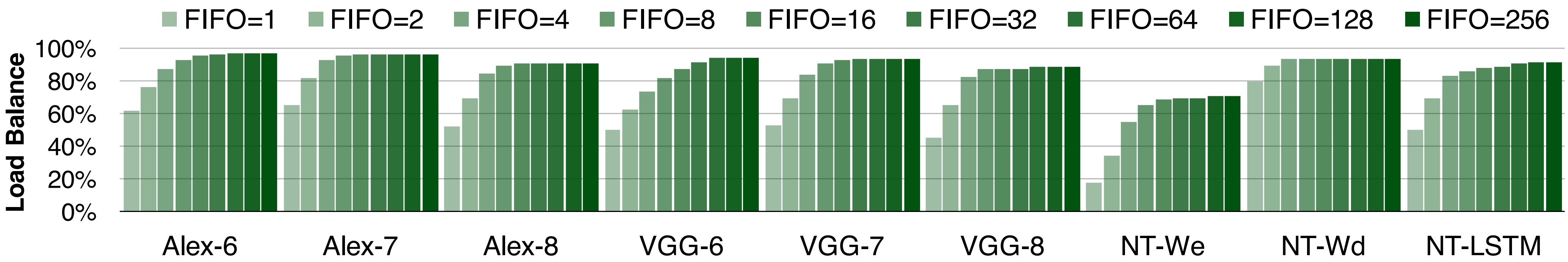


#PEs ~ Speedup

- 64PEs: 64x
- 128PEs: 124x
- 256PEs: 210x

Han et al. "EIE: Efficient Inference Engine on Compressed Deep Neural Network", ISCA 2016, Hotchips 2016

Load Balancing



- Imbalanced non-zeros among PEs degrades system utilization.
- This load imbalance could be solved by FIFO.
- With FIFO depth=8, ALU utilization is > 80%.

Han et al. “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016, Hotchips 2016

Remaining Questions

- Can we do better with load imbalance?
- Feedforward => Recurrent neural network?

Agenda

- Deep Compression (size)
  - Pruning
  - Trained Quantization
  - Huffman Coding

✦ **Hardware Acceleration (speed, energy)**
  - EIE Accelerator (ASIC)
  - ESE Accelerator (FPGA)

✦ **Efficient Training (accuracy)**
  - Dense-Sparse-Dense Regularization

Accelerating Recurrent Neural Networks

speech recognition

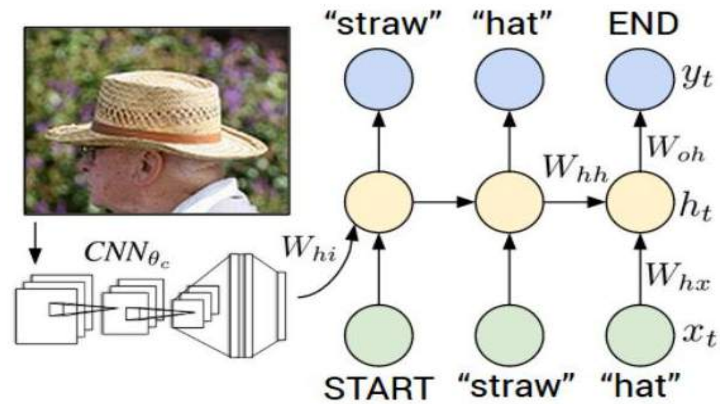
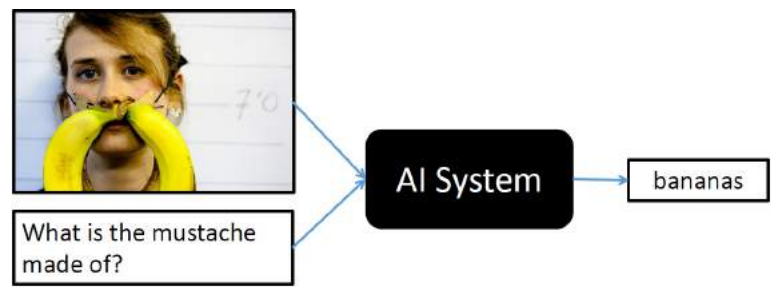


image caption

Google Translate

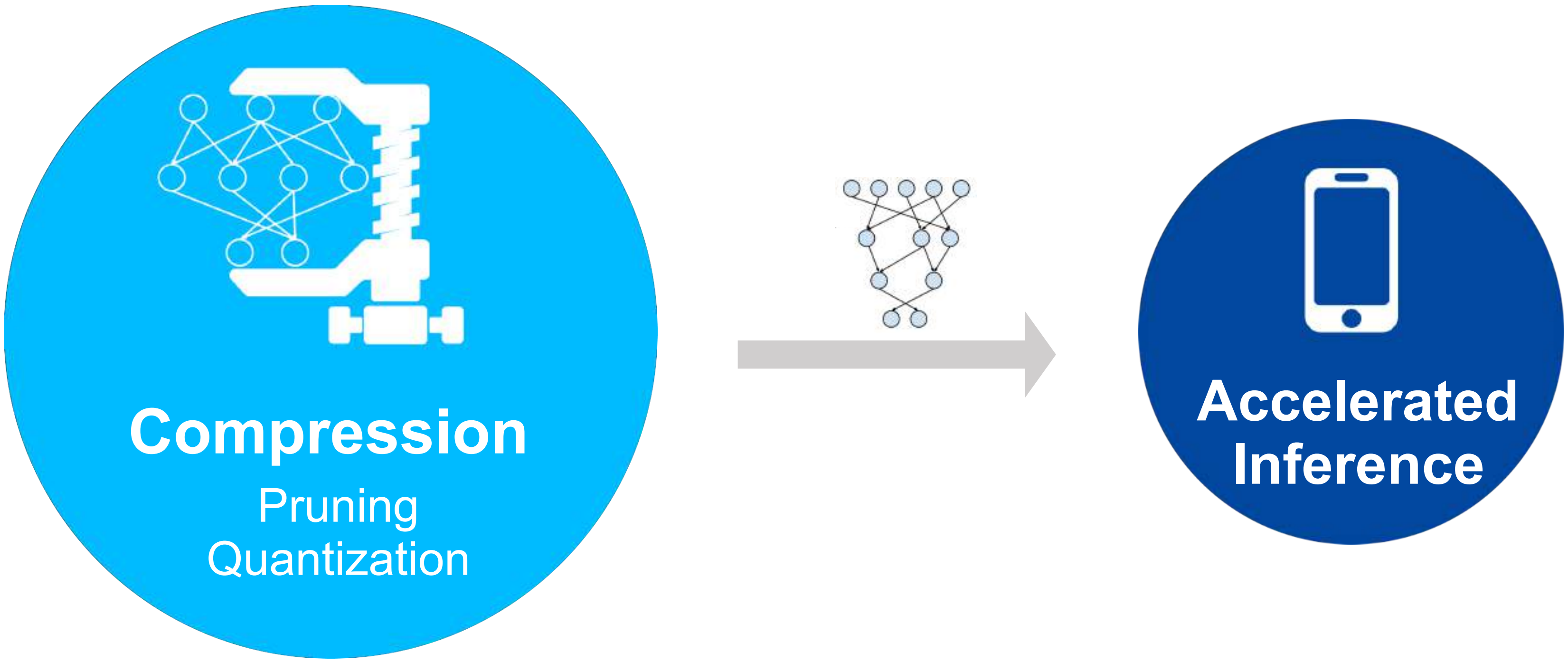
machine translation



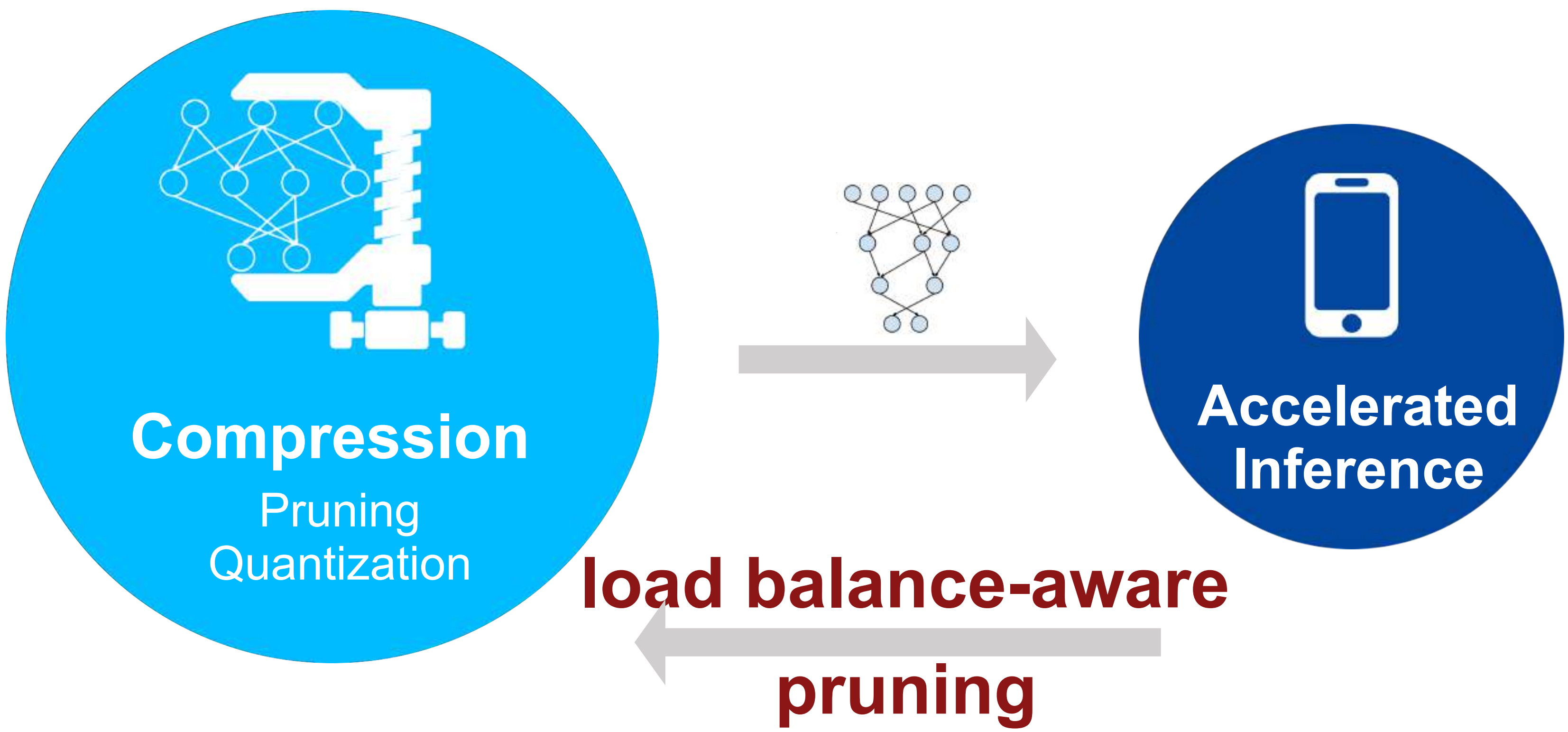
visual question answering

The recurrent nature of RNN/LSTM produces complicated <u>data dependency</u>, which is more challenging than feedforward neural nets.

Rethinking Model Compression



Rethinking Model Compression

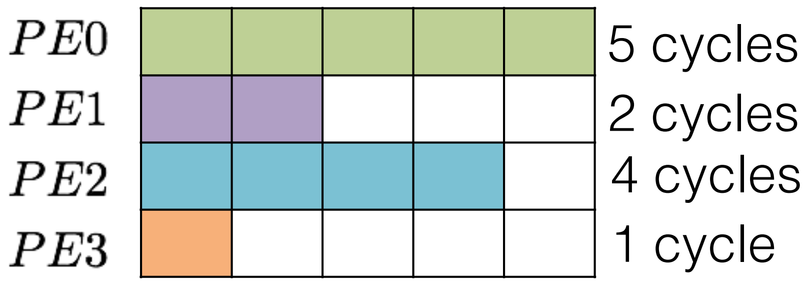


Han et, al, “ESE: Efficient Speech Recognition Engine for Compressed LSTM”, NIPS’16 workshop; FPGA’17

Pruning Lead to Load Imbalance

$$
\begin{array}{l}
PE0 \\ PE1 \\ PE2 \\ PE3 \\ \\ \\ \\ \\
\end{array}
\begin{pmatrix}
w_{0,0} & w_{0,1} & 0 & w_{0,3} \\
0 & 0 & w_{1,2} & 0 \\
0 & w_{2,1} & 0 & w_{2,3} \\
0 & 0 & 0 & 0 \\
0 & 0 & w_{4,2} & w_{4,3} \\
w_{5,0} & 0 & 0 & 0 \\
w_{6,0} & 0 & 0 & w_{6,3} \\
0 & w_{7,1} & 0 & 0
\end{pmatrix}
$$

Unbalanced



Overall: 5 cycles

Load Balance Aware Pruning

$$
\begin{array}{c}
PE0 \\ PE1 \\ PE2 \\ PE3 \\ \\ \\ \\ \\
\end{array}
\begin{pmatrix}
w_{0,0} & w_{0,1} & 0 & w_{0,3} \\
0 & 0 & w_{1,2} & 0 \\
0 & w_{2,1} & 0 & w_{2,3} \\
0 & 0 & 0 & 0 \\
0 & 0 & w_{4,2} & w_{4,3} \\
w_{5,0} & 0 & 0 & 0 \\
w_{6,0} & 0 & 0 & w_{6,3} \\
0 & w_{7,1} & 0 & 0
\end{pmatrix}
\qquad
\begin{array}{c}
PE0 \\ PE1 \\ PE2 \\ PE3 \\ \\ \\ \\ \\
\end{array}
\begin{pmatrix}
w_{0,0} & 0 & 0 & w_{0,3} \\
0 & 0 & w_{1,2} & 0 \\
0 & w_{2,1} & 0 & w_{2,3} \\
0 & 0 & w_{3,2} & 0 \\
0 & 0 & w_{4,2} & 0 \\
w_{5,0} & 0 & 0 & w_{5,3} \\
w_{6,0} & 0 & 0 & 0 \\
0 & w_{7,1} & 0 & w_{7,3}
\end{pmatrix}
$$

Unbalanced

$PE0$	5 cycles
$PE1$	2 cycles
$PE2$	4 cycles
$PE3$	1 cycle

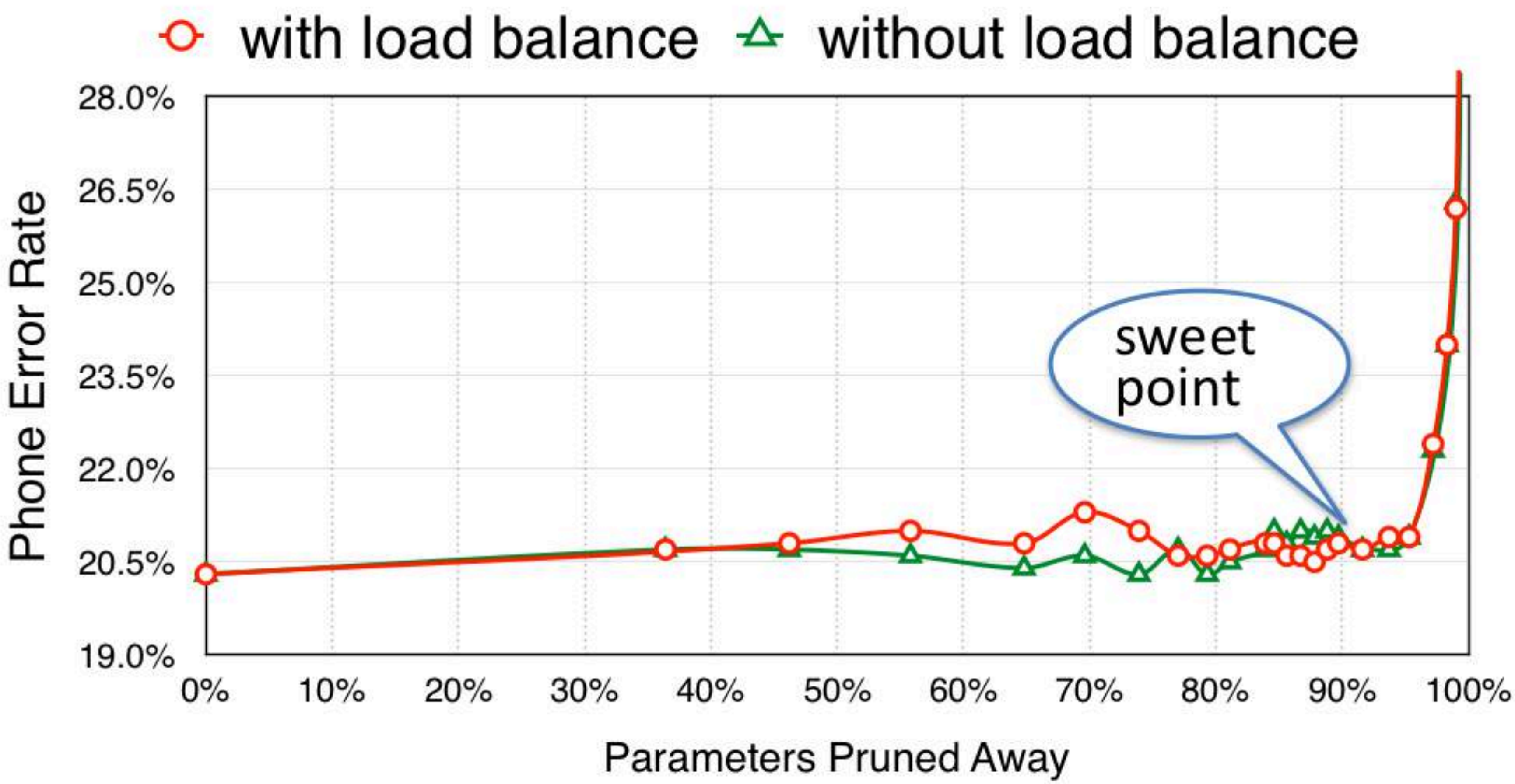
Overall: 5 cycles

Balanced

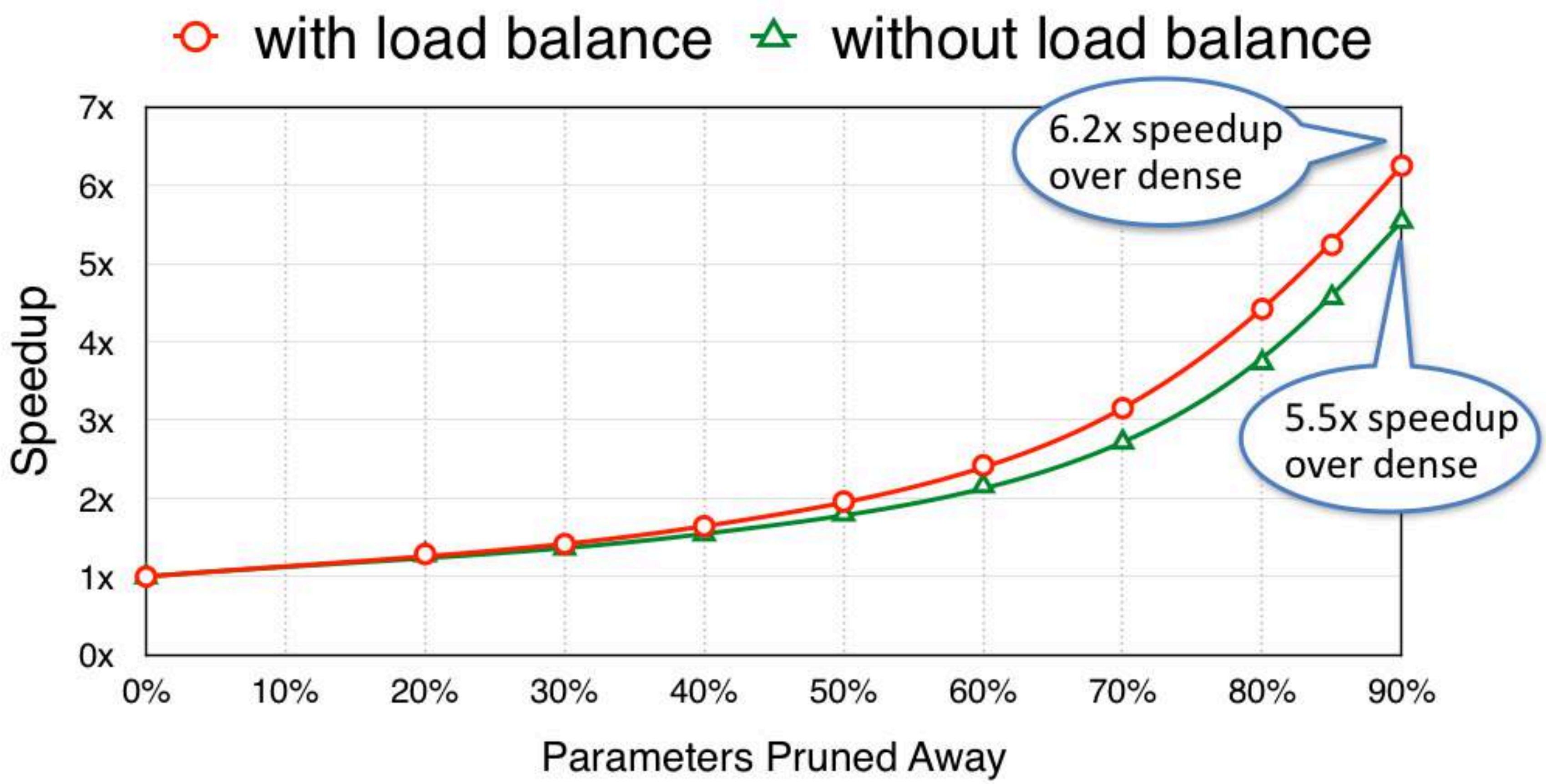
$PE0$	3 cycles
$PE1$	3 cycles
$PE2$	3 cycles
$PE3$	3 cycles

Overall: 3 cycles

Load Balance Aware Pruning: Same Accuracy



Load Balance Aware Pruning: Better Speedup



Han et, al, "ESE: Efficient Speech Recognition Engine for Compressed LSTM", NIPS'16 workshop; FPGA'17

From Compression to Acceleration

- ✦ **Challenge 1:**
  memory access is expensive.

- ✓ **Deep Compression:**
  10x-49x smaller, no loss of accuracy

- ✦ **Challenge 2:**
  sparsity, indirection, load balance.

- ✓ **EIE / ESE Accelerator:**
  energy-efficient accelerated inference

What about Training?

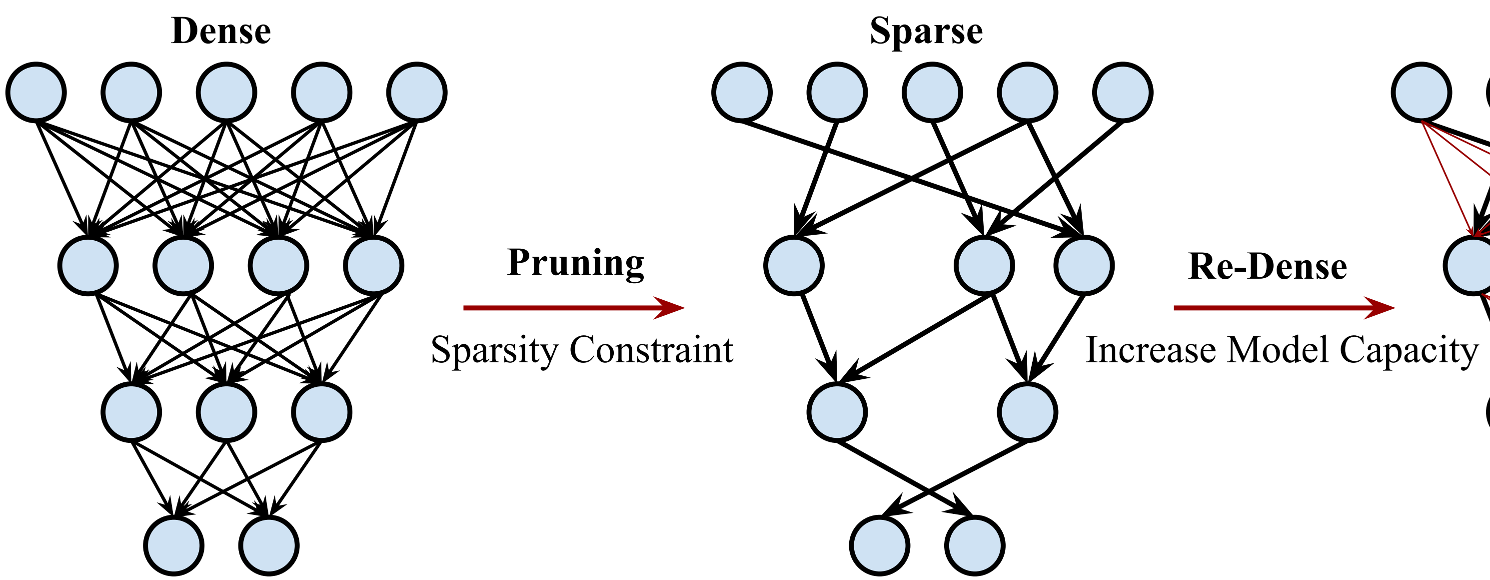
Compressed Model Size: Same accuracy

=> Original Model Size: Higher accuracy

Agenda

- **Deep Compression (size)**
  - Pruning
  - Trained Quantization
  - Huffman Coding
- **Hardware Acceleration (speed, energy)**
  - EIE Accelerator (ASIC)
  - ESE Accelerator (FPGA)
- **Efficient Training (accuracy)**
  - Dense-Sparse-Dense Regularization

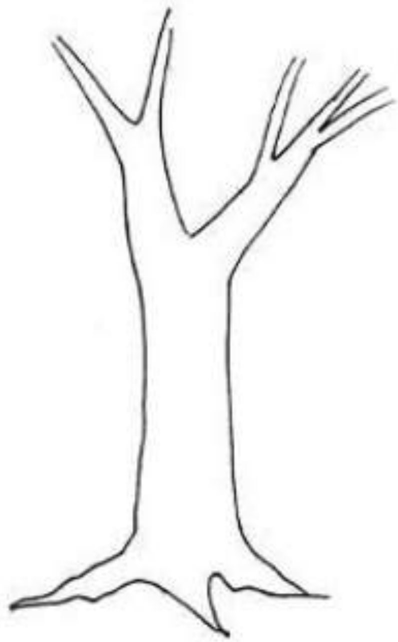
DSD: Dense Sparse Dense Training



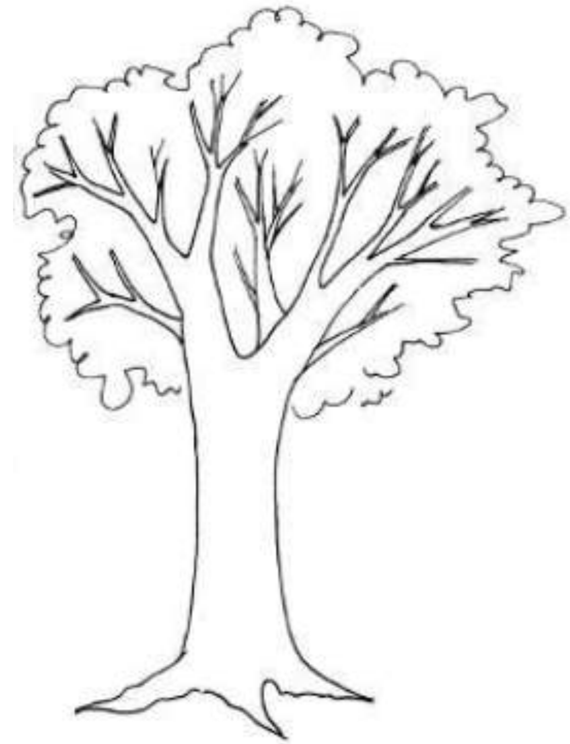
DSD produces same model architecture but can find better optimization solution, arrives at better local minima, and achieves higher prediction accuracy across a wide range of deep neural networks on CNNs / RNNs / LSTMs.

Han et al. "DSD: Dense-Sparse-Dense Training for Deep Neural Networks", ICLR 2017

DSD: Intuition



Learn the trunk first



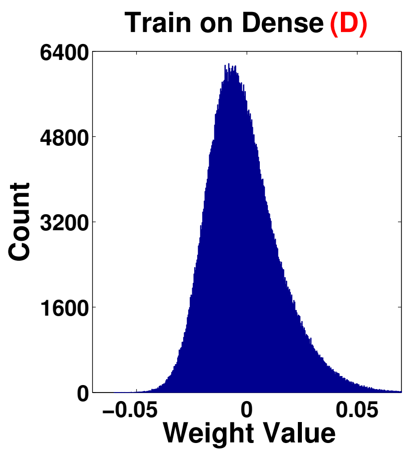
Then learn the leaves

Related Work

- **Dropout and DropConnect**
  - Dropout use a *random* sparsity pattern.
  - DSD training learns with a *deterministic* data driven sparsity pattern.

- **Distillation**
  - Transfer the knowledge from the cumbersome model to a small model
  - Both DSD and Distillation don't incur architectural changes.

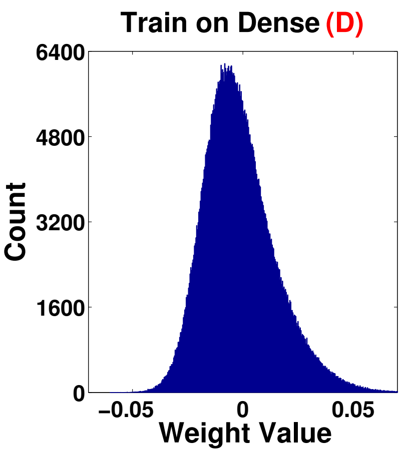
Weight Distribution



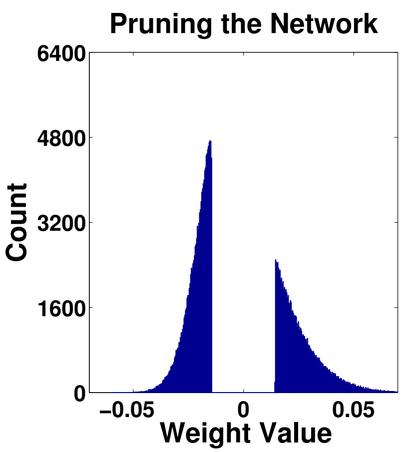
(a)

Han et al. “DSD: Dense-Sparse-Dense Training for Deep Neural Networks”, ICLR 2017

Weight Distribution



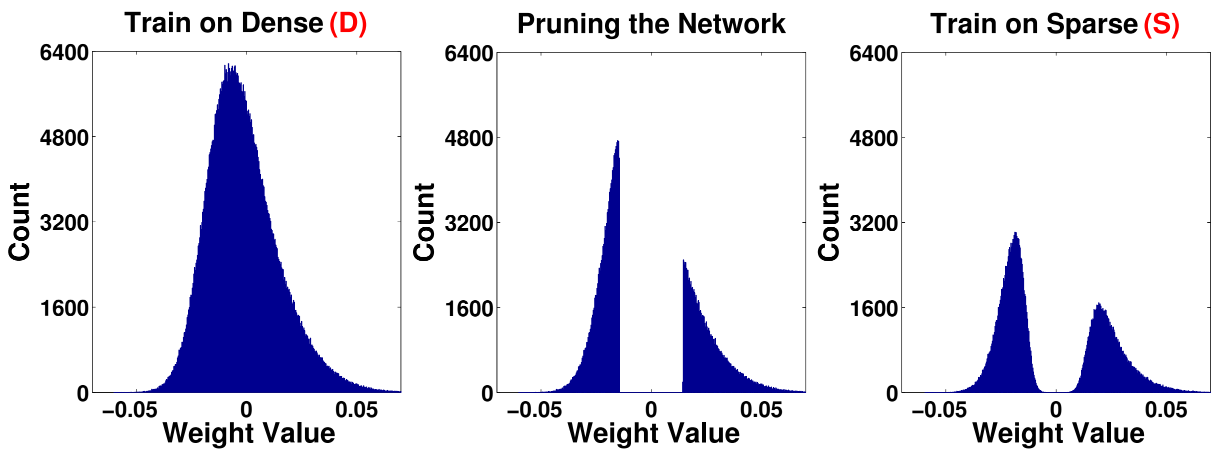
(a)



(b)

Han et al. “DSD: Dense-Sparse-Dense Training for Deep Neural Networks”, ICLR 2017

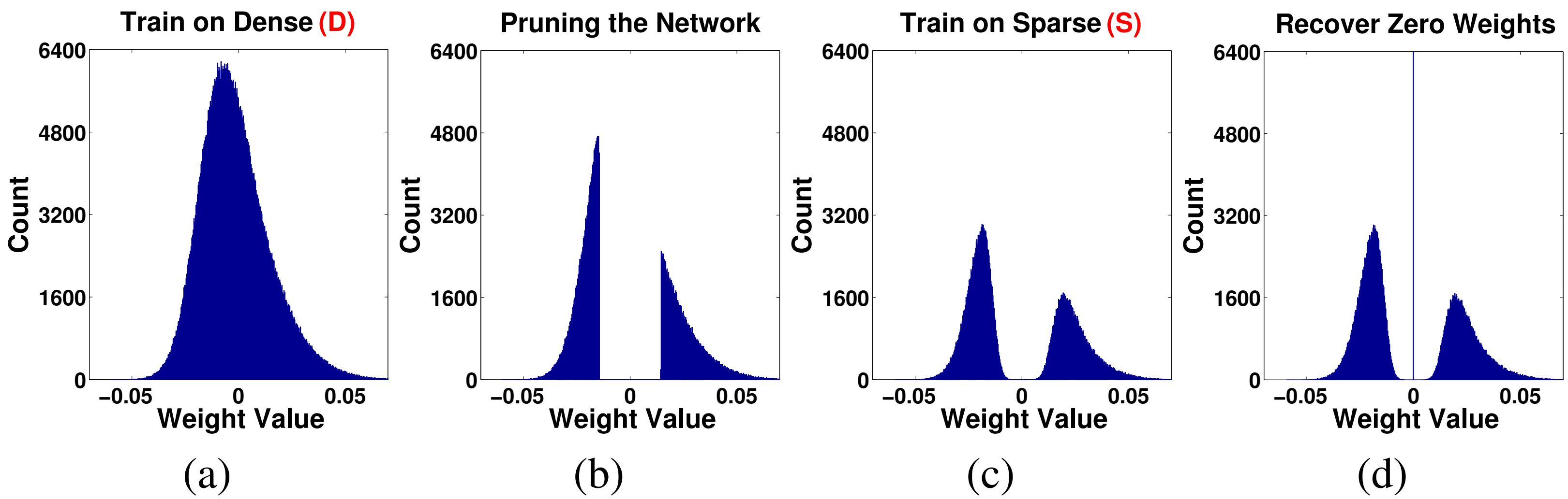
Weight Distribution



(a) (b) (c)

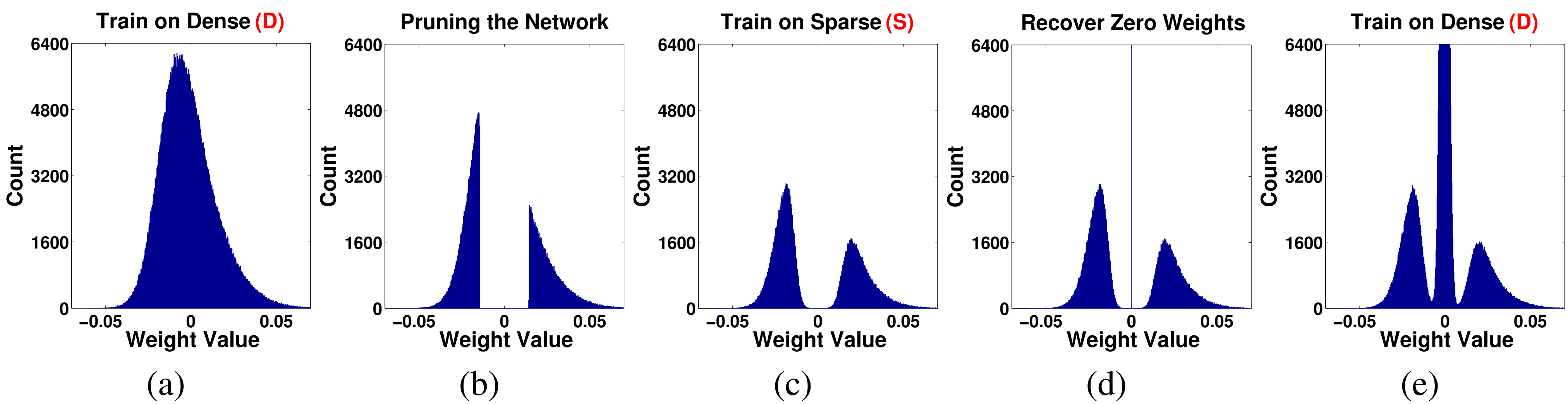
Han et al. “DSD: Dense-Sparse-Dense Training for Deep Neural Networks”, ICLR 2017

Weight Distribution



Han et al. “DSD: Dense-Sparse-Dense Training for Deep Neural Networks”, ICLR 2017

Weight Distribution



Han et al. “DSD: Dense-Sparse-Dense Training for Deep Neural Networks”, ICLR 2017

DSD is General Purpose: Vision, Speech, Natural Language

Table 1: Overview of the neural networks, data sets and performance improvements from DSD.

Neural Network	Domain	Dataset	Type	Baseline	DSD	Abs. Imp.	Rel. Imp.
GoogLeNet	Vision	ImageNet	CNN	31.1%[1]	**30.0%**	1.1%	3.6%
VGG-16	Vision	ImageNet	CNN	31.5%[1]	**27.2%**	4.3%	13.7%
ResNet-18	Vision	ImageNet	CNN	30.4%[1]	**29.3%**	1.1%	3.7%
ResNet-50	Vision	ImageNet	CNN	24.0%[1]	**23.2%**	0.9%	3.5%
NeuralTalk	Caption	Flickr-8K	LSTM	16.8[2]	**18.5**	1.7	10.1%
DeepSpeech	Speech	WSJ'93	RNN	33.6%[3]	**31.6%**	2.0%	5.8%
DeepSpeech-2	Speech	WSJ'93	RNN	14.5% [3]	**13.4%**	1.1%	7.4%

DSD Model Zoo is online: https://songhan.github.io/DSD

The beseline results of AlexNet, VGG16, GoogleNet, SqueezeNet are from Caffe Model Zoo.
The baseline results of ResNet18, ResNet50 are from fb.resnet.torch.

DSD on Caption Generation

✗ **Baseline**: a boy in a red shirt is climbing a rock wall.

✗ **Sparse**: a young girl is jumping off a tree.

✓ **DSD**: a young girl in a pink shirt is swinging on a swing.

○ **Baseline**: a basketball player in a red uniform is playing with a ball.

○ **Sparse**: a basketball player in a blue uniform is jumping over the goal.

✓ **DSD**: a basketball player in a white uniform is trying to make a shot.

✓ **Baseline**: two dogs are playing together in a field.

✓ **Sparse**: two dogs are playing in a field.

✓ **DSD**: two dogs are playing in the grass.

✗ **Baseline**: a man and a woman are sitting on a bench.

○ **Sparse**: a man is sitting on a bench with his hands in the air.

○ **DSD**: a man is sitting on a bench with his arms folded.

✗ **Baseline**: a person in a red jacket is riding a bike through the woods.

✓ **Sparse**: a car drives through a mud puddle.

✓ **DSD**: a car drives through a forest.

Baseline model: Andrej Karpathy, Neural Talk model zoo.
Han et al. “DSD: Dense-Sparse-Dense Training for Deep Neural Networks”, ICLR 2017

DSD on Caption Generation

Baseline: a boy is swimming in a pool.
Sparse: a small black dog is jumping into a pool.
DSD: a black and white dog is swimming in a pool.

Baseline: a group of people are standing in front of a building.
Sparse: a group of people are standing in front of a building.
DSD: a group of people are walking in a park.

Baseline: two girls in bathing suits are playing in the water.
Sparse: two children are playing in the sand.
DSD: two children are playing in the sand.

Baseline: a man in a red shirt and jeans is riding a bicycle down a street.
Sparse: a man in a red shirt and a woman in a wheelchair.
DSD: a man and a woman are riding on a street.

Baseline: a group of people sit on a bench in front of a building.
Sparse: a group of people are standing in front of a building.
DSD: a group of people are standing in a fountain.

Baseline: a man in a black jacket and a black jacket is smiling.
Sparse: a man and a woman are standing in front of a mountain.
DSD: a man in a black jacket is standing next to a man in a black shirt.

Baseline: a group of football players in red uniforms.
Sparse: a group of football players in a field.
DSD: a group of football players in red and white uniforms.

Baseline: a dog runs through the grass.
Sparse: a dog runs through the grass.
DSD: a white and brown dog is running through the grass.

Han et al. "DSD: Dense-Sparse-Dense Training for Deep Neural Networks", ICLR 2017

Baseline model: Andrej Karpathy, Neural Talk model zoo.

Summary

✦ **Deep Compression (size)**
- Pruning
- Trained Quantization
- Huffman Coding

✦ **Hardware Acceleration (<u>speed, energy</u>)**
- EIE Accelerator (ASIC)
- ESE Accelerator (FPGA)

✦ **Efficient Training (<u>accuracy</u>)**
- Dense-Sparse-Dense Regularization

Summary

Training

Inference

Summary

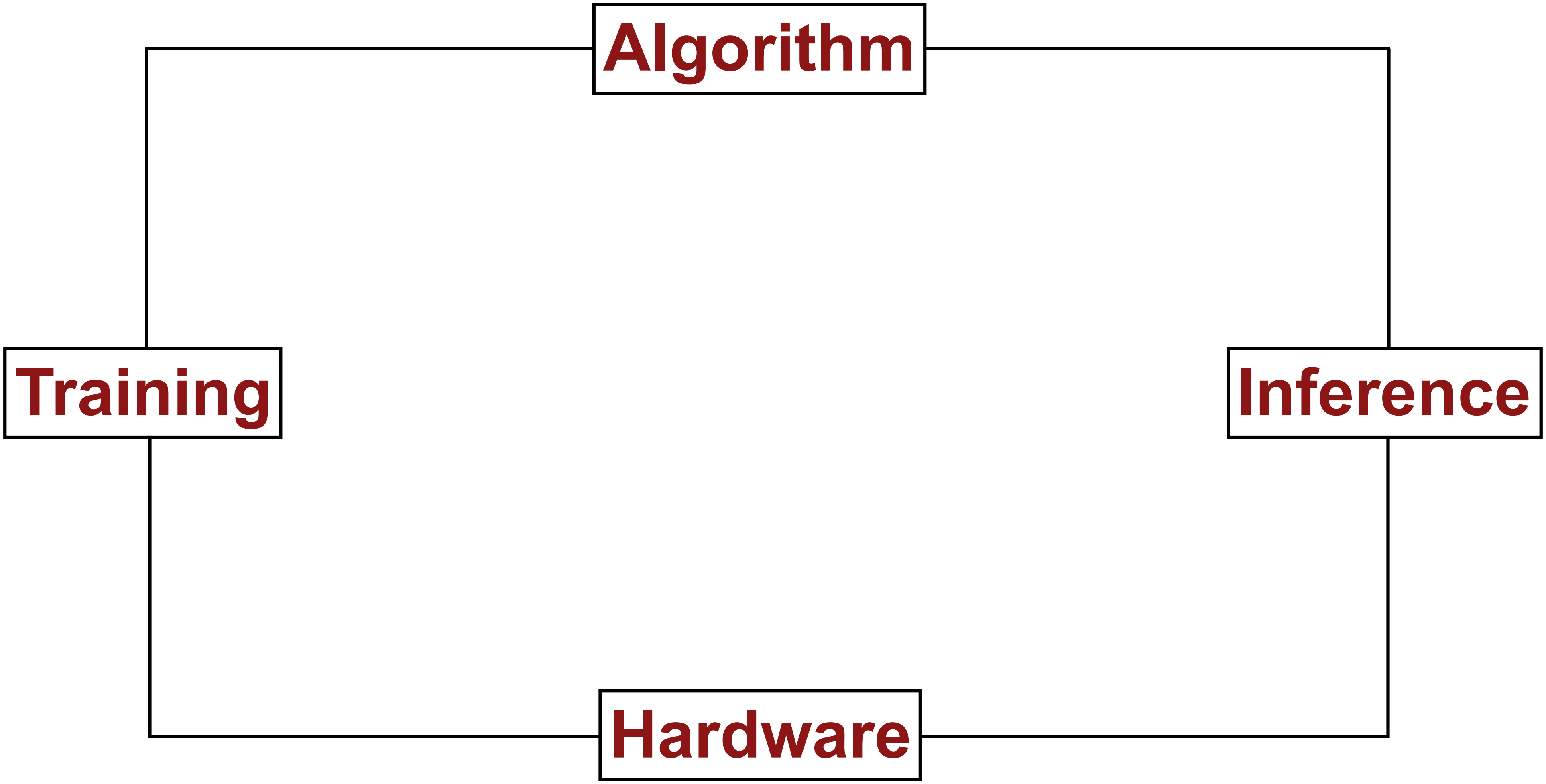
Algorithm

Training

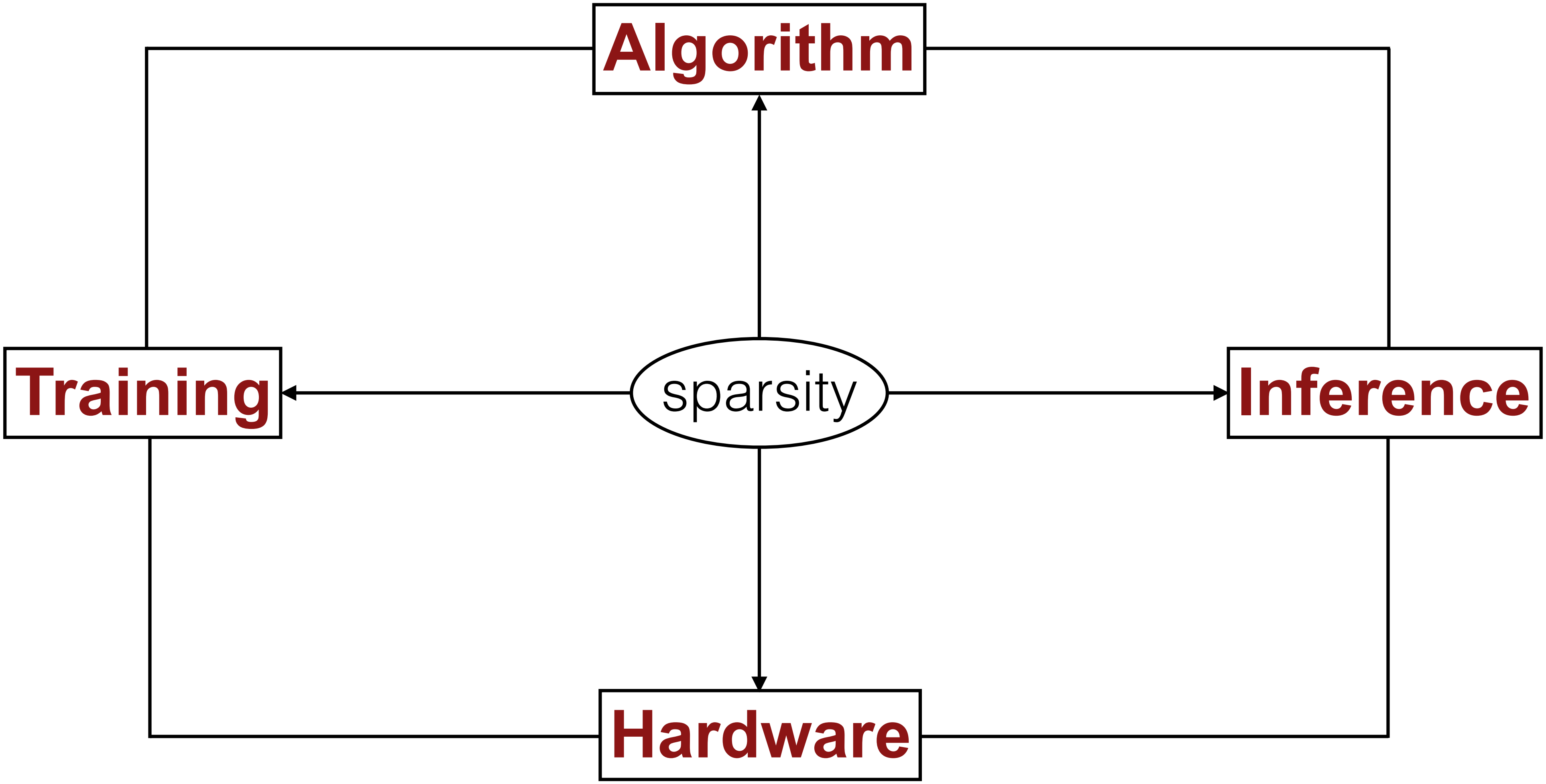
Inference

Hardware

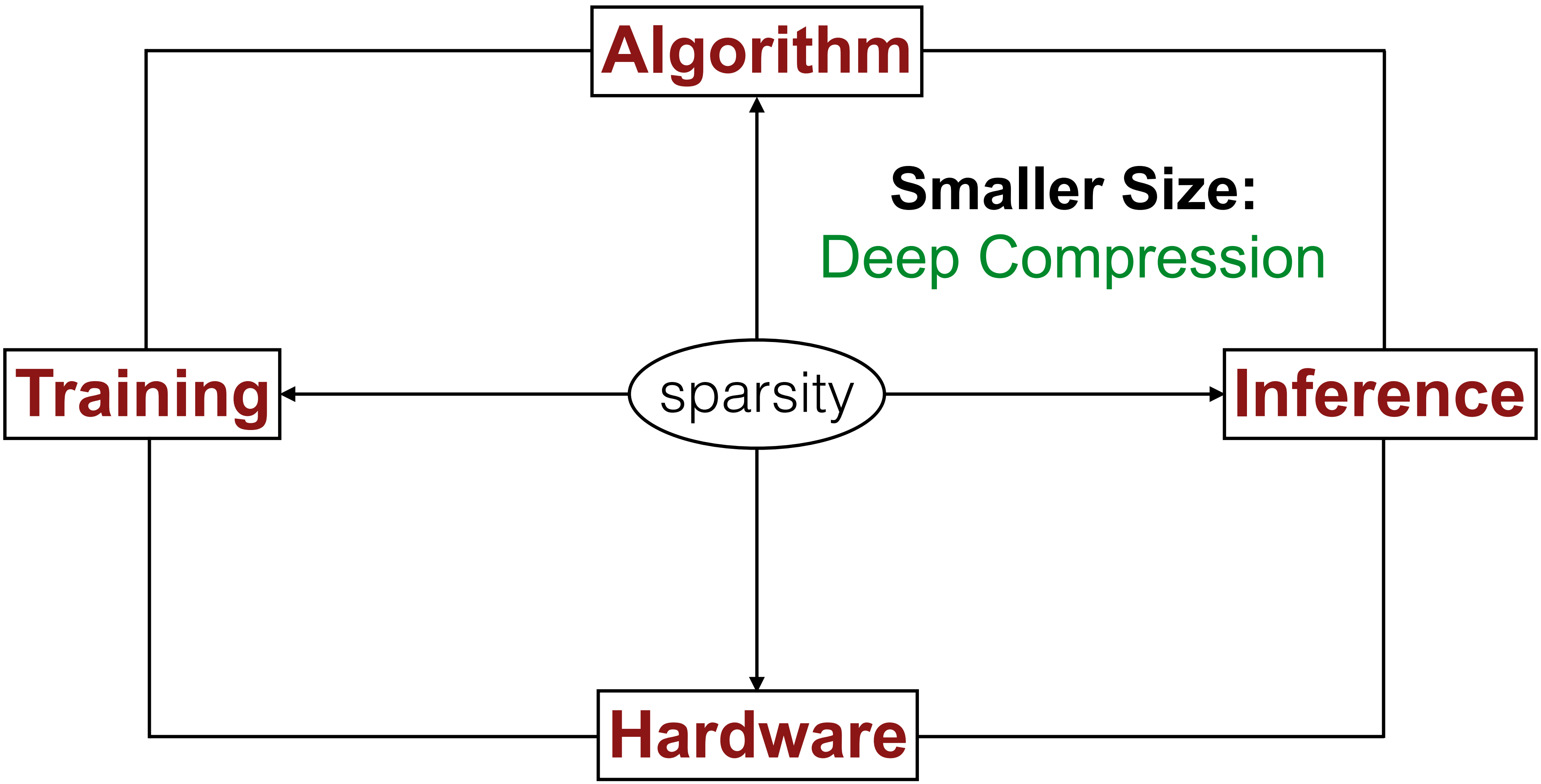
Summary



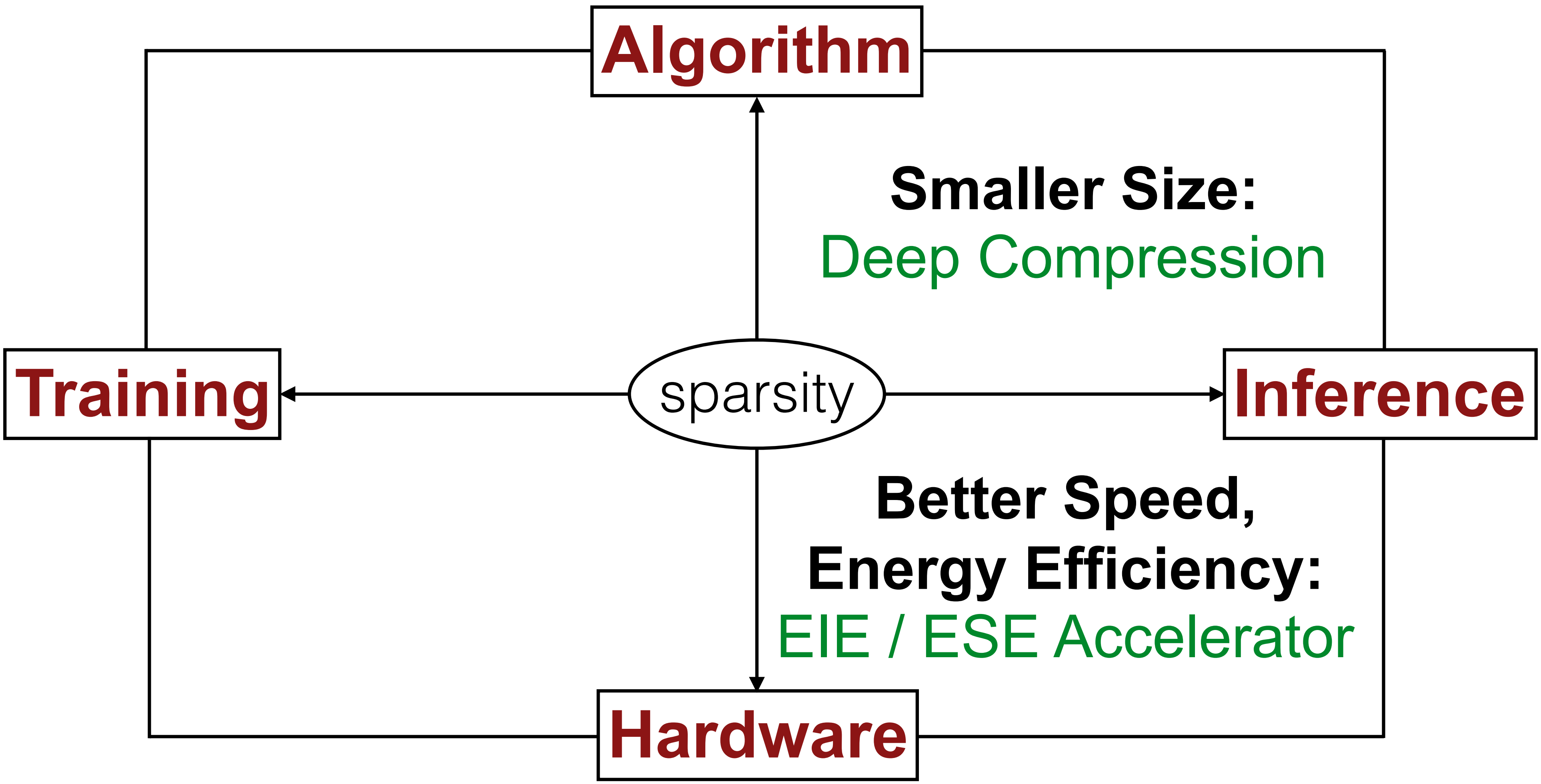
Summary



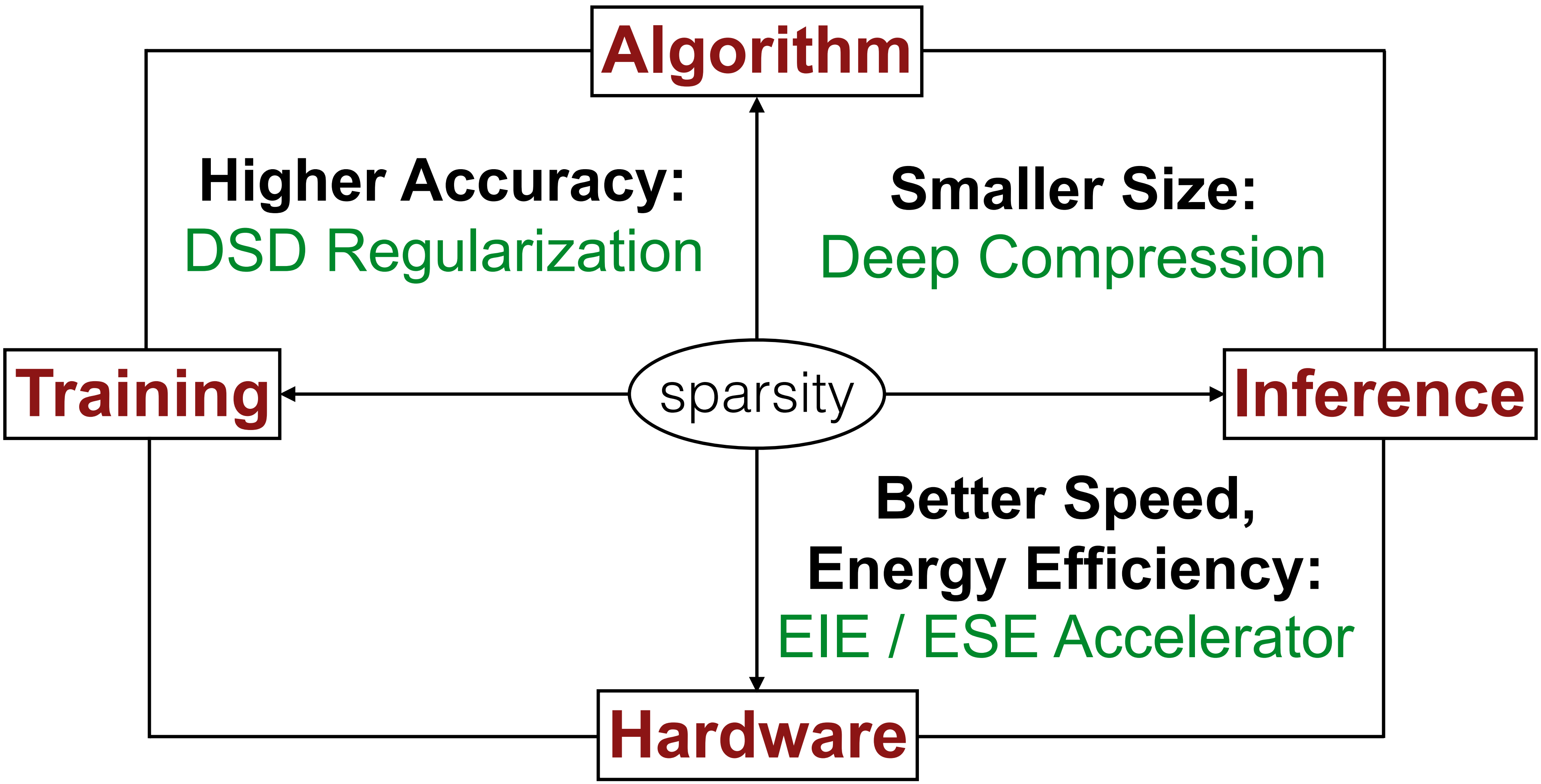
Summary


Algorithm
Smaller Size:
Deep Compression
Training
sparsity
Inference
Hardware

Summary


Algorithm
Smaller Size:
Deep Compression
Training
sparsity
Inference
Better Speed,
Energy Efficiency:
EIE / ESE Accelerator
Hardware

Summary


Algorithm
Higher Accuracy:
DSD Regularization
Smaller Size:
Deep Compression
Training
sparsity
Inference
Better Speed,
Energy Efficiency:
EIE / ESE Accelerator
Hardware

Detection with Low Precision

Discover the philisophy behind
DEEP LEARNING

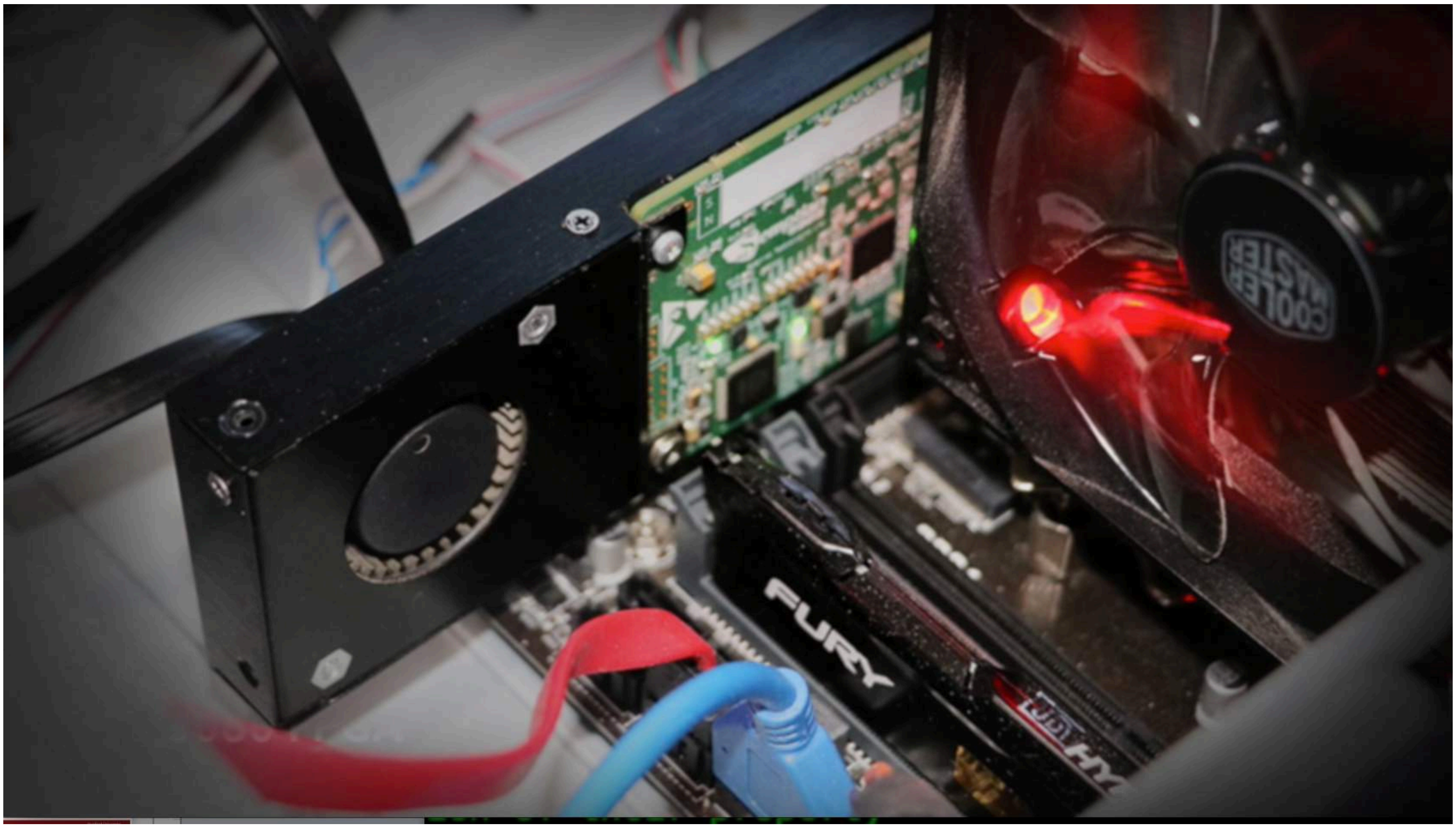
DEEPHi

ESE for Speech Recognition

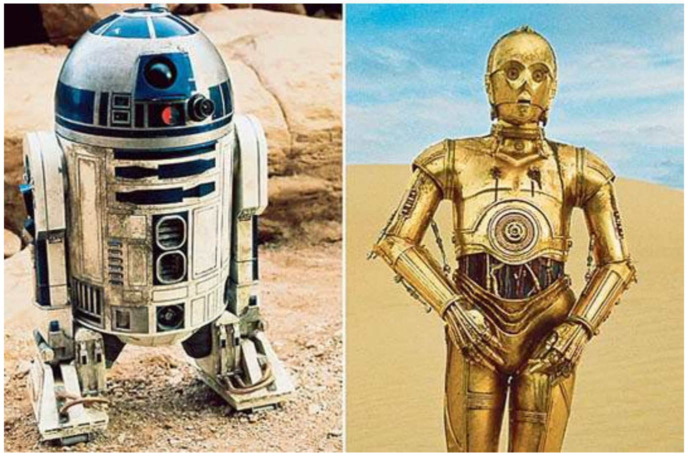
Discover the philisophy behind DEEP LEARNING

DEEPHi

Efficient Speech Recognition Engine on Sparse LSTM



Outlook: the Path for Computation



PC | **Mobile-First** | **AI-First**

Computation → Mobile Computation → Cognitive Computation

Sundar Pichai, Google IO, 2016

Thank you!

stanford.edu/~songhan

Model Compression

[1]. Han et al. “Learning both Weights and Connections for Efficient Neural Networks”, NIPS 2015

[2]. Han et al. “Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding”, Deep Learning Symposium, NIPS 2015; ICLR 2016, **(best paper award)**

[3]. Chen, Han, et.al, “Trained Ternary Quantization”, ICLR 2017

Model Regularization

[3]. Han et al. “DSD: Regularizing Deep Neural Networks with Dense-Sparse-Dense Training ”, ICLR 2017

Hardware Acceleration

[6]. Han et al. “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016

[7]. Han et al. “ESE: Efficient Speech Recognition Engine for Compressed LSTM”, NIPS’16 workshop; FPGA 2017

[8]. Guo et al. “Angel-Eye: A Complete Design Flow for Mapping CNN onto Customized Hardware”, ISVLSI 2016

[9]. Guo, Han et al. “Software-Hardware Co-Design for Efficient Neural Network Acceleration”, IEEE Micro, 2017

CNN Design Space Exploration

[4]. Iandola, Han, et al. “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size” arXiv’16

[5]. Yao, Han, et al. “Hardware-friendly convolutional neural network with even-number filter size” ICLR 2016 workshop