Outlier Channel Splitting

Improving DNN Quantization without Retraining

Ritchie Zhao, Yuwei Hu, Jordan Dotzel,
Christopher De Sa, Zhiru Zhang

School of Electrical and Computer Engineering
Cornell University

Cornell University

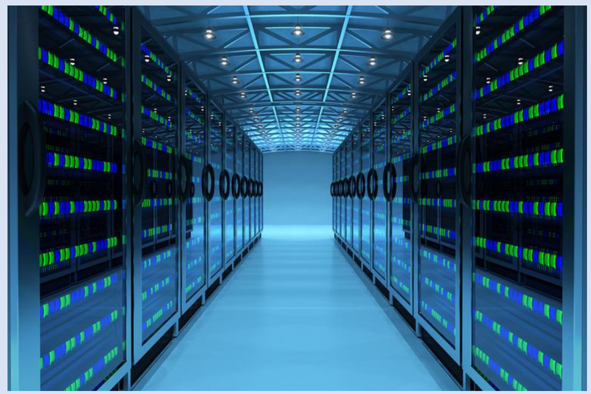
CSL

Specialized DNN Processors are Ubiquitous

Mobile

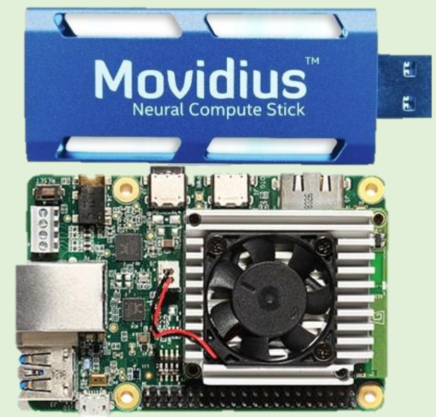
Apple (A12)
Samsung (Exynos 9820)
Huawei (Kirin 970)
Qualcomm (Hexagon)

Cloud



Google (TPU)
Microsoft (Brainwave)
Xilinx (EC2 F1)
Intel (FPGAs, Nervana)
AWS Offerings

Embedded



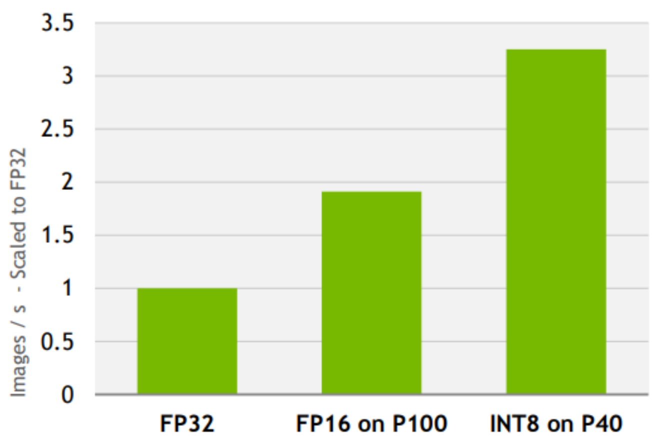
Google (Edge TPU)
Intel (Movidius)
Deephi/Xilinx (Zynq)
ARM (announced)
Many Startups

Quantization is Key to Hardware Acceleration

Lower Precision → less energy and area per op
→ fewer bits of storage per data

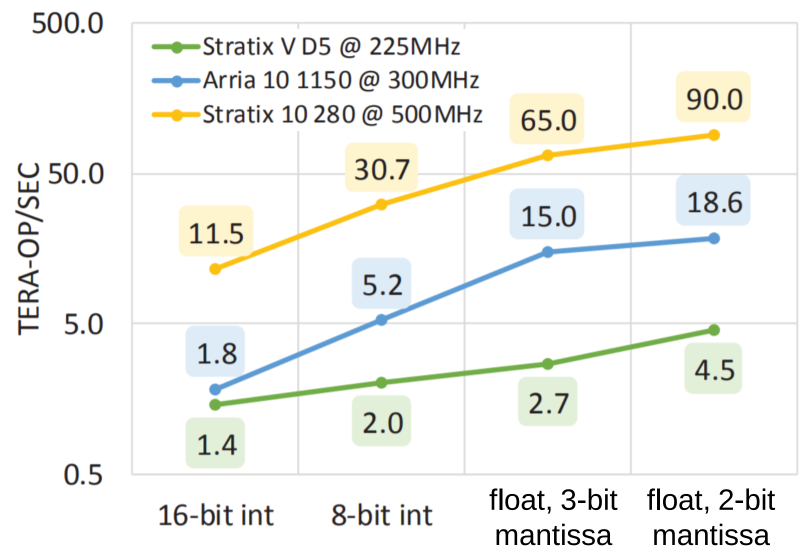
GPU Performance

ResNet-50



https://developer.nvidia.com/tensorrt

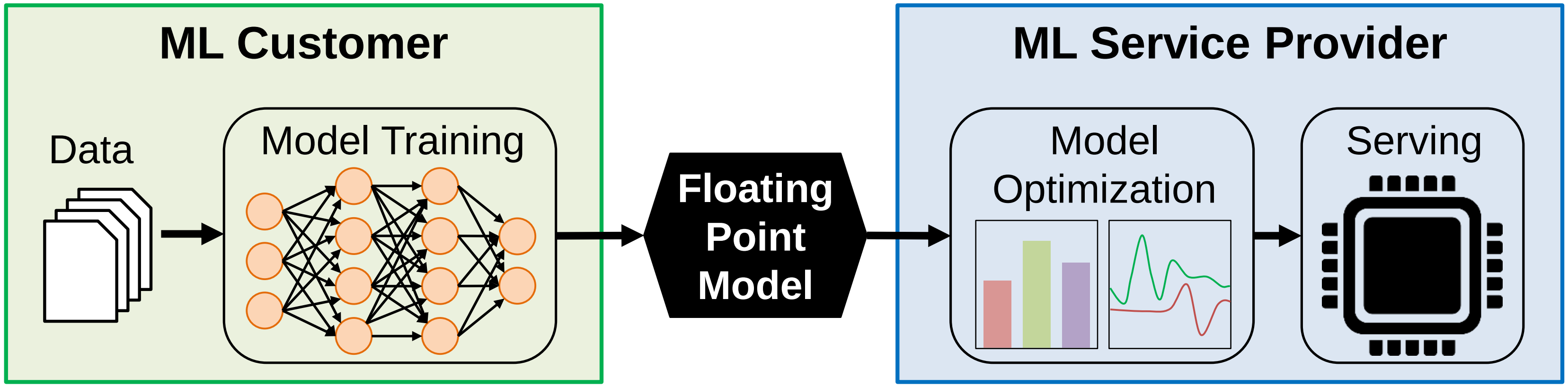
FPGA Performance



E. Chung, J. Fowers et al. **Serving DNNs in Real Time at Datacenter Scale with Project Brainwave**, *IEEE Micro*, April 2018.

Data-Free Quantization

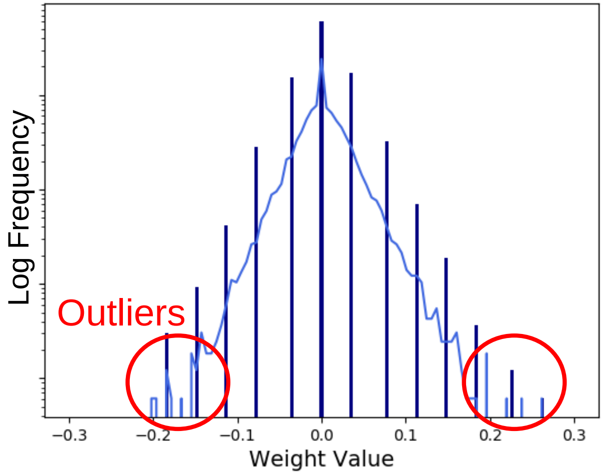
- DNN quantization techniques that require training are discouraged by the current ML service model



- Reasons to prefer **data-free quantization**:
  1. ML providers typically cannot access customer training data
  2. Customer is using a pre-trained **off-the-shelf model**
  3. Customer is unwilling to retrain a **legacy model**
  4. Customer **lacks the expertise** for quantization training

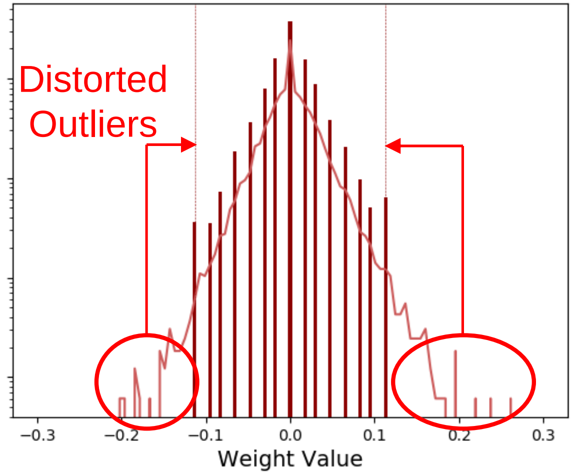
Paper Summary

Baseline
Linear Quantizer



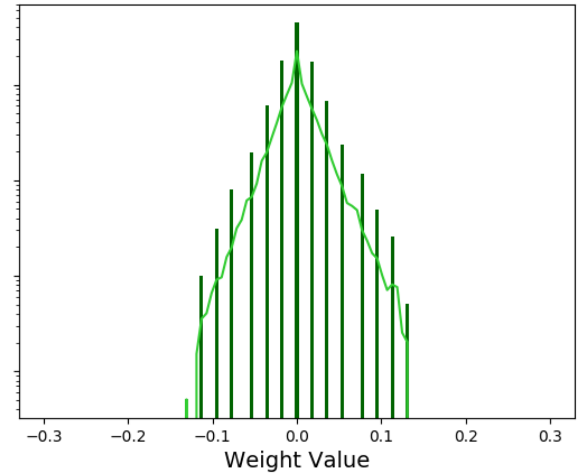
− Poor quantizer resolution due to outliers

Prior Art
Clipping



+ Reduces quantization noise
+ Used in NVIDIA TensorRT
− Distorts outliers

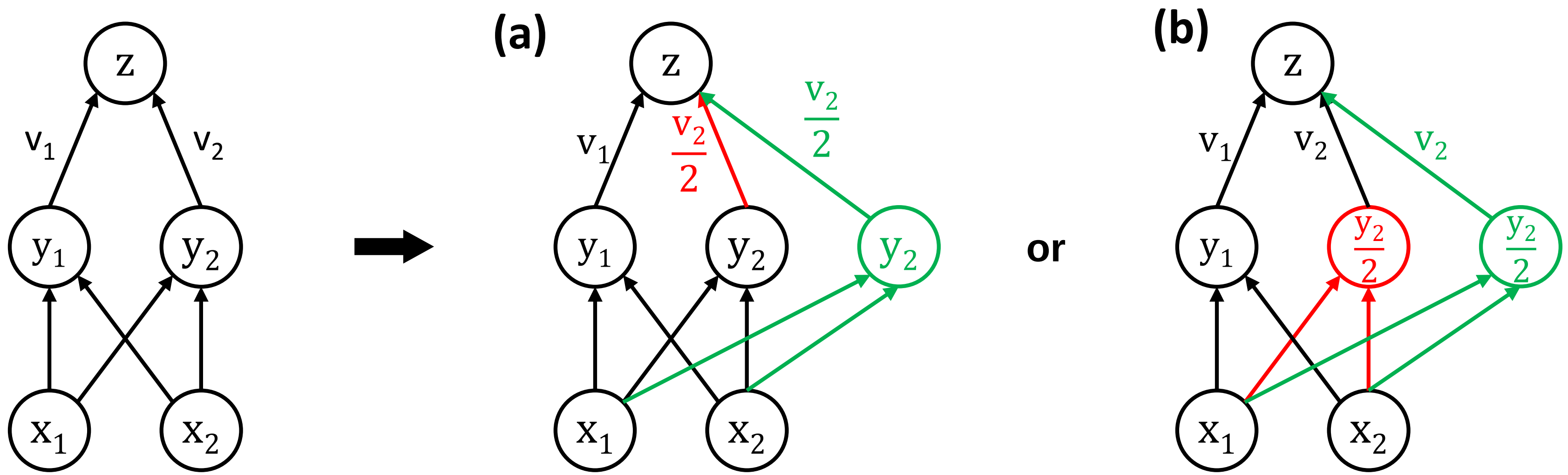
Our Method
Outlier Channel Splitting



+ Reduces quantization noise
+ Removes outliers
− Model size overhead

- OCS improves quantization without retraining
- OCS can outperform existing methods with negligible size overhead (<2%) in both CNNs and RNNs
- We also perform a comprehensive evaluation of different clipping methods in literature

Outlier Channel Splitting



$z = v_1 y_1 + v_2 y_2$

$z = v_1 y_1 + \frac{v_2}{2} y_2 + \frac{v_2}{2} y_2$

$z = v_1 y_1 + v_2 \frac{y_2}{2} + v_2 \frac{y_2}{2}$

- OCS **splits** weights or activations, halving them
  - **(a)** Duplicate node y_2 to halve the weight v_2
  - **(b)** Duplicate weight v_2 to halve the activation y_2
  - Inspired by *Net2Net*, a paper on layer transformations

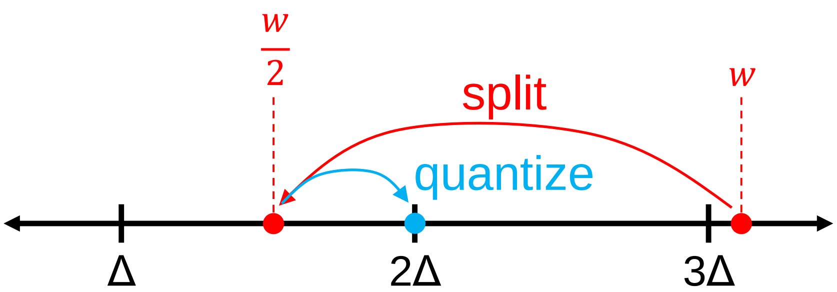
T. Chen, I. Goodfellow, J. Shlens, **Net2Net: Accelerating Learning via Knowledge Transfer**. *ICLR'16*, May 2016.

Quantization-Aware Splitting

Naïve Splitting (*Net2Net*)

$w \to (\frac{w}{2}, \frac{w}{2})$

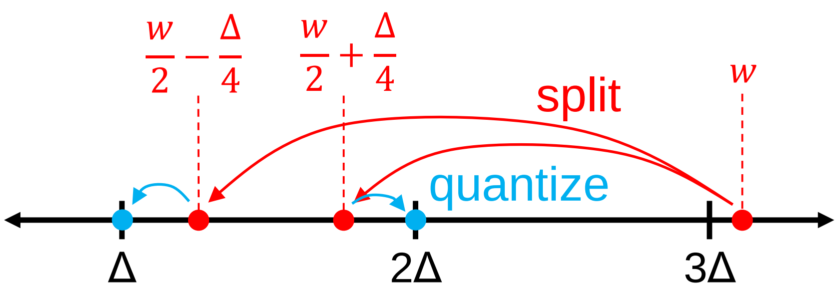
Halves round in the same direction



Quantization-Aware Splitting

$w \to (\frac{w}{2} - \frac{\Delta}{4}, \frac{w}{2} + \frac{\Delta}{4})$

Halves can round in opposite directions to help cancel out quantization noise



- In the paper, we show that QA splitting preserves the expected quantization noise on a single value

Results on CNNs

In these results OCS is constrained to ~2% size overhead.

Blue = +1% or better
Red = −1% or worse

Network (Float Acc.)	Wt. Bits	Quantized Acc. (± vs. Best Clipping Result)	
		OCS	**OCS + Clip**
VGG-16 BN (73.4)	6	+1.0	+0.5
	5	+3.3	+2.6
	4	−33.1	+4.4
ResNet-50 (76.1)	6	+0.4	+0.5
	5	+2.0	+2.0
	4	−26.8	+4.2
DenseNet-121 (74.4)	6	+1.6	+1.7
	5	+4.3	+5.3
	4	−5.1	+13.9
Inception-V3 (75.9)	6	+5.6	+5.5
	5	+13.5	+19.5
	4	−1.4	+0.7

- **OCS constrained to 2% overhead** outperforms Clipping at 6-5 bits
- **OCS + Clipping** outperforms Clipping alone at 4 bits

Thank you!

Ritchie Zhao, Yuwei Hu, Jordan Dotzel, Zhiru Zhang. Improving Neural Network Quantization without Retraining using Outlier Channel Splitting. *ICML*, June 2019

Code available at:
https://github.com/cornell-zhang/dnn-quant-ocs