Towards Cheaper Inference in Deep Networks with Lower Bit-Width Accumulators

YANIV BLUMENFELD, ITAY HUBARA, DANIEL SOUDRY

International Conference On Learning Representation- 2024

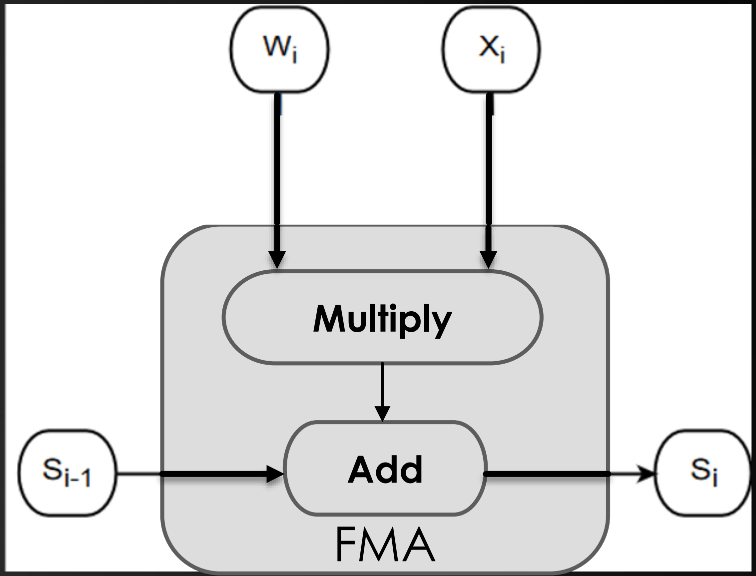
Making Neural Networks More Efficient

- There are many existing methods to make neural networks more hardware efficient using **quantization**-- of weights, activation and gradients.

- Quantization helps reducing the cost of the Fused-Multiply-Add (**FMA**) operation, by reducing the size of its inputs.

- In this work, we will focus on the **internal parts of the FMA** computation, (like **product accumulation)**, and propose a way to make them more efficient in hardware.

- **Our goal**: Allow inference for models using low-bit-accumulators (LBAs) without loss of accuracy.



Fine-Tuning for Inference with 12-Bits LBAs

Core Ideas:

- Use Floating-Point Representation, with non-trivial exponent-biases.
- Start from a pretrained, full-precision model.
- Dual-Stage finetuning, with different treatment for different types of quantization error.

Resnet: FP8 W/A Quantization

Model	Data Type	Weights	Activations	Accumulator	Top-1 Accuracy
ResNet18					
Baseline	FP	32	32	32	70.23%
Baseline (FP8)	FP	8	8	32	69.90%
Wang et al. (2018)	FP	8	8	16	66.95%
Ni et al. (2020)	INT	7	2	12	63.84%
Ours (1-stage)	FP	8	8	12	69.54%
Ours (dual-stage)	FP	8	8	12	69.70%
ResNet34					
Baseline	FP	32	32	32	73.87%
Baseline (FP8)	FP	8	8	32	73.49%
Ours (1-stage)	FP	8	8	12	73.18%
Ours (dual-stage)	FP	8	8	12	73.42%
ResNet50					
Baseline	FP	32	32	32	76.80%
Baseline (FP8)	FP	8	8	32	76.25%
Wang et al. (2018)	FP	8	8	16	71.72%
Ours (1-stage)	FP	8	8	12	74.15%
Ours (dual-stage)	FP	8	8	12	76.22%

Resnet: No W/A Quantization

Model	Baseline	1-stage	no UF*	no UF → with UF
ResNet18	70.23%	69.94%	70.01%	70.06%
ResNet34	73.87%	73.64%	73.61%	73.45%
ResNet50	76.80%	74.70%	76.60%	76.40%

	Baseline		LBA ($M7E4$) b_{acc},b_{prod}=7,9		LBA ($M7E4$) b_{acc},b_{prod}=8,10	
Model	Exact (%)	f1 (%)	Exact (%)	f1 (%)	Exact (%)	f1 (%)
Bert-Small	71.32	80.96	70.88	80.24	71.35	80.59
Bert-Base	79.84	87.53	79.60	87.62	79.80	87.52
Bert-Large	83.22	90.40	82.97	89.97	83.25	90.66

Below 12 Bits

- The previous method doesn't work with less than 12 bits.
- The culprit: For extreme quantization, the naïve Gradient estimator we used is too ``far away''.
- Our solution: A novel implementation for Straight Through Estimator (STE), which is **internal to the FMA computation graph.**
- We suggest several alternatives for estimating the gradients in this setup.
- We implement this method when training transformers, closing much of the gap with full-precision training.

MNIST with Naïve STE:

LBA Format	Underflow	Top-1 Accuracy
FP32	-	98.64%
M6E3	Yes	42.28%
M4E3	Yes	18.28%
M4E3	No	18.28%

MNIST with M4E3 accumulation

STE Type	Underflow	Top-1 Accuracy
IM/OF	Yes	98.47%
IM/DIFF	Yes	11.35%
IM/DIFF	NO	97.67%
R/OF	Yes	98.46%

Summary

- Modern Neural Networks can be finetuned to operate with lower cost FMA hardware, with relative ease.

- Accumulation precision can be reduced to 12 bits without accuracy degradation

- For a 12 bits setup, a dedicated hardware can reduce the cost of inference by approximately 63%. (Based on Gate-Count analysis, included in the paper)

- Going below that would require a more careful approach, that utilizes special gradient computation kernels with dedicated STEs.

Thank You