Algorithm-Hardware Co-Design of Adaptive Floating-Point Encodings for Resilient Deep Learning Inference

Thierry Tambe[1], En-Yu Yang[1], Zishen Wan[1], Yuntian Deng[1]
Vijay Janapa Reddi[1], Alexander Rush[2], David Brooks[1], Gu-Yeon Wei[1]
[1]Harvard University, Cambridge, MA, [2]Cornell University, New York, NY

A Startling Observation

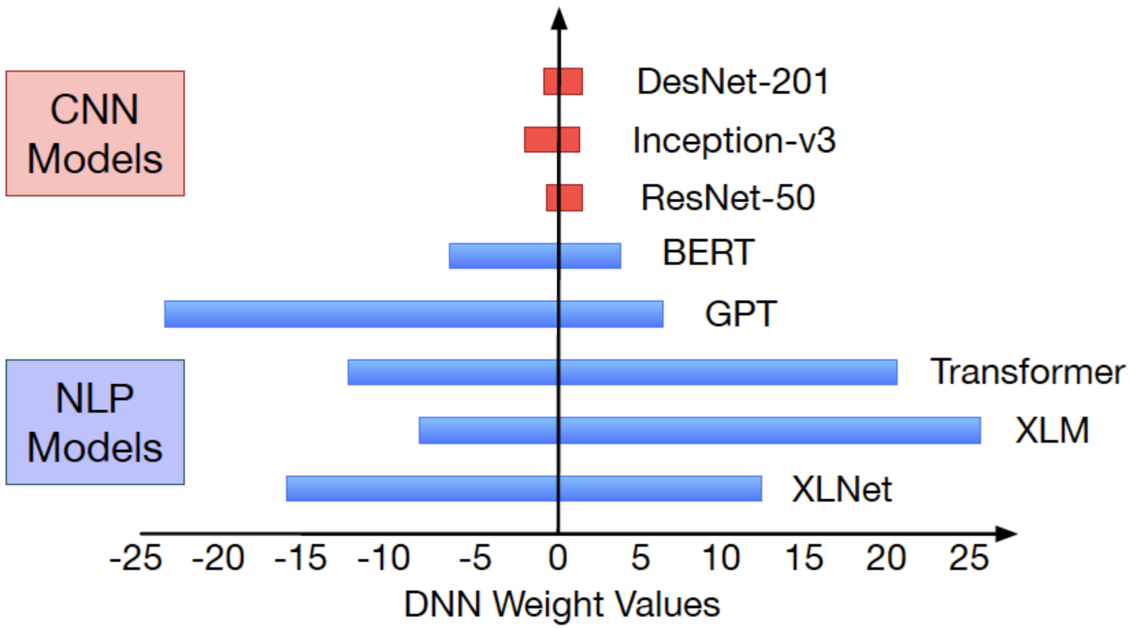
There is a plethora of DNN quantization techniques out there

- Most are fixed-point based
- Evaluated solely on CNNs such as ResNet

A Startling Observation

There is a plethora of DNN quantization techniques out there

- Most are fixed-point based
- Evaluated solely on CNNs such as ResNet



- Perform poorly on models with wide weight distribution such as in NLP
  - Due to an inherent lack of dynamic range

This work

A generalized DNN numerical encoding blueprint, AdaptivFloat, that is:

- Adaptive to the statistical distribution of the DNN parameters
- Resilient to aggressive bit width compression
- Hardware-friendly with low energy overheads

This work

A generalized numerical DNN encoding blueprint, AdaptivFloat, that is:

Adaptive to the statistical distribution of the DNN parameters

Resilient to aggressive bit width compression

Hardware-friendly with low energy overheads

The AdaptivFloat Algorithm

W_{fp} : full precision weight matrix

-1.17	2.71	-1.60	0.43
-1.14	2.05	1.01	0.07
0.16	-0.03	-0.89	-0.87
-0.04	-0.39	0.64	-2.89

max

Find exp_{bias} to fit max absolute value of W_{fp} and get representable datapoints

Adaptivfloat params

$exp_{max} = -2$

$abs(\mathrm{min}) = 0.375$

$abs(\mathrm{max}) = 3$

+0	-0
+0.375	-0.375
+0.5	-0.5
+0.75	-0.75
+1	-1
+1.5	-1.5
+2	-2
+3	-3

Get quantized $W_{adaptiv}$ by rounding to nearest datapoints

-1	3	-1.5	0.375
-1	2	1	0
0	-0	-1	-0.75
-0	-0.375	0.75	-3

- Floating-point based
- Performed at a per-layer granularity
- Maximizes dynamic range by formulating an exponent bias, exp_{bias}, from maximum absolute tensor value
    - Then uses exp_{bias} to shift the exponent range of datapoints

Handling Denormals

Floating points w/o denormals

+0.25	-0.25
+0.375	-0.375
+0.5	-0.5
+0.75	-0.75
+1	-1
+1.5	-1.5
+2	-2
+3	-3

Floating points w/o denormals, but sacrifice ±min for ±0

+0	-0
+0.375	-0.375
+0.5	-0.5
+0.75	-0.75
+1	-1
+1.5	-1.5
+2	-2
+3	-3

- AdaptivFloat break from IEEE754 standard compliance by not encoding floating-point denormals → leaner hardware design
  - We sacrifice the positive and negative minimum representable datapoints to allocate for the "zero" slot
- AdaptivFloat clamps unrepresentable small and large values

AdaptivFloat is Lightweight and Self-Supervised

Algorithm 1: AdaptivFloat Quantization

Input: Matrix W_{fp}, bitwidth n and number of exponent bits e

`// Get Mantissa bits`

$m := n - e - 1$

`// Obtain sign and abs matrices`

$W_{sign} := sign(W_{fp})$; $W_{abs} := abs(W_{fp})$

`// Determine` exp_{bias} `and range`

Find normalized exp_{max} for $max(W_{abs})$ such that

$2^{exp_{max}} \leq max(W_{abs}) < 2^{exp_{max}+1}$

$exp_{bias} := exp_{max} - (2^e - 1)$

$value_{min} := 2^{exp_{bias}} * (1 + 2^{-m})$

$value_{max} := 2^{exp_{max}} * (2 - 2^{-m})$

`// Handle unrepresentable values`

Round $value < value_{min}$ in W_{abs} to 0 or $value_{min}$

Clamp $value > value_{max}$ in W_{abs} to $value_{max}$

`// Quantize` W_{fp}

Find normalized W_{exp} and W_{mant} such that

$W_{abs} = 2^{W_{exp}} * W_{mant}$, and $1 \leq W_{mant} < 2$

$W_q :=$ quantize and round W_{mant} by $scale = 2^{-m}$

`// Reconstruct output matrix`

$W_{adptiv} := W_{sign} * 2^{W_{exp}} * W_q$

return W_{adptiv}

- Relies only on the unlabeled data distributions in the network
- Can be easily plugged into any ML framework at learning or inference time
- User just needs to provide the input tensor, and the required word size and exponent bit width

This work

A generalized numerical DNN encoding blueprint, AdaptivFloat, that is:

Adaptive to the statistical distribution of the DNN parameters

Resilient to aggressive bit width compression

Hardware-friendly with low energy overheads

Experimental Setup

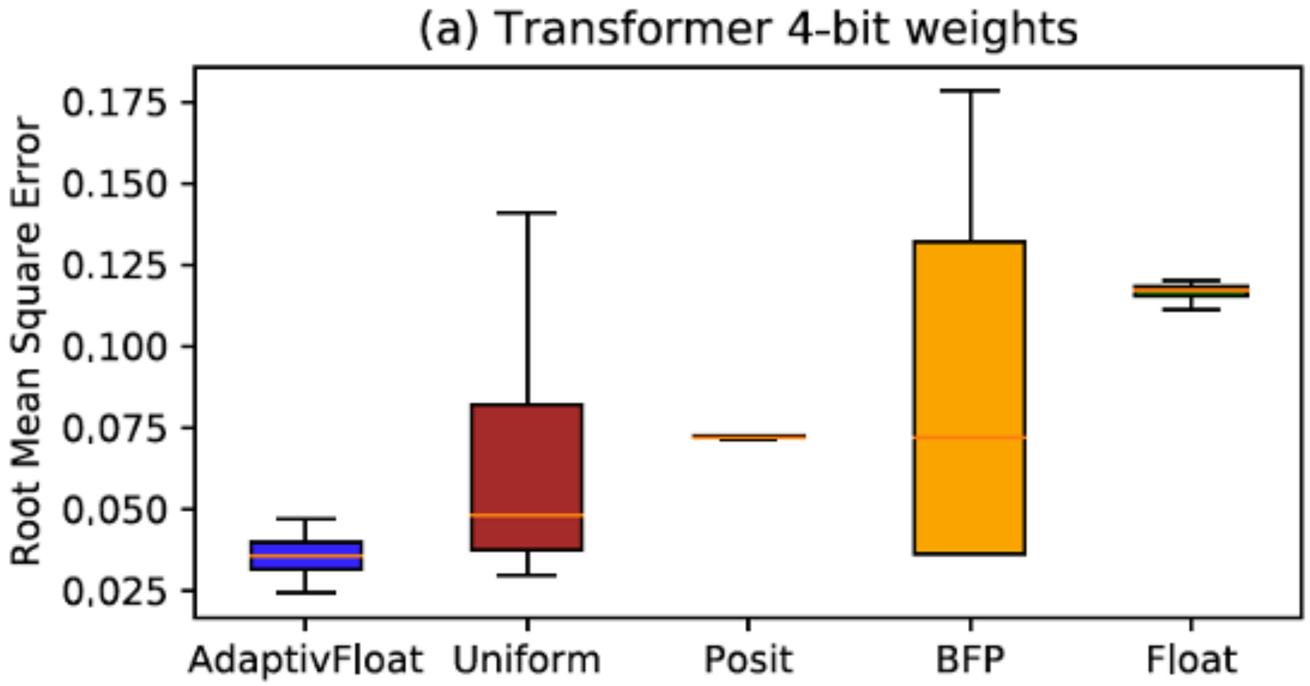
Model	Application	Dataset	# of Params	Range of Params	FP32 Performance
Transformer	Machine Translation	WMT'17 EN-to-DE	93M	[-12.46, 20.41]	BLEU: 27.40
Seq2Seq	Speech-to-Text	LibriSpeech 960H	20M	[-2.21, 2.39]	WER: 13.34
ResNet-50	Image Classification	ImageNet	25M	[-0.78, 1.32]	Top-1 Acc: 76.2

Selected models have narrow to wide weight distribution

Data types being evaluated	Adaptive Dynamic Range?
AdaptivFloat	Yes
Uniform/Integer	Yes
Posit	No
Block Floating-Point	Yes
IEEE-like Floating-Point	No

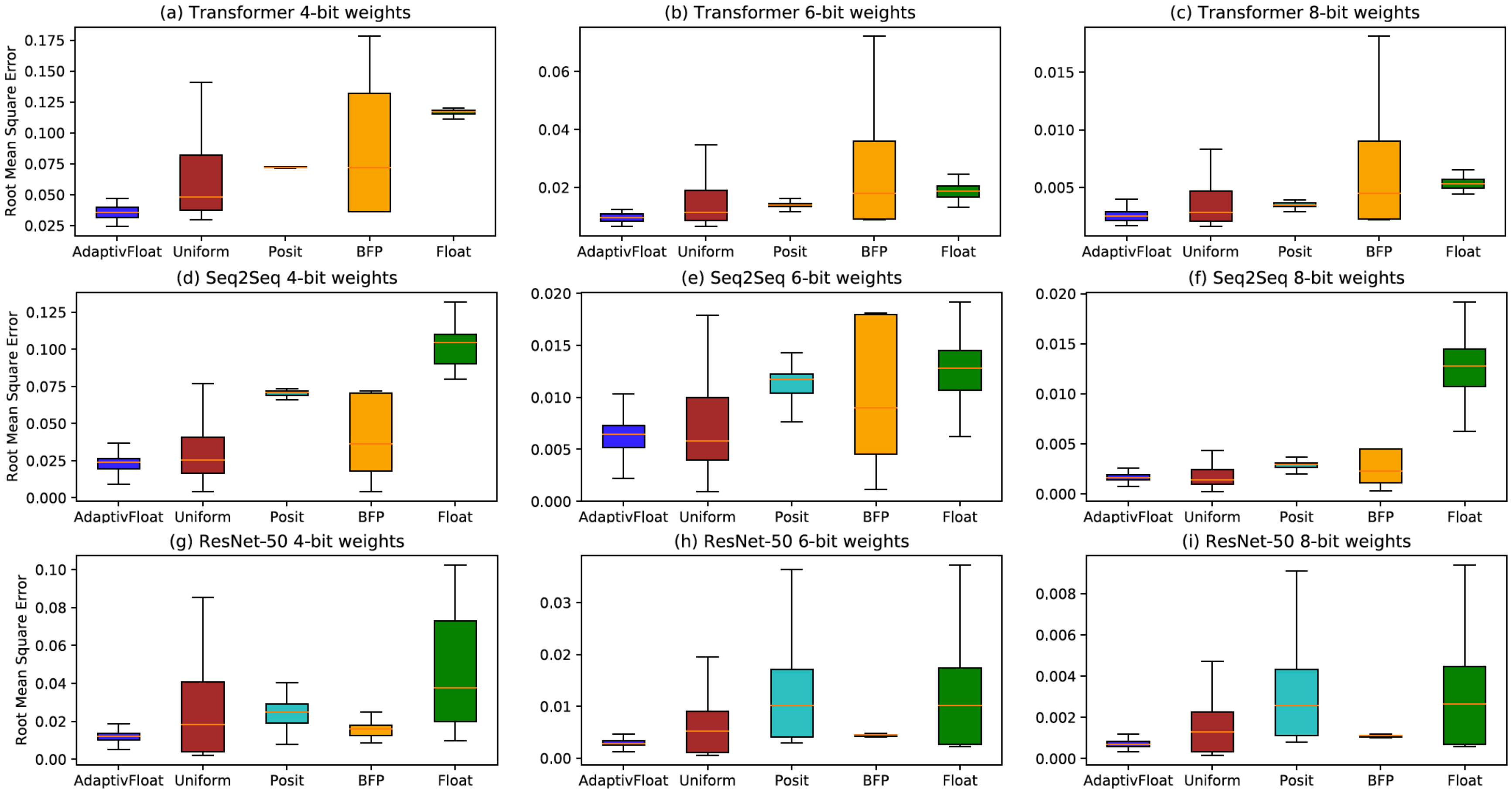
Evaluating against prominent datatypes commonly used in deep learning

Root Mean Square Error



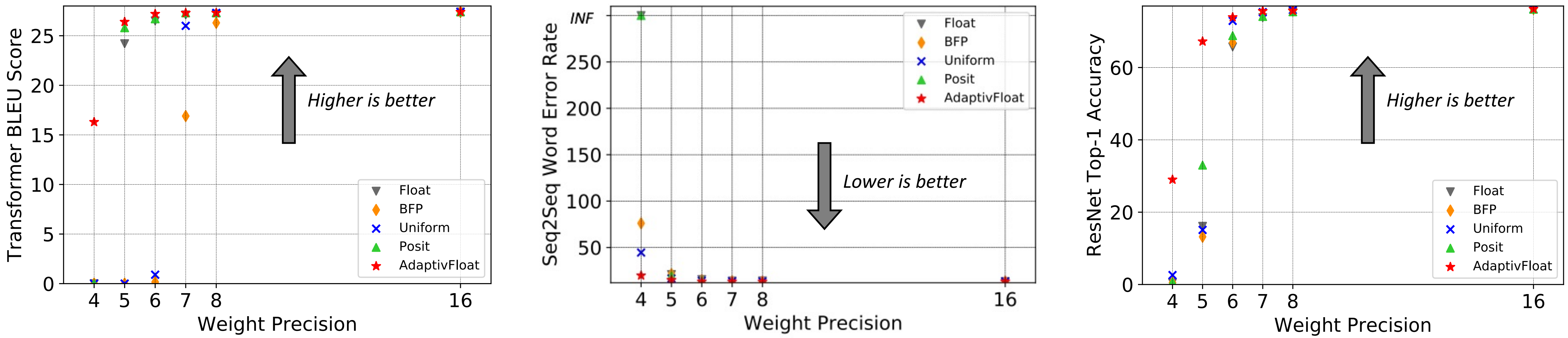
AdaptivFloat produces lowest average quantization error

Root Mean Square Error



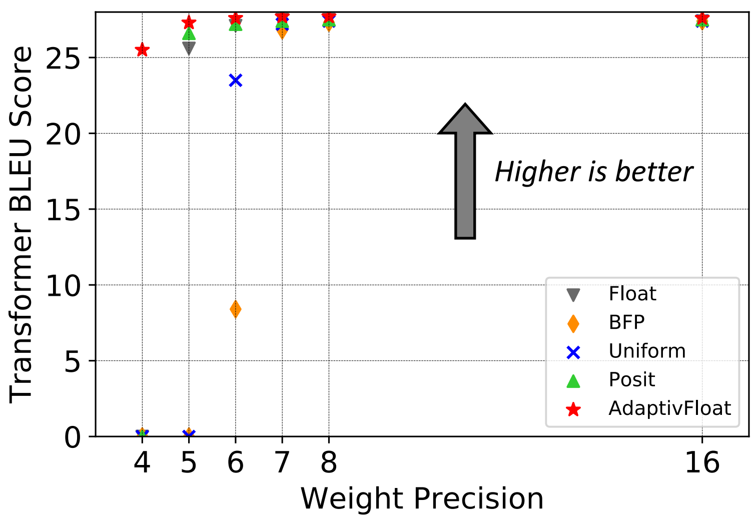
AdaptivFloat produces lowest average quantization error across models, data types, and bit precisions

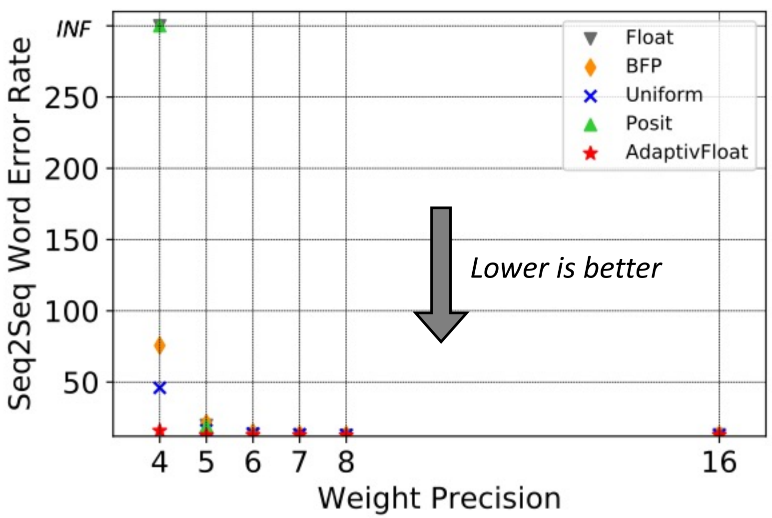
Post-Training Quantization

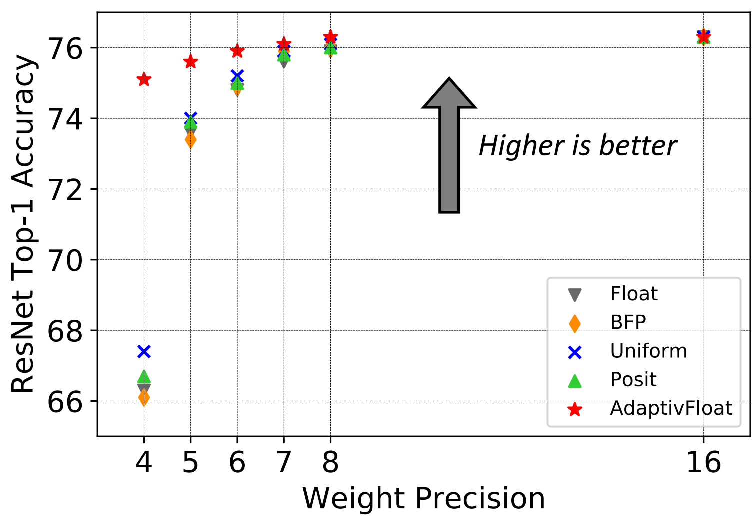


AdaptivFloat demonstrates much greater resiliency towards low word sizes

Quantization-Aware Retraining

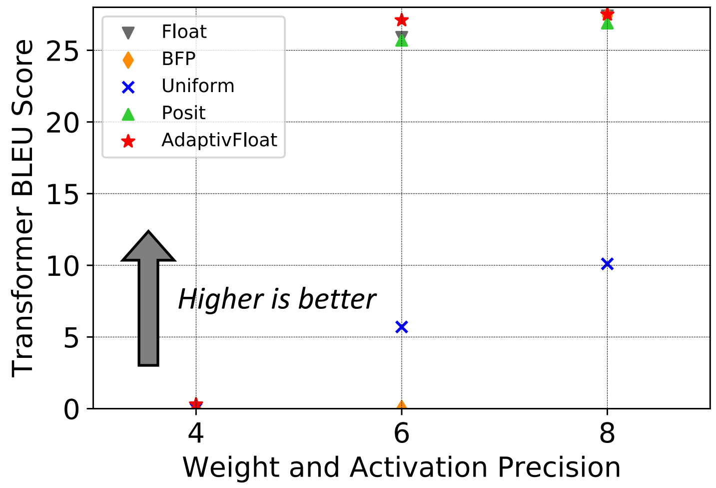


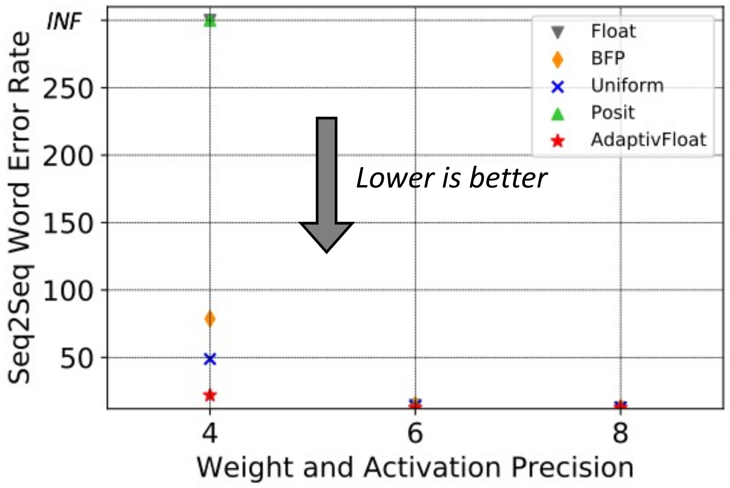


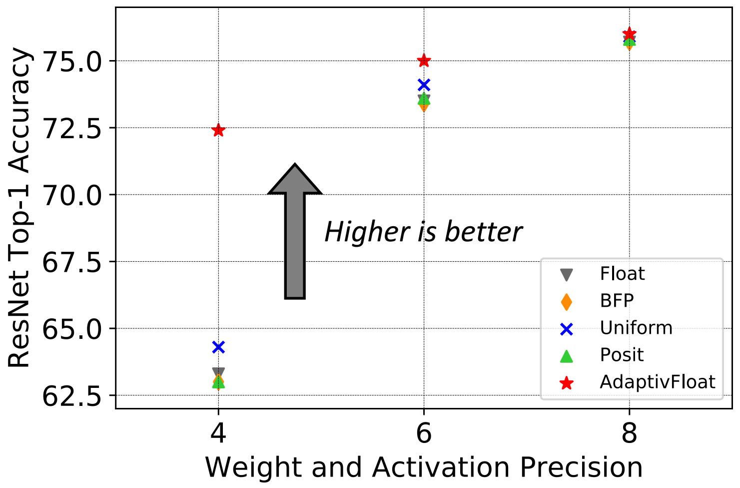


AdaptivFloat maintains much greater resiliency towards low word sizes

Compressing both Weights and Activations







AdaptivFloat maintains greater resiliency when both weights and activations are quantized

This work

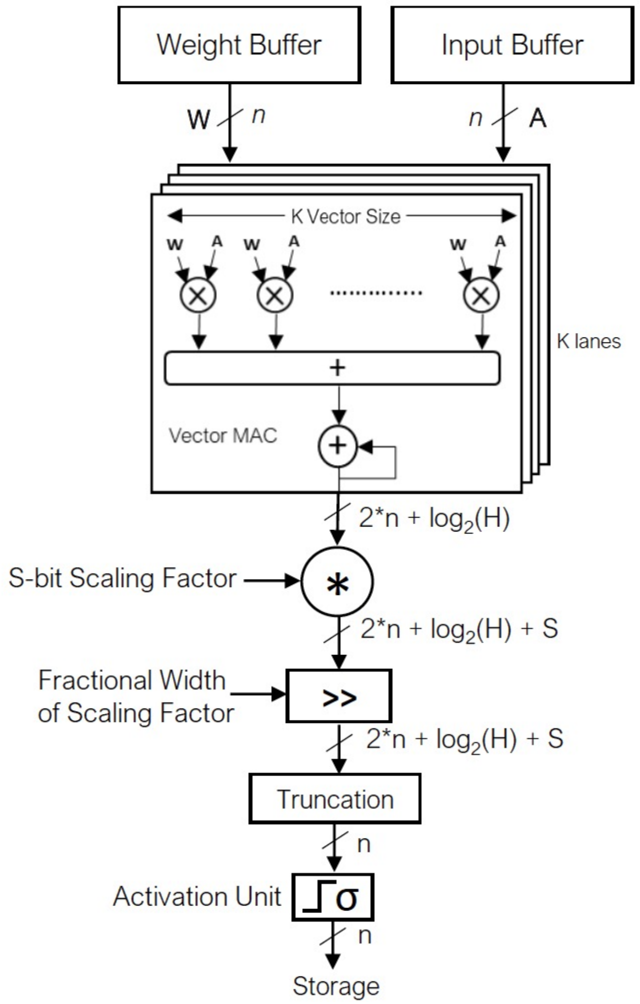
A generalized numerical DNN encoding blueprint, AdaptivFloat, that is:

Adaptive to the statistical distribution of the DNN parameters

Resilient to aggressive bit width compression

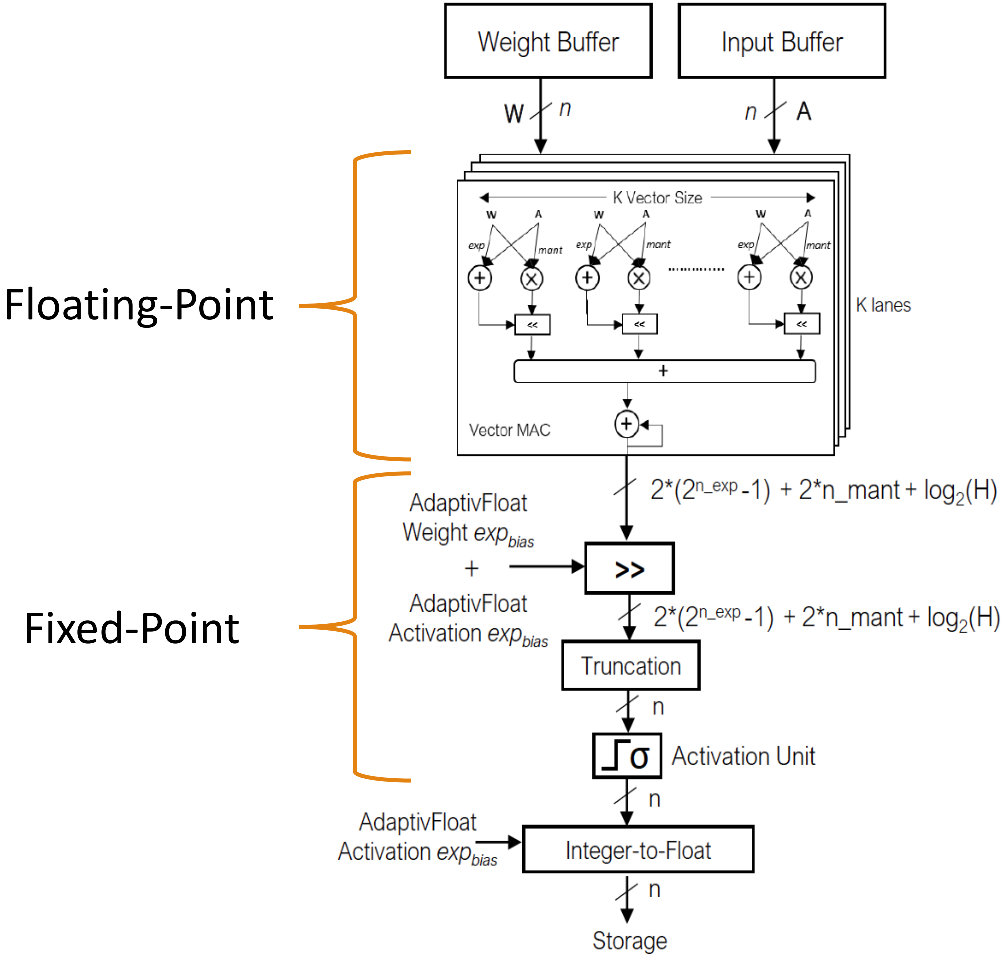
Hardware-friendly with low energy overheads

NVDLA-like n-bit Integer-based PE (INT PE)



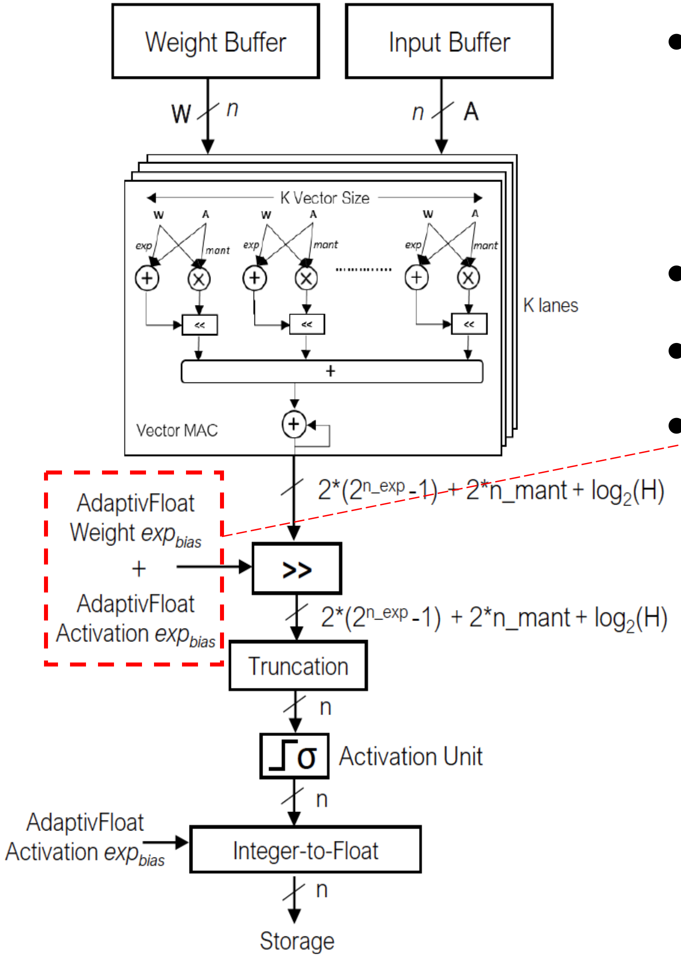
- Weights and input activations are stored in integer format in their respective buffers
- Fixed-point vector MACs
- High precision scaling factor required to scale post-MAC results
  - Scaling factor and fractional width stored in a PE register

Proposed n-bit Hybrid Float-Integer PE (HFINT PE)



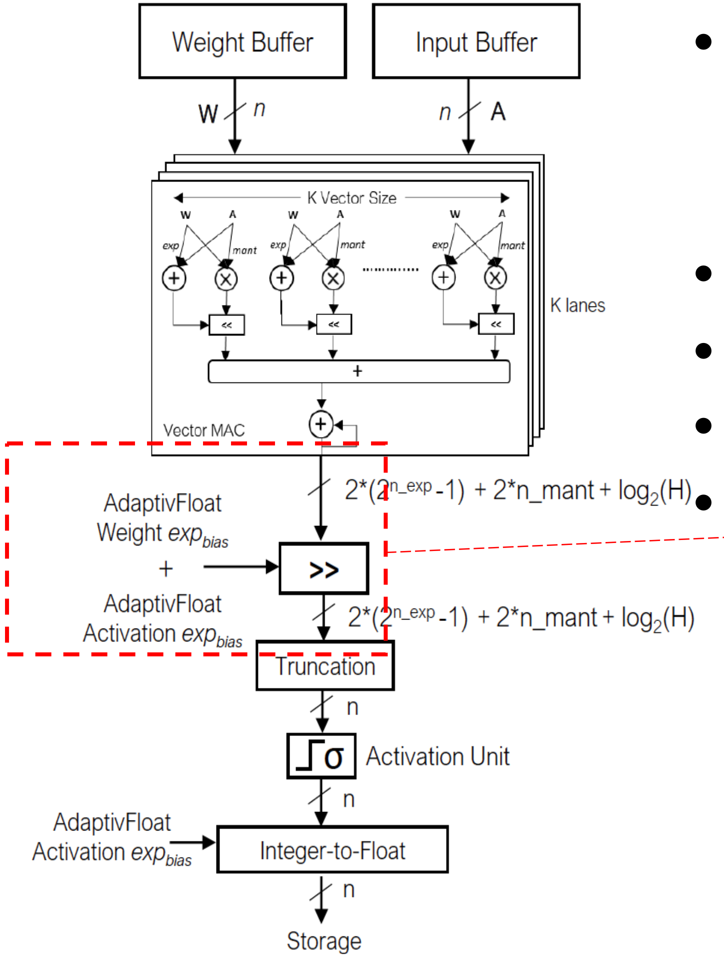
- Weights and input activations are stored in AdaptivFloat format in their respective buffers
- Floating-point vector MACs
- Fixed-point post-processing

Proposed n-bit Hybrid Float-Integer PE (HFINT PE)



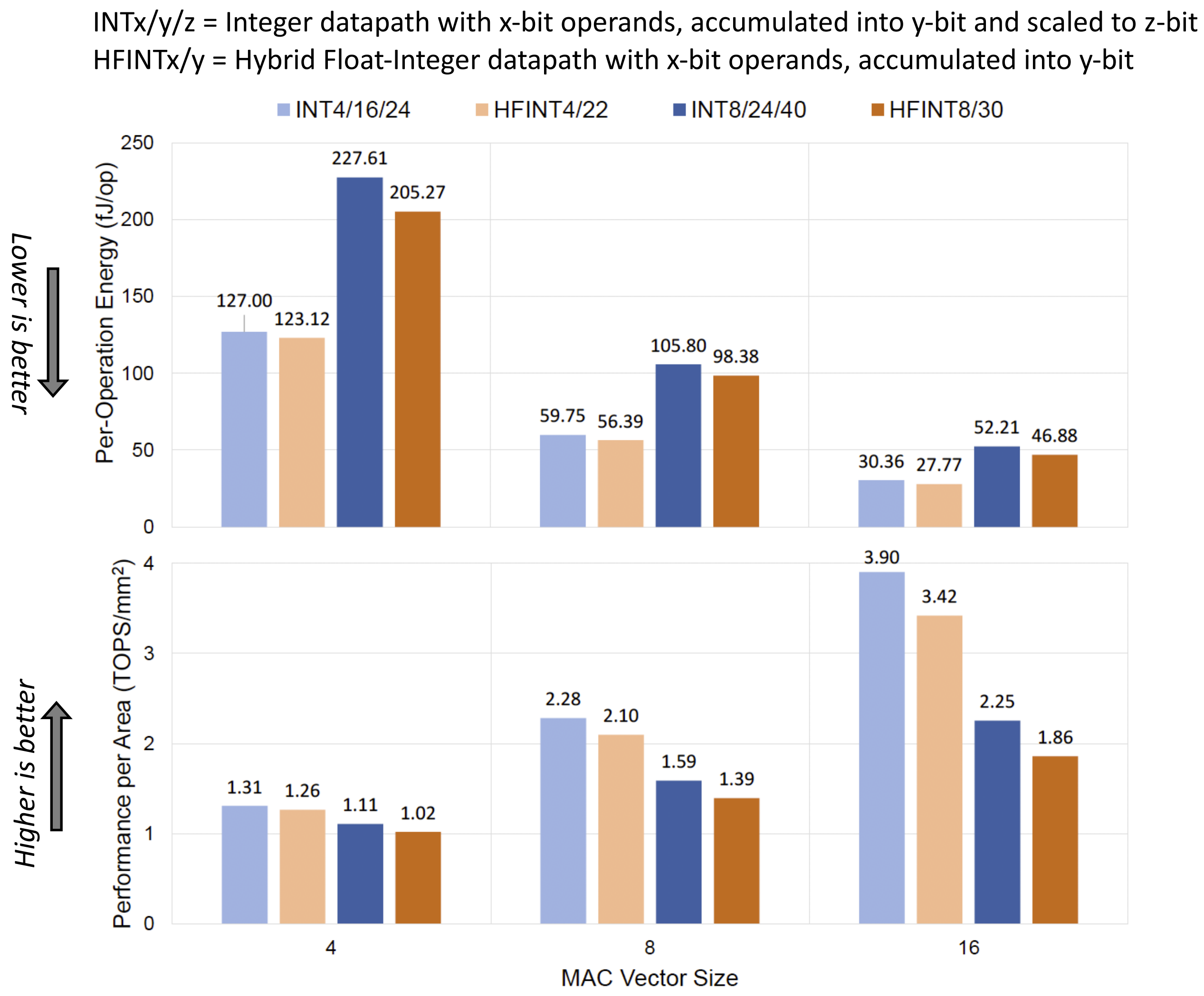
- Weights and input activations are stored in AdaptivFloat format in their respective buffers
- Floating-point vector MACs
- Fixed-point post-processing
- exp_{bias} values stored in a PE register

Proposed n-bit Hybrid Float-Integer PE (HFINT PE)



- Weights and input activations are stored in AdaptivFloat format in their respective buffers
- Floating-point vector MACs
- Fixed-point post-processing
- exp_{bias} values stored in a PE register
- Exponent-shift of partial sums by exp_{bias}

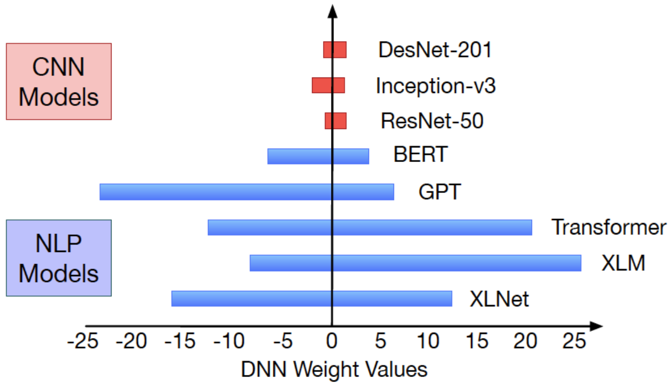
Hardware Performance



+ HFINT produces lower per-operation energy compared to an integer-based PE

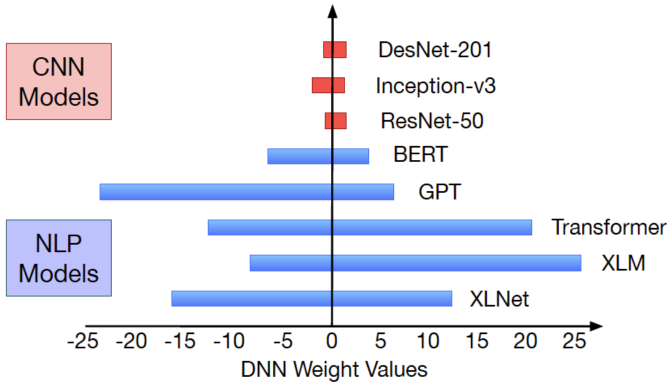
- HFINT generates higher area compared to an integer-based PE

In this talk, “Algorithm-Hardware Co-Design of Adaptive Floating-Point Encodings for Resilient Deep Learning Inference”



Deep learning quantization algorithms need to provide adequate dynamic range to faithfully encode DNNs of various parameter statistics

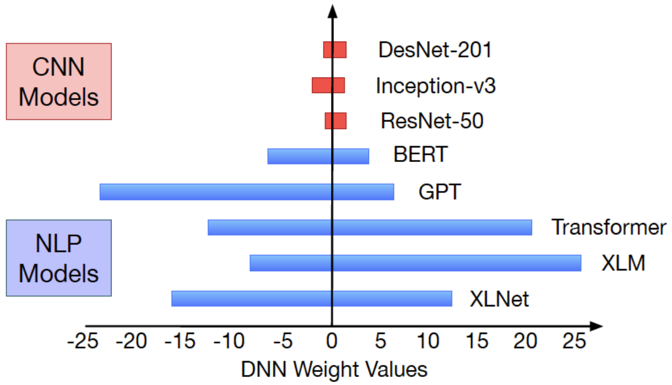
In this talk, “Algorithm-Hardware Co-Design of Adaptive Floating-Point Encodings for Resilient Deep Learning Inference”



Deep learning quantization algorithms need to provide adequate dynamic range to faithfully encode DNNs of various parameter statistics

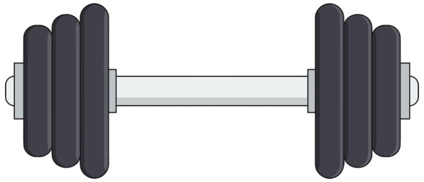
The AdaptivFloat algorithm adapts to DNN parameters by shifting its exponent range based on the max absolute value in the layer matrix

In this talk, "Algorithm-Hardware Co-Design of Adaptive Floating-Point Encodings for Resilient Deep Learning Inference"



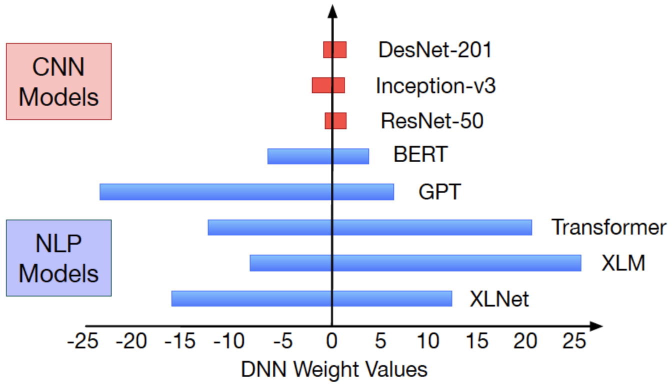
Deep learning quantization algorithms need to provide adequate dynamic range to faithfully encode DNNs of various parameter statistics

The AdaptivFloat algorithm adapts to DNN parameters by shifting its exponent range based on the max absolute value in the layer matrix



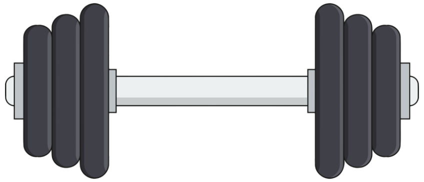
AdaptivFloat is found to be resilient to aggressive bit compression and wide data distribution

In this talk, “Algorithm-Hardware Co-Design of Adaptive Floating-Point Encodings for Resilient Deep Learning Inference”

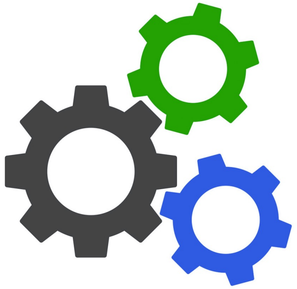


Deep learning quantization algorithms need to provide adequate dynamic range to faithfully encode DNNs of various parameter statistics

The AdaptivFloat algorithm adapts to DNN parameters by shifting its exponent range based on the max absolute value in the layer matrix



AdaptivFloat is found to be resilient to aggressive bit compression and wide data distribution



AdaptivFloat yields higher energy efficiencies in HW compared to fixed-point solutions

Thank you
Any Question?