应用物理实践探究 3

戚一嘉豪 2200012732

June 28, 2024

Paper Reading Part

1 Preface

As we all know, deep neural networks (DNNs) are becoming increasingly important in both our daily lives and academic work, offering stunning and irreplaceable accuracy in many image recognition and natural language processing (NLP) tasks, sometimes even surpassing human performance. However, achieving this beyond-human accuracy requires computationally intensive training and inference involving millions or billions of parameters (e.g., 25.6 million for ResNet-50 [He et al., 2016] and 175 billion for GPT-3 [Brown et al., 2020], which hinders further deployment of DNNs on power-efficient edge platforms like FPGAs and ASICs.

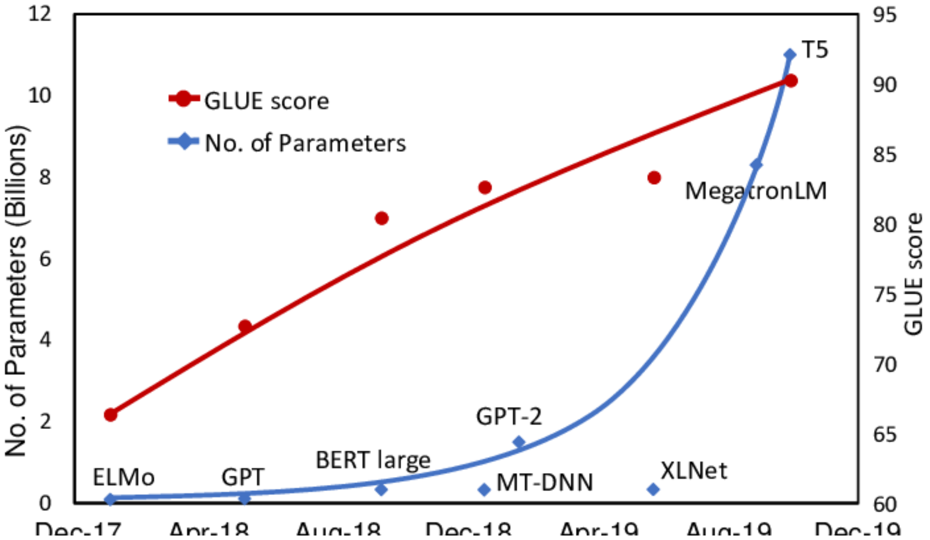

Figure 1: Language Model Size and GLUE Performance[Ahmet and Abdullah, 2020]

As a result, to support DNNs' implementation on edge platforms, we should compress the model's weight and activation from usual 32-bit floating point to a 8-bit(or less) integer times a scaling factor, namely quantization. For instance, compressing ResNet-50 from around 97 MB to about 25 MB can yield a 2× inference speed boost with only a 0.3%

accuracy loss [Zanvari, 2021]. However, more aggressive compression of ResNet-50 to 3.19 MB can result in an unacceptable 9.99% accuracy loss [**?**].

In conclusion, it remains a challenge to strike a balance between model's size(compression ratio) and model's performance. And in [Chang et al., 2020], the author proposes an algorithm-hardware co-design method that applies more flexible quantization schemes and a parameterized architecture that fully utilizes LUT and DSP resources onto two FPGA platforms, i.e., Zynq XC7Z020 and XC7Z045.

2 Contribution1 SP2 Quantization Scheme

2.1 Background:

2.1.1 Uniform Interval Quantization Schemes:

Uniform interval quantization schemes include binary or ternary quantization and fixed-point quantization.

- **Binary and Ternary Quantization**:

In these schemes, weights and activations are represented by very low bits combined with a scaling factor(s). Specifically, binary quantization uses values such as $(-1, +1) \times s$, and ternary quantization uses values such as $(-1, 0, +1) \times s$.However, they often result in an unacceptable accuracy loss ranging from 3 $\sim$ 5%.

- **Fixed-Point Quantization**:

This method is more moderate and flexible. It introduces only negligible accuracy loss compared to binary and ternary quantization.

In m-bit fixed-point quantization scheme, quantized weight can be represented by scaling factor s $\times$ quantized levels:

$$Q^{FP}(m,s) = s \times \frac{1}{2^{m-1}}\{-2^{m-1}, -2^{m-1}+1, ..., 0, ..., 2^{m-1}-1, 2^{m-1}\}$$

and the quantized level can be determined by:

$$[w,s] = \begin{cases} -1, & \text{if } w < -s \\ \frac{round(\frac{w}{s})}{2^{m-1}}, & \text{if } -s \leq w \leq s \\ 1, & \text{if } w > s \end{cases}$$

where w stands for weight, round(x) stands for nearest integer from x[Chang et al., 2020].

2.1.2 Non-Uniform Interval Quantization Schemes:

Power-of-2 is a representative non-uniform quantization method, which replaces necessary multiplications in Fixed-point quantization with bit shifting operations.

$$2^b \times a = \begin{cases} a \ll b, & b > 0 \\ a, & b = 0 \\ a \gg b, & b < 0 \end{cases}$$

In m-bit power-of-2 quantization, quantized weight can be represented a s:

$$Q^{P2}(m,s) = s \times \{-1, -\frac{1}{2}, -\frac{1}{4}, ..., -\frac{1}{2^{2^{m-1}}}, 0, +\frac{1}{2^{2^{m-1}}}, ..., +\frac{1}{2}, +1\}$$

However, increasing precision by raising m does not effectively enhance overall accuracy. Because higher m values only increase resolution around the mean, while the precision in the tails remains low, 4-bit power-of-2 quantization results in an accuracy loss of 1% $\sim$ 2%.

2.2 Sum-of-power-of-2(SP2) Quantization Scheme

The author proposes a brand-new hardware-friendly quantization scheme SP2 that combines all the advantages of quantization schemes without obvious accuracy loss.

$$Q^{SP2}(m,s) = \pm s \times \{q_1 + q_2\},$$

$$q_1 \in \left\{0, \frac{1}{2^{2^{m_1}-1}}, \frac{1}{2^{2^{m_1}-2}}, \cdots, \frac{1}{2}\right\},$$

$$q_2 \in \left\{0, \frac{1}{2^{2^{m_2}-1}}, \frac{1}{2^{2^{m_2}-2}}, \cdots, \frac{1}{2}\right\},$$

where $m_1 + m_2 + 1 = m$ and $m_1 > m_2$.

Although similar to power-of-2 quantization, m-bit SP2 quantization have$2^{m_1} \times 2^{m_2} \times 2 - 1 = 2^m - 1$quantization levels in total, it has more sparse and evenly-scattered quantization levels, which are able to capture weight and activation variance even in the tails.

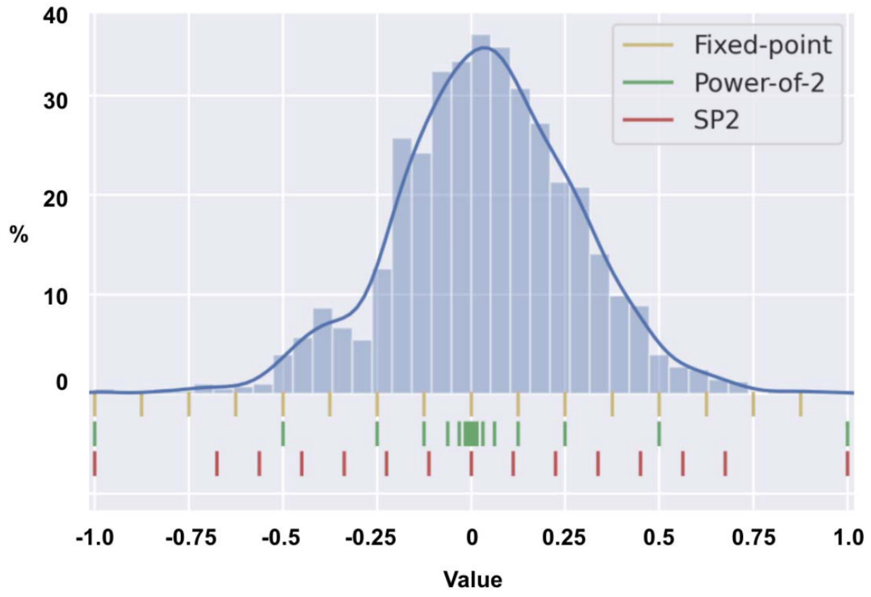

Figure 2: Quantization levels by fixed-point, power-of-2, and SP2 in 4-bit weight representation precision, and weight probability distribution of the 4th layer in MobileNet-V2.[Chang et al., 2020]

For m-bit SP2 quantization weight operand, we use an m_1-bit unsigned integer and an m_2-bit unsigned integer to encode the quantization level of the quantized weight, where $m_1 + m_2 = m - 1$ because of 1-bit sign bit. The quantization level is then $2^{b_1} + 2^{b_2}$, with $2^{b_1} = m_1$ and $2^{b_2} = m_2$

The weight-activation multiplication is implemented by: 1. Shifting the activation operand by b_1 bits, 2. Shifting the activation operand by b_2 bits, 3. Adding the two shifted operands.

Operations 1 and 2 involve shifts of at most $2^{m_1} - 2$ and $2^{m_2} - 2$ bits, respectively. The shifted activation operands will be $n + 2^{m_1} - 2$ and $n + 2^{m_2} - 2$ bits, respectively. Therefore, only one $(n + 2^{m_1} - 2)$-bit addition is needed for multiplication operand, which is quite efficient compared with traditional multiplication.

Result from different quantization schemes for the ResNet-18 and MobileNet-v2 DNN models on CIFAR10, CIFAR100, and ImageNet datasets.[Chang et al., 2020]

Quantization Scheme	Bit width	CIFAR10		CIFAR100		ImageNet	
		ResNet-18	MobileNet-v2	ResNet-18	MobileNet-v2	ResNet-18	MobileNet-v2
	(Wght./Actv.)	Top1	Top5	Top1	Top5	Top1	Top5
Baseline (FP)	32/32	93.62	92.51	74.49	92.70	69.76	90.29
P2	4/4	92.97 (-0.65)	91.34 (-1.17)	73.88 (-0.61)	90.06 (-1.92)	68.20 (-1.56)	88.63 (-1.66)
Fixed	4/4	93.43 (-0.19)	92.34 (-0.17)	74.37 (-0.12)	91.63 (-0.35)	69.72 (-0.04)	90.18 (-0.11)
SP2	4/4	93.47 (-0.15)	92.72 (+0.21)	74.33 (-0.17)	91.69 (-0.29)	69.74 (-0.02)	90.17 (-0.12)
MSQ (half/half)	4/4	93.53 (-0.09)	92.57 (+0.06)	74.58 (+0.09)	91.74 (-0.24)	70.11 (+0.35)	90.04 (-0.25)
MSQ (optimal)	4/4	93.65 (+0.03)	92.55 (+0.04)	74.60 (+0.11)	91.82 (-0.16)	70.27 (+0.51)	90.11 (-0.18)

Upon this tablet, we can realize that Power-of-2 (P2) quantization results in significant accuracy degradation, typically around 1% to 2%, with an extreme case of 2.80% Top-5 accuracy loss for MobileNet-v2 on CIFAR100.

For ImageNet, both Fixed-point (Fixed) and Sum-of-Powers-of-2 (SP2) schemes exhibit negligible accuracy loss: $\leq$ 0.41% for ResNet-18 and $\leq$ 0.62% for MobileNet-v2 across all datasets. These two schemes achieve comparable accuracy for quantized models, while in some cases such as MobileNet-v2 on CIFAR10, SP2 quantization even outperforms Baseline(FP32).

3 Contribution2 FPGA-Centric Mixed Scheme Quantization

The author proposes a mixed scheme quantization(MSQ), which means for the same layer, some weights are quantized by fixed-point quantization and the other by SP2 quantization. The motivation is that different lines in a single weight matrix typically have different weight distribution. And as discussed above2, fixed-point quantization is better for uniform distribution and SP2 is preferable for more Gaussian-like distribution, thus using mixed quantization scheme offers enough flexibility to minimize quantization error.

Apart from that, concerning General Matrix Multiplication(GEMM), traditional multiplication for fixed-point quantization should be processed on DSPs, which is comparatively limited resource on FPGA, while bit-shifting for SP2 quantization can be done via abundant LUT resources. Using mixed quantization scheme can drive up the overall hardware utilization, so optimizing the ratio of fixed-point to SP2 quantization in accordance with the DSP to LUT resource ratio on a given FPGA can significantly enhance processing throughput.

Through analyzing the available DSP and LUT resources on a specific FPGA platform, we can determine an optimal threshold θ for the variance in the weight distribution of a DNN. For lines with variance below this threshold, SP2 GEMM is applied, while fixed-point GEMM is used otherwise, achieving high parallelization and throughput.

Performance of different implementations on FPGA platforms[Chang et al., 2020]

Impl.	Device	Bat	Blk_{in}	Blk_{out}	Ratio (fixed/SP2)	Peak Thrpt. (GOPS)
D1-1	XC7Z020	1	16	16	1:0	52.8
D1-2		1	16	16	1:1	106
D1-3		1	16	16	1:1.5	132
D2-1	XC7Z045	4	16	16	1:0	208
D2-2		4	16	16	1:1	416
D2-3		4	16	16	1:2	624

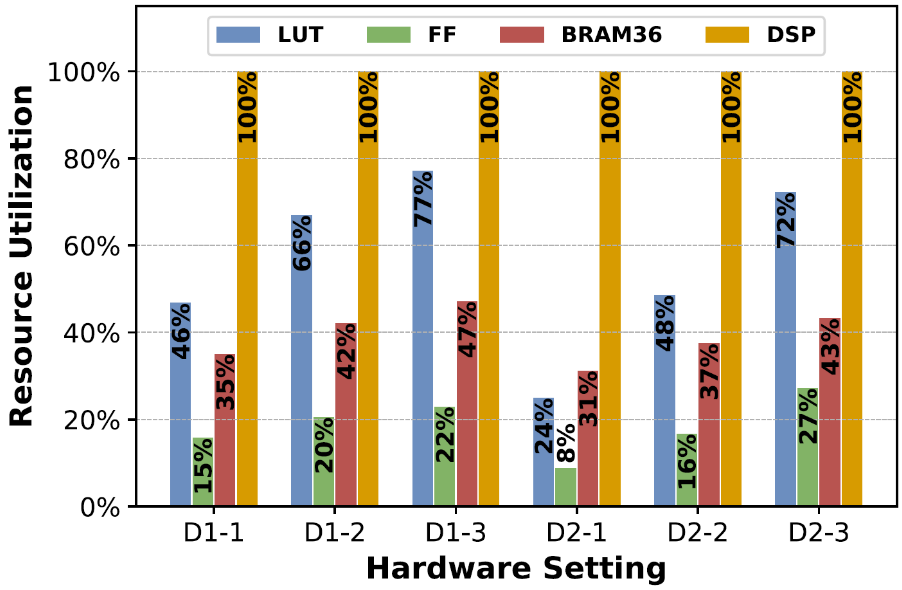

Figure 3: FPGA resource utilization with different devices and settings[Chang et al., 2020]

As SP2 ratio increases, peak throughput (GOPS) increases due to better usage of LUT resources. This can be observed with a 2.5× increase in the XC7Z020's peak throughput, from 52.8 to 132 GOPS, and a 3× increase in the XC7Z045's peak throughput, from 208 to 624 GOPS. Correspondingly, with the optimal ratio of fixed/SP2 on XC7Z045, an single image-recognition inference latency decreases 2.49× from 25.1ms to 10.1ms on ResNet-18. And on XC7Z020, this latency decreases 2.13× from 100.7ms to 47.1ms.

Comparisons with existing works with ResNet-18 model on ImageNet dataset.

Methods	Bit-width (W/A)	Top-1 (%)	Top-5 (%)
Baseline(FP)	32/32	69.76	89.08
Dorefa [38]	4/4	68.10	88.10
PACT [39]	4/4	69.20	89.00
DSQ [40]	4/4	69.56	N/A
QIL [41]	4/4	70.10	N/A
μL2Q [42]	4/32	65.92	86.72
LQ-NETS [44]	4/4	69.30	88.80
MSQ	4/4	**70.27**	**89.42**

Comparisons with existing works with MobileNet-v2 model on ImageNet dataset.

Methods	Bit-width (W/A)	Top-1 (%)	Top-5 (%)
Baseline(FP)	32/32	71.88	90.29
PACT [39]	4/4	61.40	N/A
DSQ [40]	4/4	64.80	N/A
MSQ	4/4	**65.64**	**86.98**

We can see MSQ outperforms other quantization schemes maybe because combining SP2 and Fixed quantization allows the quantized DNN weights to capture original weight distribution better and the quantization noise can act as regularization, which further improves generalization ability and prevents over-fitting. Although quantizing the lightweight MobileNet-v2 model with 4-bit weights and activations is particularly challenging, MSQ still outperforms other quantized models.

4 Summary and Prospect

The author proposes the first per row quantization scheme that applies SP2 quantization for rows with more-Gaussian like distribution and fixed-point quantization for uniformly distributed rows. As $GEMM_{SP2}$ simply uses LUTs, more DSPs can be assigned to process traditional multiplication in $GEMM_{fixed}$, thus improving overall throughput and decreasing processing latency.

At last, the proposed solution is designed for FPGA use only and can not be applied directly onto general-use hardware such as CPUs or GPUs. And determining the optimal partition ratio between different quantization schemes based on different FPGA architectures requires characterization of FPGA resources and extremely time-consuming fine-tuning to achieve optimal performance. These two reasons combined may hinder it from extensive deployment.

References

[Ahmet and Abdullah, 2020] Ahmet, A. and Abdullah, T. (2020). Real-time social media analytics with deep transformer language models: A big data approach. In *2020 IEEE 14th International Conference on Big Data Science and Engineering (BigDataSE)*, pages 41–48.

[Brown et al., 2020] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. *CoRR*, abs/2005.14165.

[Chang et al., 2020] Chang, S., Li, Y., Sun, M., Shi, R., So, H. K., Qian, X., Wang, Y., and Lin, X. (2020). Mix and match: A novel fpga-centric deep neural network quantization framework. *CoRR*, abs/2012.04240.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

[Zanvari, 2021] Zanvari (2021). Resnet50 quantization for inference speedup in pytorch. `https://github.com/zanvari/resnet50-quantization`. Accessed: 2024-06-27.