Ultra-low Bit Quantization for Visual Wake Word Challenge

Zhen Dong, Yaohui Cai, Amir Gholami, Tianjun Zhang, Kurt Keutzer
University of California at Berkeley, Peking University
{zhendong, amirgh, tianjunz, keutzer}@berkeley.edu caiyaohui@pku.edu.cn

1. Contact Email ID. Zhen Dong <zhendong@berkeley.edu>

2. Description of model architecture.

- **Model architecture**
  Our model is based on ShuffleNetv2 [1] $\times 0.5$ version. We replace the final 1024×1000 fully connected layer with a 1024×2 fully connected layer to solve the binary classification problem.
  Please refer to the detailed model architecture and performance metrics in the appendix Table 1.
- **Data Preprocessing**
  In order to satisfy the memory size and memory usage constraints, we resize all input images to 192×192. Apart from resize, we use random affine, random horizontal flip and normalization as preprocessing techniques during training. However, for inference we only use normalization.
- **Quantization Scheme** The original model in floating point precision has 352K parameters which does not fit into a 250KB SRAM as required by the competition. To address this we perform mixed-precision quantization using a recently proposed Hessian AWare Quantization (HAWQ) method from our group [2].
  The reason for mixed-precision quantization is two-folds: (i) different layers have different sensitivity to quantization, and a uniform quantization of all layers may lead to significant accuracy degradation, and (ii) some of the stages have considerably larger activation memory requirement, as shown in Table 1. Performing mixed-precision quantization, allows us to reduce the bits for these layers and keep the quantization bits for other layers at higher values to avoid unnecessary loss of performance.
  Another important point is that, we restrict the quantization bits for both weights and activations to be less than or equal to 8-bits. Although most layers in our model can satisfy the constraints even with floating point activations, we still quantized them because we expect an actual hardware implementation in the future. Models with 8-bit weights and 32-bit activations still need floating point arithmetics and cannot benefit from low-precision operators and modules.

3. Performance Metrics.

- **Accuracy on the COCO minival set**
  We first train the model using full precision on the combined training and validation dataset (excluding minival) from COCO 2014 challenge[3]. We achieve 91.48% accuracy on minival set before quantization. Accuracy after quantization is 91.04%, which is pretty close to the original accuracy.
- **Model size**
  The model has 336K parameters which is equivalent to 1312 KB in floating points. After performing the quantization using the approach described in [2], the model size is reduced to **246KB**.
- **Peak memory usage**
  According to the definition in the documents, the peak memory usage of our model is 1293 KB before quantization. This is reduced to **243 KB** after quantization.
- **Multiply-adds per inference**
  The total MACs per inference is **29.0M** which is well below the limit of 60M MACs. One can trade-off the MACs with accuracy by using larger kernels or more conv layers at the beginning of the network.

REFERENCES

[1] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 116–131, 2018.
[2] Z. Dong, Z. Yao, A. Gholami, M. Mahoney, and K. Keutzer, "HAWQ: Hessian aware quantization of neural networks with mixed-precision," *arXiv preprint arXiv:1905.03696*, 2019.
[3] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

Appendix A. Methods to compute performance metrics.

- For MAC calculation, we didn't include element-wise operation, such as Relu, bias addition, pooling into total MACs.

- For activation, each ShuffleNet v2 block has two branches. We sum over the maximum input plus output activation size on each branch.

Appendix B. Model architecture and performance metrics.

Stage	Output Size	Output Channels	Peak Memory	Parameter Size (SRAM)	MACs
Conv3 × 3	96 × 96	24	1296KB	3KB	6.0M
MaxPool	48 × 48	24	1080KB	0	-
Stage1	24 × 24	48	540KB	24KB	4.6M
Stage2	12 × 12	96	276KB	168KB	7.2M
Stage3	6 × 6	192	138KB	341KB	4.1M
GlobalPool	1 × 1	192	27KB	0	-
FC			∼0KB	∼0KB	∼0.0M
Total/Max			1296KB	536KB	21.9M

Table 1: *Model architecture and performance metrics before HAWQ. Max Memory refers to the largest activation memory usage in each stage, according to the method described in the competition.*

Stage	Output Size	Output Channels	Peak Memory	Parameter Size (SRAM)	MACs
Conv3 × 3	96 × 96	24	1296KB	3KB	6.0M
MaxPool	48 × 48	24	1080KB	0	-
Stage1	24 × 24	48	540KB	24KB	4.6M
Stage2	12 × 12	96	276KB	168KB	7.2M
Stage3	6 × 6	192	138KB	341KB	4.1M
Conv1 × 1	6 × 6	192	54KB	144KB	1.3M
GlobalPool	1 × 1	192	27KB	0	-
FC			∼0KB	∼0KB	∼0.0M
Total/Max			1296KB	680KB	23.2M

Table 2: *Model architecture and performance metrics before HAWQ. Max Memory refers to the largest activation memory usage in each stage, according to the method described in the competition.*

Stage	Output Size	Output Channels	Scheme	Peak Memory	Parameter Size (SRAM)	MACs
Conv3 × 3	96 × 96	24	W8 A6	243KB	1KB	6.0M
MaxPool	48 × 48	24	A6	203KB	0	-
Stage1	24 × 24	48	W8 A8	207KB	6KB	4.6M
Stage2	12 × 12	96	W6/8 A8	69KB	32KB	7.2M
Stage3	6 × 6	192	W5/6/8 A8	35KB	63KB	4.1M
GlobalPool	1 × 1	192	A8	7KB	0	-
FC			W8 A8	∼0KB	∼0KB	∼0.0M
Total/Max				243KB	102KB	21.9M

Table 3: *Model architecture and performance metrics after HAWQ. Max Memory refers to the largest memory usage in that stage. Scheme refers to quantization scheme, where W stands for weight and A stands for activation. For example, W6/8 A6 means quantizing all weights in this stage to 6 or 8 bits depending on the layer while quantizing output activations of this stage to 6 bits.*

Stage	Output Size	Output Channels	Scheme	Peak Memory	Parameter Size (SRAM)	MACs
Conv3 × 3	96 × 96	24	W8 A6	243KB	1KB	6.0M
MaxPool	48 × 48	24	A6	203KB	0	-
Stage1	24 × 24	48	W8 A8	207KB	6KB	4.6M
Stage2	12 × 12	96	W6/8 A8	69KB	32KB	7.2M
Stage3	6 × 6	192	W5/6/8 A8	35KB	63KB	4.1M
Conv1 × 1	6 × 6	192	W6 A8	14KB	27KB	1.3M
GlobalPool	1 × 1	192	A8	7KB	0	-
FC			W8 A8	∼0KB	∼0KB	∼0.0M
Total/Max				243KB	129KB	23.2M

Table 4: *Model architecture and performance metrics after HAWQ. Max Memory refers to the largest memory usage in that stage. Scheme refers to quantization scheme, where W stands for weight and A stands for activation. For example, W6/8 A6 means quantizing all weights in this stage to 6 or 8 bits depending on the layer while quantizing output activations of this stage to 6 bits.*