PROFIT: A Novel Training Method for sub-4-bit MobileNet Models

Supplementary Material

Eunhyeok Park[1][0000−0002−7331−9819] and Sungjoo Yoo[2][0000−0002−5853−0675]

[1] canusglow@gmail.com and [2]sungjoo.yoo@gmail.com
[1] Inter-university Semiconductor Research Center (ISRC)
[2] Department of Computer Science and Engineering
[2] Neural Processing Research Center (NPRC)
[1,2]Seoul National University, Seoul, Korea

1 Training and validation loss curves with and without PROFIT

Figure 1 shows the training and validation loss curves regarding the existance of PROFIT. The loss curves are obtained while fine-tuning the quantized MobileNet-v3 with teacher-student and progressive quantization technique. The orange line represents the loss curve of PROFIT, while the blue line represents the loss curve without PROFIT. As shown in the figure, without PROFIT, the validation loss curve heavily fluctuates. When we apply PROFIT, the fine-tuning process is greatly stabilized by removing, from the training process, the source of AIWQ problem, i.e., the sensitive layers to AIWQ. Under PROFIT, both training and validation loss curves are lower than the curves without PROFIT, and the fluctuation of the validation loss is minimized, which shows that PROFIT can offer better convergence in the training of low-precision networks.

2 Training and validation loss curves of PACT versus DuQ

Figure 2 shows the training and loss curves of 4-bit MobileNet-v3 for ImageNet with PACT and DuQ with negative padding. In progressive quantization, we first train activation quantizatioin parameters and then weight quantization ones. At the end of fine-tuining, DuQ with negative padding gives lower loss than PACT for both activation and weight quantization. This is because DuQ is designed to support asymmetric distribution and consider both truncation and rounding errors. In addition, the negative padding helps DuQ to fully utilize the given quantization levels. However, in the case of weight quantization, the validation loss of DuQ in the early epochs is larger than that of PACT. This is because PACT with SAWB is designed under a distribution prior to preserve the scale of input distribution, while DuQ has additional parameters for input,

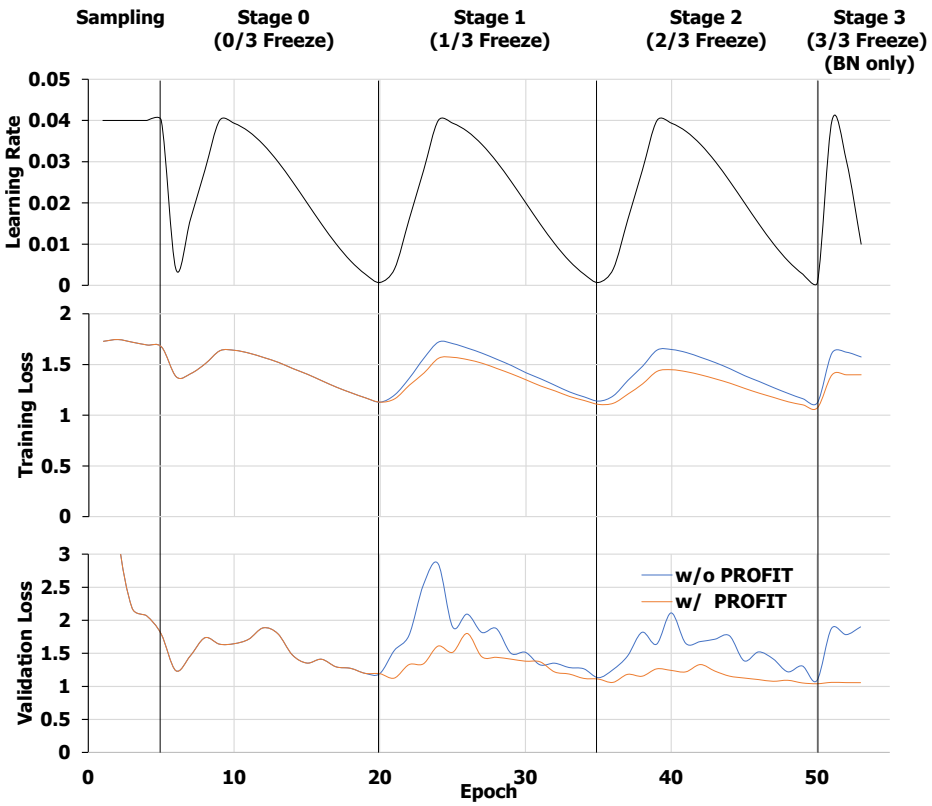

Fig. 1. The training and validation loss curves with and without PROFIT for 4-bit quantization of MobileNet-v3 for ImageNet.

i.e., in post transformation, which may incur different scales of input and output distribution under DuQ in the early epochs and disturb the running mean and variance of normalization layers. However, having additional parameters in post transformation eventually helps increase the accuracy of network by providing additional degree of freedom as the validation loss shows at the end of training.

3 H-swish output distributions and DuQ with negative padding

H-swish function has a constant negative minimum, but its maximum value depends on the input data. As shown in the Figure 3, the output distributions of h-swish function are highly different in terms of layer and feature map. Thus, learned step-size quantization [2], which have pre-defined number of negative and positive quantization levels, are not appropriate to address these various shapes of distribution. Meanwhile, as the figure shows, DuQ with negative padding is

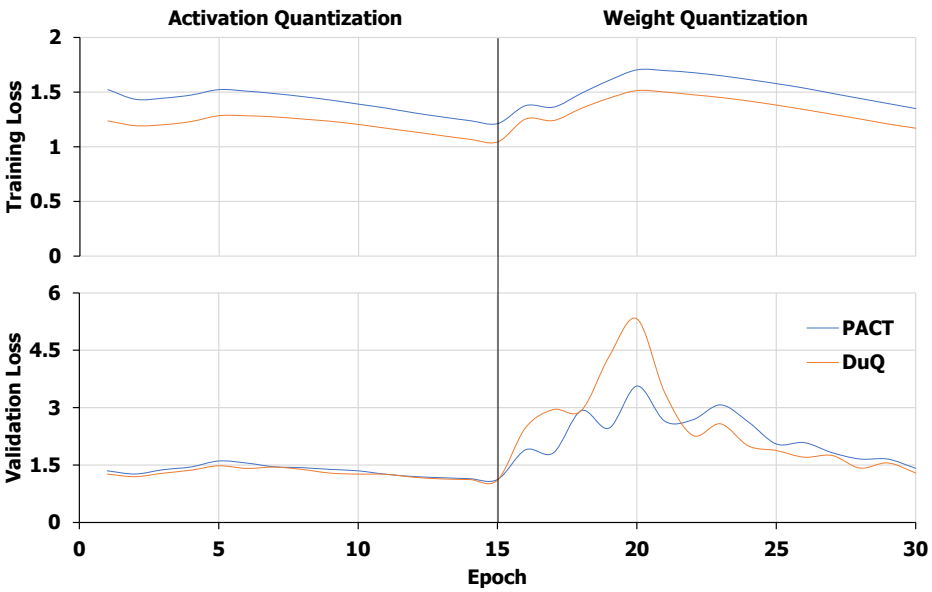

Fig. 2. Loss curves of PACT and DuQ with negative padding obtained during fine-tuning MobileNet-v3 for ImageNet dataset.

flexible enough to learn quantization levels through back-propagation for those distributions with diverse characteristics.

4 Detailed analysis of computation cost and model size

Figure 4 shows the best results of the computation cost and model size of the quantized MobileNet-v1, v2 and v3. As shown in the figure, our proposed methods enable 4-bit quantization of optimized networks at high accuracy thereby pushing mobile networks towards more resource-efficient regime compared with the state-of-the-art quantization solutions.

Our 4-bit MobileNet-v1 model offers 4.06 % better accuracy with 37.7 % computation cost reduction compared to the model of 4-bit activation and 8-bit weight from [6]. In the case of MobileNet-v2, our 4-bit MobileNet-v2 model outperforms the previous best 4-bit model [3] by 6.76 % with the same computation cost, and the 4-bit activation and 8-bit weight model [6] by 9.56 % with 38.0 % less computation cost. 4-bit Our MobileNet-v3 model reduces computation cost by 63.8 % at almost the same accuracy (a difference of 0.1 %) as the 8-bit quantized model [4]. In terms of model size, our 4-bit quantized MobileNet-v2 and v3 almost halve over the 8-bit models [6, 4] but have negligible accuracy loss or even better accuracy.

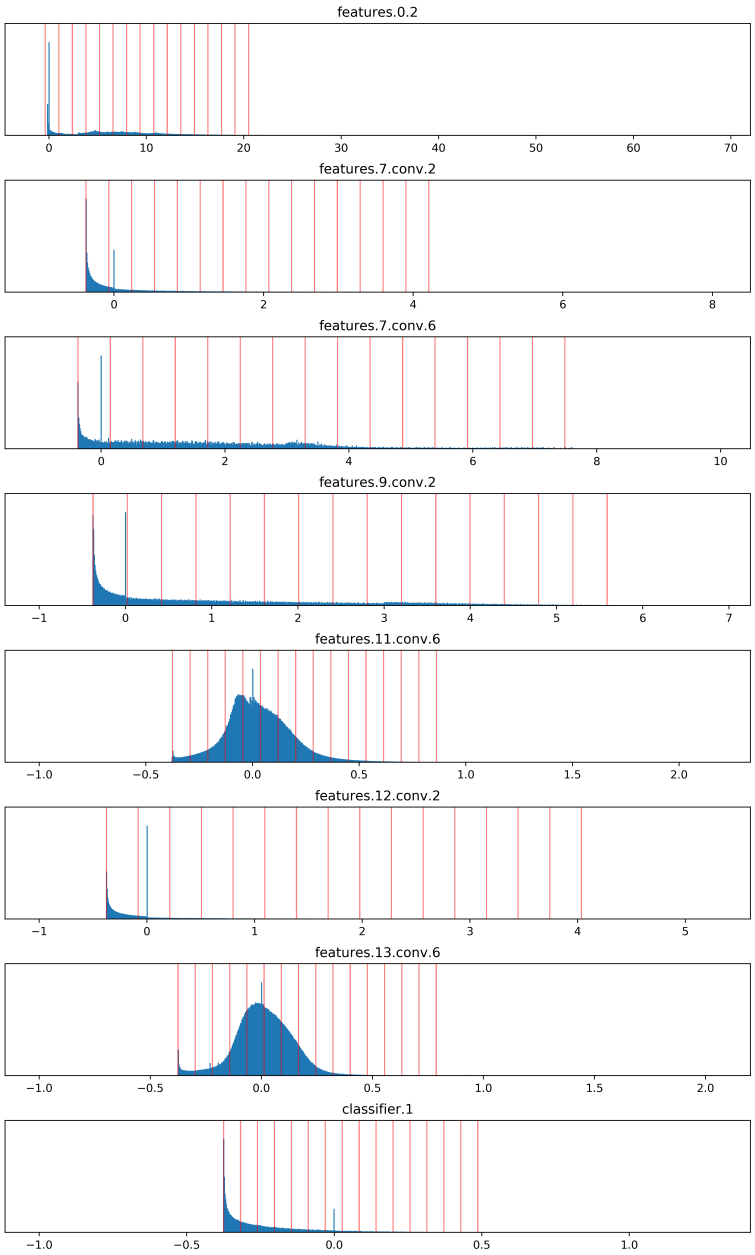

Fig. 3. The layer-wise h-swish output histogram of MobileNet-v3 for ImageNet. The red lines represent the trained 4-bit quantization levels under DuQ with negative padding.

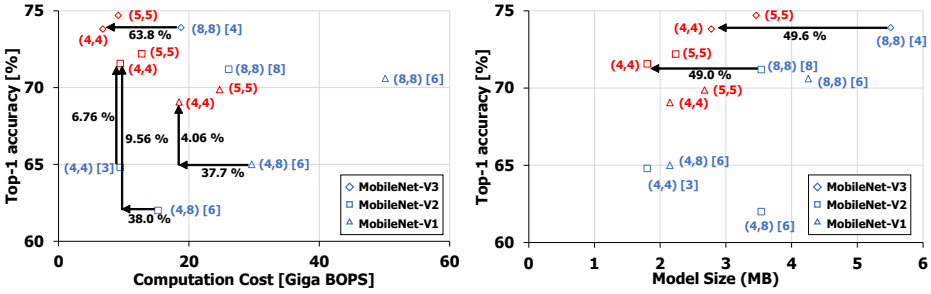

Fig. 4. Comparison of accuracy and estimated computation cost based on the HW accelerator model (the bit-operations, BOPS [1]), and comparison of accuracy and model size of the quantized network. The tuple (a,w) represents the bit-width of activation and weight, respectively. The red markers represent our results, and the blue markers the results of state-of-the-art methods [3, 4, 6, 8].

5 Discussion of fused-batchNorm and skip-connection

Batch normalization layer normalizes input activation using batch statistics, i.e., mean and variance, and applies scale and shift (Eqn. 1) [5]. After training, the running mean and variance of the batch normalization layer can be absorbed to scale and shift. In addition, the combined scale and shift terms can be absorbed by scaling convolution kernel weights and adding to the bias of the prior convolution layer (Eqn. 2). This technique, called fused-batchnorm [10], was proposed to remove the overhead of batch normalization layer in inference.

$$\hat{x} = \gamma \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta. \tag{1}$$

$$x = W \otimes a, \ \hat{x} = (\frac{\gamma}{\sqrt{\sigma^2 + \epsilon}} W) \otimes a + \beta - \frac{\gamma \mu}{\sqrt{\sigma^2 + \epsilon}}. \tag{2}$$

$$\hat{x}' = Q(\frac{\gamma}{\sqrt{\sigma^2 + \epsilon}} W) \otimes a + \beta - \frac{\gamma \mu}{\sqrt{\sigma^2 + \epsilon}}. \tag{3}$$

When we quantize weights under fused-batchnorm, we need to apply quantization to the weight with batch norm scaling, as shown in Eqn. 3. However, according to our observation, all models under fused-batchnorm failed to converge when 4-bit quantization is applied to the weights with fused-batchnorm.

It is because the weights under fused-batchnorm tend to have wider value ranges, due to the additional scaling, than the original weights. We think that, in order to exploit fused-batchnorm in 4-bit and lower precision, it is desirable to apply channel-wise quantization, which is the beyond of the scope of this paper and left as future work.

Besides, our 4-bit MobileNet-v3 model performs 4-bit convolution and matrix multiplication operation for the entire network except first convolution layer having 8-bit input and squeeze-excitation module having 8-bit activation. On

the other hand, we adopt the precision-highway [7, 9] for the skip-connection path. This helps to maintain the high-precision dataflow through identity path thus greatly reduces the impact of quantization error of the residual path with negligible overhead.

References

1. Baskin, C., Schwartz, E., Zheltonozhskii, E., Liss, N., Giryes, R., Bronstein, A.M., Mendelson, A.: Uniq: Uniform noise injection for non-uniform quantization of neural networks. arXiv:1804.10969 (2018)
2. Esser, S.K., McKinstry, J.L., Bablani, D., Appuswamy, R., Modha, D.S.: Learned step size quantization. arXiv:1902.08153) (2019)
3. Gong, R., Liu, X., Jiang, S., Li, T., Hu, P., Lin, J., Yu, F., Yan, J.: Differentiable soft quantization: Bridging full-precision and low-bit neural networks. arXiv:1908.05033 (2019)
4. Howard, A., Sandler, M., Chu, G., Chen, L., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q.V., Adam, H.: Searching for mobilenetv3. International Conference on Computer Vision (ICCV) (2019)
5. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. International Conference on Machine Learning (ICML) (2015)
6. Krishnamoorthi, R.: Quantizing deep convolutional networks for efficient inference: A whitepaper. arXiv:1806.08342 (2018)
7. Liu, Z., Wu, B., Luo, W., Yang, X., Liu, W., Cheng, K.: Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. European Conference on Computer Vision (ECCV) (2018)
8. Nagel, M., van Baalen, M., Blankevoort, T., Welling, M.: Data-free quantization through weight equalization and bias correction. arXiv:1906.04721 (2019)
9. Park, E., Kim, D., Yoo, S., Vajda, P.: Precision highway for ultra low-precision quantization. arXiv:1812.09818 (2018)
10. Tulloch, A., Jia, Y.: Quantization and training of neural networks for efficient integer-arithmetic-only inference. Conference on Computer Vision and Pattern Recognition (CVPR) (2018)