Universally Slimmable Networks and Improved Training Techniques (Supplementary)

Jiahui Yu Thomas Huang
University of Illinois at Urbana-Champaign

1. Discussion

In this section, we mainly discuss three topics with some experimental results.

Nonuniform Universally Slimmable Networks. For all trained US-Nets so far, the width ratio is uniformly applied to all layers (*e.g.*, MobileNet $0.25\times$ means width in all layers are scaled by 0.25). Can we train a nonuniform US-Net where each layer can independently adjust its own ratio using our proposed methods? This requirement is especially important for related tasks like network slimming. Our answer is YES and we show a simple demonstration on how the nonuniform US-Net can help in network slimming.

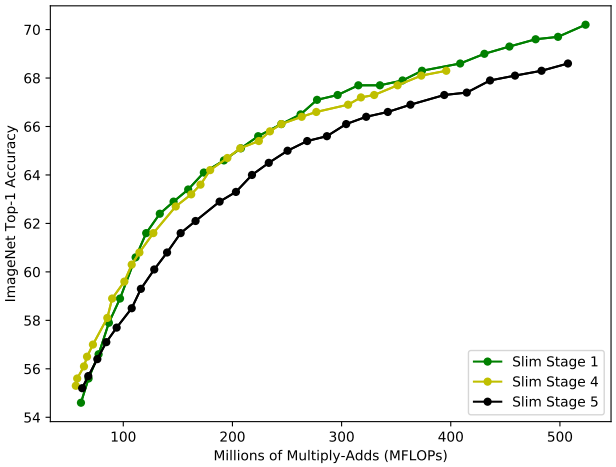

Figure 1. FLOPs-Accuracy spectrum of nonuniform US-MobileNet v1 tested with different slimming strategies. Note that each layer can adjust its own width ratio. The result suggests that *slimming the stage 5 of MobileNet v1 is not a good choice*.

In this demonstration, we first train a nonuniform US-MobileNet v1. The architecture of MobileNet v1 has 5 resolution stages with base channel number as 64, 128, 256, 512, 1024 in each stage. After training, we apply an additional width ratio 0.6 to one of five stages and get five models. Along with global width ratio, we can draw their FLOPs-Accuracy spectrum in Figure 1. For simplicity we only show performances of slimming stage 1, 4 and 5. Slimming stage 2 and 3 have curves close to that of slimming stage 1, while slimming stage 1 achieves the best results. Figure 1 shows that the stage 5 of MobileNet v1 may require more channels because slimming stage 5 has worst accuracy under same FLOPs. The result suggests *slimming the stage 5 of MobileNet v1 is not a good choice*. It further implicitly indicates that the stage 5 of MobileNet v1 network architecture needs a larger base channel number.

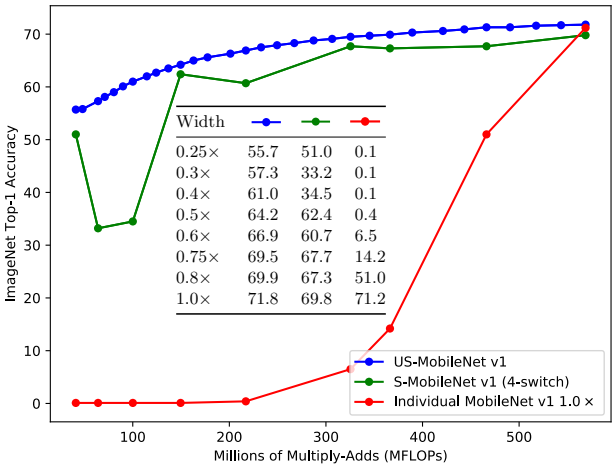

Figure 2. FLOPs-Accuracy spectrum of US-MobileNet v1, 4-switch S-MobileNet v1 and individual MobileNet v1 $1.0\times$ tested on different widths after BN calibration. The results suggest that deep neural networks are not naturally slimmable.

Naturally Slimmable? Perhaps the question is naive, but are deep neural networks naturally slimmable? We have proposed training methods and improved techniques for universally slimmable networks, yet we have not presented any result if we directly evaluate a trained neural network at arbitrary width either with naive training algorithm or slimmable training algorithm in [1]. If we can calibrate post-statistics of BN in these trained models (instead of using our proposed US-Nets training algorithm), do they have good performances? The answer is NO, both naively trained models and slimmable models [1] have very low accuracy at arbitrary widths even if their BN statistics are calibrated.

In Figure 2, we show results of a US-MobileNet v1, 4-

switch S-MobileNet v1 $[0.25, 0.5, 0.75, 1.0]\times$ and individually trained MobileNet v1 $1.0\times$. For individually trained MobileNet v1 $1.0\times$, it achieves good accuracy at width $1.0\times$, but fails on other widths especially when its computation is below 200 MFLOPs. For 4-switch S-MobileNet v1 $[0.25, 0.5, 0.75, 1.0]\times$, it achieves good accuracy at widths in $[0.25, 0.5, 0.75, 1.0]\times$, but fails on other widths that are not included in training. Our proposed US-MobileNet v1 achieves good accuracy at any width in the range from 40 MFLOPs to 570 MFLOPs consistently.

Averaging Output by Input Channel Numbers. In slimmable networks [1], private scale and bias γ, β are used as conditional parameters for each sub-network, which brings slight performance gain. These parameters comes for free because after training, they can be merged as $y' = \gamma' y + \beta', \gamma' = \frac{\gamma}{\sqrt{\sigma^2+\epsilon}}, \beta' = \beta - \gamma'\mu$.

In US-Nets, by default we share scale and bias. Additionally we propose an option that mimics conditional parameters: averaging the output by the number of input channels. It also brings slight performance gain as shown in Table 1. In this way, to some extent the *feature aggregation* can be viewed as *feature ensemble* in each layer.

Table 1. Performance comparison (top-1 error) of our default model (US-MobileNet v1) and model trained with **output averaging** (US-MobileNet v1 +).

Name	$0.25\times$	$0.5\times$	$0.75\times$	$1.0\times$	AVG
US-MobileNet v1	44.3	35.8	30.5	28.2	34.7
US-MobileNet v1 +	43.3	35.5	30.6	27.9	**34.3** (0.4)

In practice, it is important not to average depthwise convolution, because the actual input to each output channel in depthwise convolution is always single-channel. For networks with batch normalization, the proposed output averaging also come for free since these constants can be merged into BN statistics after training. At runtime when switch to different widths, a switch cost (*e.g.*, fusing new BN to its previous convolution layer) will be applied. But for networks without batch normalization, we should notice that if we do not use output averaging, there is no switch cost. Thus, the proposed output averaging is optional and is not used by default.

References

[1] Jiahui Yu, Linjie Yang, Ning Xu, Jianchao Yang, and Thomas Huang. Slimmable neural networks. *arXiv preprint arXiv:1812.08928*, 2018. 1, 2