MLO Research Notes
Monday 14th December, 2020

MLO Project Proposal – Spring 2021

Federated Slimmable/Reconfigurable Neural Network

Tao LIN*

1 Introduction

Recent trends in light-weight mobile network design consider training slimmable networks with different *width/depth/kernel size* configurations. For example, [YYX$^+$19] train a shared network with switchable batch normalization: at runtime, the network can adjust its width on the fly according to on-device benchmarks and resource constraints, rather than downloading and offloading different models. The primary training objective of [YYX$^+$19] (i.e., train a slimmable neural network) is to optimize its accuracy averaged from all switches: the loss of the model is computed by taking an unweighted sum of all training losses of different switches.

[CGW$^+$20] further extend [YYX$^+$19] and enable a much more diverse architecture space (depth, width, kernel size, and resolution) and a significantly larger number of architectural settings. Instead of directly optimizing the once-for-all network from scratch, [CGW$^+$20] first train the largest once-for-all network with maximum *depth*, *width*, and *kernel size*, then progressively fine-tune the once-for-all network to support smaller sub-networks that share weights with the larger ones.

Federated Learning (FL) is a machine learning setting where many devices collaboratively train a machine learning model while keeping the training data decentralized (and localized). [LKSJ20] first propose to utilize unlabeled dataset on the server-side for the ensemble distillation over heterogeneous neural architectures. However, the proposed scheme in [LKSJ20] only applies to the distinct model architectures with non-shared weights (e.g. ResNet-20, ResNet-32, and ShuffleNet-V2, are considered in their experiments).

Training a federated slimmable neural network allows flexible training and inference on edge devices with diverse hardware capacities. However, it remains unclear (1) how to train a slimmable neural network (a.k.a. once-for-all network) [YYX$^+$19, CGW$^+$20] in a federated fashion, (2) if such trained slimmable neural networks have more performance gain, or better compute/memory efficiency, than the naive choice in [LKSJ20].

In this project, we would like to explore federated slimmable neural network training:

- we should first identify the limitations (e.g. memory cost, inference efficiency, test performance) of the federated slimmable neural network training (or once-for-all network), e.g. by considering training different subnetworks through the scheme in [LKSJ20].
- can we use the optimal transport idea in [SJ20] for federated slimmable neural network training[2]?

[*] Machine Learning and Optimization Laboratory (MLO), EPFL

[2] [SJ20] present a layer-wise model fusion algorithm for neural networks that utilize optimal transport to (soft-) align neurons across the models before averaging their associated parameters. It also provides a principled way to combine

- can we design an efficient and effective federated learning algorithm?

References

[CGW+20] Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-all: Train one network and specialize it for efficient deployment. In *International Conference on Learning Representations*, 2020.

[LKSJ20] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

[SJ20] Sidak Pal Singh and Martin Jaggi. Model fusion via optimal transport. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

[YYX+19] Jiahui Yu, Linjie Yang, Ning Xu, Jianchao Yang, and Thomas Huang. Slimmable neural networks. In *International Conference on Learning Representations*, 2019.

the parameters of neural networks with different widths.