Weight Excitation: Built-in Attention Mechanisms in Convolutional Neural Networks

Niamul Quader[1], Md Mafijul Islam Bhuiyan[1], Juwei Lu[1], Peng Dai[1], and Wei Li[1]

Huawei Noah's Ark Lab, Canada
{niamul.quader1,juwei.lu,peng.dai,wei.li.crc}@huawei.com,
mbhuiyan@ualberta.ca

1 Robustness to weight initialization approaches

In our work, we used PyTorch's [9] default initialization method [7] for the fully connected layers of LWE. To verify robustness to initialization, we change inialization approach to Xavier initialization [1] for ResNet50[4] training on ImageNet [10]. We find similar top-1 accuracy of 77.1%. This shows preliminary evidence that the benefits of LWE may not be sensitive to the initialization of the weights of LWE.

2 Robustness to exploding and vanishing gradients

We trained ResNet50 without BN [5] (or other normalization) to introduce vanishing/exploding gradient - this resulted in failed optimization after a few iterations in first epoch; however, ResNet50-LWE is still able to optimize without such normalization modules and reaches 52.4% accuracy at 30 epoch. Thus, LWE is robust to exploding and vanishing gradients problems.

3 Effectiveness of LWE in pruning

Removing the most location-wise unimportant filter kernels (i.e., removing $h \times w$ filters having the lowest m values) in each layer results in 0.03% accuracy improvement on ResNet50-LWE trained on ImageNet. As a preliminary investigation on effectiveness in pruning using LWE-based training, on CIFAR10 [6], we use LWE-based convolution blocks in training the recent Dynamic Sparse Training pruning algorithm that jointly trains and prunes a ConvNet [8]. Using this pruning strategy on WideResNet-16-8-LWE reduces model parameters to around 10% of baseline WideResNet-16-8 without compromising much accuracy (0.1% loss in accuracy), whereas using the Dynamic Sparse Training pruning algorithm with WideResNet-16-8 reduces model parameters to around 11% of baseline at the loss of 0.13% accuracy. This preliminary analysis suggests that using LWE-based convolution during training can be helpful in pruning algorithms as well (e.g. [8, 3, 2]).

References

1. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 249–256 (2010)
2. Han, S., Pool, J., Tran, J., Dally, W.: Learning both weights and connections for efficient neural network. In: Advances in neural information processing systems. pp. 1135–1143 (2015)
3. Hassibi, B., Stork, D.G.: Second order derivatives for network pruning: Optimal brain surgeon. In: Advances in neural information processing systems. pp. 164–171 (1993)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
5. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
6. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Tech. rep., Citeseer (2009)
7. LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.R.: Efficient backprop. In: Neural networks: Tricks of the trade, pp. 9–48. Springer (2012)
8. LIU, J., XU, Z., SHI, R., Cheung, R.C.C., So, H.K.: Dynamic sparse training: Find efficient sparse network from scratch with trainable masked layers. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=SJlbGJrtDB
9. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
10. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)