Compact and Efficient Networks Through Weight Sampling

Hongzhi Liu

May 27, 2018

Abstract

Today, I read a thesis written by Jianchao Yang, a currently director of Toutiao AI Lab in Silicon Valley, whose expertise and research interests are on computer vision, machine learning, deep learning, and image/video processing. Before joining Toutiao, Dr. Yang was a manager and Principal Research Scientist at Snap and has published over 80 technical papers on top conferences and journals, which have attracted over 10K citations from the community. Besides, he is the receipt of Best Student Paper Award in ICCV 2011 and Best Paper Final List in ECCV 2016.

In this paper, Dr. Yang and his team present a novel network architecture, termed WSNet, for learning compact and efficient deep convolutional neural networks. Their new architecture can learn smaller and more efficient networks with competitive accuracy compared to the baseline conventional networks. Extensive experiments on multiple audio classification datasets and ImageNet verifies the effectiveness of WSNet.

1. Overview of WSNet

Deep neural networks (DNNs) usually suffer following two problems that stem from their inherent huge parameter space despite remarkable successes in various applications. One is over-fitting and the other is large amount of storage memory and energy consuming. Existing approaches learn model parameters independently, and when needed, apply different ways of model compression to reduce model size which lead to large performance drop.

In this paper, Dr. Yang proposes a Weight Sampling deep neural network (*i.e.* WSNet) to significantly reduce both the model size and computation cost [3], achieving better performance than the baseline (*i.e.* conventional networks that learn filters independently). Alternatively, WSNet can learn model parameters by sampling from a compact set of learnable parameters, which naturally enforces parameter sharing throughout the learning process. they demonstrate that such a novel weight sampling approach promotes both weights and computation sharing favorably. By employing this method, the team can more efficiently learn much smaller networks with competitive performance compared to baseline networks with equal numbers of convolution filters.

2. Method of Weight Sampling

WSNet uses a condensed filter per convolutional layer. To quantize the advantage of WSNet in achieving compact networks, Dr. Yang defines the compactness of K in a learned layer in WSNet w.r.t. the conventional layer with independently learned weights as Equation (1):

$$compactness\ K = \frac{LMN}{L*M*}. \tag{1}$$

where K is the 1D convolution kernel used in the actual convolution of WSNet which has the shape of $(L; M; N)$ where L is the kernel size. In WSNet, instead of learning each weight independently, K is obtained by sampling from a learned *condensed filter* Φ which has the shape of $(L*, M*)$. The goal of training WSNet is thus cast to learn more compact DNNs which satisfy the condition of $L*M* < LMN$.

In conventional CNNs, the filters in a layer are learned independently which presents two disadvantages that is a large number of parameters and overfitting. To solve these two problems, a novel weight sampling method is proposed to efficiently reuse the weights among filters. Specifically, in each convolutional layer of WSNet, all convolutional filters K are sampled from the *condensed filter* Φ, as illustrated in Figure 1. By scanning the weight sharing filter with a window size of L and stride of S, they could sample out N filters with filter size of L. Formally, the equation between the filter size of the condensed filter and the sampled filters is as Equation (2):

$$L* = L + (N-1)S. \tag{2}$$

The compactness along spatial dimension is $\frac{LM*N}{L*M*} \approx \frac{L}{S}$. Note that since the minimal value of S is 1, the minimal value of $L*$ (*i.e.* the minimum spatial length of the condensed filter) is $L + N - 1$ and the maximal achievable compactness is therefore L.

To eliminate such redundancy in convolution and speed-up WSNet, Dr. Yang proposes a novel integral image method to enable efficient computation via sharing computations. They first calculate an inner product map $P \in \mathbb{R}^{T \times L*}$ which stores the inner products between each row vector in the input feature map (*i.e.* F) and each column vector in the condensed filter (*i.e.* Φ) as Equation (3):

$$P(u,v) = \begin{cases} F_{u,:}\Phi_v, & u \in [0, T-1] \; and \; v \in [0, L*-1], \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

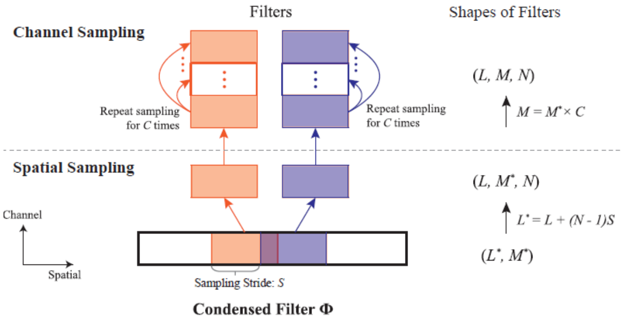

Figure 1. Illustration of WSNet that learns small condensed filters with weight sampling along two dimensions: spatial dimension (the bottom panel) and channel dimension (the top panel). The figure depicts procedure of generating two continuous filters (in pink and purple respectively) that convolve with input. In spatial sampling, filters are extracted from the condensed filter with a stride of S. In channel sampling, the channel of each filter is sampled repeatedly for C times to achieve equal with the input channel.

Finally, the team note that the integral image method applied in WSNet naturally takes advantage of the property in weight sampling: redundant computations exist between overlapped filters and input patches.

Table 1. Test error rates (in %) of WSNet and HashNet on CIFAR10 and MNIST. The baselines used for MNIST/CIFAR10 are simple 3-layer fully connected network and 5-layer convolutional network respectively. The model size is provided for the baseline.

Model	Model size	Error rate	Model size	Error rate
	CIFAR10		MNIST	
baseline	1.2M(×)	14.91	800K(1×)	1.37
HashNet	16(×)	21.42	8(×)	1.43
HashNet	64(×)	30.79	64(×)	2.41
WSNet	16(×)	17.82	8(×)	1.29
WSNet	64(×)	23.59	64(×)	1.97

3. Experimental Results of WSNet on 2D CNNs

Since both WSNet and HashNet [2] explore weights tying, Dr. Yang compares them on MNIST and CIFAR10. For fair comparison, the team use the same baselines used in [1]. For each dataset, they hold out 20% of training samples to form a validation set.

The comparison results between WSNet and HashNet on MNIST/CIFAR10 are listed in Table 1, from which one can observe that when learning networks with the same sizes, WSNet achieves significantly lower error rates than HashNet on both datasets. Above results clearly demonstrate the advantages of WSNet in learning compact models.

References

[1] W. Chen, J. Wilson, S. Tyree, K. Q. Weinberger, and Y. Chen. Compressing convolutional neural networks in the frequency domain. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016. 2

[2] W. Chen, J. T. Wilson, S. Tyree, K. Q. Weinberger, and Y. Chen. Compressing neural networks with the hashing trick. *CoRR*, 2015. 2

[3] X. Jin, Y. Yang, N. Xu, J. Yang, J. Feng, and S. Yan. Wsnet: Compact and efficient networks with weight sampling. *CoRR*, 2017. 1