Supplementary Material of 'Filter Grafting for Deep Neural Networks'

Fanxu Meng[1,2*], Hao Cheng[2*†], Ke Li[2], Zhixin Xu[1], Rongrong Ji[3,4], Xing Sun[2], Guangming Lu [1†]
[1] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
[2] Tencent Youtu Lab, Shanghai, China
[3] Department of Artificial Intelligence, School of Informatics, Xiamen University, China
[4] Peng Cheng Laboratory, China
{louischeng, tristanli, winfredsun}@tencent.com, {18S151514,xuzhixin}@stu.hit.edu.cn, luguangm@hit.edu.cn, rrj@xmu.edu.cn

Abstract

This is the supplementary material for the paper "Filter Grafting for Deep Neural Networks". Section 1 proves the locations of invalid filters are statistically different among networks. Section 2 shows layer consistency is essential for grafting algorithm. Section 3 further proves to keep layer consistency, the layer's information should be calculated from Equation (6) rather than Equation (5) from the main paper. Section 4 compares adaptive weighting strategy with fixed weighting strategy for grafting algorithm. Section 5 shows the sensitivity of grafting algorithm regarding to the hyper-parameters. Section 6 discusses the differences between grafting and swa. Section 7 proves the Theorem 1 from main paper.

1. Locations of Invalid Filters

We mentioned in Section 3.1.3 from the main paper that since two networks are initialized with different weights, the locations of invalid filters are statistically different. In this part, we perform an experiment to verify our claim. Specifically, we parallelly train two networks with the same structure and record the invalid filters in each layer of each network (20% filters are counted as 'invalid' in each layer). Then by calculating IoU (Intersection over Union) for the positions of invalid filters, we could verify our statement. A small IoU means that the locations of invalid filters are mostly different between two networks.

model	layer-5	layer-10	layer-15
ResNet32	0.00	0.00	0.20
MobileNetV2	0.05	0.14	0.17

Table 1. IoU for invalid filters' location.

From Table 1, the results have proved that the locations of invalid filters are statistically different between networks. Thus there exists little chance that the weight of an invalid filter is grafted into another invalid filter.

2. Layer Consistency

In Section 3.1.3 of the paper, we mentioned that to keep layer consistency, we should graft the weight in layer level instead of filter level. We perform an experiment on two networks M_1 and M_2 to verify our claim. For filter level grafting, we sort filters by entropy in M_2 to get the invalid ones, and graft corresponding filters from M_1 into M_2. To get a fair comparison, hyper-parameters are equally deployed for two methods. From Table 2, layer level grafting performs better than filter level grafting.

model	method	CIFAR-10	CIFAR-100
ResNet32	filter level	93.49	70.79
	layer level	**93.94**	**71.28**
ResNet56	filter level	94.33	72.29
	layer level	**94.73**	**72.83**
ResNet110	filter level	94.09	74.24
	layer level	**94.96**	**75.27**
MobileNetv2	filter level	92.66	72.70
	layer level	**94.20**	**74.15**

Table 2. Filter level grafting vs. layer level grafting

3. Two forms of the Layer Information

When calculating the layer information, we propose two forms (Equation (5) and Equation (6)) in the main paper. Equation (5) calculates the layer information as the sum of all the filter's information in a certain layer. But when two filters are identical in the same layer, one is redundant for the other. Equation (5) merely sums all filters' information, which neglects the correlation among filters, while Equation (6) takes such correlation into consideration and perform entropy calculation on the whole layer. We perform an experiment with different entropy calculations and results are listed in Table 3.

model	method	CIFAR-10	CIFAR-100
ResNet32	Equation (5)	93.89	70.95
	Equation (6)	**93.94**	**71.28**
ResNet56	Equation (5)	94.40	72.03
	Equation (6)	**94.73**	**72.83**
ResNet110	Equation (5)	94.48	74.34
	Equation (6)	**94.96**	**75.27**
MobileNetv2	Equation (5)	93.41	72.86
	Equation (6)	**94.20**	**74.15**

Table 3. Different methods for calculating the layer information.

From Table 3, compared with Equation (5), Equation (6) shows more appealing performance improvements and is thus a better way to calculate the layer information.

4. Efficiency of Adaptive Weighting Strategy

We perform an experiment to compare adaptive weighting and fixed weighting strategies in Table 4. For fixed weighting, α is fixed to be 0.5 in (2) from the main paper. From Table 4, adaptive weighting performs better on each dataset and network structure, which proves the efficiency of adaptive strategy.

model	method	CIFAR-10	CIFAR-100
ResNet32	fixed weighting	93.22	70.70
	adaptive weighting	**93.94**	**71.28**
ResNet56	fixed weighting	94.54	72.25
	adaptive weighting	**94.73**	**72.83**
ResNet110	fixed weighting	94.21	73.88
	adaptive weighting	**94.96**	**75.27**
MobileNetv2	fixed weighting	93.48	73.52
	adaptive weighting	**94.20**	**74.15**

Table 4. Comparison of adaptive weighting and fixed weighting.

5. Sensitivity of Hyper-parameters

In the Equation (7) of the paper, there are two hyper-parameters A and c. We perform experiments regarding to the variations of A and c in this section. The results are listed in Table 5.

A	c	CIFAR-10	CIFAR-100
0.4	1	92.42	71.44
0.4	5	92.76	72.69
0.4	10	93.31	73.26
0.4	50	93.24	73.05
0.4	500	92.79	72.38
0	100	93.4	72.55
0.2	100	93.61	72.9
0.4	100	93.46	73.13
0.6	100	92.6	72.68
0.8	100	93.03	71.8

Table 5. Results regarding to the variation of hyper-parameters. The model is MobileNetV2.

6. The Difference Between Filter Grafting and SWA

In this section, we discuss the difference between filter grafting and swa (stochastic weight averaging) as follows:

- Our motivation is different from swa. Swa performs weight averaging from the perspective of the loss optimization, however we consider how to activate the invalid filters **from other networks** by performing weight grafting. From Section 4.1 in the main paper, we show that even grafted from noise, the network still have better perfromance. From Section 4.8 in the main paper, we show that the grafted network has more valid filters compared to the untouched state.
- Grafting is not a simple weight averaging algorithm. We propose entropy-based criterion and adaptive function in the paper to effectively perform grafting in the experiments.

7. Proof of Theorem 1

Theorem 1 *Suppose there are two filters in a certain layer of the network, denoted as random variables X and Y. Z is another variable which satisfies $Z = X + Y$, then $H(X,Y) = H(X,Z) = H(Y,Z)$, where H denotes the entropy from information theory.*

Proof 1 *We first prove $H(Z|X) = H(Y|X)$:*

$$\begin{aligned}
&H(Z|X)\\
&= -\sum_x p(X=x)\sum_z p(Z=z|X=x)\log P(Z=z|X=x)\\
&= -\sum_x p(X=x)\sum_z p(Y=z-x|X=x)\log P(Y=z-x|X=x)\\
&= -\sum_x p(X=x)\sum_y p(Y=y|X=x)\log P(Y=y|X=x)\\
&= H(Y|X)
\end{aligned}$$

Then, according to the principle of entropy:

$$\begin{aligned}
H(X,Y) &= H(X) + H(Y|X)\\
&= H(X) + H(Z|X)\\
&= H(X,Z)
\end{aligned}$$

By symmetry of entropy, the other direction also holds. Thus:

$$H(X,Y) = H(X,Z) = H(Y,Z)$$