RePr 训练方法探讨

闫心刚
21832147
控制科学与工程学院
1928847546@qq.com

摘要（Abstract）

本文介绍了RePr[1]的训练方法，并阐述了在复现论文中遇到的问题，同时表达了一些思考。

1. 介绍

RePr (Re-initializing and Pruning)是一种训练方法，概括而言，是通过一定的规则将部分卷积核剪去，重新训练恢复一定的精度，再重新初始化已经剪去的卷积核，是网络恢复到原来的大小，不断重复这样的训练过程几次后，得到的模型测试的精度将会比传统的训练方式提高几个百分点。下面的部分主要介绍RePr方法的主要步骤。

1.1. 算法流程

①训练完整的卷积神经网络，训练过程会有一定的稀疏化方法，为了使网络权重的分布稀疏，便于后面剪枝剪去不重要的卷积核和重复的卷积核。

②通过一定的剪枝规则将不重要的卷积核或者重复的卷积核剪去。

③训练剪枝后的卷积神经网络，恢复一定的精度。

④将步骤②中剪去的卷积核权重数据重新初始化，添加到剪枝后的神经网络中。训练一定的epcoh。

⑤迭代一定次数②③④步骤，完成训练过程。

1.2. 剪枝算法

论文中作者通过卷积核之间的正交性来判断应该剪去哪些卷积核。所谓的正交性就是判断卷积核之间的向量夹角。如果两个卷积核之间的向量夹角越大，说明两个卷积核正交性越强，如果两个卷积核之间的向量夹角越小，说明两个卷积核的正交性越弱。

具体的剪枝算法步骤如下：

将一层中的$k\times k\times c$的卷积核展开为$k*k*c$的向量，表示为f。一层中有$J\ell$个卷积核，$\boldsymbol{W}\ell$为一层中卷积核所组成的矩阵，矩阵的行数为该层中卷积核的数目$J\ell$ 。

对于矩阵$\boldsymbol{W}\ell$做标准化：

$$\hat{\boldsymbol{W}}_\ell = \boldsymbol{W}_\ell / \|\boldsymbol{W}_\ell\|$$

标准化的目的是将每个卷积核向量变为单位向量。后面需要求一层之间卷积核向量两两之间的夹角。

$$\boldsymbol{P}_\ell = \left| \hat{\boldsymbol{W}}_\ell \times \hat{\boldsymbol{W}}_\ell^T - I \right|$$

上式求得的P_l是一个$J\ell\times J\ell$的对称矩阵，公式中减去单位矩阵的原因在于先前对于$\boldsymbol{W}\ell$进行了标准化，每行所代表的卷积核向量为单位向量，这样$\boldsymbol{W}\ell$与其转置矩阵相乘之后，对角线元素是单位 1。

后面需要判断一个卷积核向量和该层中其他所有的卷积核向量的正交性。

$$O_\ell^f = \frac{\sum P_\ell[f]}{J_\ell}$$

$P_l[f]$表示为矩阵P_l的第i行。论文中认为如果第i行所对应的卷积核与其他卷积核正交，那么P_l中对应行相加的和是小的，如果第i行所对应的卷积核与其他卷积核不正交，那么P_l中对应行相加的和是大的。

后面论文中按照这个标准对于神经网络的卷积核进行排序，将O_l^f中值相对较大的所对应的卷积核剪去，因为O_l^f的值相对比较大，说明该卷积核与该层中其他卷积核有在向量方向上有更大的相似度。

1.3. 重新初始化已经剪去的卷积核

在完成剪枝后的重训练过程之后，会重新初始化已经剪去的卷积核。所应用的规则是，是已经剪去的卷积核和保留的卷积核正交，其中应用了QR分解去寻找null-sapce用来寻找初始化点。

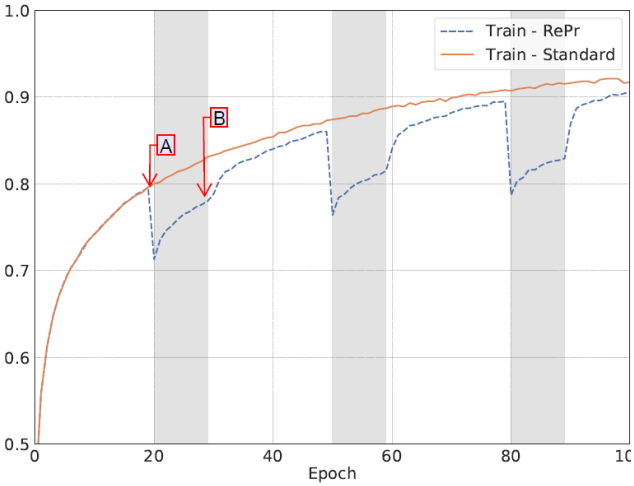
[1] Prakash et al., “RePr.”

1.4. 部分结果分析

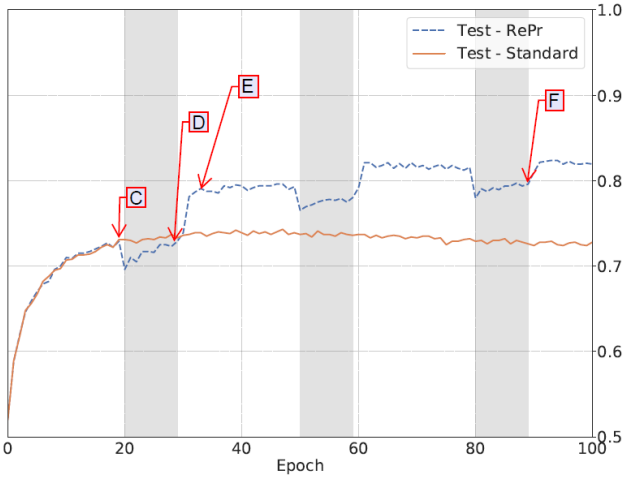
论文构造了一个 3 层的卷积神经网络，每个卷积层有 96 个卷积核，网络结构如下：

$$\text{Img} \longmapsto \left[\text{CONV}(X) \to \text{RELU}\right]^{n} \longmapsto \text{FC} \longmapsto \text{Softmax}$$

其中 n 的值为 3，训练的方法为SGD。



图一： 训练准确度变化曲线



图二：测试准确度变化曲线

从上面的图中可以有下面的发现：

①剪枝之后，精度立刻下降，但是训练几个epoch之后，精度会有一定的恢复。

②重新将已经剪去的卷积核权重重新初始化之后，比如在D点，准确度是连续的，虽然网络结构变化了，但是在此刻准确度却没有变化。

③从训练曲线和测试曲线的对比上来看，使用RePr训练方法对训练集而言，精度不如标准的训练方法高，但是在测试数据集上，使用RePr的训练方法准确度更高。这说明，RePr的训练方法也许在处理过拟合问题上有明显的效果。

④在达到一定的(剪枝-训练-重新初始化剪枝权重-训练)这样的步骤迭代之后，再增加迭代的次数，不会对准确度的变化有明显效果了。

2. 分析点评

关于卷积核的冗余，是一个已经被讨论过很多次的问题了，也有很多的剪枝算法用于卷积核的剪枝，将不重要的卷积核剪去，保留重要的卷积核，以实现模型压缩的目的。

初看来，论文的方法是一种剪枝方法，实则不是，作者是从训练的角度去思考这个问题的。Song Han的DSD中有相似的思想，是剪枝之后，将剪枝的权重值保存再零，再训练，发现性能有所提升。Song Han的这篇文章，本人尚未复现。

所以，本人看来，RePr之所以有效的关键点在于重新初始化已经剪枝的权重数据，因为前面的剪枝方法实际上是很普通的，有很多比这种剪枝算法更好的模型压缩算法。

但是，在论文的复现过程中发现了一些问题。主要问题是关于重新初始化已经剪枝的权重数据上的事情。按照文中的说明是通过QR分解求解Null Sapce来初始化正交点，但是很多情况下是求解不出Null Space的。后来用随机初始化的方式进行了一定测试，但是效果并不好，重新初始化已经剪枝的权重再训练之后并不会使得网络有更好的精度表现。所以对于文中声称的极佳的表现，我表示是要存疑的。

后面又做了下面的测试，将Network Slimming[2]的剪枝方法配合这种重训练的方法使用。与文中作者相同，构建了一个三层的卷积神经网络，训练一定的epoch之后，根据Network Slimming中的方法将不重要的卷积核剪去，保留剩余的卷积核重训练，再训练一定的epoch之后，将剪去的卷积核随机初始化之后再添加进网络中，实际的结果是效果没有明显的变化。***[github: https://github.com/Yanxingang/RePr]***

References

[1] Prakash, Aaditya, James Storer, Dinei Florencio, and Cha Zhang. “RePr: Improved Training of Convolutional Filters.” *ArXiv:1811.07275 [Cs]*, November 18, 2018. http://arxiv.org/abs/1811.07275

[2] Liu, Zhuang, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. “Learning Efficient Convolutional Networks through Network Slimming.” *ArXiv:1708.06519 [Cs]*, August 22, 2017. http://arxiv.org/abs/1708.06519.