Headline

大家好：

这是**2018年度第6篇Arxiv Weekly**。

本文是 模型研究 方向的文章。

Highlight

> 通过弹性网络架构压制梯度消失来提升部分**backbone**的性能，同时赋予网络快速**inference**的能力

Information

Title

Elastic Neural Networks for Classification

Link

https://arxiv.org/pdf/1810.00589.pdf

Source

- 坦佩雷理工大学（Tampere University of Technology）
- 马里兰大学学院市分校（University of Maryland）

Introduction

本文提出了一种通过解决梯度消失问题提升神经网络性能的架构，能够运用在任何深度神经网络当中。

这个架构通过在每个层后插入一个中间输出分支，并将输出分支的loss融合在最后的loss中，使得梯度反传的时候每个layer的输出分支都有自己的贡献。

我们将这个架构命名为弹性网络（Elastic Network），并在CIFAR上进行了综合实验。试验结果表明弹性网络架构在某些小网络（MobileNet）和某些深度网络（DenseNet）上都表现良好，能够提升网络的acc。而在另外一些backbone上弹性网络表现不佳，文章中也讨论了可能的原因。

另外值得一提的是，加入弹性架构的网络天然可以通过early exit的方式加速inference，或者说在性能和效率中方便地进行折中。这可以说是弹性网络架构的一个有趣的副产物。

Keys

可以说梯度消失的问题是阻碍神经网络获得良好性能的关键问题。而许多主流的技术之所以成为主流，正是因为它们解决了这样的问题。例如 **ReLU**激活函数（解决**softmax**、**tanh**等激活函数边缘消失的现象）、正则化

（**BN**、**dropout**、**loss**中的**L1/L2**惩罚项等）、含**identity path**的**resnet**系列架构（给梯度一个无损传递的路径）等技术。

不过有趣的是，传统的技术大多试图尽可能让梯度按原路径传递的同时保留下来，等于给梯度装上盔甲。基本没有主流技术采用将梯度直接跨过消耗性中间步骤，通过short cut直达终点的思路。

本文针对这个问题提出了另一种解决方案，将原有架构改成弹性的（Elastic）的，在每个层后面抽象出来一个中间变量层和中间变量loss，最终把所有的loss联合起来构成final loss。这就为每一层的梯度搭建了一个无损传递的桥梁。其架构示意图如下：

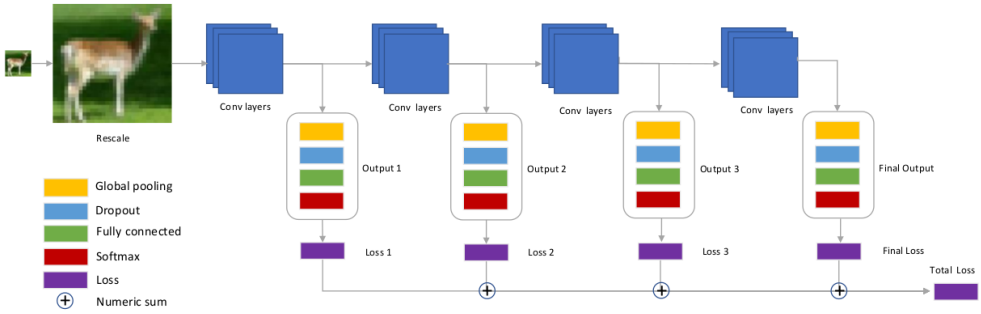

Fig.1 Elastic Network Architecture

实际上，之前的一些工作表明这样添加extra output的方法是有其优势的：

- Inception的分支结构中添加auxiliary output后，在ImageNet1k上的acc涨了0.4%。
- Deeply-supervised Nets中的实验表明在每个layer后添加SVM classifier后performance更加稳定，网络的收敛性也提高了。
- 有人尝试在AlexNet中加入两个中间输出层，结果同时提升了early和final输出的acc。

本文的工作实际上是对extra output思路的一种极限推广， 也即每个layer/block（特别深的网络在每个layer后均添加太密集了，不可行）后都附加一个extra output、每个extra output都有相同的结构（Global pooling->Dropout->Fully connected->Softmax）。示例如下图所示：

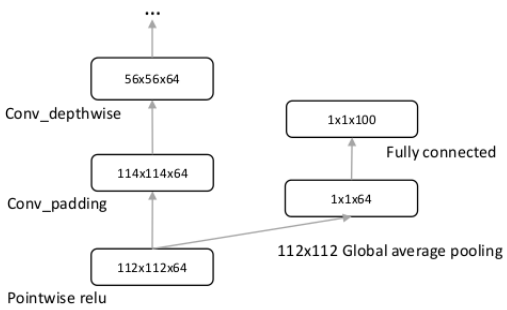

Fig.2 The intermediate output on the first “depth-conv” block in MobileNet on CIFAR 100

另外，可以提一提loss函数的细节。本文当中采用的就是softmax loss，如下图所示

$$L_i(\hat{\mathbf{y}}^{(i)}, \mathbf{y}^{(i)}) = -\frac{1}{C} \sum_{c \in \{1,\dots,C\}} y_c^{(i)} \log \hat{y}_c^{(i)},$$

Fig.3 loss function for layer i, where $\hat y$ is calculated from softmax and y is the one hot ground truth label vector.

并且在每一层完成计算后，通过加权求和给出最终的loss。文章中没有探索最佳的权重设定方式，直接都设计为1。

$$L_{total} = \sum_{i=1}^{N} w_i L_i \left(\hat{\mathbf{y}}^{(i)}, \mathbf{y}^{(i)} \right),$$

Fig.4 the final loss function.

Results

文章象征性地在CIFAR上进行了实验。数据没有增强，仅仅resize到了224×224×3的大小上。实验平台是Keras。

在训练时， 由于过多的dropout的引入，在前面的数十个epochs会固定pretrained network，只对Elastic部分的新增模块进行微调。然后再放开全部网络整体训练。

TABLE I: Testing error rate (%) on CIFAR 10 and CIFAR 100.

Model [1]	CIFAR-10			CIFAR-100		
	w/o Elastic Structure	w/ Elastic Structure	Improvement	w/o Elastic Structure	w/ Elastic Structure	Improvement
DenseNet-121	6.35	**5.44**	14.3%	24.48	**23.44**	4.3%
DenseNet-169	8.14	**5.58**	31.5%	23.06	**21.65**	6.1%
Inception V3	6.39	**4.48**	29.9%	24.98	**21.83**	12.6%
MobileNet	10.37	**7.82**	24.6%	38.60	**25.22**	34.7%
VGG-16	**7.71**	8.20	-6.4%	38.06	**33.03**	13.2%
ResNet-50	**5.54**	6.19	-11.7%	**21.96**	24.20	-10.2%
ResNet-50-5	5.54	**5.39**	2.7%	21.96	**21.54**	1.9%
PyramdNet + ShakeDrop[2]	2.3	-	-	12.19	-	-

[1] All the models we use are pretraiend on ImageNet
[2] the state of the art accuracy [15]

Fig.5 Testing Error on CIFAR.

从上图可以看出，Elastic架构在MobileNet和DenseNet这样的架构上还是相当有效的；而对比之下：在ResNet这样把梯度消失问题解决地较有力度的架构中；或者 在VGG这样的浅层网络中，密集添加Elastic模块则会适得其反。

Model	# conv layer	params	error
Elastic-DenseNet-169-output-14	14	0.39M	73.37
Elastic-DenseNet-169-output-39	39	1.47M	48.96
Elastic-DenseNet-169-output-104	104	6.71M	22.86
Elastic-DenseNet-169-output-168	168	20.90M	21.65
DenseNet-169	168	20.80M	23.06
DenseNet-121	120	12.06M	24.48

Fig.6 Testing Error on DenseNet-169 on different depth.

上图描述了另一件有意思的事情： 添加Elastic架构后，网络天然可以兼容early exit。对比相同层数的完整网络和大网络early exit，似乎在DenseNet+CIFAR的benchmark下，后者是明显占优的。

Insights

本文是比较有意思的一篇文章，值得一提的是其实Elastic这个概念是很多元化的，上网查询能够查到很多和本文中Elastic Network毫不相关的Elastic Network……

Elastic Net其实就是把early exit模块化并且密集插入训练的一种尝试，感觉上有一定的道理。

缺憾很明显：文章的实验给予CIFAR，并且对比很不完善。因此我们要进一步探究的话首先是补全实验，然后是探索里面细节的原理和改动的可能，尽量让它在ResNet系列网络中也能够work。这个想法才有大规模落地的意义。