·人工智能与模式识别·　　文章编号：1000-3428(2022)05-0067-07　　文献标志码：A　　中图分类号：TP391

融合弱层惩罚的卷积神经网络模型剪枝方法

房志远，石守东，郑佳馨，胡加钿

（宁波大学 信息科学与工程学院，浙江 宁波 315211）

摘　要：深度卷积神经网络的存储和计算需求巨大，难以在一些资源受限的嵌入式设备上进行部署。为尽可能减少深度卷积神经网络模型在推理过程中的资源消耗，引入基于几何中值的卷积核重要性判断标准，提出一种融合弱层惩罚的结构化非均匀卷积神经网络模型剪枝方法。使用欧式距离计算各层卷积核间的信息距离，利用各卷积层信息距离的数据分布特征识别弱层，通过基于贡献度的归一化函数进行弱层惩罚，消除各层间的差异性。在全局层面评估卷积核重要性，利用全局掩码技术对所有卷积核实现动态剪枝。在CIFAR-10、CIFAR-100和SVHN数据集上的实验结果表明，与SFP、PFEC、FPGM和MIL剪枝方法相比，该方法剪枝得到的VGG16单分支、Resnet多分支、Mobilenet-v1轻量化网络模型在保证精度损失较小的情况下，有效地减少了模型参数量和浮点操作数。

关键词：模型剪枝；弱层惩罚；全局掩码；欧式距离；核重要性评估

开放科学（资源服务）标志码（OSID）：

中文引用格式：房志远，石守东，郑佳馨，等.融合弱层惩罚的卷积神经网络模型剪枝方法[J].计算机工程，2022，48(5)：67-73.

英文引用格式：FANG Z Y，SHI S D，ZHENG J Q，et al.Pruning method of convolutional neural network model with weak layer penalty[J].Computer Engineering，2022，48(5)：67-73.

Pruning Method of Convolutional Neural Network Model with Weak Layer Penalty

FANG Zhiyuan，SHI Shoudong，ZHENG Jiaqing，HU Jiadian

（Faculty of Electrical Engineering and Computer Science，Ningbo University，Ningbo，Zhejiang 315211，China）

【Abstract】The extensive demand of convolutional neural networks for memory and computation makes it difficult to deploy them in resource-constrained embedded devices.To minimize the resource consumption of the deep convolutional neural network model during the inference process，this study introduces a criterion for judging the importance of the convolution kernel，based on the geometric median and further proposes a structured，non-uniform pruning method for convolutional neural network models with weak layer penalty. First，using the Euclidean distance，the algorithm calculates the information distance for each layer of the convolution kernel.Then，the data distribution characteristics of the information distance of each convolutional layer are used to identify the weak layers，and a normalization function based on the contribution degree is proposed to eliminate the difference between layers while weakening the redundant layers.Second，the importance of the convolution kernel is evaluated at the global level，and the global mask technique is used to achieve dynamic pruning. The experimental results on the CIFAR-10，CIFAR-100，and SVHN datasets demonstrate that compared with SFP，PFEC，FPGM，and MIL pruning methods，the proposed method prunes the VGG16 single-branch，Resnet multi-branch，and Mobilenet-v1 lightweight network models，effectively reducing the number of model parameters and Floating Points of Operations（FLOPs）while ensuring that the loss of precision is small.

【Key words】model pruning；weak layer penalty；global mask；Euclidean distance；kernel importance evaluation

DOI：10. 19678/j. issn. 1000-3428. 0061461

0　概述

目前，深度卷积神经网络已在计算机视觉、语音识别、自然语言处理等领域[1-3]取得了重大突破。但由于深度学习模型的计算和存储需求巨大，在不断更新任务精度的同时，模型参数量和网络深度也随之增长，因此很难在一些资源受限的嵌入式设备上进行部署。针对该问题，研究人员提出了一系列解决方法，这些方法主要包括低秩近似[4]、知识蒸馏[5]、轻量化网络结构[6]、模型剪枝[7-8]等。它们从不同的

基金项目：宁波市公益项目“基于深度学习的儿童学习姿态识别系统研究与实现”（2019C50020）。

作者简介：房志远（1995—），男，硕士研究生，主研方向为模型压缩与加速；石守东，副教授、博士；郑佳馨、胡加钿，硕士研究生。

收稿日期：2021-04-26　　**修回日期：**2021-06-19　　**E-mail：**565463192@qq.com

角度考虑如何尽可能减少模型在推理过程中所需的随机存储器(工作内存)、处理器计算(推理代价)、闪存(存放模型)等资源。

模型剪枝方法作为模型压缩的重要分支,目前发现其可在保证精度没有显著下降的同时大幅减少模型大小和浮点操作数(Floating Points of Operations,FLOPs)。需要注意的是,卷积神经网络模型的低层卷积核趋向于提取粗级别特征(如点和线),高层卷积核则趋向于提取抽象特征(如常见的目标和形状)。因此,对于一个模型而言,每一层对最终模型的精度影响或贡献是不一样的。此外,考虑到在剪枝和训练期间网络权重的重要性是动态变化的[9],在剪枝过程中对其进行动态更新在一定程度上可以提升模型精度。

本文提出一种融合弱层惩罚的结构化模型剪枝方法。在局部层面,使用欧式距离计算各层中所有卷积核的信息距离,同时利用各层相关性值的数据分布特征判别层重要性,并对弱层中的卷积核进行惩罚。在训练与剪枝过程中,通过全局掩码技术对每一个卷积核实现动态剪枝,每次剪枝算法会从全局层面评估每一个卷积核的冗余性。

1 相关工作

模型剪枝技术是通过去除模型中的冗余参数和结构来实现深度神经网络的推理加速。现有模型剪枝方法可分为结构化和非结构化模型剪枝。

非结构化剪枝也称为权重剪枝,这些方法注重于剪枝卷积核的细粒度权重。文献[10]提出使用二进制掩码来检验连接神经元是否被剪枝,并且考虑对已剪枝神经元进行恢复,从而减少过度剪枝所带来的精度影响,在一定程度上保证了精度。但该方法需要通过专用的稀疏矩阵操作库或硬件实现加速,且这样不规则的结构很难利用现有的基本线性代数子程序库[11]。文献[12]对基于“范数小重要性低”的剪枝标准进行研究,提出基于范数标准剪枝的2个依赖条件:1)核的范数分布应该足够大;2)核的最小范数值应该非常小。基于此,又提出新的基于几何中值的卷积核重要性判断标准。文献[13]为解决“硬剪枝”在训练过程中的不可恢复性,提出利用“软”方式进行动态剪枝,在训练过程中可对已剪枝核的权值进行更新。文献[14]提出一种融合卷积层和BN层双层参数信息的动态剪枝方法,该方法利用注意力机制以及BN层缩放系数选择冗余卷积核。文献[15]为加速嵌入式端的表现,采用混合网络剪枝进一步减少网络中的冗余参数并加速网络。文献[16]首先将BN层的缩放因子与输出相乘,接着联合训练网络权重和这些缩放因子,然后将较小缩放因子的通道剪枝,最后微调剪枝后的网络。但是上述方法均存在以下问题:一方面,通常层采用固定/均匀剪枝率对各卷积层实施剪枝,忽略了各层之间的差异性;另一方面,在一层内通过局部重要性评估得到卷积核,无法说明其对于整个模型的重要性。本文将考虑对各卷积层的重要性进行判断并进行惩罚,从全局层面进行评估,改善误剪导致的精度下降问题。

对于结构化剪枝,现阶段研究人员提出了自动搜索网络结构的方法,该方法考虑各层之间的差异性,自动探索和学习网络架构,最终得到一个结构化非均匀的剪枝模型。文献[17]提出一种完全可微分的稀疏性方法,可以使用随机梯度下降方法同时学习网络的权重和稀疏结构。文献[18]使用强化学习的方法实现自动剪枝权重和卷积核,该方法得到了不错的效果,但训练成本较大。文献[19]将预训练好的模型直接部署在资源受限的手机平台上进行压缩,最后通过评估压缩后的直接性能表现进行反馈。文献[20]利用生成器产生多个候选剪枝策略,每一个剪枝策略为各层剪枝率的组合,再通过基于自适应BN层的候选评估模块挑选出最有可能的候选策略并进行微调,该方法大幅降低了剪枝时间代价,但训练过程相对复杂。在多个模型和数据集上的实验结果表明,该方法在保证精度损失较小的同时,有效地减少了模型参数量和FLOPs。

本文引入基于几何中值的卷积核重要性判断标准[12],提出一种融合弱层惩罚的结构化非均匀模型剪枝方法。由于文献[12]利用几何中值理论证明了距离各层中几何中值较近的卷积核可被该卷积层中其他卷积核替代,因此对这些卷积核进行剪枝,对最终模型精度影响较小。本文利用该方法中核重要性判断标准,实现了层重要性判断和惩罚,并在全局层面进行重要性评估。

2 融合弱层惩罚的结构化模型剪枝方法

2.1 符号与定义

假设一个卷积神经网络有L层,使用C_i和C_{i+1}分别表示i_{th}卷积层的输入输出通道数,$\boldsymbol{F}_{i,j}$表示i_{th}层的j_{th}卷积核,其中:$\boldsymbol{F}_{i,j}$的维度为$\mathbb{R}^{C_i\times K\times K}$,$K$表示核的尺寸。$i_{\text{th}}$层的输入特征图$\boldsymbol{S}$和输出特征图$\boldsymbol{O}$分别为$C_i\times H_i\times W_i$和$C_{i+1}\times H_{i+1}\times W_{i+1}$,$i_{\text{th}}$层的权值$W_i$可表示为$\{\boldsymbol{F}_{i,j},1\leqslant j\leqslant C_{i+1}\}$。因此,$i_{\text{th}}$层的卷积操作可表示为$\{\boldsymbol{O}=\boldsymbol{F}_{i,j}\times\boldsymbol{S},1\leqslant j\leqslant C_{i+1}\}$,卷积神经网络可被参数化表示为$\{\boldsymbol{W}^{(i)}\in\mathbb{R}^{C_i\times K\times K\times C_{i+1}},1\leqslant i\leqslant L\}$,即$\{\boldsymbol{W}^{(i)}\in\mathbb{R}^{C_i\times Y},1\leqslant i\leqslant L,Y=K\times K\times C_{i+1}\}$,$\boldsymbol{F}_{i,j}$的权重为$\boldsymbol{W}_i^j\in\mathbb{R}^Y$。

2.2 剪枝标准

利用欧式距离计算各卷积核相对于计算当前卷积层的冗余性[12]。具体而言,卷积神经网络所有卷积层中卷积核的冗余性可以通过欧式距离求得,将其称为信息距离R。例如,i_{th}层的j_{th}卷积核$\boldsymbol{F}_{i,j}$的信息距离值可以表示如下:

$$R(\boldsymbol{F}_{i,j})=\sum_{j^*\in\left[1,C_{i+1}\right],j^*\neq j}\left\|\boldsymbol{W}_i^j-\boldsymbol{W}_i^{j^*}\right\|_2 \tag{1}$$

其中:$\boldsymbol{W}_i^j$为$\boldsymbol{F}_{i,j}$的权重,是一个$K\times K\times C_i$维向量。$R(\boldsymbol{F}_{i,j})$值越小,表明$\boldsymbol{F}_{i,j}$与其他核的相关性越大,$\boldsymbol{F}_{i,j}$所包含的其他核的信息越多;$R(\boldsymbol{F}_{i,j})$值越大,表示信息冗余程度越低。

通过式(1)所求出的信息距离值的数据分布在各卷积层中存在差异，图1(a)给出了各卷积层中卷积核信息距离R的数据分布，其中黑色圆表示该层数据的平均值，曲线为R值的分布估计。图1(b)为基于贡献度归一化的卷积核信息距离，卷积核整体向左偏移。通过全局筛选Z值较低的卷积核，并利用掩码实现全局剪枝，其中黑色圆表示惩罚后的卷积核。全局剪枝后的效果如图1(c)所示，其中黑色圆为需要剪枝的卷积核。

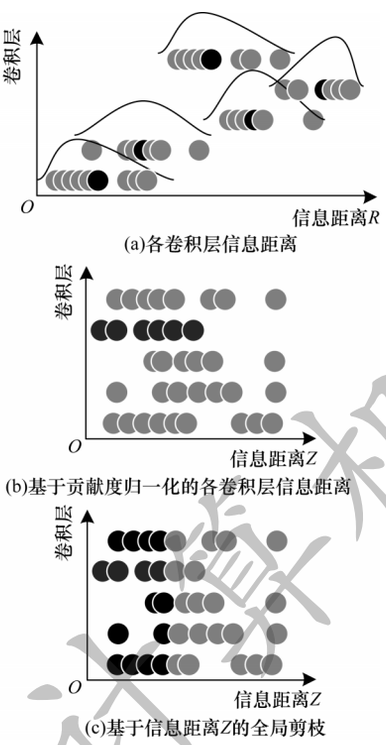

图1　卷积神经网络模型剪枝流程

Fig.1　Pruning procedure of convolutional neural network model

若直接通过信息距离R进行全局剪枝可能会剪掉某一卷积层中所有的卷积核。针对该情况，对每一层中的卷积核进行归一化处理来消除这种差异性。同时，为了考虑对弱层进行惩罚，需对每层乘上贡献度，贡献度较小的层在归一化后会增加对该层卷积核的剪枝，如式(2)所示：

$$Z(\boldsymbol{F}_{i,j^*})=\frac{R(\boldsymbol{F}_{i,j^*})}{\mathrm{Max}(R_i)-\mathrm{Min}(R_i)+\theta}\times I_{\mathrm{IMP}_i}$$
$$1\leqslant i\leqslant L,1\leqslant j^*\leqslant C_{i+1} \tag{2}$$

其中：$Z(\boldsymbol{F}_{i,j^*})$表示$i_{\mathrm{th}}$层$j_{\mathrm{th}}$卷积核归一化后的信息距离；$C_{i+1}$为$i_{\mathrm{th}}$层卷积核数量；$\mathrm{Max}(R_i)$和$\mathrm{Min}(R_i)$分别表示求$i_{\mathrm{th}}$层的最大和最小的核信息距离；$I_{\mathrm{IMP}_i}$为$i_{\mathrm{th}}$层的贡献度；$\theta$为极小的数，避免分母为0。

基于信息距离R可以较好地表明各卷积层中卷积核的冗余程度。因此，可以认为某一层卷积核信息距离R的标准偏差(STD)越小，则该层卷积核之间的信息距离越接近，卷积核之间相似的可能性越高，如图2(a)所示。相反地，如图2(b)所示，STD越大，该层卷积核之间相似的可能性越低。

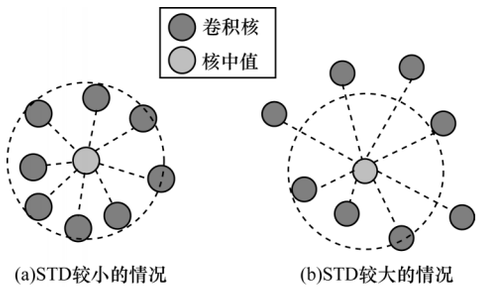

图2　卷积核信息距离与标准偏差的关系

Fig.2　Relationship of information distance and standard deviation of convolution kernel

利用这种数据分布特征对弱层进行识别。假设图1中的第2层为需要识别的弱层，首先利用式(3)计算各卷积层相关性R值的标准偏差，再使用式(4)计算所有层的平均标准偏差，最后利用式(5)和式(6)对该层进行判断并对弱层加入贡献度，贡献度较低的层中卷积核的R值会得到惩罚，最终在全局重要性评估过程中对其进行弱化。

$$S_{\mathrm{Std}_i}=\sqrt{\frac{1}{C_{i+1}}\sum_{j^*}^{C_{i+1}}[R(\boldsymbol{F}_{i,j^*})-A_{\mathrm{Avg}_i}]^2}$$
$$1\leqslant i\leqslant L,1\leqslant j^*\leqslant C_{i+1} \tag{3}$$

$$A_{\mathrm{Avg}_{\mathrm{all}}}=\frac{1}{L}\sum_{i=1}^{L}S_{\mathrm{Std}_i} \tag{4}$$

$$I_{\mathrm{IMP}_i}=\begin{cases}v, S_{\mathrm{Std}_i}<A_{\mathrm{Avg}_{\mathrm{all}}}/2\\ 1,\text{其他}\end{cases} \tag{5}$$

$$S_{\mathrm{Std}_{\mathrm{all}}}=\sqrt{\frac{1}{L}\sum_{i=1}^{L}[S_{\mathrm{Std}_i}-A_{\mathrm{Avg}_{\mathrm{all}}}]^2} \tag{6}$$

其中：S_{Std_i}为i_{th}层所有卷积核信息距离的标准偏差；A_{Avg_i}为i_{th}层的所有卷积核信息距离的平均值；C_{i+1}为i_{th}层卷积核的数量。为得到合适的贡献度，实验部分通过对比$A_{\mathrm{Avg}_{\mathrm{all}}}-S_{\mathrm{Std}_{\mathrm{all}}}$和$A_{\mathrm{Avg}_{\mathrm{all}}}/2$这2种阈值对模型精度的影响，选择$A_{\mathrm{Avg}_{\mathrm{all}}}/2$作为重要性判断阈值。

2.3　剪枝过程

考虑到训练中卷积核的重要性是动态变化的[17,21]，引入掩码实现动态剪枝。

动态剪枝是指利用全局掩码M对模型权值W进行动态更新，其中$M=\{\boldsymbol{M}_i^j,1\leqslant i\leqslant L,1\leqslant j\leqslant C_{i+1}\}$，$\boldsymbol{M}_i^j\in\{0,1\}^Y$，$Y=K\times K\times C_i$为二进制掩码，$W=\{\boldsymbol{W}_i^j,1\leqslant i\leqslant L,1\leqslant j\leqslant C_{i+1}\}$。当通过式(1)和式(2)计算出每一个卷积核信息距离Z后，根据全局剪枝率P对所有卷积核进行筛选，得到符合条件的C_{all}个卷积核，其中$C_{\mathrm{all}}=\sum_{l=1}^{L}C_{i+1}\times P$。根据所选卷积核对掩码$M$更新，并通过掩码$M$对权重$W$更新。例如，满足剪枝条件的集合为$S_{\mathrm{Set}_{\mathrm{prune}}}=\{\boldsymbol{F}_1^1,\boldsymbol{F}_1^2,\cdots,\boldsymbol{F}_i^j\}$，其中$\boldsymbol{F}_i^j$表示$i_{\mathrm{th}}$层的$j_{\mathrm{th}}$卷积核。经过式(7)计算掩码，再通过掩码对满足条件的卷积核进行剪枝，操作形式如式(8)所示。

$$\boldsymbol{M}_i^j=\begin{cases}0, \boldsymbol{F}_i^j\in S_{\mathrm{Set_{prune}}}\\ 1, \text{其他}\end{cases} \tag{7}$$

$$\boldsymbol{W}_i^j=\boldsymbol{M}_i^j\odot\boldsymbol{W}_i^j \tag{8}$$

算法 融合弱层惩罚的结构化非均匀模型剪枝算法

输入 训练数据X,全局剪枝率P,贡献度v

输出 压缩后的模型和参数W

1.初始化模型参数W和全局掩码M=1

2.For epoch = 1; epoch < $\mathrm{epoch_{max}}$; epoch++:

3.用训练集X更新模型参数W;

4.通过式(1)计算每一个卷积核的R值;

5.利用式(2)计算归一化后的Z值;

6.找到C_{all}个最小的Z值所对应的卷积核;

7.通过式(7)更新M;

8.通过式(8)对所选卷积核权重归零;

9.End for

10.构造最优稀疏分配的压缩模型;

11.获取最优稀疏分配的压缩模型权重W^*。

3 实验结果与分析

3.1 数据集与训练策略设置

为验证本文提出方法的有效性,采用的数据集包括CIFAR-10、CIFAR-100和SVHN数据集。CIFAR-10数据集包含50 000张训练图和10 000张测试图,共10个种类。CIFAR-100数据集的图像数量和CIFAR-10相同,共有100个种类。SVHN数据集为Google的街景门牌号数据集,训练集包含73 257个数字,测试集包含26 032个数字,其中每一张图像都由一组数字组成,图像分辨率为32×32像素的彩色图像。在网络结构的选择方面,包括单分支网络(VGG16)、多分支网络(Resnet20、32、56、110),轻量化网络(Mobilenet-v1),所有实验均使用深度框架PyTorch1.6.0,运行于NVIDIA 2080TI GPU。

对于数据集CIFAR-10和CIFAR-100的训练策略和文献[13]相同,输入图像分辨率为32×32像素,使用Nesterov的随机梯度下降,权重下降系数为5e-4,batch-size为128,初始学习率为0.1,在epoch为80、120、160时学习率降低10倍,共训练200个epoch。其中,数据增强策略和文献[22]相同。对于轻量化网络Moblienet-v1,修改第1个卷积操作的stride为1以适合输入图像分辨率。

在剪枝策略上,本文方法从头开始迭代训练与剪枝模型,在每次训练后选择剪枝操作。除VGG模型外,其他模型不需要额外的微调恢复精度,从而降低训练的时间开销。同时,将本文方法(Ours)与FPGM[12]、SFP[13]、GDP[21]、MIL[23]、PFEC[24]等方法进行实验对比,其中,贡献度为v,GDP为结构化非均匀剪枝方法。

3.2 CIFAR-10数据集上的实验结果分析

对于CIFAR-10数据集,选择在VGG16、Resnet20、Resnet32、Resnet56、Resnet110和Mobilenet-v1上进行实验,实验结果如表1所示,其中F.T表示用较小的学习率对模型进行训练恢复精度,“—”表示无有效实验结果。

表1 在CIFAR-10数据集上的实验结果

Table 1 Experimental results on the CIFAR-10 dataset %

模型	剪枝方法	模型精度	FLOPs减少率	参数量减少率
VGG16	Baseline	93.51	0.0	0.0
	FPGM(F.T)	92.82	80.5	80.6
	GDP(F.T)	93.29	47.2	86.7
	Ours(F.T)	93.06	87.1	80.6
Resnet20	Baseline	92.20	0.0	0.0
	SFP	90.83	42.2	41.5
	FPGM	91.09	42.2	41.5
	Ours(v=0.9)	91.38	51.1	43.7
Resnet32	Baseline	92.63	0.0	0.0
	SFP	92.08	41.5	41.1
	MIL	90.74	31.2	—
	FPGM	92.31	41.5	41.1
	Ours(v=0.9)	92.70	46.8	45.2
	Ours(v=0.7)	92.40	49.6	43.6
Resnet56	Baseline	93.59	0.0	0.0
	SFP	92.26	52.6	52.4
	PFEC	91.31	27.6	—
	FPGM	92.70	53.6	53.4
	GDP	92.73	52.7	42.2
	Ours(v=0.9)	93.09	59.4	57.0
	Ours(v=0.7)	93.15	61.0	56.4
Resnet110	Baseline	93.68	0.0	0.0
	PFEC	92.94	38.6	—
	FPGM	93.03	52.3	52.2
	Ours(v=0.9)	93.62	66.8	59.4
	Ours(v=0.7)	93.46	63.1	57.5
Mobilenet-v1	Baseline	89.59	0.0	0.0
	Ours	86.35	92.1	95.3

从VGG16实验结果可以看出:相比于FPGM方法,本文方法在各指标上均有所提高;相比GDP方法,本文方法的精度虽下降了0.23个百分点,但FLOPs却减少了39.9个百分点。

从Resnet20、Resnet32、Resnet56和Resnet110实验结果可以看出:本文方法相比其他方法具有更高的剪枝模型精度,同时参数量和FLOPs也大幅减少;对于Resnet32,当v=0.9时,本文方法在FLOPs和参数量分别减少46.8%和45.2%的情况下,精度甚至超过了基准精度;对于Resnet56,当v=0.7时,本文方法相比于GDP方法精度提升0.42个百分点的同时,FLOPs和参数量分别减少了8.3和14.2个百分点,相比于FPGM方法精度提升0.45个百分点的同时,FLOPs和参数量分别减少了7.4和3.0个百分点,相

比于PFEC方法，精度提升了1.84个百分点且FLOPs减少了33.4个百分点；对于Resnet110，当v=0.9时，本文方法在FLOPs下降66.8%的情况下，相比基准精度仅损失了0.06个百分点；对于Mobilenet-v1，当参数量和FLOPs分别减少了95.3%和92.1%的情况下，本文方法精度相比于基准精度仅损失了3.24个百分点。

综上所述，相比未考虑层差异性的SFP、PFEC、FPGM和MIL方法，本文方法可以剪枝出更好性能的模型，关键在于其考虑了对重要性较高的层减少剪枝，提高了模型精度，同时对Mobilenet-v1进行剪枝的结果表明，本文方法同样适用于轻量化网络结构剪枝，经过剪枝后的Mobilenet模型所占内存更小、推理速度更快。

3.3　CIFAR-100数据集上的实验结果分析

对于CIFAR-100数据集，选择在Resnet32、Resnet56和Resnet110上进行实验，实验结果如表2所示。

表2　在CIFAR-100数据集上的实验结果

Table 2　Experimental results on the CIFAR-100 dataset　%

模型	剪枝方法	模型精度	FLOPs减少率	参数量减少率
Resnet32	Baseline	71.21	0.0	0.0
	SFP	68.59	41.5	41.1
	FPGM	68.64	41.5	41.1
	Ours(v=0.9)	69.19	49.8	48.5
Resnet56	Baseline	71.66	0.0	0.0
	SFP	68.79	52.6	52.4
	FPGM	69.66	52.6	52.4
	Ours(v=0.9)	69.98	58.9	56.0
Resnet110	Baseline	73.75	0.0	0.0
	FPGM	70.18	52.3	52.2
	Ours(v=0.9)	71.67	60.9	57.4

从Resnet20、Resnet56和Resnet110实验结果可以看出：对于Resnet32，当v=0.9时，本文方法精度相比于基准精度仅损失2.02个百分点的情况下，FLOPs和参数量分别减少了49.8%和48.5%，相比于SFP方法精度提升了0.6个百分点，相比于FPGM方法精度提升了0.55个百分点；对于Resnet56，本文方法同样优于对比方法，例如，当v=0.9时，相比于FPGM方法精度提升了0.32个百分点，但FLOPs和参数量分别减少了6.3和3.6个百分点，相比于SFP方法精度提升了1.19个百分点；对于Resnet110，当v=0.9时，本文方法精度相比于FGPM方法提升了1.49个百分点，FLOPs和参数量分别减少了8.6和5.2个百分点，在精度和其他性能之间获得了更好的权衡。

总体而言，本文方法可以在提高精度的同时大幅减少参数量和FLOPs，这关键在于剪枝算法引入了弱层的识别与惩罚，将卷积核从局部卷积层面的重要性评估上升为全局网络层面的重要性评估。使用该处理方式，当提高全局剪枝率时，剪枝算法会增加对弱层的剪枝。因此，最终模型精度损失在很小的情况下，却可以更多地减少模型FLOPs和参数量。

为验证本文提出的剪枝方法所识别的弱层的合理性，使用Resnet32在CIFAR-100数据集上进行实验。对本文方法所识别到的弱层分别进行剪枝与训练（剪枝率为0.8），测试各层对最终模型精度的影响。实验结果如图3所示，其中基准精度为71.21%。从图3可以看出，在对弱层保留较少特征的情况下，模型依然获得了较好的精度，可以认为所识别到的弱层对最终模型的影响较小，验证了本文方法的有效性。

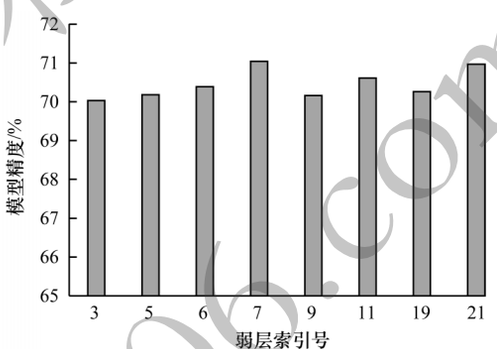

图3　弱层在较高剪枝率下训练得到的模型精度

Fig.3　Model accuracy of the weak layer trained at a higher pruning rate

3.4　SVHN数据集上的实验结果分析

对于SVHN数据集，本文选择在多分支网络Resnet32和轻量化网络Mobilenet-v1上进行实验，实验结果如表3所示。从表3可以看出：对于Resnet32，本文方法可以在精度仅损失0.65个百分点的情况下，参数量和FLOPs分别减少了80.0%和82.4%；对于Mobilenet-v1，本文方法可以在模型精度没有大幅下降的情况下，参数量和FLOPs分别减少了92.7%和95.3%；原始模型过度参数化，从而验证了本文剪枝方法的有效性。

表3　在SVHN数据集上的实验结果

Table 3　Experimental results on the SVHN dataset　%

模型	方法	模型精度	FLOPs减少率	参数量减少率
Resnet32	Baseline	96.43	0.0	0.0
	Ours	95.78	82.4	80.0
Mobilenet-v1	Baseline	95.08	0.0	0.0
	Ours	93.93	92.7	95.3

3.5　相关参数对模型性能的影响

为研究贡献度对模型性能的影响程度，在CIFAR-10数据集上对贡献度为1.0、0.9、0.7、0.5和0.1下的模型精度、FLOPs和参数量减少率进行统计，实验结果如表4

所示。从表4可以看出，Resnet32和Resnet56分别在v=0.9、v=0.7时获得最佳性能，说明贡献度v在一定程度上提升了模型性能，同时也证明了本文方法的有效性，但是过度惩罚弱层会导致精度大幅下降。

表4 层贡献度比较结果

Table 4 Comparison results of layer contribution

模型	贡献度	模型精度/%	FLOPs减少率/%	参数量减少率/%
Resnet32	1.0	92.41	47.2	44.6
	0.9	92.70	46.8	45.2
	0.7	92.42	49.6	43.6
	0.5	92.15	50.4	44.7
	0.1	90.94	57.6	51.3
Resnet56	1.0	92.66	60.1	56.7
	0.9	93.09	59.4	57.0
	0.7	93.15	61.0	56.4
	0.5	92.84	67.1	56.1
	0.1	91.17	65.4	58.2
Resnet110	1.0	92.22	64.4	57.8
	0.9	93.62	66.8	59.4
	0.7	93.46	63.1	57.5

为进一步研究剪枝算法的性能，对比Resnet110在不同FLOPs下本文方法的模型精度变化情况。如图4所示：当FLOPs减少率约小于18%时，模型精度得到了提升，说明通过本文方法进行剪枝，当剪枝率较小时，模型得到了正则化作用，增强了模型泛化能力；当FLOPs减少率约大于67%时，模型得到大幅剪枝，模型的表征能力受到影响，模型精度也因此下降明显。

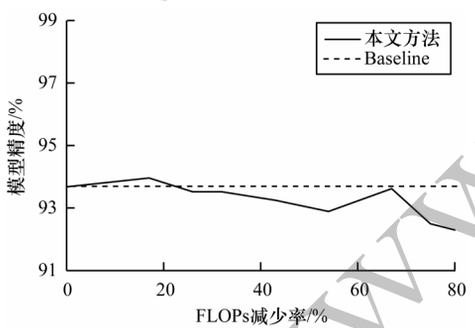

图4 不同FLOPs下Resnet110在CIFAR10数据集上的模型精度

Fig.4 Model accuracy of Resnet110 on CIFAR10 dataset under different FLOPs

对于层的重要性判断，本文采用2种重要性判断阈值作为对比，实验结果如表5所示。在实验中，相同网络设置相同的配置参数，每个实验进行3次，使用平均值加上标准差作为实验结果。从表5可以看出，$A_{\mathrm{Avg_{all}}}/2$相比于$A_{\mathrm{Avg_{all}}}-S_{\mathrm{Std_{all}}}$整体性能有所提升，验证了选择$A_{\mathrm{Avg_{all}}}/2$作为重要性判断阈值的正确性与有效性。

表5 在不同重要性判断阈值下的模型精度

Table 5 Model accuracy under different importance judgment thresholds %

模型	CIFAR-10数据集		CIFAR-100数据集	
	$A_{\mathrm{Avg_{all}}}-S_{\mathrm{Std_{all}}}$	$A_{\mathrm{Avg_{all}}}/2$	$A_{\mathrm{Avg_{all}}}-S_{\mathrm{Std_{all}}}$	$A_{\mathrm{Avg_{all}}}/2$
Resnet32	92.44±0.3	92.70±0.3	68.19±0.3	69.19±0.3
Resnet56	92.78±0.2	93.15±0.3	68.24±0.2	69.98±0.1
Resnet110	93.35±0.3	93.62±0.3	71.39±0.3	71.67±0.2

4 结束语

本文提出一种融合弱层惩罚的结构化非均匀模型剪枝方法，使用欧式距离计算各卷积层中所有卷积核的信息距离，利用各层信息距离值的数据分布特征识别层的冗余性，并通过基于贡献度的归一化函数消除各层之间的差异性，同时从全局层面评估卷积核重要性，从而筛选卷积核。在多个数据集上的实验结果表明，相比于FPGM、SFP、GDP、MIL、PFEC等方法，本文方法剪枝得到的网络模型获得了较好的性能提升，且不需要特殊的软件和硬件加速，为下一步模型部署奠定了基础。后续可将本文剪枝算法应用到基于深度学习的坐姿识别等任务中，利用其对深度学习人体姿态估计模型进行剪枝，减少人体姿态估计模型提取骨骼特征所需的计算和存储资源，使深度学习模型可在保证识别精度的情况下加快检测速度，并结合模型量化等技术，提升深度学习模型在嵌入式设备上的运行效率。

参考文献

[1] BULAT A, KOSSAIFI J, TZIMIROPOULOS G, et al. Toward fast and accurate human pose estimation via soft-gated skip connections[C]//Proceedings of the 15th IEEE International Conference on Automatic Face and Gesture Recognition. Washington D. C., USA: IEEE Press, 2020: 8-15.

[2] HAN W, ZHANG Z D, ZHANG Y, et al. ContextNet: improving convolutional neural networks for automatic speech recognition with global context[EB/OL]. [2021-03-05]. https://arxiv. org/abs/2005. 03191v3.

[3] TORFI A, SHIRVANI R A, KENESHLOO Y, et al. Natural language processing advancements by deep learning: a survey[EB/OL]. [2021-03-05]. https://arxiv. org/abs/2003. 01200.

[4] KINGSBURY B E D, SAINATH T N, SINDHWANI V. Low-rank matrix factorization for deep belief network training with high-dimensional output targets: US9262724[P]. 2016-02-16.

[5] GONG R H, LIU X L, JIANG S H, et al. Differentiable soft quantization: bridging full-precision and low-bit neural networks[EB/OL]. [2021-03-05]. https://arxiv. org/abs/1908. 05033.

[6] WANG W H, WEI F R, DONG L, et al. MiniLM: deep self-attention distillation for task-agnostic compression of pre-trained transformers[EB/OL]. [2021-03-05]. https://arxiv. org/abs/2002. 10957.

[7] MA N N, ZHANG X Y, ZHENG H T, et al. ShuffleNet V2: practical guidelines for efficient CNN architecture design[M]. Berlin, Germany: Springer, 2018: 122-138.
[8] YOU Z H, YAN K, YE J M, et al. Gate decorator: global filter pruning method for accelerating deep convolutional neural networks[EB/OL]. [2021-03-05]. https://arxiv.org/abs/1909.08174.
[9] HE Y, DING Y H, LIU P, et al. Learning filter pruning criteria for deep convolutional neural networks acceleration[C]// Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2020: 2006-2015.
[10] GUO Y W, YAO A B, CHEN Y R. Dynamic network surgery for efficient DNNs[C]//Proceedings of the 30th International Conference on Neural Information Processing Systems. New York, USA: ACM Press, 2006: 1387-1395.
[11] LUO J H, WU J X, LIN W Y. ThiNet: a filter level pruning method for deep neural network compression [C]// Proceedings of IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2017: 5068-5076.
[12] HE Y, LIU P, WANG Z W, et al. Filter pruning via geometric Median for deep convolutional neural networks acceleration[C]// Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2019: 4335-4344.
[13] HE Y, DONG X Y, KANG G L, et al. Asymptotic soft filter pruning for deep convolutional neural networks[EB/OL]. [2021-03-05]. https://arxiv.org/abs/1808.07471.
[14] 卢海伟，夏海峰，袁晓彤. 基于滤波器注意力机制与特征缩放系数的动态网络剪枝[J]. 小型微型计算机系统，2019，40(9)：1832-1838.
LU H W, XIA H F, YUAN X T. Dynamic network pruning via filter attention mechanism and feature scaling factor[J]. Journal of Chinese Computer Systems, 2019, 40(9): 1832-1838. (in Chinese)
[15] 甘岚，李佳，沈鸿飞. 面向嵌入式的残差网络加速方法研究[J]. 小型微型计算机系统，2020，41(11)：2314-2320.
GAN L, LI J, SHEN H F. Research on the acceleration method of residual network for embedded system [J]. Journal of Chinese Computer Systems, 2020, 41(11): 2314-2320. (in Chinese)
[16] LIU Z, LI J G, SHEN Z Q, et al. Learning efficient convolutional networks through network slimming[C]// Proceedings of IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2017: 2755-2763.
[17] SCHENCK C, FOX D. SPNets: differentiable fluid dynamics for deep neural networks[EB/OL]. [2021-03-05]. https://arxiv.org/abs/1806.06094.
[18] HE Y H, LIN J, LIU Z J, et al. AMC: AutoML for model compression and acceleration on mobile devices[C]// Proceedings of the 15th European Conference on Computer Vision. Berlin, Germany: Springer, 2018: 851-832.
[19] YANG T J, HOWARD A, CHEN B, et al. NetAdapt: platform-aware neural network adaptation for mobile applications[C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2018: 289-304.
[20] LI B L, WU B W, SU J, et al. EagleEye: fast sub-net evaluation for efficient neural network pruning [C]// Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2020: 639-654.
[21] LIN S H, JI R R, LI Y C, et al. Accelerating convolutional networks via global & dynamic filter pruning [C]// Proceedings of the 27th International Joint Conference on Artificial Intelligence. New York, USA: ACM Press, 2018: 2425-2432.
[22] PASZKE A, GROSS S, CHINTALA S, et al. Automatic differentiation in PyTorch[EB/OL]. [2021-03-05]. https://openreview.net/pdf?id=BJJsrmfCZ.
[23] DONG X Y, HUANG J S, YANG Y, et al. More is less: a more complicated network with less inference complexity[C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2017: 1895-1903.
[24] LI H, KADAV A, DURDANOVIC I, et al. Pruning filters for efficient ConvNets[EB/OL]. [2021-03-05]. https://arxiv.org/abs/1608.08710.

编辑　陆燕菲