软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(12):4534–4544 [doi: 10.13328/j.cnki.jos.006371]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于样本个体差异性的深度神经网络训练方法*

李 响[1,2], 刘 明[1], 刘明辉[1], 姜 庆[2], 曹 扬[2]

[1](电子科技大学 计算机科学与工程学院, 四川 成都 611731)

[2](中电科大数据研究院有限公司, 贵州 贵阳 550022)

通信作者: 李响, E-mail: lx_madcat@163.com

摘 要: 深度神经网络目前在许多任务中的表现已经达到甚至超越了人类的水平, 但是其泛化能力和人类相比还是相去甚远. 如何提高网络的泛化性, 一直是重要的研究方向之一. 围绕这个方向开展的大量卓有成效的研究, 从扩展增强训练数据、通过正则化抑制模型复杂度、优化训练策略等角度, 提出了很多行之有效的方法. 这些方法对于训练数据集来说都是某种全局性质的策略, 每一个样本数据都会被平等的对待. 但是, 每一个样本数据由于其携带的信息量、噪声等的不同, 在训练过程中, 对模型的拟合性能和泛化性能的影响也应该是有差异性的. 针对是否一些样本在反复的迭代训练中更倾向于使得模型过度拟合, 如何找到这些样本, 是否可以通过对不同的样本采用差异化的抗过拟合策略使得模型获得更好的泛化性能等问题, 提出了一种依据样本数据的差异性来训练深度神经网络的方法, 首先使用预训练模型对每一个训练样本进行评估, 判断每个样本对该模型的拟合效果; 然后依据评估结果将训练集分为易使得模型过拟合的样本和普通的样本两个子集; 最后,再使用两个子集的数据对模型进行交替训练, 过程中对易使得模型过拟合的子集采用更强有力的抗过拟合策略. 通过在不同的数据集上对多种深度模型进行的一系列实验, 验证了该方法在典型的分类任务和细粒度分类任务中的效果.

关键词: 深度神经网络; 泛化性; 正则化; 权重衰减

中图法分类号: TP181

中文引用格式: 李响, 刘明, 刘明辉, 姜庆, 曹扬. 基于样本个体差异性的深度神经网络训练方法. 软件学报, 2022, 33(12): 4534–4544. http://www.jos.org.cn/1000-9825/6371.htm

英文引用格式: Li X, Liu M, Liu MH, Jiang Q, Cao Y. Deep Neural Network Training Method Based on Individual Differences of Training Samples. Ruan Jian Xue Bao/Journal of Software, 2022, 33(12): 4534–4544 (in Chinese). http://www.jos.org.cn/1000-9825/6371.htm

Deep Neural Network Training Method Based on Individual Differences of Training Samples

LI Xiang[1,2], LIU Ming[1], LIU Ming-Hui[1], JIANG Qing[2], CAO Yang[2]

[1](School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China)

[2](CETC Big Data Research Institute Co., Ltd., Guiyang 550022, China)

Abstract: In recent years, the performance of deep neural networks in many tasks has been comparable to or even surpassed that of humans, but its generalization ability is still far from that of humans. How to improve the generalization of the network has always been an important research direction, and a lot of fruitful research has been carried out around this direction. Many effective methods have been proposed from the perspectives of expanding and enhancing training data, suppressing model complexity through regularization, and optimizing training strategies. These methods are a global strategy for the training data set, and each sample data will be treated equally. However, due to the difference in the amount of information and noise carried by each sample data, the impact on the fitting performance and generalization performance of the model during the training process should also be different. Are some samples more likely to overfit the model during repeated iterative training? How to find these samples? Can the model obtain better generalization performance by

* 基金项目: 贵州省科技计划(黔科合支撑[2020]4Y058)

收稿时间: 2020-08-15; 修改时间: 2020-11-20, 2021-02-25; 采用时间: 2021-04-02

adopting a differentiated anti-overfitting strategy for different samples? In response to these problems, a method for training deep neural networks is proposed based on individual differences in sample data. First, the pre-training model is used to evaluate each training sample to determine the fit effect of each sample to the model. Then, according to the evaluation results, the training set is divided into two subsets: samples that are easy to overfit the model and the remaining ordinary samples. Finally, two subsets of data are used to train the model. In the process, a stronger anti-overfitting strategy is adopted for the subset that is more likely to overfit the model. Through a series of experiments on various deep models on different data sets, the effect of the proposed method on typical classification tasks and fine-grained classification tasks is verified.

Key words: deep neural networks; generalization; regularization; weight-decay

深度神经网络(DNN)[1,2]是当前许多人工智能应用的基础, 近年来, 在数据可用性和高性能计算的帮助下, 各种 DNN 模型及其变体层出不穷, 并且在很多任务(例如自然语言处理、视觉对象识别、对象检测等)中表现出了达到甚至是超越人类水平的性能. 但是, 深度神经网络仍然面临许多挑战, 过度拟合、泛化性不强是常见问题. 复杂的网络结构和大量的参数赋予了深度神经网络极高自由度和非常大的容量, 可以拟合更多类型的复杂函数以将输入映射到输出. 但是, 这种优势也在训练数据集的信息量与模型复杂度、容量不匹配时使得 DNN 易于过度拟合. 当在一个小的数据集(很少的样本, 较少的信息)上训练一个复杂的模型(一个具有复杂结构、很多参数、大容量的模型)时, 神经网络的容量足以存储整个数据集[3], 从而使模型在训练数据集上效果突出, 但在未知的测试数据集上效果不好.

围绕着解决过拟合、提升模型泛化性已经开展了很多的研究, 也提出了很多的方法. 例如: 在训练中观察模型在验证集上的表现, 当模型在验证集上的表现开始下降的时候, 停止训练的 Early Stopping[4]; 在训练过程中, 随机选择某些神经网络单元以零概率激活的 Dropout[5]; 在损失函数中增加约束因子, 对模型复杂度进行惩罚的正则化; 包含一套可增强训练数据集的大小和质量的技术的数据增强等. 这些方法的基本原理是保持训练数据集中的信息量与模型的复杂性之间的平衡[6,7], 也取得了很好的效果.

当前, 处理过拟合问题的方法对于训练数据集来说基本是全局性质的, 等效地作用于训练数据集中的所有样本数据. 而我们认为, 不同的样本数据由于所蕴含的信息量、噪声等是存在个体差异性的. 所以在反复迭代的训练过程中, 不同的样本数据对模型的贡献和影响是有区别的, 对模型的泛化性能的影响是有差异的. 所以, 我们应该针对不同的样本数据采取差异化的防止过拟合的策略: 对于那些更容易造成模型过拟合的样本, 采用较为强力的措施抑制过拟合; 对于其他的普通样本, 采用相对比较柔和的策略. 避免出现欠拟合, 以提升模型的性能和泛化性. 为了验证我们的理论, 我们提出了一种通过样本数据在预训练模型上的表现来评估样本是否趋于导致模型过度拟合的方法, 并依据评估结果将训练数据分为易于过度拟合的样本和普通样本, 在训练网络的过程中, 两类样本使用不同的正则化率. 通过在一系列经典数据集上的各种深度模型进行的广泛实验, 证明了我们的方法对于典型的分类任务和细粒度分类任务是有效的[1,8].

本文第 1 节解释了我们方法的动机. 在第 2 节中, 简要概述了与本文相关工作. 第 3 节介绍了我们的方法. 第 4 节通过大量实验评估了我们的方法. 第 5 节对本文做了一个总结.

1 动 机

我们从一个简单的实验观察开始, 我们基于 $y=x^2$ 函数加上均匀分布的噪声生成了一些样本数据, 用于训练一个简单的 4 层全连接神经网络模型, 网络结构如图 1 所示, 损失函数为均方损失函数, 如公式(1)所示.

$$Loss=\frac{1}{m}\sum_{i=1}^{m}(y_i-f(x_i))^2 \tag{1}$$

其中, y_i 和 $f(x_i)$分别表示第 i 个样本的真实值和预测值, m 为样本个数.

在不加入任何防治过拟合的措施、经过若干轮的训练之后, 模型拟合出的曲线如图 2(a)所示, 其中, 红色曲线为模型输出结果. 可以看到: 为了更好地拟合部分噪声较大、偏离 $y=x^2$ 曲线较远的样本, 模型输出的曲线方差(variance)较大, 非常曲折, 体现出了明显的过拟合现象.

为了抑制这种过拟合, 我们引入 L2 正则化(L2 regularization), 通常也称为权重衰减(weight decay), 即是

在损失函数中加入一个由所有参数权重的平方和组成的惩罚项, 由此来抑制模型的复杂度. 引入 L2 正则化后, 损失函数如公式(2)所示:

$$Loss=\frac{1}{m}\sum_{i=1}^{m}(y_i-f(x_i))^2+\lambda\sum_{w}w^2 \tag{2}$$

其中, λ为权重衰减系数, w 为权重参数. 引入 L2 正则化后, 模型输出的拟合曲线如图 2(b)绿色曲线所示. 我们可以看到: 曲线相比图 2(a)中平滑了许多, 方差降低, 过拟合的现象得到了有效的抑制. 然而, 相对的, 也是由于正则化对参数的抑制, 输出曲线相对于 $y=x^2$ 函数曲线偏差(bias)更大, 部分样本出现了欠拟合的现象.

接下来我们选取部分噪声较大、偏离 $y=x^2$ 曲线较远的样本, 如图 2(c)中标注所示, 将这部分样本作为一个子集 A, 其余噪声较小的样本作为另一个子集 B, 我们使用子集 A, B 轮流对模型进行训练, 同样引入 L2 正则化处理. 在使用子集 A 训练时, 我们使用较大的λ系数(50 倍于图 2(b)中使用的λ); 在使用子集 B 训练时, 我们使用较小的λ系数(与图 2(b)中使用的λ相等), 得到的结果如图 2(c)所示. 我们能观察到: 模型拟合的情况有了明显的提升, 模型输出的曲线方差降低了, 相对比较平滑, 偏差也相对减少了, 更加贴近 $y=x^2$ 函数曲线.

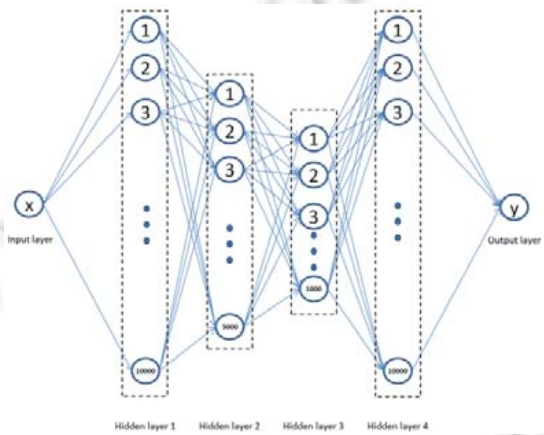
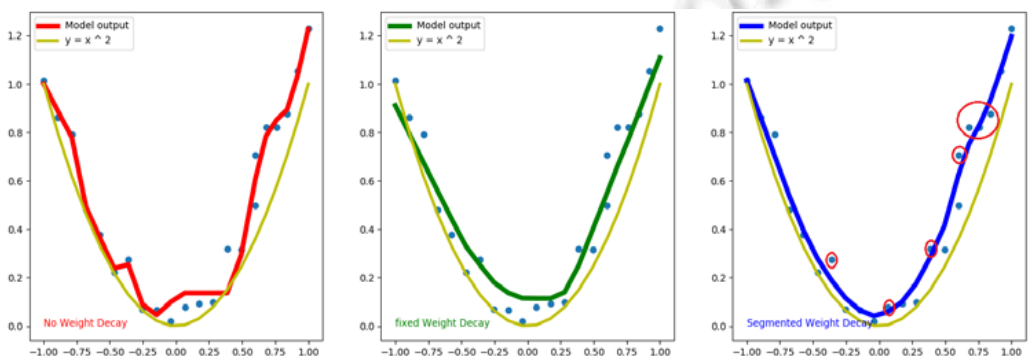


图 1 全连接神经网络结构



(a) Without Weight Decay (b) With Fixed Weight Decay (c) With Segmented Weight Decay

图 2 不同的 L2 正则化策略的拟合效果对比

基于以上观察, 我们尝试探索样本数据与模型过度拟合之间的关系. 如何判断样本数据是否容易导致模型过度拟合? 对于在训练过程中可能导致过度拟合的样本, 是否可以采用更强有力的防治过拟合的策略来提高泛化性能? 在下文中, 我们将讨论这些问题, 并通过实验进行验证.

2 相关工作

避免过度拟合并改善模型的泛化性能一直是深度学习中的一个基本问题. 关于此问题, 已经开展了大量的研究. 其中, 正则化是可以防止神经网络过度拟合的一种常用技术. L1 正则化主要用于稀疏网络[9]或二进制神经网络[10,11]训练, 目的是提高计算速度. 权重衰减等价于 L2 正则化, 通过为模型损失函数添加惩罚项,使得学习的模型参数值较小, 是常用的过拟合的常用手段. L2 范数正则化是在模型原损失函数基础上添加 L2 范数惩罚项, 以防止模型具有较高的复杂性[12]. 一些工作研究了如何调整权重衰减系数[13,14]. 分层权重衰减[15]引入了逐层权重衰减. 自适应权重衰减[16]提出了一种使用损耗梯度的大小来确定每次优化迭代时权重衰减率的方法. 贝叶斯权重衰减[17]通过提出的目标函数, 结合贝叶斯概率分布, 为衰减参数制定了解析解.

Dropout 也是一种正则化技术, 在训练过程中, 随机选择某些神经网络单元以零概率激活(连同它们的连接)[18]. Standout[19]提出了一种自适应 Dropout 方法, 该方法可学习二进制置信网络以产生 Dropout 率. Variational dropout[20]引入了局部重新参数化技术, 以减少随机梯度的波动. 通过噪声对深度神经网络进行正则化[21]提出了一种使用噪声的权重随机梯度下降法作为 Dropout 的优化方法. 注意力机制(attention)也应用于 Dropout 的优化[22], 利用自我注意力(self-attention)机制来删除目标对象的最具区分性的部分, 以更好地捕捉整体特征.

数据增强用于在深度神经网络训练期间扩展有限的训练数据集[1], 常用的数据增强方法包括裁剪、平移、翻转、旋转、添加噪声等. 随机擦除(random erasing)[23]选择图片的矩形区域以引入随机噪声, 以降低过拟合的风险并提高模型的鲁棒性. 随机图像裁剪和修补[24]将多个训练图像及其标签混合在一起, 以生成新的训练数据. 用于对象检测的数据增强策略[25]研究了在目标检测任务中使用专门的数据增强策略来提高模型的泛化性能. Mixup[26]构造了虚拟训练示例, 以在从损坏的标签中学习时, 提高神经网络的鲁棒性. MixUp 已应用于半监督学习. CutMix[27]与 Mixup 有相似之处, CutMix 通过用另一个训练图像块替换图像区域, 进一步克服了 Mixup 合成的样本趋于不自然的问题.

总的来说, 先前的研究主要是从模型和参数的角度来研究过度拟合的问题, 对于训练集样本数据来说, 都是全局性质一视同仁的. 而我们的方法主要是从样本数据的差异与模型的过度拟合之间的关系的角度进行的, 差异化的对待训练样本数据.

3 基于训练样本数据差异的网络训练方法

我们的方法简单但有效, 包括 3 个部分: 样本数据评估、生成训练数据集和模型训练.

3.1 样本数据评估

首先要评估模型对训练集中每一个样本的拟合性能. 以分类任务为例, 分类任务中通常使用交叉熵[28]作为损失函数, 交叉熵描述了两个概率分布之间的距离. 交叉熵越小, 神经网络的输出与标签数据越接近, 说明模型与样本数据拟合得越好. 交叉熵可以表示为公式(3):

$$H(p,q)=\sum_{\forall}p(x)\log(q(x)) \tag{3}$$

其中, p 为真实分布, q 为非真实分布. 尽管交叉熵描述了两个概率分布之间的距离, 但是神经网络的输出不一定是概率分布. 因此, 我们经常使用 *Softmax* 将神经网络的正向传播结果转化为概率分布. *Softmax* 是在多分类任务中经常使用的激活函数. 标准 *Softmax* 可以表示为公式(4):

$$\sigma(z)_i=\frac{e^{Z_i}}{\sum_{j=1}^{K}e^{Z_j}} \tag{4}$$

其中: i 为输出层神经元下标次序; 而 Z_i 为输出层 i 神经元输出, 也即是一个线性分类器的输出作为自然常数 e 的指数, 它将多个神经元的输出归一化为(0,1)区间, 且和为 1. 因此, *Softmax* 的输出可以视为分类的置信度.

我们通过独热编码(one-hot)将样本标签转化为 One-hot 向量, 在目标类别的索引位置是 1, 在其他位置是

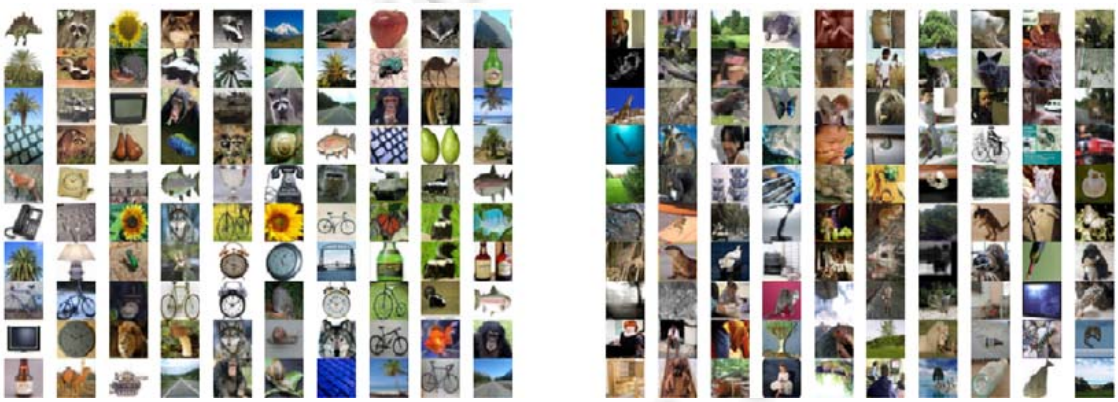
0. 交叉熵损失比较 *Softmax* 输出的概率分布与标签的 One-hot 向量匹配的程度.

由于我们仅计算单个样本数据的模型输出和标签 One-hot 编码的交叉熵, 将公式(4)带入公式(3), 我们可以得出单个样本数据模型输出的分类结果与标签数据的交叉熵, 如公式(5)所示:

$$H(p,q)=\log\left(\frac{e^{Z_{label\_index}}}{\sum_{j=1}^{K}e^{Z_j}}\right) \tag{5}$$

其中, Z_{label_index} 为网络输出层对应正确标签位的神经元输出. 通过上式可以得出: 目标类别索引对应的模型输出概率越接近 1, 则交叉熵损失函数越接近 0, 表明模型输出的结果与样本拟合得越好. 所以我们可以使用 *Softmax* 输出的标签类别索引对应的置信度概率来作为我们评估单个训练样本对于模型而言的易拟合程度的分值, 分值越高, 说明模型对该样本的拟合性能越好.

我们使用预训练模型对 CIFAR100 数据集中的训练集数据进行评分. 图 3 显示了我们评估结果中 100 个高分样本与 100 个普通样本的对比. 相比之下, 高分样本具有相对较少的信息、简单的背景、更清晰的边界和突出的特征, 是模型相对比较容易识别的简单样本. 我们认为: 对于这些简单的、信息量较少的样本, 在多轮反复迭代的训练过程中, 高自由度的复杂模型更容易依靠自身的容量记住这些样本, 而非从这些样本中去学习潜在的规律[3], 这就导致过拟合现象的产生. 所以对于我们评估的高分样本, 我们应该采用相对于其他普通样本更强有力的防治过拟合的策略.



(a) High score sample (b) Ordinary sample

图 3 CIFAR100 数据集中的高分样本和普通样本

3.2 训练数据分集

使用预训练模型对训练数据集中每个样本数据与正确标签对应的概率进行评分及排序. 将训练集划分为两个子集: 子集 *A* 部分为排序最前高分数据; 子集 *B* 部分为训练集中的剩余数据. 子集 *A* 和子集 *B* 比例的选择是通过系列实验数据得到的经验值, 在第 4 节的实验中会介绍.

3.3 模型训练

在完成数据集的分集之后, 开始模型训练. 我们按顺序交替训练两个数据子集, 当训练具有更高分数的子集 *A* 时, 使用较大的权重衰减系数λ_A来增强对模型复杂度的抑制; 反之, 较低分数的子集 *B* 使用较小的权重衰减系数λ_B. 权重衰减因子的确定在第 4 节实验中进行介绍.

整个过程如下算法 1 所示.

算法 1. 基于训练样本数据差异的网络训练流程

1. **For** *epoch* in *range*(*Epochs*):
2. 使用训练集对模型进行训练
3. **Endfor**

4. 保存模型为预训练模型
5. **For** *i* in *range*(训练集):
6. 使用预训练模型计算每一个训练样本输出的标签位 *Softmax* 值作为分值:

$$\sigma(z)_{label\_index}=\frac{e^{Z_{label\_index}}}{\sum_{j=1}^{K}e^{Z_j}}.$$

7. **Endfor**
8. 依据分值对训练集的样本进行排序, 分值 Top 10%的样本划分为训练子集 *A*, 其余样本为训练子集 *B*.
9. **For** *epoch* in *range*(2**Epochs*):
10. **if** (*epoch*%2==0):
    权重衰减因子=λ_A
    使用子集 *A* 训练模型
    else:
    权重衰减因子=λ_B
    使用子集 *B* 训练模型
    Endif
11. **Endfor**

4 实 验

为了验证我们方法的效果, 我们在典型的分类任务数据集, 包含: 1) SVHN[29]; 2) CIFAR10 和 CIFAR100[30]; 3) Tiny ImageNet[31]以及 4) 细粒度分类数据集 CUB-200-2011 鸟类数据集上, 使用 ResNet[32], PreResNet, VGG[33], DensNet[34], Wide ResNet[35]这些流行的深度神经网络模型进行训练及验证. 在实验过程中, 我们使用 Top-1 平均分类精度进行评估.

4.1 参数设置

实验中, 所有网络都训练 300 个 *Epoch*, 将学习率设置为 0.1, 然后在第 150 个 *Epoch* 和第 225 个 *Epoch* 分别除以 10.使用 0.0001 的权重衰减系数作为基线. 为了公平比较, 学习率等所有训练参数都是相同设置.

在实验过程中, 我们主要有两个超参需要调节: 一个是高分子集 A 占整个训练集的比例, 另一个是高分子集 A 的权重衰减因子. 我们首先使用训练数据集对待测试模型进行训练, 得到基线数据, 同时也得到预训练模型; 然后, 我们使用预训练模型对训练数据集的每一个样本进行评估打分, 通过评估分值将训练数据集分为 *A*, *B* 两个子集. 通过系列实验得出的经验, 如图 4 所示, 我们使用 CIFAR100 数据集训练 ResNet56 网络, 在相同的参数设置下, 使用不同的高分样本集比例进行分集训练实验.

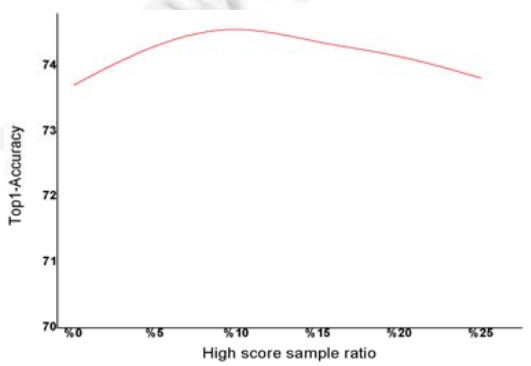


图 4 不同的高分样本分集比例在 CIFAR100 数据集、ResNet56 模型场景的性能表现

通过实验结果我们能够发现: 高分样本比例在 5%–15%的区间都能够有比较明显地增益, 10%左右较为突出, 所以在实验中采用高分样本分集 10%这个策略. 我们用子集 A 和 B 轮流训练待测试模型, 每个子集 300 个 *Epoch*, 学习率的策略同上. 使用子集 A 训练时, 采用较大权重衰减系数; 子集 B 采用较小的权重衰减系数.

我们对占训练集样本数 90%的子集 B 仍然采用基线的权重衰减因子 $1e^{-4}$; 而对于子集 A, 通过系列的实验, 我们发现 λ_A 在 $5e^{-4}$ 到 $2.5e^{-3}$ 之间都能够有比较显著的增益. λ_A 的取值应由模型的复杂度和训练数据集的信息量的相对关系来决定: 相对于数据集的信息量拥有较复杂结构, 较多参数的模型应使用较大的 λ_A; 反之, 应使用相对较小的 λ_A. 这个推论在后续的实验中基本得到验证.

4.2 SVHN数据集

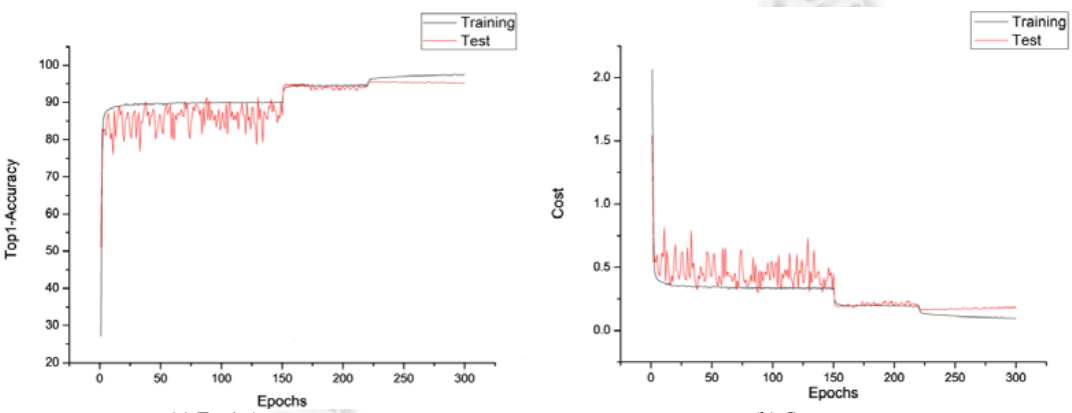
SVHN 是一个现实世界的图像数据集, 用于开发机器学习和对象识别算法, 而对数据预处理和格式化的要求最低. SVHN 是从 Google 街景图像中的门牌号获得的, 它包含 73 257 个训练样本和 26 032 个测试样本.

在 SVHN 数据集上, 我们使用 ResNet20, ResNet56 进行验证. 子集 A 为 7 325 个训练样本, 子集 B 为 65 932 个训练样本, 实验结果如表所示.

表 1 SVHN 上不同网络的 TOP-1 准确性(%)

模型		基线	λ(子集 A)	λ(子集 B)	我们的方法
ResNet	ResNet20	95.76	5e−4	1e−4	96.81
			1.5e−3	1e−4	95.78
			2e−3	1e−4	95.69
	ResNet56	96.63	1e−3	1e−4	96.68
			1.5e−3	1e−4	96.72
			2e−3	1e−4	96.76

从表 1 可以看出: 无论是 ResNet20 还是 ResNet56, 我们的方法在 SVHN 数据集上的增益作用都比较有限. 我们认为, 这两个模型在 SVHN 数据集上本身表现就比较优秀(如图 5 所示). 从 Cost 曲线和准确率曲线来看, 在训练集和测试集的表现已经非常接近了, 过度拟合现象并不突出. 因此, 抑制过度拟合的策略无法带来显著的收益.



(a) Top1-Accuracy (b) Cost curves

图 5 Cost 和准确率曲线、SVHN 数据集、ResNet20

4.3 CIFAR10/CIFAR100数据集

CIFAR10 数据集包含 10 个类别的 60 000 个 32×32 彩色图像, 每个类别 6 000 张图像, 50 000 张训练图像和 10 000 张测试图像. CIFAR100 与 CIFAR10 相似, 只是它有 100 个类别, 每个类别包含 600 张图像, 500 张训练图像和 100 张测试图像.

在 CIFAR10 数据集上, 我们使用 ResNet56, ResNet110, PreResNet56, PreResNet110 进行验证. 子集 A 为 5

000 个训练样本, 子集 *B* 为 45 000 个训练样本, 实验结果如表 2 所示.

表 2 CIFAR10 上不同网络的 TOP-1 准确性(%)

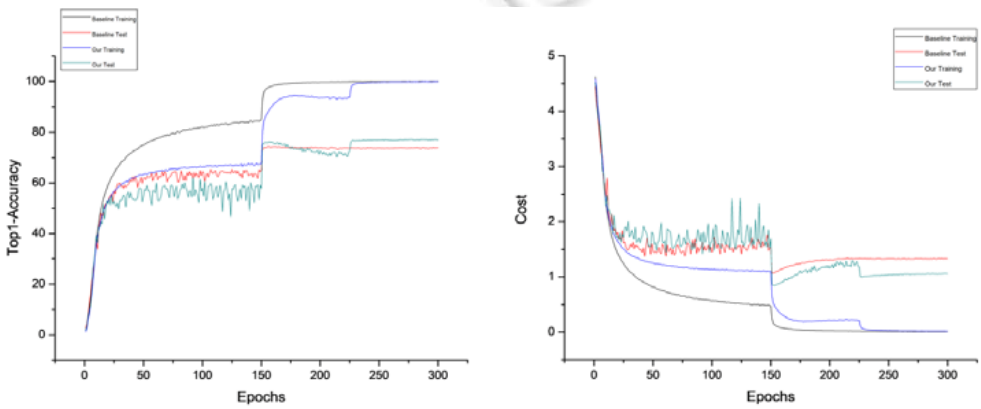
模型		基线	λ(子集 *A*)	λ(子集 *B*)	我们的方法
ResNet	ResNet56	93.82	1e−3	1e−4	94.36
	ResNet110	94.35	1.5e−3	1e−4	94.87
PreResNet	PreResNet56	94.06	1e−3	1e−4	94.49
	PreResNet110	95.08	1e−3	1e−4	95.15

在 CIFAR100 数据集上, 我们使用 ResNet56, ResNet110, PreResNet56, PreResNet110, VGG16, VGG19, DenseNet100, Wide ResNet28 进行验证. 子集 *A* 为 5 000 个训练样本, 子集 *B* 为 45 000 个训练样本, 实验结果如表 3 所示.

表 3 CIFAR100 上不同网络的 TOP-1 准确性(%)

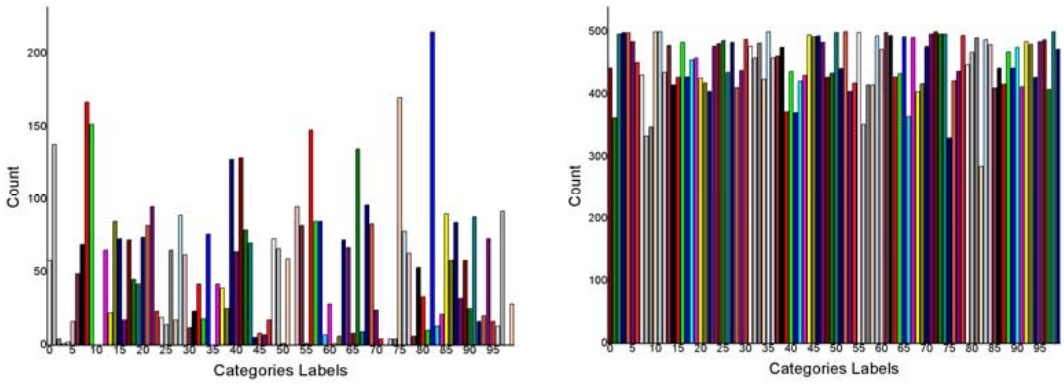
模型		基线	λ(子集 *A*)	λ(子集 *B*)	我们的方法
ResNet	ResNet56	73.71	1e−3	1e−4	74.52
			1.5e−3	1e−4	74.66
			2e−3	1e−4	74.57
	ResNet110	74.6	1.8e−4	1e−4	76.07
			2.5e−3	1e−4	76.41
PreResNet	PreResNet56	74.18	1e−3	1e−4	74.46
			1.5e−3	1e−4	74.24
	PreResNet110	76.09	1.5e−3	1e−4	76.74
VGG	VGG16	72.45	1.5e−3	1e−4	74.67
	VGG19	73.14	1.5e−3	1e−4	73.74
DenseNet	DenseNet100	77.21	1e−3	1e−4	77.30
			5e−4	1e−4	77.59
Wide ResNet	Wide ResNet28	79.07	2e−3	1e−4	80.12
			2.5e−3	1e−4	80.07

图 6 展示了在 CIFAR100 数据集上, ResNet110 模型在我们的方法和基线的 Cost 曲线及准确率曲线的对比. 可以看出: 我们的方法相对于基线最终在训练集和测试集的曲线更加接近, 泛化性更好. 我们也注意到:图中展示的训练过程中, 我们方法的测试准确率曲线和测试 Cost 曲线对比基线的训练过程整体波动较大, 特别是在学习率下降之前. 这种现象我们认为是由于子集 *A*, *B* 中, 各类别的样本数量并不均匀, 不是 1:1 的关系. 两个子集中的样本数量构成如图 7 所示, 各类别进入到评分 Top 10%的数量是不一样的, 甚至一些类别都没有样本数据评分进入前 10%. 这样对两个分集差异化交替训练的前期会导致一些模型对不同类别识别性能的不均衡, 这个现象在学习率下降后逐步收敛改善乃至消失.



(a) Top1-Accuracy (b) Cost curves

图 6 Cost 和准确率曲线, CIFAR100 数据集、Renst110、基线和本文方法对比



(a) Subset A (b) Subset B

图 7 CIFAR100 训练子集 *A*, *B* 中的样本类别数量分布

4.4 Tiny-ImageNet数据集

Tiny-ImageNet 数据集与完整的 ImageNet 数据集相似. Tiny-ImageNet 包含 200 个类别, 每个类别有 500 张图片. 测试集包含 10 000 张图像, 所有图像均为 64×64 彩色图像. 由于测试集标签是非公开的, 因此将验证集视为实验中的测试集.

在 Tiny-ImageNet 数据集上, 我们使用 ResNet50, PreResNet56, PreResNet110, DenseNet121 进行验证. 子集 *A* 为 10 000 个训练样本, 子集 *B* 为 90 000 个训练样本, 实验结果如表 4 所示. 从实验结果能看出, 我们提出的方法有效地提高了这些模型在 Tiny-ImageNet 数据集上的性能.

表 4 Tiny-ImageNet 上不同网络的 TOP-1 准确性(%)

模型		基线	λ(子集 *A*)	λ(子集 *B*)	我们的方法
ResNet	ResNet50	60.47	2e−3	1e−4	62.04
PreResNet	PreResNet56	58.60	1e−3	1e−4	60.50
	PreResNet110	60.97	1e−3	1e−4	64.56
DenseNet	DenseNet121	63.67	2e−3	1e−4	65.90

图 8 显示了网络中最后一个卷积层的输出的激活映射图[36], 与基线网络激活情况相比, 我们激活的位置更准确.

图 8 Tiny ImageNet 数据集、ResNet50、我们的方法与原始网络的类激活映射的对比

4.5 Caltech-UCSD Birds-200-2011数据集

细粒度分类是近年来的热门课题. “细粒度”是指在常见物种分类下的更细粒度分类, 例如特定鸟类物种的鉴定和狗品种的鉴定, 这对模型的性能提出了更高的要求. 我们使用 Caltech-UCSD Birds-200-2011[37]数据集来验证我们的方法在细粒度分类领域的有效性.

Caltech-UCSD Birds-200-2011(CUB-200-2011)涵盖了 200 种鸟类, 包括 5 994 张训练图像和 5 794 张测试图像. 除类别标签外, 每张图像还将使用 1 个边界框、15 个关键点和 312 个属性进行注释.

在 CUB-200-2011 数据集上, 我们使用 ResNet50 进行验证. 子集 *A* 为 616 个训练样本, 子集 *B* 为 5 378 个训练样本, 实验结果如表 5. 从实验结果可以看出: 在细粒度分类任务中, 我们的方法仍然具有明显的效果.

表 5 CUB-200-2011 上不同网络的 TOP-1 准确性(%)

模型		基线	λ(子集 *A*)	λ(子集 *B*)	我们的方法
ResNet	ResNet50	76.58	5e−4	1e−4	77.27
			1e−3	1e−4	77.70

5 结 论

在本文中, 我们研究了训练样本数据中个体差异对模型泛化的影响, 并提出了对不同样本数据使用不同正则化处理的策略. 我们介绍了一种基于训练样本的评估结果使用不同权重衰减率的方法, 而不是在整个深度神经网络模型训练过程中使用恒定权重衰减率的方法. 该方法简单、轻巧、有效, 可广泛应用于深度神经网络的训练过程中. 实验结果证明了其有效性, 它可以有效地抑制过度拟合并提高模型的泛化性.

References:

[1] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Proc. of the Advances in Neural Information Processing Systems. 2012. 1097−1105.

[2] LeCun Y, Boser B, Denker JS, *et al*. Backpropagation applied to handwritten zip code recognition. Neural Computation, 1989, 1(4): 541551.

[3] Zhang CY, Bengio S, Hardt M, *et al*. Understanding deep learning requires rethinking generalization. arXiv:1611.03530, 2016.

[4] Prechelt L. Early stopping-but when? In: Proc. of the Neural Networks: Tricks of the Trade. Springer, 1998. 5569.

[5] Srivastava N, Hinton GE, Krizhevsky A, *et al*. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 2014, 15(1): 192991958.

[6] Cook JA, Ranstam J. Overfitting. British Journal of Surgery, 2016, 103(13): 1814.

[7] Lawrence S, Giles CL. Overfitting and neural networks: Conjugate gradient and backpropagation. In: Proc. of the IEEE-INNS-ENNS Int'l Joint Conf. on Neural Networks (IJCNN 2000). Neural Computing: New Challenges and Perspectives for the New 325 Millennium. Vol.1. 2000. 114119.

[8] He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition. 2016. 770−778.

[9] Zhang YC, Lee JD, Jordan MI. l_1-regularized neural networks are improperly learnable in polynomial time. In: Proc. of the Int'l Conf. on Machine Learning. 2016. 993101001.

[10] Rastegari M, Ordonez V, Redmon J, *et al*. Xnor-Net: Imagenet classifification using binary convolutional neural networks. In: Proc. of the European Conf. on Computer Vision. Springer, 2016. 525542.

[11] Tang W, Hua G, Wang L. How to train a compact binary neural network with high accuracy? In: Proc. of the 31st AAAI Conf. on Artifificial Intelligence. 2017.

[12] Krogh A, Hertz JA. A simple weight decay can improve generalization. In: Proc. of the Advances in Neural Information Processing Systems. 1992. 950−957.

[13] Snoek J, Rippel O, Swersky K, *et al*. Scalable bayesian optimization using deep neural networks. In: Proc. of the Int'l Conf. on Machine Learning. 2015. 2171 180.

[14] Shahriari B, Bouchard-Côté A, Freitas N. Unbounded Bayesian optimization via regularization. In: Proc. of the Artifificial Intelligence and Statistics. 2016. 1168−1176.

[15] Ishii M, Sato A. Layer-Wise weight decay for deep neural networks. In: Proc. of the Pacifific-Rim Symp. on Image and Video Technology. Springer, 2017. 276−289.

[16] Nakamura K, Hong BW. Adaptive weight decay for deep neural networks. IEEE Access, 2019, 7: 118857118865.

[17] Park JG, Jo SH. Bayesian weight decay on bounded approximation for deep convolutional neural networks. IEEE Trans. on Neural Networks And Learning Systems, 2019, 30(9): 28662875.

[18] Srivastava N, Hinton GE, Krizhevsky A, *et al*. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 2014, 15(1): 1929−1958

[19] Ba LJ, Frey B. Adaptive dropout for training deep neural networks. In: Proc. of the Advances in Neural Information Processing Systems. 2013. 3084–3092.

[20] Kingma DP, Salimans T, Welling M. Variational dropout and the local reparameterization trick. In: Proc. of the Advances in Neural Information Processing Systems. 2015. 2575–2583.

[21] Noh H, You T, Mun J, *et al*. Regularizing deep neural networks by noise: Its interpretation and optimization. In: Proc. of the Advances in Neural Information Processing Systems. 2017. 5109–5118.

[22] Choe JS, Shim HJ. Attention-Based dropout layer for weakly supervised object localization. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition. 2019.

[23] Zhong Z, Zheng L, Kang G, *et al*. Random Erasing Data Augmentation. arXiv:1708.04896v2, 2017.

[24] Takahashi R, Matsubara T, Uehara K. Data augmentation using random image cropping and patching for deep cnns. IEEE Trans. on Circuits and Systems for Video Technology, 2019.

[25] Zoph B, Cubuk ED, Ghiasi G, *et al*. Learning data augmentation strategies for object detection. arXiv:1906.11172, 2019.

[26] Zhang HY, Cisse M, Dauphin YN, *et al*. *mixup*: Beyond empirical risk minimization. arXiv:1710.09412, 2017.

[27] Yun SD, Han DY, Oh SJ, *et al*. Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proc. of the IEEE Int'l Conf. on Computer Vision. 2019.

[28] Kline D, Berardi V. Revisiting squared-error and cross-entropy functions for training neural network classifiers. Neural Computing and Applications, 2005, 14(12): 310–318.

[29] Netzer Y, Wang T, Coates A, *et al*. Reading digits in natural images with unsupervised feature learning. 2011.

[30] Krizhevsky A. Learning multiple layers of features from tiny images. Technical Report, 2009.

[31] http://tiny-imagenet.herokuapp.com/

[32] Ren SQ, Sun J, He KM, Zhang XY. Deep residual learning for image recognition. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition. 2016.

[33] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: Proc. of the Int'l Conf. on Learning Representations. arXiv:1409.1556v6, 2015.

[34] van der Maaten L, Weinberger KQ, Huang G, *et al*. Densely connected convolutional networks. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: Hawaii Convention Center, 2017.

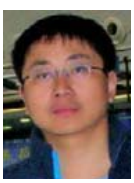
[35] Zagoruyko S, Komodakis N. Wide residual networks. In: Proc. of the British Machine Vision Conf., 2016.

[36] Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features for discriminative localization. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). 2016. 2921–2929.

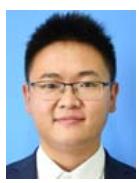
[37] Catherine W, Branson S, Welinder P, *et al*. The caltech-UCSD birds-200-2011 dataset. 2011.

李响(1982－), 男, 硕士, 工程师, CCF 会员, 主要研究领域为人工智能, 大数据.

姜庆(1982－), 男, 硕士, 工程师, 主要研究领域为大数据, 人工智能.



刘明(1972－), 男, 博士, 教授, 博士生导师, CCF 会员, 主要研究领域为人工智能, 大数据, 泛在网络.



曹扬(1981－), 男, 硕士, 高级工程师, 主要研究领域为大数据, 人工智能.

刘明辉(1990－), 男, 博士, 主要研究领域为人工智能, 大数据, 对抗生成网络.