高技术通讯 2023 年 第33 卷 第12 期:1276-1285

doi:10.3772/j.issn.1002-0470.2023.12.005

面向知识蒸馏的自动梯度混合方法①

曹灵宣②*** 常 明*** 张 蕊③*** 支 天*** 张曦珊***

(*中国科学技术大学 合肥 230026)

(**中国科学院计算技术研究所 北京 100190)

(***中科寒武纪科技股份有限公司 北京 100191)

摘 要 在知识蒸馏(KD)中,学生网络会同时受到真实数据的监督和来自教师网络的监督,因此在训练中,其损失函数包含有来自真实标签的任务损失和来自教师网络的蒸馏损失,而如何有效配置损失函数的权重至今仍是一个未解决的问题。为了克服这个问题,本文提出了一种自动梯度混合(AGB)方法,通过搜索这2个损失的最佳混合梯度来自动有效地找到合适的损失权重。在知识蒸馏的原始设计中,蒸馏损失是用来辅助任务损失进行训练,因此本文将混合梯度的模长约束为任务损失对应梯度模长,仅仅只搜索梯度向量的方向,从而显著缩减了搜索空间。在搜索得到最佳混合梯度后,2个损失的损失权重可以被自动计算出来,从而避免了耗时的手动调节过程。本文在13种不同的师生网络组合以及10种不同的知识蒸馏方法间进行了大量的实验。结果表明,自动梯度混合方法能够在使用更少计算资源的条件下,在70%的蒸馏方法上比手动调节方法结果更优。

关键词 深度神经网络(DNN);知识蒸馏(KD);超参数优化(HPO);图像分类

0 引 言

近年来,得益于越来越深的网络层数和越来越大的参数量,深度神经网络(deep neural networks,DNN)在各类任务中取得了显著的成功。然而,DNN有着较高的计算复杂度和较大的参数储存要求,因此将其部署到运算资源有限的设备或者对即时性要求较高的应用场景变得比较困难,例如智能手机、嵌入式设备和边缘计算。因此压缩大的模型并提高其运行速度变得非常重要。知识蒸馏(distilling the knowledge,KD)就是一种十分有效的模型压缩方法,其通过从大型教师网络中提取有用的知识转移给小型学生网络从而提高小型网络的性能。知识蒸馏的损失函数包含2个部分,一个是来自于真实标签的任务损失,另一个部分则是来自于教师网络的蒸馏损失。

因此,如何有效地找到2个损失函数的权重成为了一个待解决的问题。换言之,如何在训练过程中更合理地混合这2个损失的梯度。现在大多数已有的知识蒸馏方法都是手动调整损失权重,这种方法既繁琐又十分浪费计算资源,并且往往无法达到最佳性能。手动搜索权重的问题主要在于权重的搜索空间范围特别大,而且往往是连续的。例如,根据框架RepDistiller[1],在相同的数据集和师生网络组合下,蒸馏损失权重在0.02(基于概率的知识转移方法(probabilistic knowledge transfer,PKT)[2])到30 000(计算相关性的知识蒸馏方法(correlation congruence for knowledge,CC)[3])之间变化。

① 国家重点研发计划(2020AAA0103802),国家自然科学基金(61925208,61906179,62102399,U20A20227),中国科学院战略性先导科技专项(XDB32050200)和中国科学院稳定支持基础研究领域青年团队计划(YSBR-029)资助项目。

② 男,1998年生,硕士生;研究方向:计算机视觉;E-mail:jzdcjx@mail.ustc.edu.cn。

③ 通信作者,E-mail:zhangrui@ict.ac.cn。

(收稿日期:2022-09-15)

针对这个问题,可以采用超参数优化(hyperparameter optimization,HPO[4])和多任务学习(multi-task learning,MTL)来确定2个损失的权重,但将这2种方法应用到知识蒸馏的训练时存在着一些缺陷。在知识蒸馏训练中,存在2个优化目标:用于任务损失的真实标签以及用于蒸馏损失的教师网络。而知识蒸馏设计的初衷就是使用蒸馏损失作为辅助,帮助作为主要目标的任务损失降低到最小。但超参数优化和多任务学习方法会认为这2个损失处于一个平等情况,因此会产生大量冗余的搜索空间导致参数调节过程的效率十分低下,并且过分平衡用于辅助的蒸馏损失也有一定可能损害到主优化目标(即任务损失),后续多任务学习的实验也证明了这一点。

为了解决上述问题,本文提出了一种新颖的自动梯度混合方法,该方法可以自动地为知识蒸馏训练找到合适的损失函数权重。本文将寻找合适的损失函数权重的问题转换为寻找2个损失通过反向传播得到的最佳混合梯度的问题。考虑到在知识蒸馏中,蒸馏损失是任务损失的辅助这一重要的先验知识,自动梯度混合方法可以显著减少混合梯度的搜索空间。通过找到混合梯度的模长和方向从而确定用于更新模型参数的混合梯度。在具体训练过程中,混合梯度的模长用来控制模型参数更新速度,而方向则是决定着模型最终的训练结果。因此自动梯度混合方法通过固定混合梯度模长与任务损失产生的梯度模长相同,用来保证模型迭代的稳定性。在只需要搜索方向的情况下,可以有效地减少混合梯度的搜索空间并提高搜索效率。在确定了混合梯度的模长和方向后,就可以计算出2个损失函数的权重,从而避免了复杂的手动调节过程。

与现有的手动调节方法相比,本文提出的自动梯度混合方法有效利用了知识蒸馏的先验知识,具有以下几个优点:首先,自动梯度混合方法将混合梯度的模长约束到与任务梯度模长相同,这样能够保证模型训练的收敛稳定性,解耦了梯度向量模长和方向,只需要在方向上进行搜索,显著减少了搜索空间;此外,在进行了该梯度模长的约束后,早期训练轮次的结果与最终训练轮次的结果具备一个较好的保序性,从而通过一个极短时间的预训练即可找到较优的混合方向,从而实现了比手动设置权重更好的性能;最后,自动梯度混合方法是一种简单易用的方法,能够适用于绝大部分的知识蒸馏方法,可以对某种蒸馏方法在某类应用场景下是否有效进行一个快速验证。

为了证明自动梯度混合方法的效果,本文在CIFAR-100[5]和ImageNet-1k[6]数据集上使用Rep-Disitiller[1]框架进行实验,自动梯度混合方法在130个组别中表现超过70%的手动调节结果。在时间上,与超参数优化方法相比,自动梯度混合方法只需要1/10或者更少的时间就能达到与超参数优化方法相当的精度。

1 相关工作

1.1 知识蒸馏

知识蒸馏将大的、笨重的教师网络的知识转移给更小、更敏捷的学生网络中,从而能够有效提高学生网络的性能。Hinton等人[7]提出了这种方法,该方法使用温度来修正教师网络输出的softmax,使其作为软标签来指导小型的学生网络。目前有3种不同类型的知识蒸馏,分别是基于响应、基于特征和基于关系的知识蒸馏方法[8]。基于响应的方法[7]旨在通过使用教师网络的logits作为知识来直接模拟教师网络的最终预测。基于特征的方法[9-12]则是专注于匹配教师网络和学生网络中间层的特征。基于关系的方法[1,3,13-14]认为不同层或数据样本间的关系能有助于蒸馏。然而,现有绝大部分方法都使用手动调整来找到合适的任务损失权重和蒸馏损失权重,这既繁琐又十分耗时,而且往往无法达到最佳性能。

1.2 超参数优化

超参数优化(HPO)方法是一类寻找最优的超参数组合的方法。这些方法可以分成3类。第1类是穷举搜索,例如随机搜索和网格搜索。网格搜索将超参数空间划分为不同的网格并运行每个网格对应的参数组合以此找到最佳参数。这种遍历式的搜索方法由于没有对搜索空间进行任何裁剪,因此非

常耗时。为了使得搜索过程效率更高,研究人员提出了第 2 类启发式搜索方法,该类搜索方法可以在搜索过程中根据可用信息(例如之前训练的结果)选择后续最佳的搜索分支。超参数优化方法中包含有一些经典的启发式搜索方法,例如朴素进化和模拟退火。最近,研究人员也提出了 Hyperband[15]、Popluation-Based Training[16]等新的启发式方法。第 3 类是贝叶斯优化,它通过条件概率建模来预测给定超参数的最终性能,例如序列贝叶斯优化(sequential Bayesian optimization hyperband, BOHB)[17]、树形 Parzen 估计方法(tree-structured Parzen estimator approach, TPE)[18]等。与手动设置参数相比,超参数优化方法的调节器理论上可以节省一些搜索时间,但是仍然非常耗时。

1.3 多任务学习

多任务学习(multi-task learning, MTL)是指通过使用所有任务和其他一些任务中包含的知识来共同学习多个任务,以此来提高每个任务性能的一种训练方法。多任务学习方法包括 2 个方面[19]。一些多任务学习方法设计深度学习多任务架构,包含有设计侧重于编码器[20-21]或侧重于解码器[22-23]的架构。其他的一些多任务学习方法则是侧重于平衡多个任务的训练优化,例如 Uncertainly[24]、GradNorm[25]、DWA[26]、DTP[27]、Multi-Objective Optim[28]。绝大部分多任务学习方法都会等权重优化所有任务或者是所有损失函数,因此它可能会和知识蒸馏中将任务损失视为主要损失、将蒸馏损失视为辅助的理念相冲突。

2 自动梯度混合

2.1 问题定义

为了提高小型学生网络的性能,知识蒸馏除了利用来自于真实数据的监督外,还额外引入了来自于大型的教师网络中的有益的知识。因此,总的损失函数由来自于真实标签的任务损失和来自于教师网络的蒸馏损失构成,公式为

$$L_{\mathrm{kd}} = \alpha L_{\mathrm{task}} + \beta L_{\mathrm{distill}} \tag{1}$$

这里 L_{kd} 是总的知识蒸馏损失函数, L_{task} 是任务损失, L_{distill} 是蒸馏损失。α 和 β 是任务损失和蒸馏损失的缩放系数。为了获得合适的系数 α 和 β, 绝大部分已有的知识蒸馏方法都是通过手动调节方法来进行搜索,这类方法非常繁琐又耗时,并且往往无法使学生网络拥有最佳的性能。为了解决这个问题,本文提出了一种自动梯度混合方法来自动高效地找到损失权重。

假设在整个训练过程中,第 t 轮的模型参数更新迭代时,损失函数对模型参数求导后得到的梯度被用来迭代模型参数,公式为

$$W^{t+1} = W^t - \eta G_{\mathrm{kd}} \tag{2}$$

这里 $G_{\mathrm{kd}} = \dfrac{\partial L_{\mathrm{kd}}}{\partial W^t}$ 是损失函数的梯度。W^t 是学生网络处在第 t 轮迭代时的模型参数, η 是学习率。基于式(1),总损失函数的梯度可以表示为任务损失的梯度和蒸馏损失梯度的混合,公式为

$$G_{\mathrm{kd}} = \alpha G_{\mathrm{task}} + \beta G_{\mathrm{distill}} \tag{3}$$

这里 $G_{\mathrm{task}} = \dfrac{\partial L_{\mathrm{task}}}{\partial W^t}$ 是任务损失的梯度,而 $G_{\mathrm{distill}} = \dfrac{\partial L_{\mathrm{distill}}}{\partial W^t}$ 是蒸馏损失的梯度。因此,本文将找到合适的损失系数 α 和 β 的问题转换为搜索梯度 G_{task} 和 G_{distill} 的最优混合梯度的问题。

2.2 高效搜索混合梯度

为了有效搜索最优混合梯度,需要尽可能地缩小搜索空间。在这项工作中,本文利用了知识蒸馏中的一个重要的先验知识,即任务损失是主要优化目标,而蒸馏损失是任务损失的辅助。因此,混合梯度 G_{kd} 应当与任务梯度 G_{task} 更加相关,蒸馏梯度 G_{distill} 用来做一个细化调整。

本文通过确定混合梯度 G_{kd} 的方向和模长来找到这个混合梯度。一般而言,在使用梯度来更新模型参数的过程中,梯度向量具有 2 个自变量,一个是方向,另一个则是模长,两者的功能具有一定差异。梯度的模长主要影响着模型参数的更新速度,从而控制模型收敛,当模长太长时,会出现梯度爆炸使得模型无法收敛或者是在最优值附近徘徊的情况;而模长过短时,模型收敛会非常缓慢,找到最优值的时间过长,也有可能陷入到某个局部最优点中。梯度的方向则是决定着模型参数的更新方向,决定模型

最终的收敛位置能否在相应的指标上取得好的效果(如分类任务中的准确率,检测任务中的 mAP 等)。在非蒸馏训练中,模型仅使用任务损失产生的梯度就能训练出来一个稳定的结果。本文基于上述先验知识,为了提高效率减小搜索空间,以及保证模型训练的收敛稳定性,自动梯度混合方法将混合梯度的模长约束到与任务损失梯度模长相同,公式为

$$\| G_{\text{kd}} \| = \| G_{\text{task}} \| \tag{4}$$

在实现该约束后,可以很方便地将学生网络的非蒸馏训练版本的超参数,如学习率、权重衰减等,方便应用到本文中使用的蒸馏训练上。因此可以通过对 G_{kd} 的模长约束得到一个稳定的训练过程。

在确定了模长大小后,自动梯度混合方法只需要在搜索空间中搜索 G_{kd} 梯度方向,该梯度方向由任务梯度 G_{task} 和蒸馏梯度 G_{distill} 决定。如图 1 所示,G_{kd} 方向的搜索空间为 G_{task} 和 G_{distill} 之间的角度空间。θ 为 G_{task} 和 G_{distill} 夹角大小:

$$\cos\theta = \frac{G_{\text{task}} \cdot G_{\text{distill}}}{\| G_{\text{task}} \| \| G_{\text{distill}} \|} \tag{5}$$

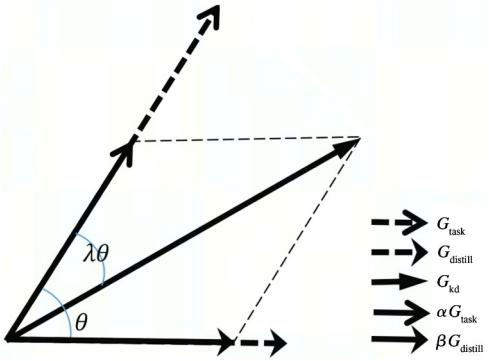

图 1　梯度混合示意图

假设 G_{task} 和 G_{kd} 的夹角为 $\lambda\theta$, 只需要在 $\lambda \in [0,1]$ 这个范围内进行搜索。在这种方式下,由于不需要对 G_{kd} 的模长进行搜索,整个搜索空间得到大幅度缩减,同时对最优方向的搜索可以保证混合梯度 G_{kd} 的有效性。

通过搜索得到 λ 后,可以用如下公式表示 G_{kd} 的方向:

$$\cos\lambda\theta = \frac{G_{\text{task}} \cdot G_{\text{kd}}}{\| G_{\text{task}} \| \| G_{\text{kd}} \|} \tag{6}$$

使用式(3)和(4),可以得到:

$$\| G_{\text{task}} \| = \| \alpha G_{\text{task}} + \beta G_{\text{distill}} \| \tag{7}$$

$$\cos\lambda\theta = \frac{G_{\text{task}} \cdot (\alpha G_{\text{task}} + \beta G_{\text{distill}})}{\| G_{\text{task}} \| \| \alpha G_{\text{task}} + \beta G_{\text{distill}} \|} \tag{8}$$

联立式(5)~(7),可以解得损失权重系数 α 和 β 为

$$\alpha = \cos\lambda\theta - \frac{\sin\lambda\theta}{\tan\theta},\ \beta = \frac{\| G_{\text{task}} \| \sin\lambda\theta}{\| G_{\text{distall}} \| \sin\theta} \tag{9}$$

2.3　热身策略

如式(9)所示,损失权重系数 α 和 β 取决于 λ。λ 的有效值为[0,1]。当 λ 等于 0 时,蒸馏损失对混合梯度没有任何影响;当 λ 等于 1 的时候,混合梯度方向会完全遵循蒸馏梯度的方向。此外,实验结果表明自动梯度混合方法在训练早期和后期的性能(在分类任务中为准确率)有着良好的保序性。因此,为了进一步提高搜索过程中实验的效率,本文使用训练早期的训练效果来预测最终的性能。在具体操作中,本文在搜索空间中对 λ 进行一个早期的搜索来作为预热训练模型。然后选择性能最佳的一个作为 λ 的最佳值。之后可以采用式(9)来计算损失权重 α 和 β,并且使用它们来完成训练。搜索和训练模型的整个过程如算法 1 所示。

算法 1　自动梯度混合算法训练流程

```
1.  阶段 1:
2.  设置 acc_best = 0, λ_select
3.  for λ = 0.1 to 0.9, stepsize = 0.2 do
4.      for epoch = 0 to epocs_warm-up do
5.          train model
6.      end for
7.      if acc > acc_best then
8.          λ_select = λ
9.      end if
10.     end for
11. 阶段 2:
12. λ = λ_select
13. for epoch = 0 to epocs_train do
14.         train model
15. end for
```

3　实验和结果

本节中,本文将提出的自动梯度混合方法应用在被广泛使用的图像分类数据集 CIFAR-100[5] 和 ImagNet LSVRC 2012[6] 上。此外,本文使用的 Rep-

Distiller[1]框架基于 Pytorch,其模型库中包含有 13 种流行的蒸馏方法。在实验中,本文遵循 RepDistiller 默认的超参数设置,如训练轮次、学习率、优化器等。在自动梯度混合方法中,预热轮次设置为 5。作为对比实验,本文使用 RepDistiller 中给出的手动调整的损失权重的训练结果作为基线。

3.1 CIFAR-100 上的实验结果

本文在 KD[7]、Fitnets[11]、SP[29]、AT[12]、CC[3]、VID[29]、RKD[13]、PKT[3]、FT[10]和 NST[9]这 10 种蒸馏方法上进行实验。此外,实验还包含有 7 个相似架构的师生网络组合和 6 个不同架构的师生网络架构,即整个实验包含有 10×13 个小的实验。

结果如表 1 所示,可以发现自动梯度混合方法和手动方法比较,无论是在教师网络架构和学生网络架构相似的 VGG13-VGG8 和 ResNet110-ResNet32 亦或者是 ResNet32x4-ShuffleNetV2 和 VGG13-MobileNetV2 这类架构差异很大的网络上都有比较好的效果。总结表 1 的结果可以发现,自动梯度混合方法在 70% 的蒸馏组合上都要比手动调节的方法表现得更好。

表 1 在数据集 CIFAR-100 上使用手动调节(Manual)和 自动梯度混合方法(AGB)在 10 种不同的蒸馏方法和 13 种不同的师生网络组合的 Top-1 准确率(%)

教师网络	学生网络	方法	KD	FitNet	SP	AT	CC	VID	RKD	PKT	FT	NST
WRN-40-2	WRN-16-2	Manual	74.92	73.58	74.08	73.83	73.56	74.11	73.35	74.54	73.25	73.68
(75.61)	(73.26)	AGB	74.79	73.24	73.90	74.29	73.65	74.27	73.10	75.95	74.14	73.84
WRN-40-2	WRN-40-1	Manual	73.54	72.24	72.43	72.77	72.21	73.30	72.22	73.45	71.59	72.24
(75.61)	(71.98)	AGB	74.10	71.16	72.26	72.73	72.47	73.92	73.47	73.89	73.23	72.63
ResNet56	ResNet20	Manual	70.66	69.21	70.55	69.67	69.63	70.38	69.61	70.34	69.82	69.60
(72.34)	(69.06)	AGB	71.87	69.67	71.11	70.70	69.50	70.19	69.28	71.01	71.15	70.12
ResNet110	ResNet20	Manual	70.67	68.99	70.22	70.04	69.48	70.16	69.25	70.25	70.22	69.53
(74.31)	(69.06)	AGB	70.65	69.47	70.63	69.98	69.64	69.95	69.92	70.85	69.96	69.51
ResNet110	ResNet32	Manual	73.08	71.06	72.31	72.69	71.48	72.61	71.82	72.61	72.37	71.96
(74.31)	(71.14)	AGB	73.55	71.21	72.89	72.84	71.74	72.38	72.38	73.48	73.14	71.05
ResNet32x4	ResNet8x4	Manual	73.33	73.50	73.44	72.94	72.97	73.09	71.90	73.64	72.86	73.30
(79.42)	(72.50)	AGB	73.39	72.59	73.30	73.32	73.39	73.13	72.31	74.18	73.60	73.64
VGG13	VGG8	Manual	72.98	71.02	72.68	71.43	70.71	71.23	71.48	72.88	70.58	71.53
(74.64)	(70.36)	AGB	73.25	72.14	72.90	72.48	71.08	74.24	71.46	73.79	70.97	71.96
VGG13	MobileNetV2	Manual	67.37	64.14	66.30	59.40	64.86	65.56	64.52	67.13	61.78	58.16
(74.64)	(64.60)	AGB	68.48	64.01	65.84	67.28	65.65	65.82	65.66	68.96	66.00	62.82
ResNet50	MobileNetV2	Manual	67.35	63.16	58.58	68.08	65.43	67.57	64.43	66.52	60.99	64.96
(79.34)	(64.60)	AGB	67.66	63.06	64.46	65.99	65.77	62.41	65.37	68.42	66.31	64.83
ResNet50	VGG8	Manual	73.81	70.69	71.84	73.34	70.25	70.30	71.50	73.01	70.29	71.28
(79.34)	(70.36)	AGB	73.01	67.40	71.47	71.58	71.22	71.92	71.39	74.11	71.87	69.62
ResNet32x4	ShuffleNetV1	Manual	74.07	73.59	71.73	73.48	71.14	73.38	72.28	74.10	71.75	74.12
(79.42)	(70.50)	AGB	73.13	72.58	73.75	73.27	71.46	77.13	72.70	74.54	73.18	75.81
ResNet32x4	ShuffleNetV2	Manual	74.45	73.54	74.56	72.73	71.29	73.40	73.21	74.69	72.50	74.68
(79.42)	(71.82)	AGB	75.08	74.29	74.18	73.95	73.95	73.64	73.56	75.54	74.11	76.08
WRN-40-2	ShuffleNetV1	Manual	74.83	73.73	73.32	74.52	71.38	73.89	72.21	73.89	72.03	74.89
(75.61)	(70.50)	AGB	76.05	73.93	74.66	73.14	72.39	73.89	72.88	73.60	73.20	74.57

3.2 ImageNet-1K 上的实验结果

本文使用 KD、CC、对比表示知识蒸馏方法(contrastive representation distillation,CRD)和注意知识蒸馏方法(attention on distillation,AT)在 Image Net-1K 数据集进行实验。因为 RepDistiller 框架没有 ImageNet-1K 对应代码,所以本文在 ImageNet-1K 上复现了这 4 种方法。超参数和手动调整的损失权重是按照另一个蒸馏框架 TorchDistil 设置的。本文使用

Pytorch 团队发布的模型 ResNet34 和 ResNet18 作为教师和学生网络,并遵循 TorchDistill 的 ImageNet 训练设置。

表 2 展示了自动梯度混合方法和手动参数设置方法在以 ResNet34 和 ResNet18 作为师生网络组合上的 top-1 准确度。对于 KD、CC 和 AT 方法,自适应梯度混合方法可以获得更好的性能,对于 CRD 方法,自动梯度混合方法也可以达到和手动设置接近的性能。因此,ImageNet-1K 上的实验有效证明了自动梯度混合方法的有效性。

表 2　自动梯度混合方法(AGB)和手动调整(Manual)在 ImageNet-1k 上的 Top-1 准确度(%),其中教师网络是 ResNet34(top-1 准确度 73.314%),学生网络是 ResNet18(top-1 准确度 69.76%)

方法	KD	CC	CRD	AT
Manual	71.37	70.49	70.80	70.63
AGB	71.74	70.60	70.73	71.04

3.3　和超参数优化方法的比较

本文在 CIFAR100 上使用自动梯度混合方法和 Microsoft Neural Network Intelligence (NNI)的 3 个不同的超参数优化调节器进行了对比。这些超参数优化方法包括有启发式搜索方法模拟退火(simulated annealing)、Hyperband[15] 和贝叶斯优化方法 TPE[18]。选择 VGG13 和 VGG8 作为师生网络,并使用 AT 蒸馏方法进行实验,在超参数优化方法中,参照式(1),设置 α 等于 1,β 的搜索空间为 0.02 到 30 000。

图 2 显示了 3 个超参数优化调节器和自动梯度混合方法的比较实验。可以观察到自动梯度混合方法只需要极少训练的时间就能达到非常高的精度。相比之下,在运行同样的时间中,超参数优化方法只能实现更低的精度。尽管超参数优化方法在最终的结果中达到了与自动梯度混合方法相当或者略高的精度,但它们需要更多的时间来进行搜索,这是非常低效的。

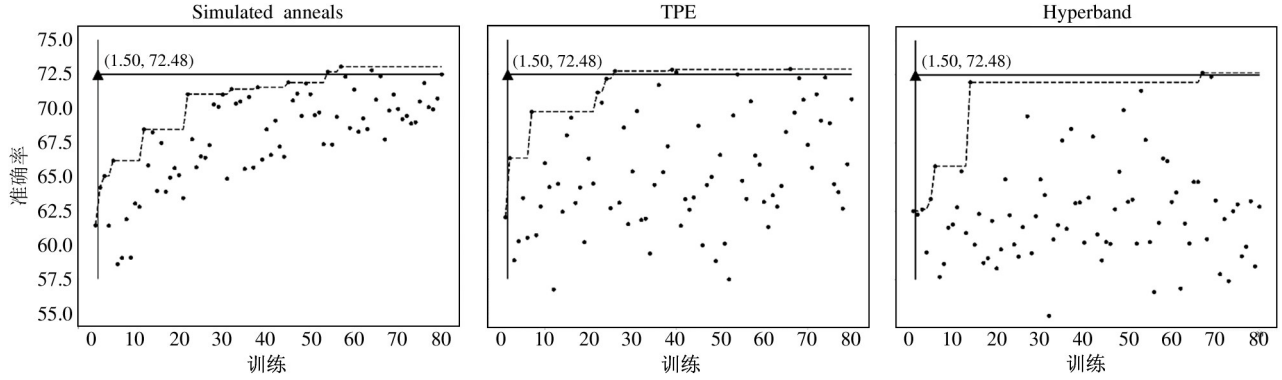

图 2　CIFAR-100 上超参数优化方法和自动梯度混合方法的精度变化

分析超参数优化方法出现的问题,可以发现无论是手动调节、超参数优化或者是一些简单约束情况,都会导致超参数搜索过程变得漫长而复杂。本质上,这是由于这类方法在搜索超参数时会将总梯度向量模长和方向进行耦合,同时去搜索梯度向量的方向和模长,会影响模型的收敛性,并出现两类冗余搜索的情况:(1)搜索到合适的方向而模长过长或过短,导致出现模型无法收敛;(2)搜索到合适的模长而方向不对,这样会影响模型最终的收敛位置,即影响模型最终的结果。而当一些更为奇怪的约束使得总梯度向量的方向与模长耦合得更加紧密时,甚至无法搜索到对应合适方向。

3.4　和多任务学习方法的比较

将 Uncertainly 和 GradNorm 这 2 种无超参数的多任务学习方法与自动梯度混合方法进行对比实验。本文对所有的 10 种蒸馏方法进行了实验,所有的 13 种教师学生网络组合与 3.1 节中的相同。

如表 3 所示,自动梯度混合方法应用到绝大多数蒸馏方法中都优于这 2 种多任务学习方法。多任务学习方法将蒸馏损失和任务损失平等对待,忽略

表 3 多任务学习方法 GradNorm 和 Uncertainly 在 CIFAR-100 上与自动梯度混合方法(AGB)相比的 Top-1 测试准确度(%)。由于训练过程中的梯度爆炸,一些方法显示出非常差的准确性或无法训练出有效的结果(用\表示)。null 表示此蒸馏方法不支持多任务学习方法。

教师网络	学生网络	方法	KD	FitNet	SP	AT	CC	VID	RKD	PKT	FT	NST
WRN-40-2 (75.61)	WRN-16-2 (73.26)	Uncertain	**74.88**	**73.62**	null	73.30	6.84	27.19	**73.80**	73.14	74.03	**74.44**
		GradNorm	75.14	\	\	null	\	18.20	\	2.27	3.58	15.28
		AGB	74.79	73.24	**73.90**	**74.29**	**73.65**	**74.27**	73.10	**75.95**	**74.14**	73.84
WRN-40-2 (75.61)	WRN-40-1 (71.98)	Uncertain	73.57	**71.50**	null	68.37	13.80	71.08	71.54	71.36	72.15	72.07
		GradNorm	73.64	\	3.72	null	\	18.20	\	2.27	3.58	15.28
		AGB	**74.10**	71.16	**72.26**	**72.73**	**72.47**	**73.92**	**73.47**	**73.89**	**73.23**	**72.63**
ResNet56 (72.34)	ResNet20 (69.06)	Uncertain	71.45	69.50	null	69.69	11.08	67.66	**69.49**	67.56	69.56	70.07
		GradNorm	71.51	5.04	null	20.82	\	\	1.63	29.34	\	11.89
		AGB	**71.87**	**69.67**	**71.11**	**70.70**	**69.50**	**70.19**	69.28	**71.01**	**71.15**	**70.12**
ResNet110 (74.31)	ResNet20 (69.06)	Uncertain	70.32	68.90	null	67.61	12.37	69.13	69.35	68.01	69.30	**69.77**
		GradNorm	54.92	1.82	\	null	\	\	1.88	2.24	54.66	\
		AGB	**70.65**	**69.47**	**70.63**	**69.98**	**69.64**	**69.95**	**69.92**	**70.85**	**69.96**	**69.51**
ResNet110 (74.31)	ResNet32 (71.14)	Uncertain	73.43	**71.34**	null	71.04	9.94	**72.58**	73.32	53.91	Null	null
		GradNorm	56.61	\	63.69	null	\	\	46.27	59.38	\	4.39
		AGB	**73.55**	71.21	**72.89**	**72.84**	**71.74**	72.38	**72.38**	**73.48**	**73.14**	**71.05**
ResNet32x4 (79.42)	ResNet8x4 (72.50)	Uncertain	73.36	**73.22**	null	72.55	69.93	71.78	71.91	69.50	73.09	73.05
		GradNorm	**73.96**	13.84	1.01	null	\	2.31	57.56	\	\	4.39
		AGB	73.39	72.59	**73.30**	**73.32**	**73.39**	**73.13**	**72.31**	**74.18**	**73.60**	**73.64**
VGG13 (74.64)	VGG8 (70.36)	Uncertain	72.09	71.24	null	70.94	69.71	70.71	71.45	70.18	**71.03**	71.39
		GradNorm	53.89	17.93	null	8.60	\	20.48	28.38	\	\	\
		AGB	**73.25**	**72.14**	**72.90**	**72.48**	**71.08**	**74.24**	**71.46**	**73.79**	70.97	**71.96**
VGG13 (74.64)	MobileNetV2 (64.60)	Uncertain	67.69	63.78	null	66.73	63.82	65.59	63.65	65.25	65.76	**63.58**
		GradNorm	67.93	13.46	\	null	\	1.56	1.08	1.32	8.18	7.85
		AGB	**68.48**	**64.01**	**65.84**	**67.28**	**65.65**	**65.82**	**65.66**	**68.96**	**66.00**	62.82
ResNet50 (79.34)	MobileNetV2 (64.60)	Uncertain	67.93	**65.29**	null	**66.07**	65.29	**65.35**	63.13	null	null	null
		GradNorm	**68.94**	15.98	1.47	null	\	2.01	1.28	1.43	null	null
		AGB	**67.66**	63.06	**64.46**	65.99	**65.77**	62.41	**65.37**	**68.42**	**66.31**	64.83
ResNet50 (79.34)	VGG8 (70.36)	Uncertain	72.80	**70.65**	null	70.81	49.65	53.08	71.30	70.17	71.07	null
		GradNorm	62.78	20.30	\	null	\	\	1.46	2.39	2.90	null
		AGB	**73.01**	67.40	**71.47**	**71.58**	**71.22**	**71.92**	**71.39**	**74.11**	**71.87**	69.62
ResNet32x4 (79.42)	ShuffleNetV1 (70.50)	Uncertain	**74.17**	**72.83**	null	72.10	67.84	71.58	**73.01**	72.08	71.83	65.83
		GradNorm	74.51	1.12	1.76	null	\	\	17.66	1.04	2.17	1.12
		AGB	73.13	72.58	**73.75**	**73.27**	**71.46**	**77.13**	72.70	**74.54**	**73.18**	**75.81**
ResNet32x4 (79.42)	ShuffleNetV2 (71.82)	Uncertain	74.96	74.03	null	73.90	73.38	73.56	6.77	73.14	73.18	74.11
		GradNorm	75.86	1.14	8.19	null	1.55	2.98	2.18	1.56	8.15	4.44
		AGB	**75.08**	**74.29**	**74.18**	**73.95**	**73.95**	**73.64**	**73.56**	**75.54**	**74.11**	**76.08**
WRN-40-2 (75.61)	ShuffleNetV1 (70.50)	Uncertain	74.16	71.73	null	72.28	68.75	68.62	**73.51**	71.35	72.25	null
		GradNorm	41.60	7.11	\	null	\	\	4.29	1.16	5.51	null
		AGB	**76.05**	**73.93**	**74.66**	**73.14**	**72.39**	**73.89**	72.88	**73.60**	**73.20**	**74.57**

了知识蒸馏的重要先验知识,即任务损失是起到主导作用的,而蒸馏损失是用于辅助的。因此,多任务学习方法可能会为了最大限度地降低蒸馏损失而牺牲了性能。还可以发现,当使用 GradNorm 时,大多数蒸馏方法的性能都很差。这是因为 GradNorm 完全忽略了任务损失应该为主导地位。而且,与任务损失相比,蒸馏损失通常非常大或者非常小。例如,在 CC 中,蒸馏梯度的模长约为任务梯度的 100 倍,

而在 PKT 中,蒸馏梯度的模长约为任务梯度的0.001倍。因此,GradNorm 简单地平衡 2 个损失将会导致整个训练过程不稳定。相比之下,自动梯度混合方法将混合梯度的模长限制为与任务梯度的模长相同。因此,自动梯度混合方法在获得稳定训练过程的同时,可以保留任务梯度占据主导地位这一重要信息。

3.5 保序性证明

本文验证了在自动梯度混合方法中训练早期和训练后期准确率的保序性。在 CIFAR-100 上使用 AT 蒸馏方法进行这些实验,在 NNI 上用 VGG13 作为教师网络,VGG8 作为学生网络。计算早期(第 5 轮)的准确率和整个训练结束的最终准确率之间的相关系数。本文还对手动调节方法进行了这些实验,α 设置为 1,β 从 0.003 变化到 30 000。为了公平地比较,本文选择结果接近收敛时的最后 80 次实验来验证相关性。

如图3所示,下图为自动梯度混合方法,其相关

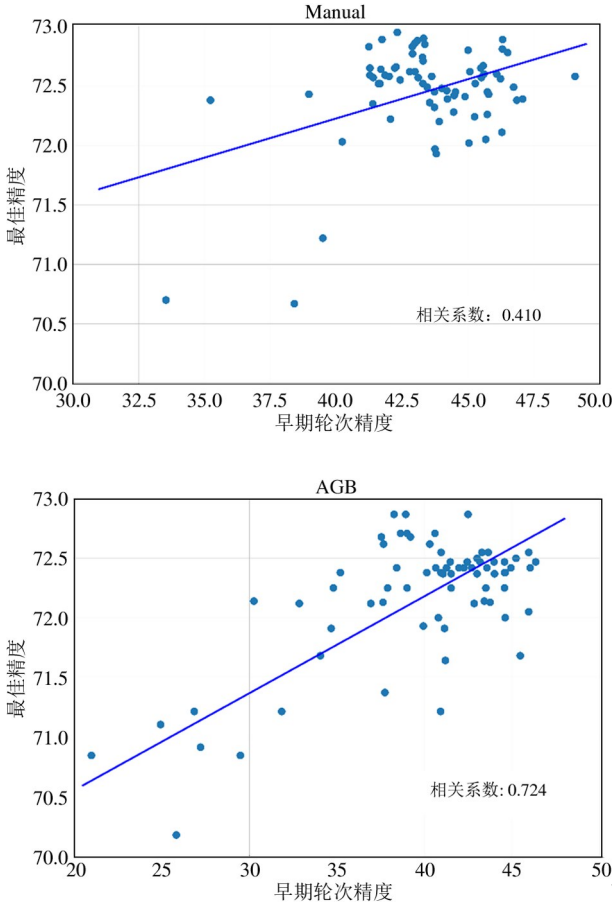

图 3 最佳精度与早期轮次精度之间的相关性

系数为0.724,远高于上图中手动调节方法的0.410。这个实验说明了使用自动梯度混合方法时早期轮次表现较好的设置同样可以运用到晚期轮次。因此,预热策略可以在不损失性能的前提下大幅提升自动梯度混合方法的效率。

3.6 消融实验

本文就预热阶段设置的热身轮次和预热阶段用于离散化的步长进行了消融实验。在 CIFAR-100 上使用 KD 蒸馏方法进行实验,教师网络为 ResNet32x4,学生网络为 ResNet8x4。

图 4 显示了准确率、时间开销与步长的关系。可以看到,当步长从 0.2 变小后,时间开销增大,对应的结果略有上升;而当步长变大后,实际上的节省的时间相当有限,而性能也会出现一定程度的下降。图 5 则显示了准确率、时间和热身轮次的关系。可以发现,与前面步长类似,选取更小的热身轮次并不会导致运行时间有一个显著的变小。而当热身轮次提升后,时间开销增大了,对于实验的准确率也没有提升太多。因此本文取的热身轮次和步长并不具备特殊性,取附近的几个值结果差异不会太大,这也说明了是前面模长约束在方法中起到了主要的作用而预热的等间距选取最优的策略只是用于辅助的。

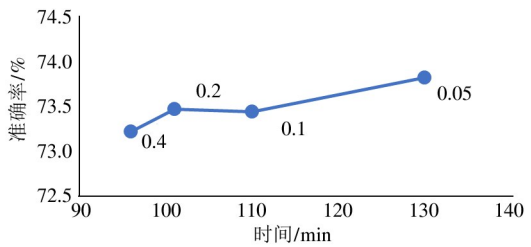

图 4 准确率、时间与步长之间的关系

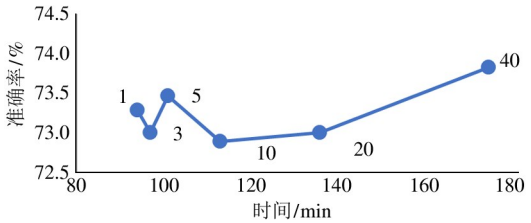

图 5 准确率、时间与热身轮次之间的关系

3.7 自动梯度混合方法的高效性

图 2 中的结果也显示了自动梯度混合方法的高

效性。图 2 中圆点表示一次超参数优化方法实验的准确性。随着训练实验的增加,每条虚线表示 超参数优化方法的最佳准确性。三角形标记表示自动梯度混合方法的结果,该方法需要大约 1.50 次实验时间才能达到 72.48% 的准确率。在知识蒸馏中寻找损失权重时,手动调整会受到大的搜索空间的影响。通过使用贝叶斯优化或者是其他算法改进搜索过程,超参数优化方法会高效一些,但是仍然有着比较大的搜索空间。相比之下,自动梯度混合方法通过约束混合梯度的模长并仅仅在预热阶段在方向上进行搜索,从而显著减少了搜索空间。如图 2 所示,超参数优化方法需要 10 次以上的实验才能达到与自动梯度混合方法相当的精度。因此,与超参数优化方法相比,自动梯度混合方法效率更高。

4 结 论

本文提出了一种自动梯度混合方法,可以有效地为绝大部分知识蒸馏方法找到合适的损失权重。利用蒸馏损失是用于辅助任务损失这一先验知识,自动梯度混合方法通过减少超参数搜索空间来优化搜索过程。自动梯度混合方法只搜索梯度方向,即 2 个损失梯度之间的角度,同时将混合梯度的模长约束为与任务损失梯度模长相同。本文在 13 种不同的师生网络组合之间对 10 种不同的知识蒸馏方法进行了实验。自动梯度混合方法在使用更少的运算资源的前提下在 70% 的蒸馏方法上性能超过了手动调节方法,这说明自动梯度混合方法具有更好的效果以及更高的效率。本文工作的前提是假设当有多个蒸馏损失时,所有的蒸馏损失共享相同的权重。未来,可以将本文工作扩展到具有多种蒸馏损失的情况。

参考文献

[1] TIAN Y, KRISHNAN D, ISOLA P. Contrastive representation distillation[C]// International Conference on Learning Representations. New Orleans: ICLR, 2019:1325-1334.

[2] PASSALIS N, TEFAS A. Learning deep representations with probabilistic knowledge transfer[C]// Proceedings of the European Conference on Computer Vision. Munich: ECCV, 2018:268-284.

[3] PENG B, JIN X, LIU J, et al. Correlation congruence for knowledge distillation[C]// International Conference on Computer Vision. Seoul: ICCV, 2019:5007-5016.

[4] JAMES B, REMI B, YOSHUA B, BALAZS K. Algorithms for hyper-parameter optimization[C]// Neural Information Processing Systems. Granada: NeurIPS, 2011: 2546-2554.

[5] ALEX K. Learning multiple layers of features from tiny images[D]. Toronto: University of Toronto, 2009.

[6] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017,60(6): 84-90.

[7] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network[EB/OL]. (2015-03-09)[2022-03-30]. https://arxiv.org/pdf/1503.02531.pdf.

[8] GOU J, YU B, MAYBANK S J, et al. Knowledge distillation: a survey[J]. International Journal of Computer Vision, 2021,129:1789-1819.

[9] HUANG Z, WANG N. Like what you like: knowledge distill via neuron selectivity transfer[EB/OL]. (2017-12-18)[2022-03-30]. https://arxiv.org/pdf/1707.01219.pdf.

[10] KIM J, PARK S U, KWAK N. Paraphrasing complex network: network compression via factor transfer[EB/OL]. (2020-07-22)[2022-03-30]. https://arxiv.org/pdf/1802.04977.pdf.

[11] ROMERO A, BALLAS N, KAHOU S E, et al. Fitnets: hints for thin deep nets[EB/OL]. (2015-03-27)[2022-03-30]. https://arxiv.org/pdf/1412.6550.pdf.

[12] KOMODAKIS N, ZAGORUYKO S. Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer[EB/OL]. (2017-01-24)[2022-03-30]. https://arxiv.org/pdf/1612.03928v2.pdf.

[13] PARK W, KIM D, LU Y, et al. Relational knowledge distillation[EB/OL]. (2019-04-10)[2022-03-30]. https://arxiv.org/pdf/1904.05068v1.pdf.

[14] TUNG F, MORI G. Similarity-preserving knowledge distillation[C]// International Conference on Computer Vision. Seoul: ICCV, 2019:1365-1374.

[15] LI L, JAMIESON K, DESALVO G, et al. Hyperband: a novel bandit-based approach to hyperparameter optimization[J]. The Journal of Machine Learning Research, 2017,18(1):6765-6816.

[16] JADERBERG M, DALIBARD V, OSINDERO S, et al. Population based training of neural networks[EB/OL]. (2017-11-28)[2022-03-30]. https://arxiv.org/pdf/1711.09846v2.pdf.

[17] FALKNER S, KLEIN A, HUTTER F. BOHB: robust

and efficient hyperparameter optimization at scale[C]// International Conference on Machine Learning. Vancouver: ICML, 2018:1437-1446.
[18] BERGSTRA J, BARDENET R, BENGIO Y, et al. Algorithms for hyper-parameter optimization[J]. Advances in Neural Information Processing Systems, 2011,24:1-9.
[19] VANDENHENDE S,GEORGOULIS S,VAN GANSBEKE W, et al. Multi-task learning for dense prediction tasks: a survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021,44(7):3614-3633.
[20] MISRA I, SHRIVASTAVA A, GUPTA A, et al. Cross-stitch networks for multi-task learning[C]// Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 3994-4003.
[21] RUDER S, BINGEL J, AUGENSTEIN I, et al. Latent multi-task architecture learning[C]// Proceedings of the AAAI Conference on Artificial Intelligence. Honolulu: AAAI Press, 2019:4822-4829.
[22] XU D, OUYANG W, WANG X, et al. Pad-net: multi-tasks guided prediction and-distillation network for simultaneous depth estimation and scene parsing[C]// Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018:675-684.
[23] ZHANG Z, CUI Z, XU C, et al. Joint task-recursive learning for semantic segmentation and depth estimation [C]// Proceedings of the European Conference on Computer Vision. Munich: ECCV, 2018:235-251.
[24] KENDALL A, GAL Y, CIPOLLA R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics[C]// Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018:7482-7491.
[25] CHEN Z, BADRINARAYANAN V, LEE C Y, et al. Gradnorm: gradient normalization for adaptive loss balancing in deep multitask networks[C]// International Conference on Machine Learning. Stockholm: IMLS, 2018: 794-803.
[26] LIU S, JOHNS E, DAVISON A J. End-to-end multi-task learning with attention[C]// Computer Vision and Pattern Recognition. Los Angeles: IEEE, 2019:1871-1880.
[27] GUO M, HAQUE A, HUANG D A, et al. Dynamic task prioritization for multitask learning[C]// European Conference on Computer Vision. Munich: ECCV, 2018:270-287.
[28] SENER O, KOLTUN V. Multi-task learning as multi-objective optimization[EB/OL]. (2019-01-11)[2022-03-30]. https://arxiv.org/pdf/1810.04650.pdf.
[29] AHN S, HU S X, DAMIANOU A, et al. Variational information distillation for knowledge transfer[C]// Computer Vision and Pattern Recognition. Los Angeles: IEEE, 2019:9163-9171.

Automatic gradient blending for knowledge distillation

CAO Jiongxuan[*,**,***], CHANG Ming[***], ZHANG Rui[**,***], ZHI Tian[**,***], ZHANG Xishan[**,***]
([*]University of Science and Technology of China, Hefei 230026)
([**]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)
([***]Cambricon Technologies Corporation Limited, Beijing 100191)

Abstract

Since the loss function of knowledge distillation (KD) contains a task loss from the ground truth and a distillation loss from the teacher network, how to efficiently find the suitable weights of the two losses remains an unsolved issue. To overcome this issue, this paper proposes an automatic gradient blending (AGB) method to automatically and efficiently find the suitable loss weights by searching the optimal blending gradient of the two losses. We mainly consider the original design of knowledge distillation that the distillation loss is the auxiliary of the task loss. AGB efficiently searches the blending gradient by only searching the gradient direction from the search space, which is the span of the gradient directions of the two losses, meanwhile constraining the norm of blending gradient the same as the gradient norm of task loss to significantly reduce the search space. The loss weights of two losses can be automatically computed from the optimal blending gradient, avoiding the time-consuming manual tuning process. Extensive experiments on 10 different knowledge distillation methods between 13 different teacher-student combinations show the effectiveness and efficiency of AGB, which outperforms manual tuning methods over 70% combinations with a fewer computational resource.

Key words: deep neural network(DNN), knowledge distillation (KD), hyperparameter optimization(HPO), image classification