有效神经结构搜索的多项式分布学习

Zheng 1厦门大学信息学院人工智能系媒体分析与计算实验室，2鹏城实验室，深圳，中国，3北京航空航天
大学， 4华为诺亚方舟实验室， 5德克萨斯大学圣安东尼奥分校计算机科学系{zhengxiawu,langt}
@stu.xmu.edu.cn, rrji@xmu.edu.cn bczhang@buaa.edu. cn, liu.jianzhuang@huawei.com,
qitian@cs.utsa.edu

抽象的

通过神经架构搜索（NAS)获得的架构在各种计算机视觉任务中取得了极具竞争力的性能。然而，深度神经网络和搜索算法中前向后向传播的高计算需求使得 NAS 难以在实践中应用。在本文中，我们提出了一种针对极其有效的 NAS 的多项式分布学习，它将搜索空间视为联合多项式分布，即从该分布中对两个节点之间的操作进行采样，并通过以下方式获得最优网络结构在此分布中具有最可能概率的操作。因此，NAS可以转化为一个多项式分布学习问题，即对分布进行优化，使其具有较高的性能期望。此外，提出并证明了性能排名在每个训练时期都是一致的假设，以进一步加速学习过程。 CIFAR-10 和 ImageNet 上的实验证明了我们方法的有效性。在 CIFAR-10 上，我们的方法搜索的结构实现了 2.55% 的测试错误，同时比最先进的 NAS 算法快 6.0 倍（在 GTX1080Ti 上仅 4 个 GPU 小时）。在 ImageNet 上，我们的模型在 MobileNet 设置（MobileNet V1/V2）下达到了 75.2% 的 top 1 精度，同时在测量的 GPU 延迟下速度提高了 1.2 倍。带有预训练模型的测试代码可在https://github.com/tanglang96/MDENAS获得

一、简介

给定数据集，神经架构搜索 (NAS) 旨在通过搜索算法在巨大的搜索空间中发现高性能卷积架构。

NAS 在各种深度学习任务的自动化架构工程中取得了很大的成功，例如图像分类[19、34、32]、语言建模[20、33]和语义分割[18、6]。如[9]中所述，NAS 方法由三部分组成：搜索空间、搜索策略和性能估计。传统的 NAS 算法通过搜索策略对特定的卷积架构进行采样并估计性能，这可以被视为更新搜索策略的目标。尽管取得了显着进步，但传统的 NAS 方法因密集的计算和内存成本而受阻。

例如， [34]中的强化学习 (RL) 方法在4 天内跨 500 个 GPU 训练和评估了 20,000 多个神经网络。 [20]最近的工作通过以一种可区分的方式制定任务来提高可扩展性，其中搜索空间放松到连续空间，因此可以通过梯度下降对验证集的性能优化架构。然而，可微分 NAS 仍然存在 GPU 显存消耗高的问题，它随着候选搜索集的大小呈线性增长。

实际上，大多数 NAS 方法[34、18]使用标准训练和验证对每个搜索到的架构执行性能估计，通常，必须训练架构收敛以获得对验证集的最终评估，这在计算上昂贵且有限它的搜索探索。但是，如果不同架构的评估可以在几个epoch内进行排序，为什么我们需要在神经网络收敛后评估性能呢?考虑图1 中的示例，我们对具有不同层的不同架构（LeNet [17]、AlexNet [16]、 ResNet-18 [11]和 DenseNet [14])进行 domly 采样，训练和测试中的性能排名是一致的（即在不同网络和训练epoch上的性能排名是ResNet-18 > DenseNet-BC > AlexNet > LeNet）。基于这一观察，我们对性能排名提出以下假设：

*通讯作者。

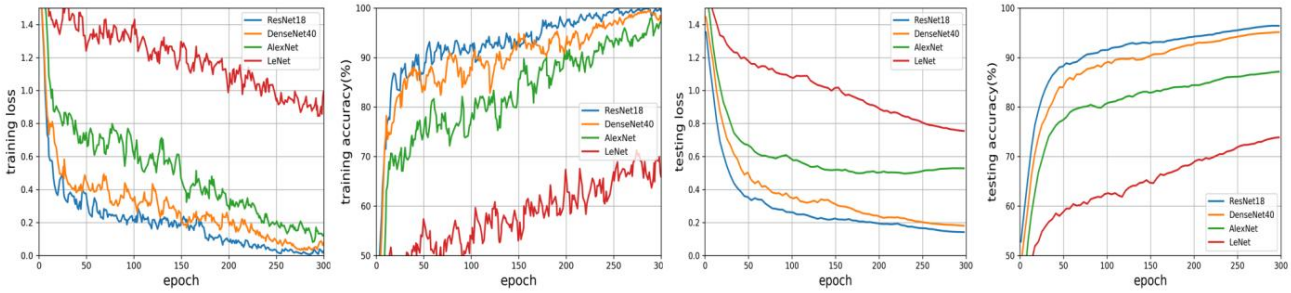

图 1. 我们随机选择广泛使用的 LeNet [17]、 AlexNet [15]、 ResNet-18[11]和 DenseNet-BC(k = 40) [14]来说明所提出的性能排名假设。训练和测试在 CIFAR-10 上进行。我们报告了训练集和测试集上的 top1 错误和损失学习曲线。正如我们在图中看到的,测试损失和准确性的排名在每个训练时期保持一致,即好的架构往往在整个训练过程中具有更好的性能。

绩效排名假设。如果 Cell A 在特定网络和训练时期具有比 Cell B 更高的验证性能,则在这些网络训练后,Cell A 在不同网络上往往优于 Cell B

收敛。

这里,一个单元是一个完全卷积的有向无环图（DAG）,它将一个输入张量映射到一个输出张量,最终的网络是通过堆叠不同数量的单元得到的,其细节在第2节中描述。 3.

该假设说明了神经架构搜索中一个简单但重要的规则。不同架构的比较可以在早期阶段完成,因为不同架构的排名就足够了,而最终的结果是不必要且耗时的。基于这个假设,我们提出了一种简单而有效的神经架构搜索解决方案,称为高效神经架构搜索的多项分布（MdeNAS）,它直接将 NAS 表示为分布学习过程。具体来说,两个节点之间的操作候选概率被初始化为相等的,这可以被认为是一个多项式分布。在学习过程中,分布的参数通过每个时期的当前性能进行更新,从而将不良操作的概率转移到更好的操作。通过这种搜索策略,MdeNAS 能够在丰富的搜索空间中快速有效地发现具有复杂图拓扑的高性能架构。

在我们的实验中,MdeNAS 设计的卷积细胞取得了很强的定量结果。搜索模型在 CIFAR-10 上以较少的参数达到 2.55% 的测试误差。在 ImageNet 上,我们的模型在 MobileNet 设置（MobileNet V1/V2 [12、26]）下实现了 75.2% 的 top 1 精度,同时在测量的 GPU 延迟下速度提高了 1.2 倍。本文的贡献总结如下：

· 我们为网络架构师介绍了一种新颖的算法

自然搜索,适用于各种大规模数据集,因为内存和计算成本与普通神经网络训练相似。

· 我们提出了一个性能排名假设,可以将其合并到现有的NAS 算法中以加速其搜索。

· 所提出的方法实现了显着的搜索效率,例如,使用 1 GTX1080Ti（与最先进的算法相比快 6.0 倍)在 4 小时内在 CIFAR-10 上的测试错误率为 2.55%,这归因于使用我们的分布学习完全不同于基于 RL 的[2, 34]方法和可微分方法[20, 29]。

二、相关工作

正如[33、34] 中首次提出的那样,预定义架构空间中的自动神经网络搜索在过去几年中受到了极大的关注。为此,已经提出了许多搜索算法来使用特定的搜索策略找到最佳架构。由于大多数手工制作的 CNN 是通过堆叠缩减（即输入的空间维度减少)和范数（即输入的空间维度被保留）单元[14、11、13] 构建的， [33， [34]建议在相同设置下搜索网络以减少搜索空间。 [33、34、2]中的作品使用强化学习作为元控制器来探索架构搜索空间。 [33, 34]中的工作采用递归神经网络 (RNN) 作为策略,对编码特定神经架构的字符串进行顺序采样。可以使用策略梯度算法或近端策略优化来训练策略网络。 [3,4,19]中的工作将架构搜索空间视为网络转换的树结构,即网络由更远的网络生成一些预定义的操作,这减少了搜索空间和

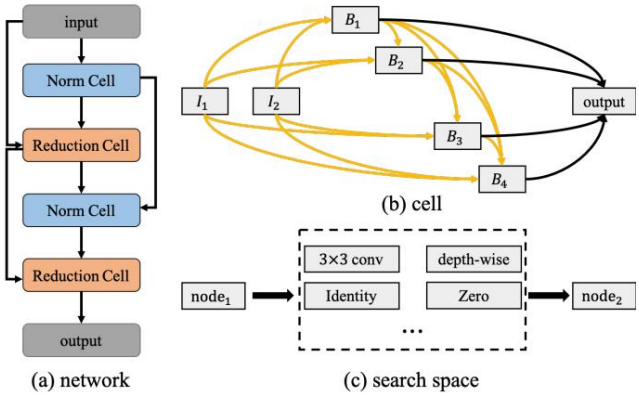

图 2. 搜索不同规模的网络。 (a) 网络由堆叠的单元组成,每个单元将两个先前单元的输出作为输入。 (b) 一个单元包含 7 个节点,两个输入节点I1和I2,四个对输入节点和上层节点应用采样操作的中间节点B1、 B2、 B3、 B4 ,以及一个连接四个节点输出的输出节点中间节点。 (c) 两个节点之间的边表示根据搜索空间中的多项式分布的可能操作。

加快搜索速度。基于 RL 的方法的替代方法是进化方法,它通过进化算法优化神经架构[28、24]。

然而,上述架构搜索算法仍然是计算密集型的。因此,最近提出了一些通过一次性设置来加速 NAS 的工作,其中网络通过超表示图进行采样,并且可以通过参数共享来加速搜索过程[23]。例如,DARTS [20]通过连续松弛联合优化超图中两个节点内的权重。因此,可以通过标准梯度下降来更新参数。然而, one-shot 方法存在 GPU 内存消耗大的问题。为了解决这个问题, ProxylessNAS [5]通过路径二值化[7]在没有特定代理的情况下探索搜索空间。然而,由于 ProxylessNAS 的搜索过程仍然在一次性方法的框架内,它可能具有相同的复杂性,即,在 ProxylessNAS 中获得的好处是探索和利用之间的权衡。也就是说,在搜索过程中需要更多的历元。此外， [5]中的搜索算法与之前的工作类似,无论是差分方法还是基于 RL 的方法[20、34]。

与之前的方法不同,我们将路径/操作选择编码为分布采样,并通过分布学习实现控制器/代理的优化。我们的学习过程进一步整合了所提出的假设,以估计每个操作/路径的价值,从而实现极其高效的 NAS 搜索。

3.架构搜索空间

在本节中,我们描述了架构搜索空间和构建网络的方法。我们遵循与之前 NAS 作品[20、19、34]相同的设置以保持一致性。如图2 所示,网络以不同的尺度定义:网络、小区和节点。

3.1.节点

节点是组成细胞的基本元素。 i每个节点 x
是一个特定的张量（例如,卷积神经网络中的特征图)并且从操作中采样的每个有向边（i,j） （i,j)表示搜索空间的操作以变换图2中所点:输入节点xI 、中间节点xB和输出节点xO 。每个ce单元格的输出张量作为输入节点,通过apply (i,j)到之前的节点(xI and ing sampled operations作为oj ∈ [1, i))生成中间节点x。所有中间节点的串联被视为最终输出节点。

乙

x j 乙,

以下[20]组可能的操作,表示为 O,由以下 8 个操作组成： (1) 3 × 3 max pooling。 (2) 无连接（零）。 (3) 3×3 平均池化。 (4) 跳过连接（身份）。 (5) 速率为 2 的 3 × 3 扩张卷积。 (6) 速率为 2 的 5 × 5 扩张卷积。 (7) 3×3 深度可分离卷积。 (8) 5×5 深度可分离卷积。

我们只是在具有多个操作（边)的节点的输入处使用逐元素加法。例如,在图2 （b)中， B2有三个操作,其结果按元素相加,然后被视为B2。

3.2.细胞

一个cell被定义为一个微小的卷积网络映射有两种类型的cells,norm一个 H × W × F 张量到另一个 H × W × F . cell和reduction cell。范数单元使用步幅为 1 的操作,因此 H = H 和 W = W。缩减单元使用步幅为 2 的操作,因此 H = H/2 和 W = W/2。对于过滤器的数量 F

和F ，在大多数人类设计的卷积神经网络[11、14、16、27、10、31]中,一个常见的启发式方法是在空间特征图减半时将 F 加倍。因此,步幅 1 的 F = F,步幅 2 的 F = 2F。

如图2(b) 所示,单元由具有 7 个节点的 DAG 表示（两个输入节点I1和I2,四个中间节点B1、 B2、 B3、 B4 ,它们在输入节点和上层节点上应用采样操作,以及连接中间节点的输出节点）。两个节点之间的边表示根据搜索空间中的多项式分布p(node1,node2)的可能操作。在训练中,当一个中间节点有多个边时,其输入是通过逐元素相加得到的（操作

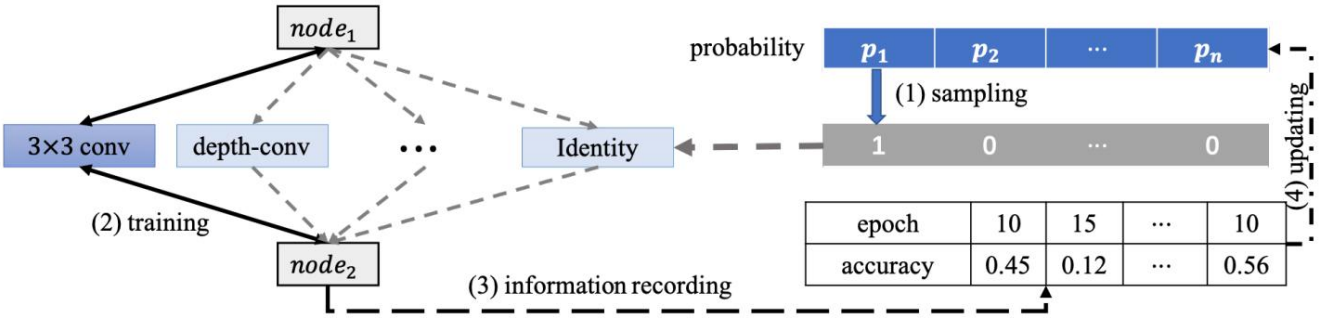

图 3. 总体搜索算法：（1）根据相应的参数为 θ 的多项式分布在搜索空间中采样一个操作。（2）用一次前向和反向传播训练生成的网络。（3）在验证集上测试网络并记录反馈（epoch 和accuracy）。（4）根据提出的分布学习算法更新分布参数。右表中操作1的epoch数为10，表示该操作在所有epoch中被选中了10次。

化）。在测试中，我们选择前 K 个概率来生成最终的单元格。因此，整个搜索空间的大小为 2 × 8 |EN |其中EN是具有 N 个中间节点的可能边的集合。在我们的例子中，N = 4，细胞结构的总数是 2 × 8 2+3+4+5

，

$$14 = 2 \times 8,$$

这是一个非常大的搜索空间，因此需要有效的优化方法。

3.3.网络

如图2(a) 所示，网络由预定数量的堆叠单元组成，这些单元可以是标准单元或缩减单元，每个单元都将两个先前单元的输出作为输入。在网络的顶部，全局平均池化后跟一个 softmax 层用于最终输出。

基于 Performance Ranking Hypothesis，我们在相关数据集上训练一个小型（例如 6 层）堆叠模型来搜索范数和缩减单元，然后生成更深的网络（例如 20 层）进行评估。整个 CNN 构建过程和搜索空间与[20] 相同。但请注意，我们的搜索算法不同。

4. 方法论

在本节中，介绍了我们的 NAS 方法。我们首先描述如何对第 1 节中提到的网络进行采样。 3减少训练过程中的 GPU 内存消耗。然后，我们提出了多项式分布学习，以使用提出的假设有效地优化分布参数。

4.1.采样

正如在第二节中提到的。 3.1，网络结构的多样性是由每两个节点对 M 条可能路径（在这项工作中，M = 8）的不同选择产生的。这里我们在开始时将这些路径的概率初始化为pi =

宁探索。在采样阶段，我们遵循

在[5]中工作并用二元门{gi}转换 M 个实值概率{pi }：

$$g = \begin{matrix} [1, 0, ..., 0] \text{ 概率为p1} \\ * \\ \cdots \\ [0, 0, ..., 1] \text{ 概率为pM} \\ * \end{matrix} \quad (1)$$

节点 i 和 j 之间的最终操作通过以下方式获得：

$$o^{(i,j)} = o^{(i,j)}_{g} = \begin{matrix} \text{o1概率为p1} \\ \cdots \\ \text{oM概率为pM。} \end{matrix} \quad (2)$$

如前面的等式所示，我们在运行时仅对一个操作进行采样，与[20]相比，这有效地降低了内存成本。

4.2.多项分布学习

以前的 NAS 方法很耗时和内存。强化学习的使用进一步禁止了网络训练中带有延迟奖励的方法，即通常在网络训练收敛后才对结构进行评估。另一方面，如第二节所述。 1、根据Performance Ranking Hypothesis，我们可以在训练网络时对一个cell进行评价。如图3 所示，记录了搜索空间中每个操作的训练时期和准确性。如果操作 A 具有更少的训练时期和更高的准确性，则操作 A 优于 B。

形式上，对于两个节点之间的特定边，我们将操作概率定义为 p，将训练时期定义为He ，将准确度定义为Ha ，其中每一个都是长度为 M = 8 的实值列向量。为了清楚地说明我们的学习，

ing方法,我们进一步定义epoch的微分为:

$$\Delta 他 = \begin{matrix} (1 \times 他_{.} - 他)^{吨} \\ \dots \\ (1 \times 他_{米} - 他)^{吨} \end{matrix}, \quad (3)$$

精度差为:

$$\Delta 哈 = \begin{matrix} (1 \times 公顷 - 哈)^{吨} \\ \dots \\ (1 \times 公顷 - 哈)^{吨} \end{matrix}, \quad (4)$$

其中 1 为长度为 8 且其元素均为 1 的列向量,ΔHe和ΔHa为 8×8 矩阵,其中 ΔHe i,j = 他 − He ΔHa ij , i,j =哈我 −哈 j。 经过一个epoch的训练,相应的变量He由评价结果计算得出。, Ha ,ΔHe和ΔHa是多项分布的参数可以通过以下方式更新:

$$pi \leftarrow pi + \alpha * \left(\sum_{j} (\Delta 他_{i,j} < 0, \Delta Ha_{i,j} > 0) - \sum_{j} (\Delta 他_{i,j} > 0, \Delta 哈_{i,j} < 0) \right), \quad (5)$$

其中 α 是超参数,表示条件为真时等于 1 的指示函数。

正如我们在方程式中看到的那样。 5、特定的概率 < 0) 操作 i 用更少的 epochs (ΔHe i ,i和更高的性能 (ΔHa i ,j) > 0).同时,> 0)

< 0).由于等式。 5是 ap 在每个训练 epoch 之后重复,搜索空间中的概率可以在几个 epoch 后有效地收敛和稳定。连同提出的性能排名假设(在第5 节中证明),我们的 NAS 多项式分布学习算法非常有效,并且在相同设置下与其他最先进的方法相比实现了更好的性能。考虑到性能排名根据假设由不同的层组成,为了进一步提高搜索效率,我们将[20]中的搜索网络替换为另一个较浅的网络(仅 6 层),仅需要 4 GPU 小时的搜索CIFAR-10。

为了生成最终网络,我们首先选择所有边中概率最高的操作。对于具有多输入的节点,我们采用具有前 K 个概率的逐元素加法。最终网络由预定数量的堆叠单元组成,使用规范单元或缩减单元。我们的多项式分布学习算法在算法中给出。 1.

5.实验

在本节中,我们首先在 CIFAR-10 上进行一些实验来证明所提出的假设。然后,

算法 1:多项式分布学习

输入:训练数据: Dt;验证数据: Dv; CNN 模型:F。输出:细胞操作概率:P。对于t= 1,...,T纪元

根据公式1 对操作进行采样;
用 1 个 epoch 训练网络;
在Dv上验证网络;
根据式3和式4计算epoch和accuracy的微分;

用公式5更新概率; 8结束

我们将我们的方法与最先进的方法在两个广泛使用的分类数据集(包括 CIFAR-10 和 ImageNet)上的搜索有效性和效率方面进行了比较。

5.1.实验设置

5.1.1 数据集

我们在他们的实验数据集和评估指标中遵循大多数 NAS 作品[20、4、34、19] 。特别是,我们在 CIFAR-10 [15]上进行了大部分实验,其中有 50, 000 个训练图像和 10, 000 个测试图像。在架构搜索中,我们随机选择训练集中的 5, 000 张图像作为验证集来评估架构。

彩色图像大小为 32 × 32,有 10 个类别。图像的所有颜色强度都归一化为 [−1, +1]。
为了进一步评估泛化,在 CIFAR-10 上发现一个好的单元后,将架构转移到更深的网络中,因此我们也在 ILSVRC 2012 ImageNet [25] 上进行分类。该数据集由 1, 000 个类别组成,其中有 128 万张训练图像和 50, 000 张验证图像。这里我们考虑移动设置,其中输入图像大小为 224 × 224,并且模型中的乘加运算次数限制在 600M 以下。

5.1.2 实施细节

在搜索过程中,根据假设,层数与细胞结构的评估无关。

因此,我们在网络中总共考虑 L = 6 个单元,其中在第二层和第三层中插入缩减单元,一个单元有 4 个节点。该网络训练了 100 个 epoches,批量大小为 512(由于网络工作较浅且操作采样较少),初始通道数为 16。我们使用带动量的 SGD 来优化网络权重 w,其中初始学习率为 0.025(按照余弦计划退火至零),动量为 0.9,权重衰减为 3 × 10−4 。

多项

式参数的学习率设置为 0.01。

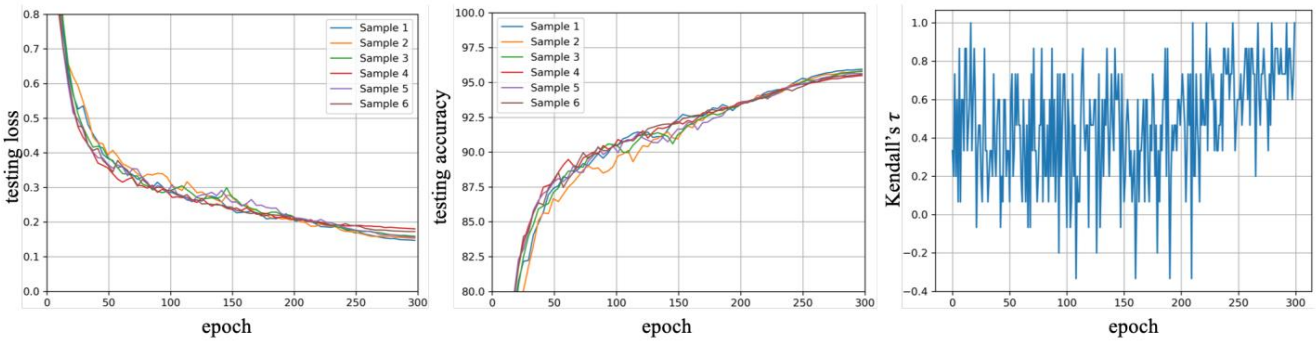

图 4. 不同架构的测试误差（左）、top 1 准确率（中)和 Kendall 的 τ（右）。误差和准确度曲线是纠缠在一起的,因为它们是从第3 节中定义的相同搜索空间中采样的。因此,我们进一步计算每个时期和最终结果之间的 Kendall s τ。请注意,Kendall 的 τ > 0 可以认为是一个很高的值,这意味着超过一半的排名是一致的。

在 CIFAR-10 上仅使用一个 NVIDIA GTX 1080Ti 进行搜索仅需 4 个 GPU 小时。

在架构评估步骤中,实验集类似于[20,34,23]。一个由 20 个单元组成的大型网络训练了 600 个时期,批量大小为 96,并进行了额外的正则化处理,例如 cutout [8] 和概率为 0.3 [20] 的路径丢失。我们实现的所有实验和模型都在 PyTorch [22] 中。

在 ImageNet 上,我们保持与 CIFAR-10 上相同的搜索超参数。在训练过程中,我们遵循具有相同实验设置的先前 NAS 方法[20、34、23] 。该网络训练了 250 个 epoch,批量大小为 512,权重衰减为 3×10−5 ,初始 SGD 学习率为 0.1（每个 epoch 衰减 0.97 倍）。

,

5.1.3 基线

我们将我们的方法与人工设计的网络和其他 NAS 网络进行比较。手动设计的网络包括 ResNet [11]、 DenseNet [14]和 SENet [13]。对于 NAS 网络，我们根据不同的搜索方法对它们进行分类,例如 RL（NASNet [34]、 ENAS [23]和 Path-level NAS [4]）、进化算法（AmoebaNet [24]）、基于序列模型的优化（ SMBO）（PNAS [19])和基于梯度的（DARTS [20]）。

我们进一步在 ImageNet 上的移动设置下比较我们的方法以证明泛化。我们的算法在 CIFAR-10 上生成的最佳架构被传输到 ImageNet,它遵循与上述作品相同的实验设置。由于我们的算法占用更少的时间和内存,我们也直接在 ImageNet 上搜索,并将其与另一个类似的无代理 NAS 基线（低计算消耗） [5] 进行比较。

5.2.假设的评估

我们首先进行实验来验证所提出的性能排名假设的正确性。为了对假设有一些直观的认识,我们引入了 Kendall 等级相关系数,又名 Kendall s τ [1]。给定 m 个项目的两个不同等级,Kendall 的 τ 计算如下：

$$\tau = \frac{P - QP}{+ Q \text{ 其}}, \tag{6}$$

中 P 是一致的对数（在两个排名中顺序相同）,Q 表示不一致的对数（顺序相反）。 τ ∈ [−1, 1],其中 1 表示排名相同，-1 表示排名相反。两个等级中的一对一致的概率是pτ = 2 。因此,τ = 0 意味着 50% 的对是一致的。

τ+1

我们在搜索空间中随机抽取不同的网络架构,并报告测试集上不同时期的损失、准确性和 Kendall s τ。将每个时期的性能排名与不同网络架构的最终性能排名进行比较。如图4 所示,由于采样网络的同质性,即所有网络都是从同一空间生成的,因此精度和损失几乎没有区别。另一方面,Kendall 系数在大多数 epoch 中保持较高的值（τ > 0， pτ > 0.5）,随着 epoch 数量的增加通常接近 1。说明架构评价排名在每个epoch都具有很强的概率能力,一般会越来越接近最终排名。请注意,每个时期的 Kendall τ 的平均值为 0.474。因此,假设成立的概率为 0.74。此外,我们发现假设与多项式分布学习的结合可以相互增强。该假设保证了选择好的架构时的高期望,而分布学习降低了概率

建筑学	测试误差 (%) 3.53	参数 (M) 11.1	搜索成本（GPU 天数）	搜索方法手册
ResNet-18 [11]	4.77	1.0	-	手册手册
密集网[14]	4.05	11.2	-	
SENet [13]	2.65	3.3 3.2	-	
NASNet-A [34]	3.34	2.8	1800	RL
变形虫网-A [24]	2.55	3.2	3150	进化进化
变形虫网络-B [24]	3.41	4.6	3150	
美国国家科学院院刊[19]	2.89	5.7	225	SMBO
易纳斯[23]	2.49	3.1	0.5	RL
路径级NAS [4]	2.94	3.4	8.3	RL
飞镖（一阶） [20]	2.83	3.1	1.5	gradient-based
飞镖（二阶） [20]	3.49 2.55	3.61	4	基于梯度
随机样本[20]			-	-
MdeNAS（我们的）			0.16	MDL

表 1. 我们发现的架构、人工设计的网络和其他 NAS 架构在 CIFAR-10 上的测试错误率。公平起见，我们选择具有相似参数（< 10M）和训练条件（相同时期和正则化）的架构和结果。

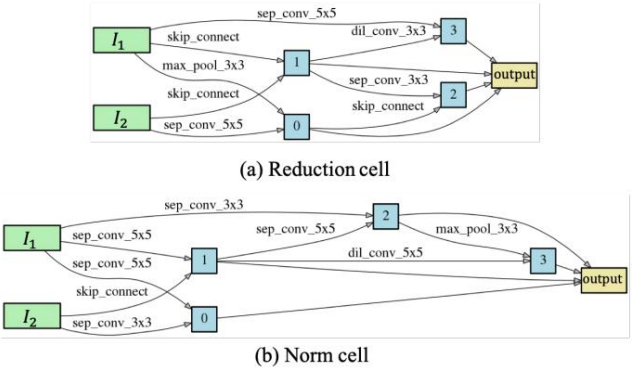

图 5. 在 CIFAR 10 上发现的最佳单元格的详细结构。边缘操作的定义在第3.1 节中。
在归约单元（向上）中，对 2 个输入节点的操作步幅为 2，在范数单元（向下）中，步幅为 1。

采样不良架构的能力。

5.3. CIFAR-10 的结果

我们首先使用所提出的方法找到最佳单元架构。特别是，我们首先在过度参数化的网络上搜索神经架构，然后我们用更深的网络评估最佳架构。为了消除随机因素，该算法运行了多次。我们发现架构性能在不同时间仅略有不同，与更深层网络中的最终性能相比（<0.2），这表明所提出方法的稳定性。最佳架构如图 5 所示。

CIFAR-10 上卷积架构的总结结果在表中给出。 1. 不值得所提出的方法优于最先进的方法[34, 20]，同时计算消耗极少（ [34] 中仅 0.16 GPU 天 << 1,800）。由于性能高度依赖于不同的正则化方法（例如，cutout [8]）和层，因此选择网络架构以在相同的设置下进行同等比较。此外，其他作品使用基于差异的或黑盒优化来搜索网络。我们将优异的结果归因于我们解决分布学习问题的新方法，以及快速学习过程：当分布收敛时，可以直接从分布中获得网络架构。相反，以前的方法[34]仅在训练过程完成时才评估架构，这是非常低效的。在 Tab 中观察到的另一个值得注意的现象。 1是，即使在搜索空间中随机抽样， [20]中的测试错误率也仅为 3.49%，这与相同搜索空间中的先前方法相当。因此，我们可以合理地得出结论，先前方法的高性能部分归功于良好的搜索空间。同时，所提出的方法可以快速探索搜索空间并生成更好的架构。我们还在 Tab 中报告了手工制作的网络的结果。 1. 显然，我们的方法显示出显着的增强，这表明它在资源消耗和测试准确性方面都具有优势。

5.4. ImageNet 上的结果

我们还在 ImageNet 数据集[25] 上运行我们的算法。在现有工作的基础上，我们对不同的搜索数据集进行了两次实验，并在同一数据集上进行了测试。如选项卡中所报告的。 1，以前的工作在CIFAR-10上很耗时，搜索起来不切实际

建筑学	准确性 （%）	参数 (M)	搜索成本	搜索
	Top1 Top5 70.6	6.6 3.4	（GPU 天数）	方法手册
MobileNetV1 [12]	89.5 72.0 91.0 90.9	5	-	手册手册
MobileNetV2 [26]	73.7 74.0 74.5 74.7	5	-	手册
ShuffleNetV1 2x (V1) [30]	73.1 74.5	5.3	-	
ShuffleNetV2 2x (V2) [21]	-	5.1	-	
NASNet-A [34]	91.6	6.4	1800	RL
变形虫网-A [24]	92.0	5.1	3150	进化进化
变形虫网-C [24]	92.4	4.9 6.1	3150	
美国国家科学院院刊[19]	91.9		225	基于
飞镖[20]	91.0		4	SMBO 梯度
MdeNAS（我们的）	92.1		0.16	MDL

表 2. 与移动设备上 ImageNet 上最先进的图像分类方法的比较。所有的NAS网络都在CIFAR-10上搜索，然后直接转移到ImageNet。

模型	前一	搜索时间 GPU 天数	GPU延迟
MobileNetV2	72.0	-	6.1毫秒
ShuffleNetV2	72.6	-	7.3毫秒
无代理 (GPU) [5] 74.8 无代理		4	5.1 毫秒
(CPU) [5] 74.1 MdeNAS (GPU)		4	7.4 毫秒
75.2 MdeNAS (CPU) 74.1		2	4.9 毫秒
		2	7.1 毫秒

表 3. 与移动设备上 ImageNet 上最先进的图像分类的比较。使用 MobileNetV2 [26]主干网直接在 ImageNet 上搜索网络。

图片网。因此，我们首先考虑在 ImageNet 上进行可迁移实验，即将在 CIFAR-10 上找到的最佳架构直接迁移到 ImageNet，使用两个步幅为 2 的初始卷积层，然后堆叠 14 个尺度缩小的单元（reduction cells)为 1， 2、6和10。触发器的总数由选择的初始通道数决定。我们按照现有的 NAS 工作来比较移动设置下的性能，其中输入图像大小为 224 × 224，模型被限制为小于 600M FLOPS。我们按照[20, 34]设置其他超参数，如第 1 节所述。 5.1.2.表中的结果。图2显示 CIFAR 10 上的最佳单元架构可转移到 ImageNet。请注意，所提出的方法实现了与最先进方法相当的精度，同时使用更少的计算资源。

极少的时间和 GPU 内存消耗使我们的算法在 ImageNet 上可行。

因此，我们进一步在 ImageNet 上进行了搜索实验。我们按照[5]设计网络设置和搜索空间。特别是，我们允许一组具有各种内核 {3, 5, 7} 和扩展比率 {1, 3, 6} 的移动卷积层。为了进一步加速搜索，我们直接使用具有 CPU 和 GPU 结构的网络

在[5]中获得。这样，搜索空间中的零层和恒等层就被舍弃了，我们只搜索与卷积层相关的超参数。结果报告于表中。 3，我们发现与人工设计和自动架构搜索方法相比，我们的 MdeNAS 实现了卓越的性能，并且计算消耗更少。

六，结论

在本文中，我们介绍了 MdeNAS，这是第一个基于分布学习的卷积网络架构搜索算法。我们的算法是基于一种新的性能等级假设部署的，该假设能够进一步减少比较早期训练过程中架构性能的搜索时间。受益于我们的假设，MdeNAS 可以大大降低计算消耗，同时在 CIFAR-10 和 ImageNet 上实现出色的模型精度。此外，Mde NAS 可以直接在 ImageNet 上搜索，这优于人工设计的网络和其他 NAS 方法。

致谢。本工作得到国家重点研发计划（No.2017YFC0113000，No.2016YFB1001503），国家自然科学基金（No.U1705262,No.61772443，No.61572410），博士后创新人才支持计划BX201600094资助，中国博士后科学基金面上项目 2017M612134，国家语委科研项目（YB135-49），福建省自然科学基金（No. 2017J01125 and No. 2018J01106）。

参考

[1] 赫维·阿卜迪。肯德尔等级相关系数。 恩
测量与统计百科全书。 Sage，加利福尼亚州千橡市，第 508–510 页，2007 年。6

[2] Bowen Baker、Otkrist Gupta、Nikhil Naik 和 Ramesh Raskar。使用强化学习设计神经网络架构。 arXiv 预印本 arXiv:1611.02167, 2016. 2

[3] 韩才, 陈天耀, 张维南, 于勇, 王军.通过网络转换进行高效的架构搜索。 In Thirty-Second AAAI Conference on Artificial Intelligence, 2018. 2 [4] 蔡涵,杨佳成,张伟南,韩松,余勇。用于高效架构搜索的路径级网络转换。 arXiv 预印本 arXiv:1806.02639, 2018. 2, 5, 6, 7

[5] 蔡寒, 朱立耕, 韩松. Proxylessnas:在目标任务和硬件上直接进行神经架构搜索。 arXiv preprint arXiv:1812.00332, 2018. 3, 4, 6, 8 [6] Liang-Chieh Chen、Maxwell Collins、Yukun Zhu、George Papandreou、Barret Zoph、Florian Schroff、Hartwig Adam 和 Jon Shlens。为密集图像预测寻找有效的多尺度架构。在神经信息处理系统的进展中,第 8713–8724 页,2018年。1 [7] Matthieu Courbariaux、Yoshua Bengio 和 Jean-Pierre David。 Binaryconnect:在传播过程中使用二进制权重训练深度神经网络。 In Advances in neural information processing systems, pages 3123–3131, 2015. 3 [8] Terrance DeVries 和 Graham W Taylor。改进了带切口的卷积神经网络的正则化。 arXiv 预印本 arXiv:1708.04552, 2017. 6, 7 [9] Thomas Elsken、Jan Hendrik Metzen 和 Frank Hutter。

神经架构搜索:一项调查。 arXiv 预印本 arXiv:1808.05377, 2018. 1 [10] 范登平、王文冠、程明明和沉建兵。将更多注意力转移到视频显着目标检测上。 In the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8554– 8564, 2019. 3 [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun.

用于图像识别的深度残差学习。 In Proceedings of the IEEE conference on Computer Vision and pattern recognition, pages 770–778, 2016. 1, 2, 3, 6, 7 [12] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand、Marco An dreetto 和 Hartwig Adam。 Mobilenets:用于移动视觉应用的高效卷积神经网络。 arXiv 预印本 arXiv:1704.04861, 2017. 2, 8

[13] 胡杰,李申,孙刚.挤压和激发网络工作。在 IEEE 计算机视觉和模式识别会议记录中,第 7132–7141页， 2018.2、6、7

[14] 高煌、刘庄、劳伦斯·范德马腾和基里安·Q·温伯格。密集连接的卷积网络。在 IEEE 计算机视觉和模式识别会议记录中,第 4700–4708页， 2017年。1、2、3、6、7

[15] Alex Krizhevsky 和 Geoff Hinton。 cifar-10 上的卷积深度信念网络。未发表手稿, 40(7), 2010. 2, 5

[16] Alex Krizhevsky、Ilya Sutskever 和 Geoffrey E Hinton。使用深度卷积神经网络进行 Imagenet 分类。 In Advances in neural information processing systems, pages 1097–1105, 2012. 1, 3 [17] Yann LeCun, Leon Bottou, Yoshua Bengio, Patrick Haffner, et al.基于梯度的学习应用于文档识别。 IEEE 会刊, 86(11):2278–2324, 1998. 1, 2

[18] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan Yuille, and Li Fei-Fei. Auto-deeplab:用于语义图像分割的分层神经结构搜索。 arXiv 预印本 arXiv:1901.02985, 2019.1 [19] Chenxi Liu, Barret Zoph, Maxim Neumann, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan Yuille, Jonathan Huang, and Kevin Murphy。渐进式神经架构搜索。 In the Proceedings of the European Conference on Computer Vision, pages 19–34, 2018. 1, 2, 3, 5, 6, 7, 8 [20] Hanxiao Liu、Karen Simonyan 和 Yiming Yang。

Darts:可区分的架构搜索。 arXiv preprint arXiv:1806.09055, 2018. 1, 2, 3, 4, 5, 6, 7, 8 [21] 马宁宁,张翔宇,郑海涛,孙健。

Shufflenet v2:高效 cnn 架构设计实用指南。在欧洲计算机视觉会议记录中,第 116–131 页,2018年。8 [22] Adam Paszke、Sam Gross、Soumith Chintala、Gregory Chanan、Edward Yang、Zachary DeVito、Zeming Lin、Al ban Desmaison、Luca Antiga 和亚当·莱尔。 pytorch 中的自动微分。 2017. 6

[23] Hieu Pham、Melody Y Guan、Barret Zoph、Quoc V Le 和 Jeff Dean。通过参数共享进行高效的神经结构搜索。 arXiv 预印本 arXiv:1802.03268, 2018. 3, 6, 7 [24] Esteban Real、Alok Aggarwal、Yanping Huang 和 Quoc V Le。图像分类器架构搜索的正则化演化。 arXiv 预印本 arXiv:1802.01548, 2018.3 , 6, 7, 8 [25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, San jeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et 阿尔。 Imagenet 大规模视觉识别挑战。国际计算机视觉杂志, 115(3):211–252, 2015. 5, 7 [26] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zh moginov, and Liang-Chieh Chen. Mobilenetv2:倒置残差和线性瓶颈。在 IEEE 计算机视觉和模式识别会议记录中,第 4510–4520 页,2018年。2、8

[27] 卡伦·西蒙尼安和安德鲁·齐瑟曼。用于大规模图像识别的非常深的卷积网络。 arXiv 预印本 arXiv:1409.1556, 2014. 3 [28] Lingxi Xie 和 Alan Yuille。遗传CNN。 In the Proceedings of the IEEE International Conference on Computer Vision, pages 1379–1388, 2017. 3 [29] Sirui Xie, Hehui Zheng, Chunxiao Liu, and Liang Lin.

Snas:随机神经结构搜索。 arXiv preprint arXiv:1812.09926, 2018. 2 [30] 张翔宇,周新宇,林梦晓,孙健。

Shufflenet:一种用于移动设备的极其高效的卷积神经网络。在 IEEE 计算机视觉和模式识别会议记录中,第 6848–6856 页,2018年。8

[31] 赵嘉兴、曹阳、范登平、程明明、李宣义和张乐。对比用于 rgbd 显着对象检测的先验和流体金字塔集成。 In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 3 [32] Xiawu Zheng, Rongrong Ji, Lang Tang, Yan Wan, Baochang Zhang, Yongjian Wu, Yunsheng Wu, and Ling Shao.用于高效网络架构搜索的动态分布剪枝。 arXiv 预印本 arXiv:1905.13543, 2019.1

[33] Barret Zoph 和 Quoc V Le。具有强化学习的神经结构搜索。 arXiv 预印本 arXiv:1611.01578, 2016. 1, 2

[34] Barret Zoph、Vijay Vasudevan、Jonathon Shlens 和 Quoc V Le。学习用于可扩展图像识别的可迁移架构。在 IEEE 计算机视觉和模式识别会议记录中,第8697–8710页， 2018。1、2、3、5、6、7、8