

深度神经架构搜索综述

孟子尧^{1,2} 谷 雪^{1,2} 梁艳春^{1,2} 许 东³ 吴春国¹

¹(符号计算与知识工程教育部重点实验室(吉林大学) 长春 130012)

²(符号计算与知识工程教育部重点实验室珠海分实验室(吉林大学珠海学院) 广东珠海 519041)

³(密苏里大学哥伦比亚分校电子工程与计算机科学系 美国密苏里州哥伦比亚 MO65211)

(zy-meng@outlook.com)

Deep Neural Architecture Search: A Survey

Meng Ziyao^{1,2}, Gu Xue^{1,2}, Liang Yanchun^{1,2}, Xu Dong³, and Wu Chunguo¹

¹(Key Laboratory of Symbol Computation and Knowledge Engineering (Jilin University), Ministry of Education, Changchun 130012)

²(Zhuhai Laboratory of Key Laboratory of Symbol Computation and Knowledge Engineering (Zhuhai College of Jilin University), Ministry of Education, Zhuhai, Guangdong 519041)

³(Department of Electrical Engineering and Computer Science, University of Missouri-Columbia, Columbia, Missouri, USA MO65211)

Abstract Deep learning has achieved excellent results on data tasks with multiple modalities such as images, speech, and text. However, designing networks manually for specific tasks is time-consuming and requires a certain level of expertise and design experience from the designer. In the face of today's increasingly complex network architectures, relying on manual design alone increasingly becomes complex. For this reason, automatic architecture search of neural networks with the help of algorithms has become a hot research topic. The approach of neural architecture search involves three aspects: search space, search strategy, and performance evaluation strategy. The search strategy samples a network architecture in the search space, evaluates the network architecture by a performance evaluation strategy, and feed-back the results to the search strategy to guide it to select a better network architecture, and obtains the optimal network architecture through continuous iterations. In order to better sort out the methods of neural architecture search, we summarize the common methods in recent years from search space, search strategy and performance evaluation strategy, and analyze their strengths and weaknesses.

Key words deep learning; neural architecture search (NAS); search space; search strategy; performance evaluation

收稿日期:2019-12-11;修回日期:2020-04-17

基金项目:国家自然科学基金项目(61972174,61876069,61876207);吉林省重点研发项目(20180201045GX,20180201067GX);吉林省自然科学基金项目(20200201163JC);广东省科技计划项目(2020A0505100018);广东省应用基础研究重点项目(2018KZDXM076);广东省优势重点学科项目(2016GDYSZDXK036)

This work was supported by the National Natural Science Foundation of China (61972174, 61876069, 61876207), the Key Research and Development Project of Jilin Province (20180201045GX, 20180201067GX), the Natural Science Foundation of Jilin Province(20200201163JC), the Science and Technology Planning Project of Guangdong Province (2020A0505100018), the Guangdong Key-Project for Applied Fundamental Research (2018KZDXM076), and the Guangdong Premier Key-Discipline Enhancement Scheme (2016GDYSZDXK036).

通信作者:吴春国(wucg@jlu.edu.cn)

摘要 深度学习在图像、语音、文本等多种模态的数据任务上取得了优异的效果.然而,针对特定任务,人工设计网络需要花费大量的时间,并且需要设计者具有一定水平的专业知识和设计经验.面对如今日趋复杂的网络架构,仅依靠人工进行设计变得越来越复杂.基于此,借助算法自动地对神经网络进行架构的搜索成为了研究热点.神经架构搜索的方法涉及 3 个方面:搜索空间、搜索策略、性能评估策略.通过搜索策略在搜索空间中选择一个网络架构,借助性能评估策略对该网络架构进行评估,并将结果反馈给搜索策略指导搜索策略选择更好的网络架构,通过不断迭代得到最优的网络架构.为了更好地为读者提供一个快速了解神经网络架构搜索方法的导航地图,从搜索空间、搜索策略和性能评估策略 3 个方面对现有典型的神经架构搜索方法进行了梳理,总结讨论了近年来常见的架构搜索方法,并分析了各种方法的优势和不足.

关键词 深度学习;神经架构搜索;搜索空间;搜索策略;性能评估

中图法分类号 TP183

人工神经网络通过模拟生物神经网络信号处理的过程,解决了很多具有挑战性的任务.以图像分类为例,LeCun 等人^[1]于 1998 年设计了第 1 个用于手写字符识别的卷积神经网络 LeNet5,该网络仅有 5 层.直到 21 世纪 10 年代,GPU 代替 CPU 成为了新的神经网络加速训练工具,这使得训练更大规模的网络成为了可能.Krizhevsky 等人^[2]于 2012 年设计了有一定深度的网络 AlexNet,并取得了 ILSVRC-12(ImageNet large-scale visual recognition challenge-2012)竞赛的成功.相比于 AlexNet,Szegedy 等人^[3]于 2015 年再次加深了网络的深度和宽度,设计了一个超过 20 层的神经网络架构 GoogLeNet. He 等人^[5]于 2016 年通过引入跳跃连接的概念,设计了一个 152 层的残差网络 ResNet,取得了 ILSVRC-15 分类任务的第 1 名.随着研究的不断深入,网络的层数不断加深,截至 2016 年,最深的神经网络已经超越 1000 层^[5].为了设计性能更加优异的网络,人工设计深层的网络需要进行大量重复的实验,随着网络层数的增加,需要人工调整的超参数就越多,这消耗了大量的人力和计算资源.因此,对于神经网络架构自动搜索的研究显得尤为关键.

神经架构搜索(neural architecture search, NAS)方法近年来在图像分类^[6-8]、目标检测^[9]等任务上取得了很好的效果.但这些方法都需要评估大量的网络架构,需要的计算资源过于庞大.通过代理模型、权值共享等方法可以加速网络搜索过程.Liu 等人^[10]于 2019 年提出可微分的架构搜索,借助反向传播算法同时搜索网络的架构和权值,这极大地提升了神经架构搜索方法的效率.目前,如何快速得到性能优异的网络架构也是研究的热点问题.

神经架构搜索的流程如图 1 所示:在预先设定

的搜索空间中得到一个中间网络架构作为候选架构,通过性能评估策略对此候选架构进行性能度量,最后将测量的结果反馈给搜索策略,不断重复搜索—评估的过程直到发现最优的网络架构.

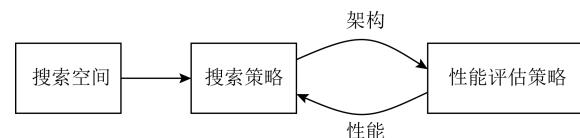


Fig. 1 The process of neural architecture search^[11]

图 1 神经架构搜索流程^[11]

本文将按图 1 的流程进行说明,对典型神经架构搜索方法的原理进行综述,对不同方法之间的关系进行讨论,并对高效的性能评估策略进行总结,最后对未来需要研究的问题进行展望.

1 搜索空间

搜索空间定义了组成网络的基本操作,通过组合不同的操作会产生不同的网络架构.为了使算法可以高效地找到性能优异的网络,就必须构建一个适合的搜索空间.Liu 等人^[12]于 2018 年指出,在构造一个适合搜索空间的前提下,即使使用简单的随机搜索策略也可以发现具有竞争力的神经网络架构.通常情况下,研究者为了提高搜索效率会根据自己的经验适当地缩小搜索空间,但这不可避免地引入了人为的偏见.许多研究者通过神经架构搜索的方法发现了之前人工设计很难相信的网络架构^[13],这证明了神经架构搜索的方法较于人工设计网络的优越性.目前,研究者涉及的搜索空间主要包括链式结构、多分支结构和基于 Cell 的结构.

1.1 链式结构

早期的深度神经网络都是较为简单的链式结构,如 LeNet5^[1], AlexNet^[2]等,即网络中的每一层

仅与其前后相邻的 2 层连接,且网络中没有跨层连接的情况,网络的整个架构呈现为链条状。如图 2 所示的 LeNet5 网络为最典型的链式结构。

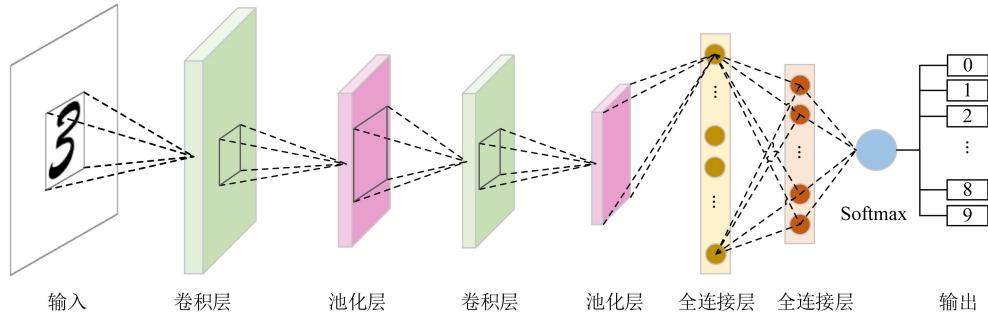


Fig. 2 A typical example of chain structures (LeNet5)

图 2 链式结构的典型例子(LeNet5)

神经网络搜索空间的设计需要从 3 个角度出发:1)网络的层数,可以根据设计者现有的资源动态调整网络的最大层数。2)网络中各层的操作及与其相关的超参数。对于卷积神经网络来说,常用的操作有卷积、池化、批归一化(batch normalization, BN)、激活函数等。具体来说,卷积操作涉及的超参数有卷积核大小、通道数、补齐方式、步长等。3)网络中各操作的排列顺序。对于神经网络来说,结构的细微变化都会对网络的性能产生影响。通常情况下,人工设计网络时会将池化操作放置在卷积操作之后,亦有研究者对多种操作进行组合视为一种操作使用。例如, Ioffe 等人^[14]于 2015 年将卷积、批归一化、激活函数组合为一个模块,将该模块作为构建网络的基本操作继续进行网络架构的生成。随着研究的不断深入,网络的层数不断加深,在训练网络过程中,链式结构易出现梯度消失、梯度爆炸等问题。为解决这一问题,研究者设计了正则化、BN 等方法缓解了梯度所带来的问题,但未从根本上解决这个问题。

1.2 多分支结构

为了缓解梯度消失、梯度爆炸等问题,研究者设计了多分支结构,它允许网络中的层可以与其前面的任意层进行连接。GoogLeNet^[3]是深度神经网络中第一个引入了多分支网络结构的概念,构建了不同的 Inception 模块以扩大网络的深度和宽度, Inception 模块中不同的分支由多种不同的操作组成,通过对输入层不同维度的特征提取,提升了网络的性能,图 3 为一个 Inception 模块的示例。ResNet^[5]在 2016 年首次提出了跳跃连接思想,浅层网络中的特征会通过跳跃连接直接传递到更深的功能层,从而克服了梯度消失对深层网络的影响。跳跃连接的

出现掀起了新一波的研究热潮。例如, DenseNet 于 2017 年^[15]将网络中的每一层都与其前面的层连接,形成了稠密的网络结构。然而该网络中的每一层都会接收其前面所有层的特征作为该层的额外输入,也就意味着需要存储大量的中间特征,因此, DenseNet 需要花费大量的内存开销。为了缓解内存的使用压力,研究者设计了共享存储空间的方法^[16],从而降低 DenseNet 模型的显存。使用多分支结构设计网络的搜索空间时,需要考虑跳跃连接的具体位置和数量。针对不同的实际问题,最佳的跳跃连接位置和数量也不尽相同,因此也为设计深度神经网络架构的自动搜索方法提出了强烈需求。

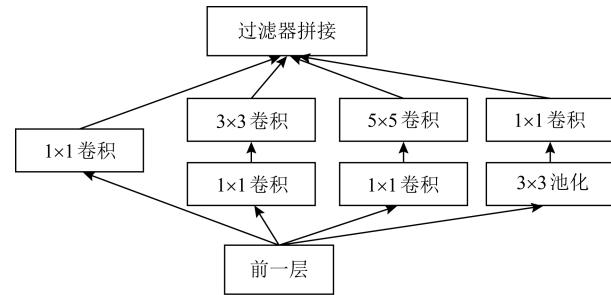


Fig. 3 Inception module of multi-branch structure GoogleNet

图 3 多分支结构 GoogLeNet 的 Inception 模块

1.3 基于 Cell 的结构

基于人工设计网络架构会重复使用相同模块的经验,研究者试图将多个操作组合成 Cell,并将 Cell 作为组成神经网络的基本单元,以此设计了基于 Cell 的网络结构。具体来说,构建基于 Cell 的网络结构分为两步,首先需要构建最优的 Cell 结构,之后将得到的 Cell 按照预先定义的规则进行堆叠,以得

到最终的网络架构.对于单个 Cell 来说,它既可以是简单的链式结构,又可以是复杂的多分支结构.Liu 等人^[12]设计了深度神经网络的层次化表示方法,该方法首先在初始的基本操作上演化得到 Cell 结构,之后将 Cell 视为基本操作进一步搜索 2 级 Cell,最后将得到的 2 级 Cell 按照预先定义的规则进行组合形成最终的网络.基于 Cell 的方法充分考虑了网络对全局与局部的设计,对搜索空间的搜索更细致,是目前最为流行的方法.

通常情况下,对基于 Cell 结构的搜索空间进行设计时需要考虑 2 个方面:1)演化 Cell 的种类.Cell 结构的不同之处在于其内部节点的数量、节点间的连接以及连接上操作的类型.一般情况下,需要演化 2 种类型的 Cell,分别是 Normal Cell 和 Reduction Cell.其中,Normal Cell 的输入和输出特征图的维度一致,而 Reduction Cell 的输出特征图的宽、高是输

入特征图的一半^[10].2)构建最终的网络架构时,不同 Cell 之间的排列顺序.Xie 等人^[17]于 2017 年在构建最终网络架构时,将演化得到的 Cell 按照定义好的方式进行组合.Rawal 等人^[18]在 2018 年将演化得到的多种 Cell 按照定义好的方法进行组合,并固定该组合重复的次数,以此得到循环神经网络.Liu 等人^[10]在 2019 年提出了可微架构搜索方法 DARTS,按照在网络的 1/3,2/3 处放置 Reduction Cell 而在其余部分放置 Normal Cell 的方式,构建了最终的网络架构.

图 4 为 DARTS 中使用的 2 种 Cell 结构.与链式结构和多分支结构相比,基于 Cell 的方法更适用于迁移学习任务,即使用该方法在小规模数据集上得到的 Cell 迁移到同类大规模数据集时,仅需要重复叠加 Cell,就可以得到适用于该大规模数据集上的深层神经网络^[9].

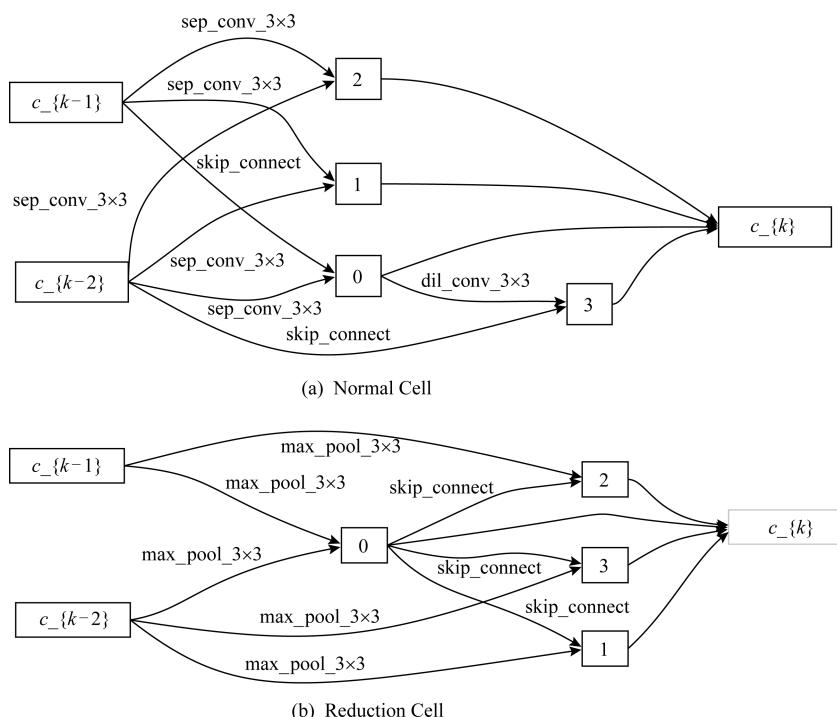


Fig. 4 Two types of DARTS cells

图 4 DARTS 的 2 种 Cell 结构

对 3 种网络结构类型来说,链式结构最为简单、直观,链式结构的网络可以通过增加深度来提升网络的性能.对于多分支结构来说,分支数量的增加意味着网络宽度的增加.多分支结构通过不断增加网络的宽度,探索了多维度特征,提升了网络的性能.基于 Cell 的结构融合了链式结构和多分支结构的优势,既可以增加网络的深度,又可以扩大网络的宽

度.从设计搜索空间的角度来说,链式结构和多分支结构需要考虑网络的整体结构,而基于 Cell 的结构只需要考虑 Cell 的内部结构.从可扩展性角度来看,链式结构和多分支结构受限于卷积神经网络(convolutional neural network, CNN),而基于 Cell 的结构可以轻松地扩展到循环神经网络(recurrent neural network, RNN).

2 搜索策略

在构建好搜索空间后,需要设计一个快速、高效的搜索策略,以实现最优网络架构的搜索。搜索策略的选择需要根据搜索时间、计算资源等要求进行选择。神经网络的架构搜索可以看作是大规模的超参数优化问题,是 AutoML 的重要组成部分。如图 5 所示,目前流行的搜索策略主要有:随机搜索、贝叶斯优化、强化学习、神经进化和基于梯度的方法。

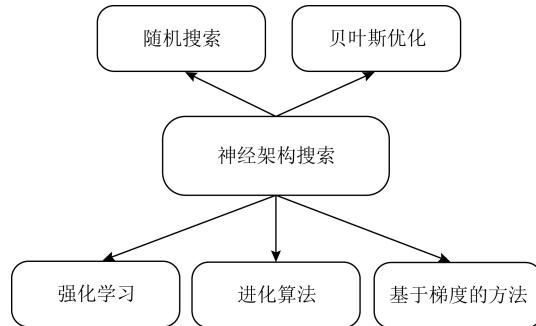


Fig. 5 Method of neural architecture search

图 5 神经架构搜索方法

2.1 贝叶斯优化

深度神经网络中存在大量的超参数,如卷积核大小、学习率等,为了找到最优的超参数组合,最直观的方法就是网格搜索。然而,网格搜索无法利用有效信息指导搜索过程,会进行大量的无效探索。而贝叶斯优化会充分学习上一次的评估结果,并建立概率模型,指导后续的参数选择过程,最常用的概率模型是高斯模型。基于这一优点,研究者设计了基于贝叶斯的神经架构搜索算法^[19-21]。

Bergstra 等人^[22]于 2013 年设计了基于贝叶斯优化的神经网络超参数搜索算法,在 3 个计算机视觉任务上取得了最优的效果。此后,Swersky 等人^[23]于 2014 年对此算法进行了改进,引入了条件参数空间的新内核,并共享了结构间的信息,简化了建模过程,提升了模型质量。Hutter 等人^[24]于 2011 年提出的基于序列模型的优化策略(sequential model-based optimization, SMBO),随后 Negrinho 等人^[25]于 2017 年设计了实验对 SMBO 和随机搜索进行比较,验证了 SMBO 方法优于随机搜索。基于此,Liu 等人^[26]于 2018 年设计了顺序渐进式神经网络架构搜索算法,在搜索过程中不断增加网络的复杂度。同样地,Perez-Rua 等人^[27]于 2019 年基于 SMBO 设计了多模态融合的神经网络架构搜索方法。此外,Jin

等人^[28]于 2018 年基于 SMBO 设计了一个多目标架构搜索框架 PPP-Net,能够自动生成最优网络架构。该方法在 CIFAR-10 数据集上进行架构搜索时,可以同时演化分类误差率、搜索时间、参数量、每秒计算次数等目标,并在移动设备上达到了最佳性能。Golovin 等人^[29]于 2017 年开发了基于贝叶斯优化的网络调参系统 Google Vizier,用于 Google 内部网络调参。

2.2 强化学习

也有研究者将强化学习算法用于神经架构搜索。具体来说,神经网络架构的生成过程可以看作 Agent 智能体选择动作的过程,神经网络的搜索空间对应着 Agent 的动作空间,最终 Agent 经过一系列动作选择后会得到最终的网络架构。针对神经网络架构搜索任务,Agent 的 Reward 奖励对应为网络架构在验证集上的准确率。目前有大量的神经架构搜索算法是基于强化学习实现的,这些方法的不同点在于 Agent 选择动作的策略及用于优化该策略的方法。

Zoph 等人^[6]于 2017 年首次将强化学习应用在神经网络架构搜索,该方法达到了与人工设计的最佳网络相似的性能,并使用 RNN 作为控制器,对设计好的搜索空间进行采样,以此得到最终的网络架构,并在搜索过程中使用 Reinforcement 方法优化 RNN 的参数,随后,Zoph 等人^[9]于 2018 年提出了 NASNet,该方法将 Reinforcement 优化方法替换为近端策略优化(proximal policy optimization, PPO),加快了搜索速度,提高了搜索结果。同时该方法通过多次堆叠在 CIFAR-10 数据集上搜索到的 Cell,很好地迁移到了 COCO 和 ImageNet 两个大数据集上。在图像分类数据集 ImageNet 上达到了 SENet 相同的精度,且模型参数只有 SENet 的一半。Baker 等人^[7]于 2017 年提出了 MetaQNN 用于神经网络架构的搜索,MetaQNN 使用带有 ϵ -greedy 贪婪探索策略和经验回放的 Q-learning 搜索网络的架构。同样,Zhong 等人^[8]于 2018 年设计了基于强化学习的神经网络架构搜索方法 BlockQNN,在该方法中,网络中的层由网络结构代码(network structure code, NSC)表示,NSC 是一个由 5 维向量表示的结构。为了加快算法的搜索进程,BlockQNN 使用了分布式异构框架和早停策略。在提升基于强化学习的神经网络架构搜索算法的搜索速度上,虽然有很多策略可以使用,但由于其搜索过程中需要评估大量的中间网络架构,会消耗大量的搜索时间。

和计算资源.基于此,Pham 等人^[30]于 2018 年提出了高效的 NAS 方法 ENAS,该方法试图将搜索过程中的各个子网络进行权值共享,避免了从头训练网络,与标准的神经架构搜索算法^[6]相比,ENAS 缩短了 1/1000 以上的 GPU 运算时间.Tan 等人^[31]于 2019 年验证了网络的深度、宽度和分辨率与网络性能的关系,对模型缩放进行了进一步探讨,其设计的 EfficientNets 有效地平衡了深度、宽度、分辨率之间的关系,在 ImageNet 上获得了 Top-1 的精度(84.4%),与 AmoebaNet 网络相比,其参数减少了 1/8,且运行速度提升了 6.1 倍.

随着边缘计算的普及,越来越多的任务需要在移动设备上实现.针对计算资源受限的移动设备,Tan 等人^[32]于 2019 年将手机设备上实时运行模型的延迟、精度和运行速度作为搜索目标,提出了适用于移动设备上的神经网络架构搜索算法 MnasNet.此外,MnasNet 为了增加搜索的灵活性,设计了基于因式分解的层级搜索空间,可以将 CNN 分解为小的模块.Yang 等人^[33]于 2018 年在资源预算受限的前提下提出了 NetAdapt 算法,通过逐层简化网络,实现了由大规模数据上的预训练模型到移动设备的迁移任务.Howard 等人^[34]于 2019 年将 MnasNet 神经网络架构搜索算法和 NetAdapt 进行结合,提出了 MobileNetV3,在图像分类、目标检测、图像分割任务上都取得了最佳的效果.Tan 等人^[35]于 2019 年在实验中发现,在同一层中组合多种尺寸的卷积和会提高模型的精度和搜索效率,设计的 MixNets 神经架构搜索方法在移动设备上取得了最佳性能.基于强化学习的神经网络架构搜索方法不仅在图像分类任务上展现了优势,在图像分割、目标检测等任务中也取得了优于人工设计网络的性能,目前基于强化学习的神经架构搜索算法是神经架构搜索领域不可缺少的方法^[36].

2.3 神经进化

神经进化的核心思想是采用进化算法来演化网络的权值、架构、激活函数乃至超参数.随着人工设计网络架构的不断发展,网络的层数不断加深,网络的结构愈加复杂,网络参数的数量不断增加.对于神经架构搜索任务来说,需要的计算资源也在不断增加.目前,基于神经进化的架构搜索算法将网络结构与权值分开进行优化,具体来说,网络的结构使用进化算法进行优化,而网络的权值使用反向传播进行优化.

针对基于进化算法的神经架构搜索算法,在 20

世纪末,研究者们的研究重点是对网络权值进行优化,其流程为:首先选定网络的架构,之后借助遗传算法对网络中的权值进行优化.这类方法在杆平衡任务上取得了很好的效果^[37-38].Kenneth 等人^[39]于 2002 年借助遗传算法的思想,提出了增强拓扑的神经进化网络 NEAT.该方法从最基础的单元结构开始演化,演化过程中引入历史标记来缓解竞争约定问题^[40],此外,NEAT 将整个种群划分为不同的物种,以确保演化过程中网络结构的多样性.NEAT 不仅实现了网络架构的演化,同时也对网络的权值进行优化,实验表明,NEAT 方法远好于最初固定网络架构仅演化权值的方法^[37].Miikkulainen 等人^[13]于 2019 年对 NEAT 方法进行了扩展,由演化拓扑结构的演化扩展到模块及超参数的演化,提出了 CoDeepNEAT.通过共同演化网络的结构和模块,在目标识别、语言建模等任务的标准数据集上达到了与人工设计网络结构性能相近的结果.Liang 等人^[41]于 2019 年将 CoDeepNEAT 与分布式计算框架进行了结合,提出了适用于演化 AutoML 框架的 LEAT,在医学影像分类、自然语言处理领域数据集上都取得了优异的性能.

随着人工设计网络架构的发展,神经架构搜索算法的搜索空间也随之变大.作为群体智能技术之一的进化算法,可以充分有效地对搜索空间进行遍历,鉴于此,基于进化算法的神经架构搜索算法逐渐流行起来.如图 6^[42] 所示,基于进化算法的神经架构搜索算法,将每一个网络架构看作种群中的个体,演化过程中对每一代产生的个体进行选择,并对优秀的个体执行交叉、变异等操作以生成下一代种群,不断重复这个过程,直到产生性能最优的个体或达到最大演化代数.进化算法包括遗传算法、进化策略、遗传编程和进化规划,这 4 种算法均有研究者扩展至神经架构搜索.

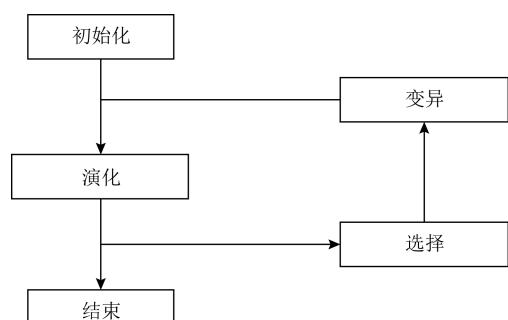


Fig. 6 Evolutionary algorithm^[42]

图 6 进化算法^[42]

基于进化算法的神经架构搜索方法主要的研究有:Salimans 等人^[43]于 2017 年通过实验证明了强化学习可以被进化策略所替代,并在多步决策任务上取得了优异的效果.Real 等人^[44]于 2017 年提出了适用于大规模图像分类的演化算法,可以在最小化人工参与的情况下,借助 250 多台 GPU 服务器,从最简单的初始结构开始,在 CIFAR-10 和 CIFAR-100 两个图像分类基准数据集上达到了与人工设计网络性能相近的效果.此后,Real 等人^[45]于 2019 年在进化算法中加入 Aging 对锦标赛选择算法进行了改进,提出了适用于图像分类任务的 AmoebaNet.该算法通过尽可能多地保留年轻个体,在 ImageNet 数据集上首次超越了人工设计网络的性能,此外,该算法还证明了与强化学习、随机搜索方法相比,进化算法的搜索速度更快.Xie 等人^[17]于 2017 年设计了定长二进制字符串表示网络架构的编码方案,并以此提出了基于遗传算法的卷积网络架构搜索算法.该算法仅仅演化卷积、池化等简单的基本操作,就能在图像分类任务上得到了较好的效果,并且得到的架构可以很好地迁移到大规模数据集上.David 等人^[46]于 2019 年对神经架构搜索的任务进行了扩展,将人工设计的 Transformer 框架引入了演化初始结构,通过锦标赛选择策略得到了自然语言处理领域性能优异的网络架构.

随着应用设备的扩展,神经架构搜索算法从服务器端扩展到移动设备端.移动设备端的神经架构搜索算法需要综合考虑网络的性能、搜索时间、复杂度、计算资源等多方面的问题,神经架构搜索由单目标搜索任务转换为多目标搜索任务.Elsken 等人^[47]于 2019 年提出了拉马克进化算法 LEMONADE 用于多目标神经架构搜索,此外,该算法设计了 Network Morphisms^[48]对神经网络空间结构的算子进行操作以保证网络结构的功能不变性.LEMONADE 借助遗传操作让子代个体可以继承其父代的信息,避免了子代从头学习,极大地减少了训练时间;为了减少网络的规模,LEMONADE 等人设计了 Approximate Network Morphisms 方法,在图像分类任务上达到了与人工设计网络相近的效果.Li 等人^[49]于 2019 年提出了偏序剪枝算法,在实现网络精度优异的前提下,搜索效率最高.此外,小米的 AutoML 团队提出了适用于移动设备的神经架构搜索算法 MoreMNAs^[50],该算法将进化算法和强化学习结合,平衡了演化过程中探索和开发的过程,充分利用了新学习的知识,降低了退化现象.

2.4 基于梯度的方法

对于 2.1~2.3 节所述的神经架构搜索算法来说,它们的搜索空间是离散不可微的,也就意味着在搜索过程中需要评估大量的网络架构,花费大量的时间.为了降低训练网络花费的时间,研究者试图将搜索空间变成连续可微的,再使用基于梯度的方法进行优化.具体来说,基于梯度的架构搜索使用反向传播算法同时对网络权值和网络结构进行优化,减少了大量训练网络消耗的时间,但搜索过程中会占用大量的内存.

基于梯度的神经架构搜索方法主要包含:Liu 等人^[10]于 2019 年提出了 DARTS 算法,首次将离散不可微的搜索空间转换为连续可微的,借助反向传播同时优化网络的架构和权值.在 DARTS 算法中,预先设定了 8 种候选操作,包含卷积核为 3×3 和 5×5 的可分离卷积、 3×3 和 5×5 扩张可分离卷积、 3×3 的最大池化、 3×3 的平均池化、保持不变和无操作.该算法在搜索过程中使用 Softmax 对每条边上的候选操作进行加权,最终只保留每条边上权值最大的操作.Hundt 等人^[51]于 2019 年对 DARTS 算法进行了改进,提出了 sharpDARTS,引入余弦幂次退火对学习率进行更新,借助 MaxW 正则化方法纠正 DARTS 算法中的偏差.Cai 等人^[52]于 2019 年为了降低基于梯度的神经架构搜索算法的内存过大的问题,提出了 ProxylessNAS,该算法在训练过程中使用路径级二值化操作,即每次只激活边上的 1 种操作.ProxylessNAS 在不使用代理模型的前提下针对大规模图像分类数据集 ImageNet 进行网络架构的搜索,并将其算法从 GPU 扩展到了手机上.该算法在 CIFAR-10 数据集上,仅用了 5.7M 个参数就达到了 2.08% 的测试误差,与 AmoebaNet-B 相比减少了 1/6 的参数量.在 ImageNet 数据集上,ProxylessNAS 比 Mobilenet V2 高 3.1% 的准确率,运行速度快了 20%,且手机实测速度是 Mobilenet V2 的 1.8 倍.Zheng 等人^[53]于 2019 年将搜索空间看作一个多项式分布,通过多项式的不同取值对应着不同的网络架构,提出了 MdeNAS,该算法在 ImageNet 数据集上搜索了 4 h 就达到了目前最佳分类效果.Dong 等人^[54]于 2019 年提出了可微架构采样方法 GDAS 用于神经架构搜索,该方法在训练过程中仅采样节点间的 1 种操作,降低了内存消耗.此外,为了缩小搜索空间,GDAS 固定了 Reduction Cell,仅对 Normal Cell 进行搜索,在 V100 显卡上搜索了 4 h 在 CIFAR-10 数据集上得到了 2.5M 个

参数、2.82%误差的网络.针对基于梯度的神经架构搜索算法占用大量内存的问题,研究者们通常会在较浅的网络中搜索最优的架构,之后不断增加层数来验证其性能.Chen 等人^[55]于 2019 年发现:基于梯度的神经架构搜索算法由于在搜索阶段和验证阶段网络层数的不同会出现在训练阶段性能优异的架构在验证阶段性能不好的现象,研究者将这种现象称为深度差距.为了缩小这种差距,Chen 等人提出了 P-DARTS,在搜索过程中逐渐增加网络的层数,并通过搜索空间近似减少了内存消耗,使用了搜索空间正则化来控制搜索的稳定性.此外,对于降低训练过程中内存消耗的问题,Xu 等人^[56]于 2019 年提出了 PC-DARTS 可以在不降低网络性能的前提下进行了高效搜索,该方法通过采样部分 Cell 减少网络中的冗余.

表 1 为基于贝叶斯优化、强化学习、神经进化、梯度的方法进行神经架构搜索的算法在 CIFAR-10 数据集上的性能比较.从搜索架构的性能和效率角度来看,基于梯度的神经架构搜索算法得到的网络性能更好,且搜索速度最快.

Table 1 Performance of Different NAS Algorithms on CIFAR-10

表 1 神经架构搜索方法在 CIFAR-10 数据集上的实验结果

搜索方法	模型	误差率/%
贝叶斯优化	PNAS ^[26]	3.41
	PPP-Net ^[28]	4.36
	NAS v3 ^[6]	3.65
	NASNet ^[9]	2.40
	MetaQNN ^[7]	6.92
	BlockQNN ^[8]	2.35
强化学习	ENAS ^[30]	2.89
	EfficientNet-B7 ^[31]	1.10
	CoDeepNEAT ^[13]	7.30
	EVOLUTION ^[44]	5.40
	AmoebaNet ^[45]	3.34
	GeNet # 2 ^[17]	7.10
神经进化	LEMONADE ^[47]	2.58
	DARTS ^[10]	2.73
	sharpDARTS ^[51]	1.98
	ProxylessNAS ^[52]	2.08
	MdeNAS ^[53]	2.55
	GDAS ^[54]	2.82
基于梯度的方法	P-DARTS ^[55]	2.25
	PC-DARTS ^[56]	2.57

3 性能评估策略

性能评估是神经架构搜索算法的必要环节,通过性能评估测量得到每个网络的性能并反馈给神经架构搜索算法,以指导搜索算法得到最优的网络架构.通常先得到一个在训练集上拟合好的网络,将网络在验证集上的误差作为网络性能的度量值.然而,这种方法需要使用反向传播对网络进行训练,需要大量的计算资源,花费大量的时间,搜索效率较低.为了加快网络架构的评估,本文总结了 4 种常见的性能评估策略,包括低保真度、早停、代理模型和权值共享等方法.

3.1 低保真度

一般情况下,神经网络的性能是网络在训练集上收敛后得到的网络精度,然而网络训练到收敛需要花费大量的时间,因此,为了加快架构搜索过程中网络的收敛过程,研究者提出了采用减少样本数量^[57]、降低图像分辨率^[58]、减少网络层数^[9,45]等方法.通过近似的数据集、近似的网络架构虽然加快了搜索过程,减少了计算成本,但是这不可避免地会引入偏差.Chrabaszcz 等人于 2017 年通过实验指出了低保真度方法会导致评估结果与收敛结果之间产生偏差.即便如此,研究者也可以通过控制近似网络与实际网络之间的差距阈值来提高训练效率.例如减少少量的训练样本、缩减部分网络.

3.2 早停

除了使用低保真度的方法加快搜索速度,也有研究者使用早停机制,即在网络未收敛时就停止训练,其采用的具体方法主要有:1) 固定训练代数.在网络的训练过程中,如果网络达到预先设定的这个阈值就会停止训练,此时的精度就代表了该网络的精度.Zheng 等人^[53]于 2019 年提出并验证了一个性能排序假设:在训练的某个时刻,如果网络 A 在某个数据集上的性能高于网络 B,那么当网络 A、网络 B 在该数据集上训练到收敛后,网络 A 的性能仍高于网络 B.该假设的成功证明极大地提高了神经架构搜索的效率.2) 学习曲线外延的方法^[20,59].在网络训练过程中,根据网络精度曲线的趋势和超参数,对该网络的收敛精度进行预测,而不是直接运行到收敛.Rawal 等人^[18]于 2018 年使用前 10 代的网络精度预测第 40 代的精度,提出的 Seq2seq 模型达到了与人工设计架构相近的结果.

3.3 代理模型

与低保真度和早停的方法不同,基于代理模型的方法采用简单的近似任务替代实际的训练任务,代理模型得到的结果即为网络的性能.基于此,Liu 等人^[26]于 2018 年设计了代理模型来评估候选 Cell 的性能,该方法减少了网络性能评估的时间.目前的研究工作中,在对 ImageNet 数据集进行架构搜索时,大多数方法都会将 CIFAR-10 数据集的架构搜索作为代理任务^[10, 26, 45, 53, 55-56],之后将得到的代理任务上最优的网络架构迁移到 ImageNet 目标任务上.

3.4 权值共享

除了上述提到的网络性能评估策略外,有研究者使用权值共享的方式进行性能评估.权值共享的方法主要有 2 类: Network Morphisms^[48, 60-61] 和 One-Shot^[10, 62].对于 Network Morphisms 方法来说,在保证网络功能不变的前提下,通过不断地扩展初始网络,提高训练效率,该方法不需要网络从头开始训练,但使用该方法得到的网络架构会越来越复杂.Chen 等人^[48]于 2016 年设计了 Approximate Network Morphisms 算法,在保证网络性能不变的情况下对网络的架构进行简化.Jin 等人^[63]于 2019 年将贝叶斯优化与 Network Morphisms 结合提出了开源的自动学习系统 Auto-keras.

与 Network Morphisms 方法不断扩展网络图不同,One-Shot 方法仅需要训练一个超图,通过不断地搜索超图中的子图实现架构的搜索.虽然 One-Shot 的共享权值方法会减少搜索时间,但整个网络的架构被限制在超图中,对于 One-Shot 方法来说,超图的选择是得到最优网络架构的前提.此外,搜索到的子图共享原超图的权值是否可以达到最佳性能也是需要进一步确认的^[49].

网络性能评估策略的提出是为了加速对网络性能进行评估,进而提高架构搜索的效率,而这一切的根源在于网络性能的获得需要对网络进行训练,如果网络无需训练就可以得到性能,那么神经架构搜索的效率会更高.基于上述想法,谷歌于 2019 年提出了权值无关的架构搜索算法^[64],该算法将整个网络共享为一个权值,在评估网络性能时,使用预先设定的权值进行训练以得到网络的性能,极大地减少了网络训练的时间,提高了搜索效率.

4 展望

准数据集上均取得了不错的成绩,但是缺少其他数据集和实际任务的检验.神经架构搜索的鲁棒性也是算法性能的一个关键指标.同时,在搜索网络的过程中,搜索时间、计算资源等影响算法性能的变量也需要作为评估最终性能的指标.神经架构搜索的可解释性也是一个需要关注的重点问题,不仅搜索过程无法科学解释,而且搜索到的网络也无法清晰解释.这些问题都使得神经架构搜索方法无法得到广泛的应用.

最后,神经架构搜索的开源化和可复现性也是限制其发展的瓶颈.通过随机几次得到的结果并不能说明算法的有效性,算法需要随时随地可以复现,而且通过开源代码可以使得更多的人加入到这个领域.总之这个领域还需要很多的努力.

参 考 文 献

- [1] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324
- [2] Krizhevsky A, Sutskever I, Hinton G. ImageNet classification with deep convolutional neural networks [C] // Proc of the 25th Int Conf on Neural Information Processing Systems. New York: ACM, 2012: 1097-1105
- [3] Szegedy C, Liu Wei, Jia Yangqing, et al. Going deeper with convolutions [C] // Proc of IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2015: 1-9
- [4] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [C/OL] // Proc of the 3rd Int Conf on Learning Representations. 2015 [2019-03-15]. <https://arxiv.org/abs/1409.1556>
- [5] He Kaiming, Zhang Xiangyu, Ren Shaoqing, et al. Deep residual learning for image recognition [C] // Proc of IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2016: 770-778
- [6] Zoph B, Le Q. Neural architecture search with reinforcement learning [C/OL] // Proc of the 5th Int Conf on Learning Representations. 2017 [2019-04-16]. <https://arxiv.org/abs/1611.01578>
- [7] Baker B, Gupta O, Naik N, et al. Designing neural network architectures using reinforcement learning [C/OL] // Proc of the 5th Int Conf on Learning Representations. 2017 [2019-09-08]. <https://arxiv.org/abs/1611.02167>
- [8] Zhong Zhao, Yan Junjie, Wu Wei, et al. Practical block-wise neural network architecture generation [C] // Proc of IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2018: 2423-2432

目前,神经架构搜索技术在图像分类的几个基

- [9] Zoph B, Vasudevan V, Shlens J, et al. Learning transferable architectures for scalable image recognition [C] //Proc of IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2018: 8697–8710
- [10] Liu Hanxiao, Simonyan K, Yang Yiming. DARTS: Differentiable architecture search [C/OL] //Proc of the 7th Int Conf on Learning Representations. 2019 [2019-05-05]. <https://arxiv.org/abs/1806.09055>
- [11] Elsken T, Metzen J, Hutter F. Neural architecture search: A survey [J]. Machine Learning Research, 2019, 20(55): 1–21
- [12] Liu Hanxiao, Simonyan K, Vinyals O, et al. Hierarchical representations for efficient architecture search [C/OL] // Proc of the 6th Int Conf on Learning Representations. 2018 [2019-02-14]. <https://arxiv.org/abs/1711.00436>
- [13] Miikkulainen R, Liang J, Meyerson E, et al. Evolving Deep Neural Networks [M]. Pittsburgh, PA: Academic Press, 2019: 293–312
- [14] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift [C] // Proc of the 32nd Int Conf on Machine Learning. New York: ACM, 2015: 448–456
- [15] Huang Gao, Liu Zhuang, Laurens V, et al. Densely connected convolutional networks [C] //Proc of IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 2261–2269
- [16] Geoff P, Chen Danlu, Huang Gao, et al. Memory-efficient implementation of DenseNets [J]. arXiv preprint, arXiv: 1707.06990, 2017
- [17] Xie Lingxi, Yuille A. Genetic CNN [C] //Proc of IEEE Int Conf on Computer Vision. Los Alamitos, CA: IEEE Computer Society, 2017: 1388–1397
- [18] Rawal A, Miikkulainen R. From nodes to networks: Evolving recurrent neural networks [J]. arXiv preprint, arXiv: 1803.04439, 2018
- [19] Ying C, Klein A, Christiansen E, et al. NAS-Bench-101: Towards reproducible neural architecture search [C] //Proc of the 36th Int Conf on Machine Learning. New York: ACM, 2019: 7105–7114
- [20] Domhan T, Springenberg J, Hutter F. Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves [C] //Proc of the 24th Int Joint Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2015: 3460–3468
- [21] Dewancker I, McCourt M, Clark S. Bayesian optimization primer [OL]. 2015 [2019-09-15]. https://app.sigopt.com/static/pdf/SigOpt_Bayesian_Optimization_Primer.pdf
- [22] Bergstra J, Yamins D, Cox D, et al. Making ascience of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures [C] //Proc of the 30th Int Conf on Machine Learning. New York: ACM, 2013: 115–123
- [23] Swersky K, Duvenaud D, Snoek J, et al. Raiders of the lost architecture: Kernels for Bayesian optimization in conditional parameter spaces [J]. arXiv preprint, arXiv: 1409.4011, 2014
- [24] Hutter F, Hoos H, Leyton-Brown K. Sequential model-based optimization for general algorithm configuration [C] // Proc of the 5th Int Conf on Learning and Intelligent Optimization. Berlin: Springer, 2011: 507–523
- [25] Negrinho R, Gordon G. DeepArchitect: Automatically designing and training deep architectures [J]. arXiv preprint, arXiv: 1704.08792, 2017
- [26] Liu Chenxi, Zoph B, Neumann M, et al. Progressive neural architecture search [C] //Proc of the 15th European Conf on Computer Vision. Berlin: Springer, 2018: 19–35
- [27] Perez-Rua J, Vielzeuf V, Pateux S, et al. MFAS: Multimodalfusion architecture search [C] //Proc of IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2019: 6966–6975
- [28] Jin Dongdong, Cheng A, Juan Dacheng, et al. PPP-Net: Platform-aware progressive search for Pareto-optimal neural architectures [C/OL] //Proc of the 6th Int Conf on Learning Representations. 2018 [2019-05-06]. <https://openreview.net/forum?id=B1NT3TAIM>
- [29] Golovin D, Solnik B, Moitra S, et al. Google Vizier: A service for black-box optimization [C] //Proc of the 23rd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2017: 1487–1495
- [30] Pham H, Guan M, Zoph B, et al. Efficient neural architecture search via parameter sharing [C] //Proc of the 35th Int Conf on Machine Learning. New York: ACM, 2018: 4092–4101
- [31] Tan Mingxing, Le Q. EfficientNet: Rethinking model scaling for convolutional neural networks [C] //Proc of the 36th Int Conf on Machine Learning. New York: ACM, 2019: 6105–6114
- [32] Tan Mingxing, Chen Bo, Pang Ruoming, et al. MnasNet: Platform-aware neural architecture search for mobile [C] // Proc of IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 2820–2828
- [33] Yang T, Howard A, Chen Bo, et al. NetAdapt: Platform-aware neural network adaptation for mobile applications [C] //Proc of the 15th European Conf on Computer Vision. Berlin: Springer, 2018: 289–304
- [34] Howard A, Pang Ruoming, Adam H, et al. Searching for MobileNetV3 [C] //Proc of the 2019 IEEE/CVF Int Conf on Computer Vision. Los Alamitos, CA: IEEE Computer Society, 2019: 1314–1324
- [35] Tan Mingxing, Le Q. MixConv: Mixed depthwise convolutional kernels [J]. arXiv preprint, arXiv: 1907.09595, 2019

- [36] Gong Xinyu, Chang Shiyu, Jiang Yifan, et al. AutoGAN: Neural architecture search for generative adversarial networks [C] //Proc of Int Conf on Computer Vision. Los Alamitos, CA: IEEE Computer Society, 2019: 3224–3234
- [37] Gomez F, Miikkulainen R. Solvingnon-Markovian control tasks with neuro-evolution [C] //Proc of the 16th Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 1999: 1356–1361
- [38] Saravanan N, Fogel D B. Evolving neural control systems [J]. IEEE Expert, 1995, 10(3): 23–27
- [39] Kenneth O S, Miikkulainen R. Evolving neural networks through augmenting topologies [J]. Evolutionary Computation, 2002, 10(2): 99–127
- [40] Montana D, Davis L. Training feedforward neural networks using genetic algorithms [C] //Proc of the 11th Int Joint Conf on Artificial Intelligence. San Francisco, CA: Morgan Kaufmann, 1989: 762–767
- [41] Liang J, Meyerson E, Hodjat B, et al. Evolutionary neural AutoML for deep learning [C] //Proc of the Geneticand Evolutionary Computation Conf. New York: ACM, 2019: 401–409
- [42] Eiben A, Smith J. From evolutionary computation to the evolution of things [J]. Nature, 2015, 521(7553): 476–482
- [43] Salimans T, Ho J, Chen Xi, et al. Evolution strategies as a scalable alternative to reinforcement learning [J]. arXiv preprint, arXiv:1703.03864v2, 2017
- [44] Real E, Moore S, Selle A, et al. Large-scale evolution of image classifiers [C] //Proc of the 34th Int Conf on Machine Learning. New York: ACM, 2017: 2902–2911
- [45] Real E, Aggarwal A, Huang Yanping, et al. Regularized evolution for image classifier architecture search [C] //Proc of the 33rd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2019: 4780–4789
- [46] David R, Le Q, Chen Liang. The evolved transformer [C] // Proc of the 36th Int Conf on Machine Learning. New York: ACM, 2019: 5877–5886
- [47] Elsken T, Metzen J, Hutter F. Efficient multi-objective neural architecture search via Lamarckian evolution [C/OL] //Proc of the 7th Int Conf on Learning Representations. 2019 [2019-07-09]. <https://arxiv.org/abs/1804.09081v2>
- [48] Chen Tianqi, Goodfellow I, Shlens J. Net2Net: Accelerating learning via knowledge transfer [C/OL] //Proc of the 4th Int Conf on Learning Representations. 2016 [2019-09-18]. <https://arxiv.org/abs/1511.05641>
- [49] Li Xin, Zhou Yiming, Pan Zheng, et al. Partial order pruning: For best speed/accuracy trade-off in neural architecture search [C] //Proc of IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 9145–9153
- [50] Chu Xiangxiang, Zhang Bo, Xu Ruijun, et al. Multi-objective reinforced evolution in mobile neural architecture search [J]. arXiv preprint, arXiv:1901.01074, 2019
- [51] Hundt A, Jain V, Hager G. sharpDARTS: Faster andmore accurate differentiable architecture search [J]. arXiv preprint, arXiv: 1903.09900, 2019
- [52] Cai Han, Zhu Ligeng, Han Song. ProxylessNAS: Direct neural architecture search on target task and hardware [C/OL] //Proc of the 7th Int Conf on Learning Representations. 2019 [2019-10-10]. <https://arxiv.org/abs/1812.00332>
- [53] Zheng Xiawu, Ji Rongrong, Tang Lang, et al. Multinomial distribution learning for effective neural architecture search [C] //Proc of the 2019 IEEE/CVF Int Conf on Computer Vision. Los Alamitos, CA: IEEE Computer Society, 2019: 1304–1313
- [54] Dong Xuanyi, Yang Yi. Searching for a robust neural architecture in four GPU hours [C] //Proc of IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 1761–1770
- [55] Chen Xin, Xie Lingxi, Wu Jun, et al. Progressive differentiable architecture search: Bridging the depth gap between search and evaluation [C] //Proc of 2019 IEEE/CVF Int Conf on Computer Vision. Los Alamitos, CA: IEEE Computer Society, 2019: 1294–1303
- [56] Xu Yuhui, Xie Lingxi, Zhang Xiaopeng, et al. PC-DARTS: Partial channel connections for memory-efficient differentiable architecture search [J]. arXiv preprint, arXiv: 1907.05737, 2019
- [57] Klein A, Falkner S, Bartels S. Fast Bayesian optimization of machine learning hyperparameters on large datasets [C] // Proc of the 20th Int Conf on Artificial Intelligence and Statistics. Brookline, MA: Microtome Publishing, 2017: 528–536
- [58] Chrabaszcz P, Loshchilov I, Hutter F. A downsampled variant of ImageNet as an alternative to the CIFAR datasets [J]. arXiv preprint, arXiv:1707.08819, 2017
- [59] Baker B, Gupta O, Raskar R, et al. Accelerating neural architecture search using performance prediction [C/OL] // Proc of the 6th Int Conf on Learning Representations. 2018 [2019-07-32]. <https://arxiv.org/abs/1705.10823>
- [60] Wei Tao, Wang Changhu, Rui Yong, et al. Network morphism [C] //Proc of the 33rd Int Conf on Machine Learning. New York: ACM, 2016: 564–572
- [61] Cai Han, Chen Tianyao, Zhang Weinan, et al. Efficient architecture search by network transformation [C] //Proc of the 32nd AAAI Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2018: 2787–2794
- [62] Gabriel B, PieterJan K, Barret Z, et al. Understanding and simplifying one-shot architecture search [C] //Proc of the 35th Int Conf on Machine Learning. New York: ACM, 2018: 549–558

- [63] Jin Haifeng, Song Qingquan, Hu Xia. Auto-keras: An efficient neural architecture search system [C] //Proc of the 25th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining. New York: ACM, 2019: 1946–1956
- [64] Gaier A, Ha D. Weight agnostic neural networks [C] //Proc of Conf on Neural Information Processing Systems. Cambridge, MA: MIT Press, 2019: 5365–5379



Meng Ziyao, born in 1995. Master. His main research interests include deep learning and evolutionary computation.

孟子尧,1995 年生.硕士.主要研究方向为深度学习和进化计算.



Gu Xue, born in 1994. Master. Her main research interests include deep learning, medical image processing and evolutionary computation.

谷 雪,1994 年生.硕士.主要研究方向为深度学习、医学图像处理、进化计算.



Liang Yanchun, born in 1953. PhD, professor, PhD supervisor. His main research interests include machine learning, evolutionary computation and bioinformatics.

梁艳春,1953 年生.博士,教授,博士生导师.主要研究方向为机器学习、进化计算和生物信息学.



Xu Dong, born in 1965. PhD, professor. His main research interests include bioinformatics and machine learning.

许 东,1965 年生.博士,教授.主要研究方向为生物信息学和机器学习.



Wu Chunguo, born in 1976. PhD, associate professor. His main research interests include machine learning and evolutionary computation.

吴春国,1976 年生.博士,副教授.主要研究方向为机器学习和进化计算.