解读百度AutoDL：打破SOTA纪录的神经架构搜索是如何炼成的

机器之心
已认证的官方帐号
已关注

机器之心原创，作者：邱陆陆。

近日，百度大数据实验室在 arXiv 上发布了两篇论文，一篇给出了任何深度学习网络在小学习率情况下的收敛性证明，包括用 AutoDL 搜出来的网络，另一篇则提供了一个正则化的方法，让 AutoDL 搜索到的网络的训练结果超过了之前所有公开报道的结果。基于 PaddlePaddle 框架实现的代码已经开源，相应功能也可以通过百度 EasyDL 免费使用，这是继去年 11 月 AutoDL 2.0 于 2018 百度世界大会上正式发布以来，AutoDL 的又一次重要更新。

机器之心就 AutoDL 各方向的设计思路和论文内容采访了百度大数据实验室主任浣军教授，以下为采访实录。

机器之心：百度开发 AutoDL 的初衷是什么？想要实现什么目标？

AutoDL 的理念，用一句话来概括，就是「开放普惠 AI」，让广大中小企业、初创企业和个人能够更方便地应用大数据和深度学习。

现在，这些能力主要掌握在大公司研发中心或者高校中间，并未向中小企业和初创企业辐射，原因在于大数据分析和深度学习对硬件、软件以及工程技术人员的能力要求都比较高。

AutoDL 所做的事情，就是用深度学习来设计深度学习，从而实现让大家都能够快速用到这项能力。我们的愿景是把如今的「深度学习模型艺术品」变成「深度学习模型工业产品」，让深度学习的模型能够像工厂的产品一样被大规模地生产出来。

机器之心：这一目标具体由哪些需求组成？如何满足这些需求？

我们从三个维度思考这件事。硬件、应用场景和模态的多样化决让 AI 算法的维度空间极为庞大。想要尽可能探索这一空间，就必然要从手工设计模型，转向自动化生产模型，快速高效地产生能够适配不同硬件、支持不同场景、适应不同模态的深度学习模型。

为了实现这些需求，我们将 AutoDL 分成三个部分，分别是 AutoDL Design，AutoDL Transfer 和 AutoDL Edge。

AutoDL Design 根据用户提供的数据集从头设计全新深度学习模型。

AutoDL Transfer 支持小数据建模，利用百度拥有的大量数据预训练好的模型迁移到用户具体的应用场景上。

AutoDL Edge 将深度学习模型部署到拥有不同算力、内存资源的硬件上，满足不同的能源消耗、响应时间需求。是 AI 和 IoT 的结合，是深度学习和边缘计算的完美结合。

AutoDL Design：更大的模型结构搜索空间带来更佳的效果

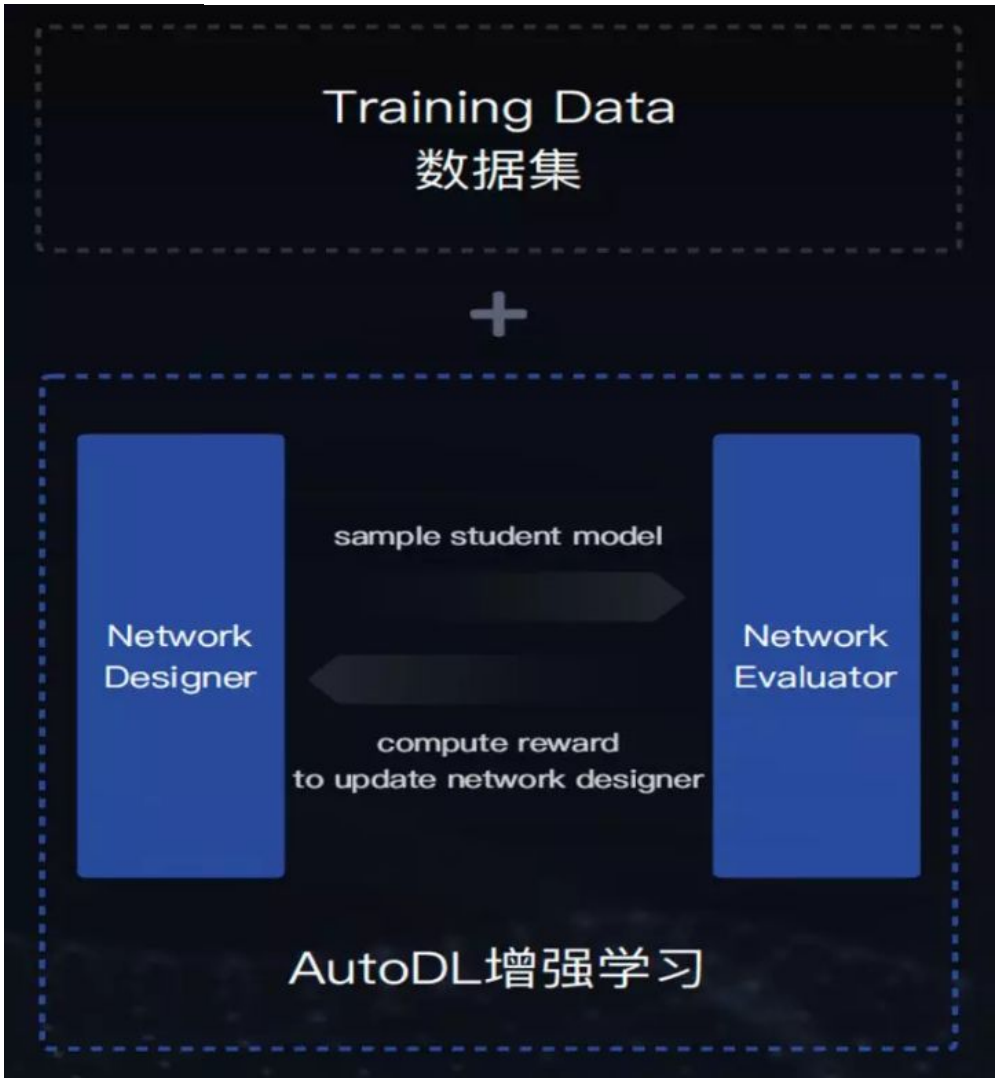
机器之心：从用户给出标注数据集到拿到自动设计好的模型结构，是一个什么样的过程？

现在 AutoDL Design 有多条技术路线，但总的来说仍然是一个端到端的训练过程。百度在模型结构优化方面选择了两条主要技术路线。

第一条技术路线利用深度增强学习完成设计。系统由两部分组成，第一部分是网络结构的编码器，第二部分是网络结构的评测器。

编码器通常以 RNN 的方式把网络结构进行编码，然后评测器把编码的结果拿去进行训练和评测，拿到包括准确率、模型大小在内的一些指标，反馈给编码器，编码器进行修改，再次编码，如此迭代。经过若干次迭代以后，最终得到一个设计好的模型。

为了性能考虑，迭代中用到的训练数据通常是几万张规模的数据集（比如 CIFAR-10），模型设计结束后，会用大规模数据（比如 ImageNet）重新训练一次，进一步优化参数。



图：AutoDL 增强学习流程

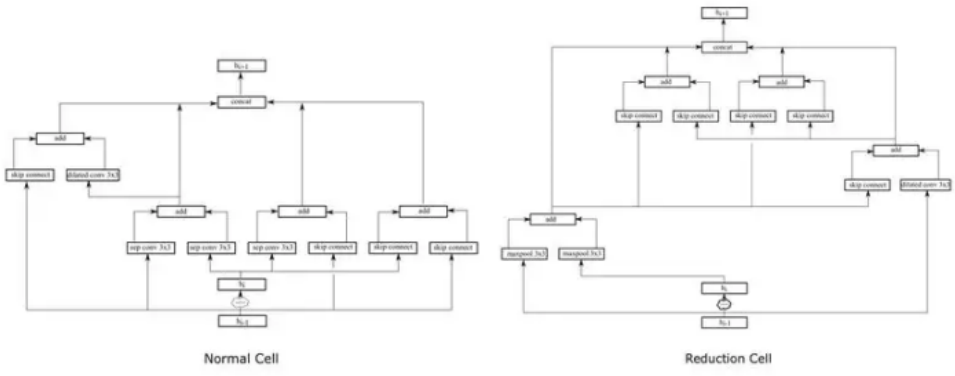
第二条技术路线是将结构图构建为可微的结构。即，连接节点的边不再是非 0 即 1 的状态，而是变成一个可求微分的概率。

除此之外，我们还进行了超参数优化，正则化训练等其他一系列优化，最终，我们在 CIFAR-10 上取得了正确率 98% 以上，这个结果优于所有有公开报道的设计网络效果，包括人类专家设计的和机器自动设计的。

机器之心：能否更详细地解释基于深度增强学习的技术路径里编码器与评测器的工作？迭代过程中计算资源消耗情况？

编码器的可以从一个随机的模型开始，也可以从一种已知的模型出发。从性能角度考虑，通常我们会选择从一个较优的模型结构出发。

模型优化分为三个层级，分别是单元格优化，单元格连接方式优化以及超参数优化。单元格（cell）是模型的基本结构，每个单元格由几个到十几个节点（node）组成。每个节点都是一种常见的操作，例如一次卷积运算就是一个节点，常见的节点中的操作有十个左右。



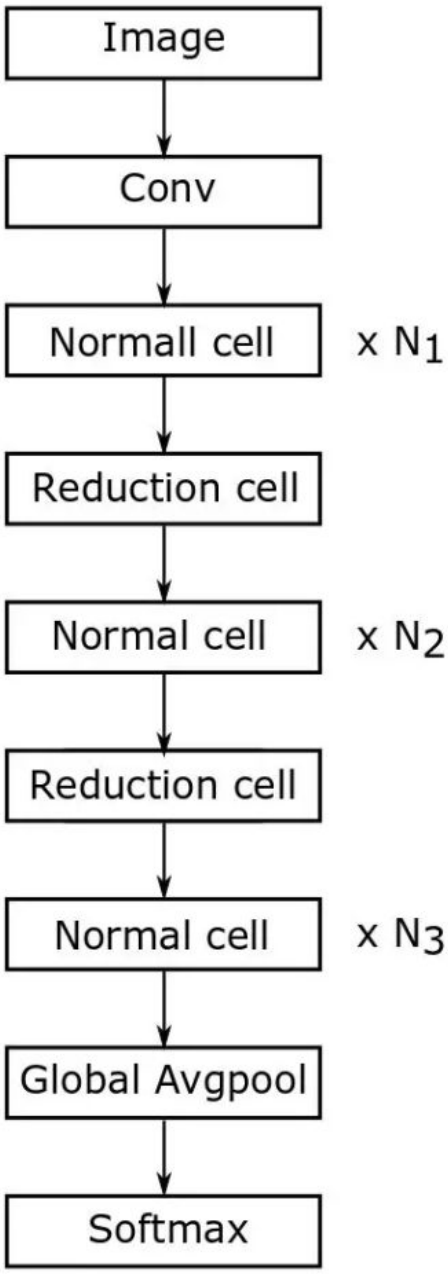
图：AutoDL Design 设计的单元格，左边为普通单元格（Normal Cell），右边为缩减单元格（Reduction Cell）

奖励函数是一个正确率的函数。这里的正确率并不是说每进行一次迭代就要在全部训练数据上训练到完全收敛。而是采用了「提前终止」（early stopping）的方法，用训练到一定程度的结果来预测最终结果。

在 CIFAR-10 级别的数据集上（数万张图片），每次迭代平均需要不到 1 GPU hour，从开始搜索到找到理想的模型结构，平均需要进行 50～200 次迭代。

机器之心：AutoDL Design 设计出的模型结构与人工设计的模型结构有什么区别?

如果把图像识别的常见模型用有向无环图表示出来，我们会发现：VGG 模型的结构是线性的；ResNet 在线性结构的基础上添加了「跳层连接」；Inception 模型的结构更偏向树状。而 AutoDL Design 的不受任何现成网络结构的约束，能够在所有可能的有向无环图空间内进行探索。



图：AutoDL Design 设计的模型结构

另外，AutoDL Design 的一个特点是可以实现多目标优化：如果目标除了效果好之外，还对模型大小、运行时间有要求，系统也可以根据要求完成设计。

AutoDL Transfer：「小数据」也可以建「大模型」

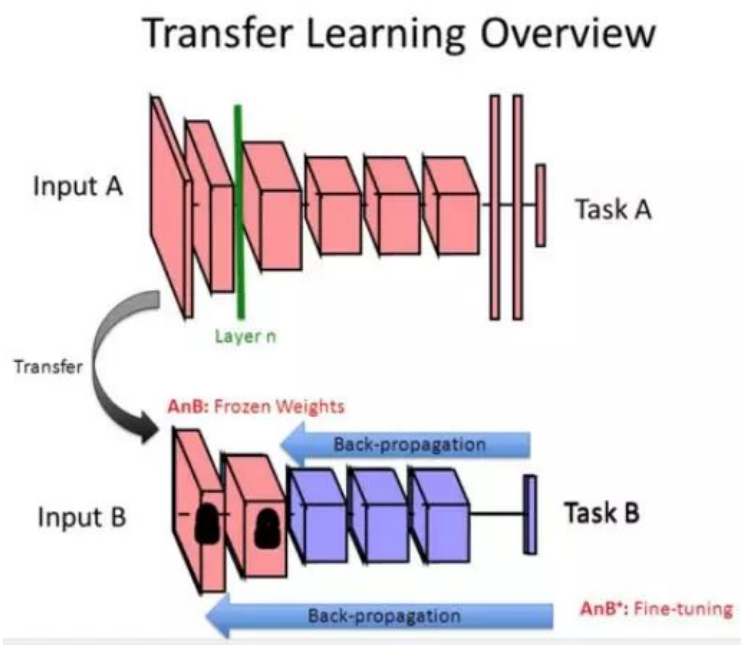
机器之心：能否介绍一下 AutoDL Transfer 的优化方式?

AutoDL Transfer 是在 ResNet、DenseNet、VGG、Inception 等经典网络的基础上，进行了一些基于人工经验的组
（bilinear）组件，

AutoDL Transfer 现在提供两种服务，分别是静态模型（Static Model）和动态模型（Dynamic Model）。

静态模型是在大量已有数据集上进行测试后，选出能够在大部分数据集上取得优异性能的模型结构，然后利用用户数据精调模型参数。

动态模型则会根据用户数据在不同组件以及超参的组合中重新进行一次搜索。和静态模型相比，动态模型能够针对用户数据进行更加精细的优化，但也需要用户提供更多数据。



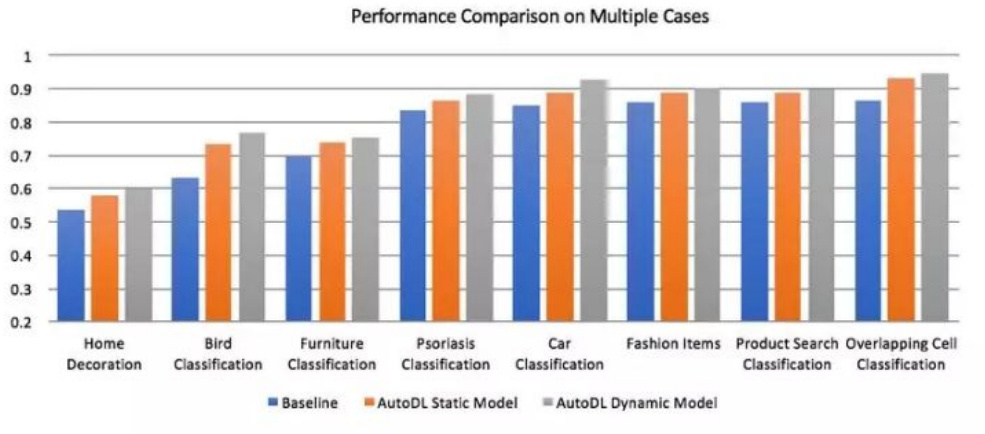
图：AutoDL Transfer 示意图

机器之心：什么样的用户场景适合选择 AutoDL Transfer？AutoDL Transfer 对用户数据规模有什么要求？

首先，数据量比较小的情景下，很难从头训练大模型，此时 AutoDL Transfer 仍然能保证一个很好的模型效果。

同时，即使用户数据量比较大，迁移学习仍然能把初始阶段从大规模数据集中习得的物体特征的知识以参数的形式带入到接下来有针对性的优化过程中，效果往往比从头训练要好。

AutoDL Transfer 还使用了一些百度自己研发的技术，包括自动数据增强、迁移过程中正则化项的优化等。这些技术都有助于在小数据条件下提升模型泛化能力，因此 AutoDL Transfer 对用户数据规模几乎没有限制，分类任务中，每个类别的数据可以只有 100 张甚至几十张。用户可以在上传数据后几分钟就拿到训练好的模型结果。关于 AutoDL Transfer 的最新进展，可以见我们在 ICLR 2019 上发表的文章。



图：AutoDL Transfer 静态模型与动态模型在不同任务中的效果示意图

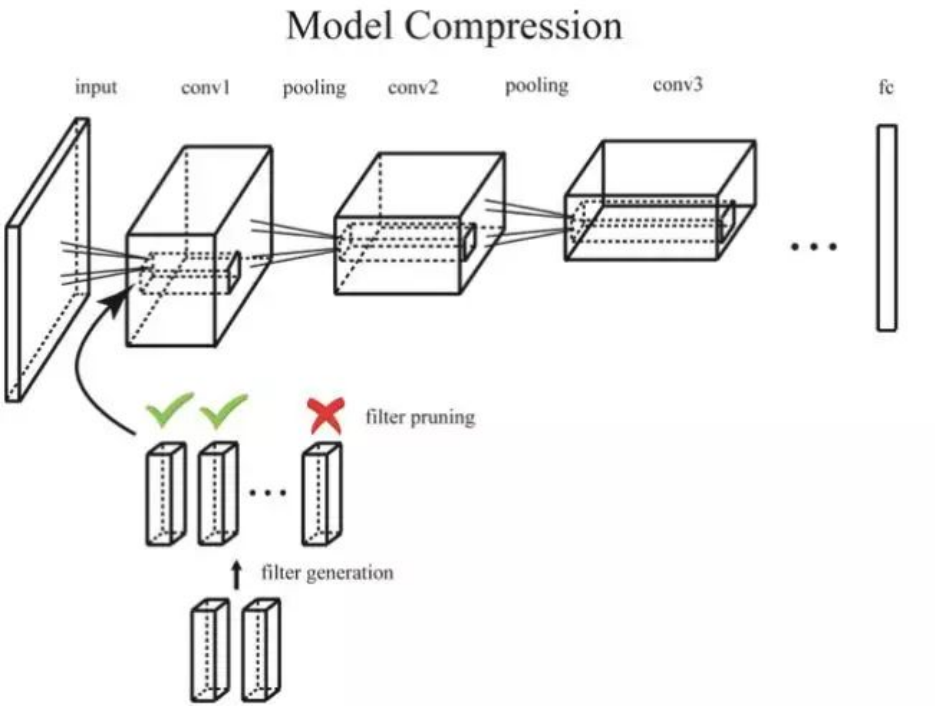
AutoDL Edge：事半功倍的终端计算

机器之心：AutoDL Edge 采用了哪些优化方法?

AutoDL Edge 主要旨在对模型进行压缩，使得同等边缘算力支持更多 AI 能力，从而拓宽应用场景。

因为市面上有非常多不同的硬件配置，因此我们的团队先研究了一些设备无关的通用的模型压缩算法，这类算法能够同时减小网络规模、提升推理速度且不改变模型的性能。

滤波器剪枝（Filter Pruning）就是其中一种典型的技术。我们会估算每一个卷积核的重要程度，在每一个卷积层中，去掉那些不那么重要的卷积核。此外，我们也会对计算资源消耗最大的全连接层做矩阵低秩分解，加速矩阵乘法。



图：滤波器剪枝示意图

观察到深度卷积网络的参数主要集中在卷积核上，我们最新采用的模型压缩算法通过参数共享来压缩卷积核的参数空间
进行参数共享。我们

使用参数共享或者其他的压缩方法。配合参数量化（quantization）方法，我们的方法在 CIFAR-10 上在准确率仅降低 0.27% 的条件下将 ResNet-18 压缩了 50 多倍。在目标检测任务上，我们的方法将 Single Shot MultiBox Detector（SSD）网络进行了有效压缩，在参数数量（0.45M）明显小于 Tiny SSD（1.13M）下，在 VOC 2007 的测试集上的平均准确率（mAP）反而有大幅提升。

此外，还有一些针对性特定芯片的优化，设备提出了算力、能耗、响应时间等约束。而算法设法在这些约束之下降低运算量，压缩模型大小。一旦将模型压缩到缓存可以容纳的大小，就可以极大加速 I/O。

值得一提的是，模型压缩的过程也是自动化的。

同时，我们也在探索用 AutoDL Design 的思路，不需要针对一个大模型进行压缩，而是从头寻找一个小模型。

机器之心：关于此次公开的两篇新论文，神经网络的收敛性证明有什么意义？

深度神经网络广泛使用随机梯度下降，其优化的特性，例如是否会离开局部最优，收敛到全局最优一直是大家关心的问题，最近这方面也有一些有意思的进展。我们的探索主要集中在构造一族损失函数。通过这样的构造，我们可以在即使学习率非常低的情况下，对于每一个局部最优，证明 SGD 都一定的概率逃逸出局部最优。逃逸概率不但与极值点的值有关而且我们也证明了与极值点附近的几何性质有关。如果我们跑 SGD 足够长时间，SGD 会以马氏链的方式遍历局部最优，可以大概率收敛到全局最优。基于这几点，我们期望未来对极值点附近的几何性质的研究会对深度学习有强有力的促进作用。

机器之心：第二篇论文提出，利用局部拉德马赫复杂度做正则化，从而提高网络泛化特性，能否详细介绍一下其做法？

深度学习的核心问题之一在于如何保证在有限样本上学到的分类器或者预测函数能在将来未观察到的数据，例如测试数据上，仍然有预测错误率的保证。因为在数据点上的 0-1 的离散错误很难精确优化，在通常的实践中，预测函数都是通过在训练数据上最小化一个损失函数得到，这个经验函数一般是预测错误率的一个上界。在统计中分类器在训练集和测试集之间的差，可以用预测函数族的一个被称之拉德马赫复杂度的标准来衡量。预测函数族越小，拉氏复杂度也越小，经验损失和泛化损失的差距也越小。拉氏复杂度在经典支持向量机有这广泛的应用。

目前深度神经网络成为广泛应用的预测函数。因为神经网络的多层结构，其所属的函数族可以逼近任意的连续函数，这使得许多经典的用函数族的拉德马赫复杂度作为正则化项的统计学习方法无法进行。但统计学习领域中的局部拉德马赫复杂度，即只考虑全函数族的一个子族上的拉德马赫复杂度，却可以避开全局拉德马赫复杂度的问题。

我们提出的局部拉德马赫复杂度方法借鉴了已有的局部拉德马赫复杂度方法，仅考虑在经验损失函数的极小值点附近的一个球内的拉德马赫复杂度。采用最近的拉德马赫复杂度的估计方法，我们对折页损失函数 (Hinge Loss) 和交叉熵（cross entropy）推得了这个固定值的表达式，并且将其称之为局部拉德马赫正则化项，并加在经验损失函数上。我们对提出的局部拉德马赫正则化方法在标准的网络结构（即 ResNet-18）和 CIFAR-10 上进行了实验，发现其可以有效降低损失函数在测试数据上的值并且提高预测准确率，体现了增强的泛化性能。我们进一步将该方法应用到被搜索出来的网络结构上，发现局部拉德马赫正则化方法和其他的提高泛化性能的方法，包括混合（mix-up）和模型集成（model ensemble），可以兼容。将我们的正则化方法作用在混合和模型集成之后，我们得到了 CIFAR-10 上目前最好的准确率。在我们的文章中也提供了基于PaddlePaddle框架实现的开源代码。

进化中的 AutoDL：剑指「一步到位」的深度学习模型

机器之心：从 Auto

主要有三方面变化。

第一，在自动设计效果上，现在的 AutoDL 设计出的神经网络已经全面超过人类专家设计的网络效果。图像识别公开数据集 CIFAR-10 上，达到了超过 98% 的正确率。这个效果优于所有有公开报道的人类专家设计的网络的效果。

第二，在模态方面，除了视觉之外，我们也增加了对语音任务的支持，包括语音模型压缩、语音模型自动建模等。

第三，在模型适配上，我们增加了一些具体的应用案例，包括对一些可以用于新零售的视觉硬件的支持。

机器之心：AutoDL 团队现在在进行哪些新方向的探索?

我们特别关心 AutoDL 三个方向的结合，换言之，能不能同时完成模型的设计、迁移和适配。

这也是我们在强化学习技术路径之外，也同时关注可微分结构路径的原因：可微分结构既可以用于自动模型搜索，也可以用于迁移学习。模型的安全性也是我们重点关注的方向。我们希望设计的网络能够抗攻击并且具有一定的可解释性。

References:
1. Wenqing Hu, Zhanxing Zhu, Haoyi Xiong, Jun Huan, Quasi-potential as an implicit regularizer for the loss function in the stochastic gradient descent, arxiv.org/abs/1901.0605...
2. Yingzhen Yang, Xingjian Li, Jun Huan, An Empirical Study on Regularization of Deep Neural Networks by Local Rademacher Complexity, arxiv.org/abs/1902.0087...
3. ICLR'19】 Xingjian Li, Haoyi Xiong, Hanchao Wang, Yuxuan Rao, Liping Liu, Jun Huan, DELTA: Deep Learning Transfer Using Feature Map with Attention for Convolutional Networks, the Seventh International Conference on Learning Representations (ICLR’ 19), New Orleans, Louisiana, USA, May 2019 openreview.net/forum?...
4. PaddlePaddle code available at: github.com/PaddlePaddle...

发布于 2019-02-12

人工智能 AutoML 神经网络

文章被以下专栏收录

机器之心
提供专业的前沿科技信息

关注专栏

推荐阅读

神经网络可解释性对具体应用的推动