DrNAS: Dirichlet Neural Architecture Search

- 作者：Xiangning Chen, Ruochen Wang, Minhao Cheng, et.al.
- 机构：UCLA, DiDi AI Labs

- 会议：ICLR2021
- 地址：https://arxiv.org/abs/2006.10355
- 代码：https://github.com/xiangning-chen/DrNAS

论文主要内容

摘要

本文将NAS建模成一个学习分布的问题，将op连续松弛的权重看作随机变量（服从Dirichlet分布）。这种方式能自然的引入探索、提高泛化能力。其次为了减轻微分NAS的内存开销，本文提出了一个简单的渐进学习策略能够直接在large-scale 任务上搜索。最后在不同数据集、搜索空间下验证DrNAS的有效性。

研究内容

方法

方法

- DARTs：

$$\min_{\theta} \ \mathcal{L}_{val}(w^*, \theta) \quad \text{s.t.} \quad w^* = \arg\min_{w} \ \mathcal{L}_{train}(w, \theta), \quad \sum_{o=1}^{|\mathcal{O}|} \theta_o^{(i,j)} = 1, \ \forall \ (i,j), \ i < j, \qquad (1)$$

- DARTs那套做法把 θ 看作一个可学习参数，直接对其优化。
  - （**点估计**）导致在 θ 在验证集上过拟合，引入巨大的泛化误差。
  - 直接优化 θ 导致搜索算法开始阶段快速收敛到次优路径，缺乏探索。
- 由此将DARTs建模成学习一个distribution的问题： θ 是一个**随机变量**，从一个可学习的分布(分布参数 β)中采样得到

$$\min_{\beta} E_{q(\theta|\beta)}\left[\mathcal{L}_{val}(w^*,\theta)\right] + \lambda d(\beta,\hat{\beta}) \ \text{ s.t. } \ w^* = \arg\min_{w} \ \mathcal{L}_{train}(w,\theta). \tag{2}$$

取 $q(\theta \mid \beta) \sim \mathrm{Dir}(\beta)$ 。设置Anchor $\hat{\beta} = 1$ ，避免 β 太小(sparse sample、high variance、unstable)太大(dense sample、low variance、insufficient exploration)。

$$\frac{d\theta_i}{d\beta_j} = -\frac{\frac{\partial F_{Beta}}{\partial \beta_j}(\theta_j|\beta_j,\beta_{tot}-\beta_j)}{f_{Beta}(\theta_j|\beta_j,\beta_{tot}-\beta_j)} \times \left(\frac{\delta_{ij}-\theta_i}{1-\theta_j}\right) \quad i,j = 1,...,|\mathcal{O}|, \tag{3}$$

获得最优架构

$$o^{(i,j)} = \arg\max_{o\in\mathcal{O}} E_{q(\theta_o^{(i,j)}|\beta^{(i,j)})}\left[\theta_o^{(i,j)}\right]. \tag{4}$$

这个期望就是Dirichlet分布均值： $\dfrac{\beta_o^{(i,j)}}{\sum_{o'}\beta_{o'}^{(i,j)}}$

正则性

- DARTs的泛化误差和验证集loss关于架构参数的hessian矩阵特征值高度相关

> Hessian矩阵的特征值就是形容其在该点附近特征向量方向的凹凸性，特征值越大，凸性越强。你可以把函数想想成一个小山坡，陡的那面是特征值大的方向，平缓的是特征值小的方向。

- 优化目标公式(2)等价于公式(5)：

$$\min_{\beta} E_{q(\theta|\beta)}\left[\mathcal{L}_{val}(w^*,\theta)\right] \ \text{ s.t. } \ w^* = \arg\min_{w} \ \mathcal{L}_{train}(w,\theta) \ , \ d(\beta,\hat{\beta}) \le \delta, \tag{5}$$

Proposition 1 *Let $d(\beta,\hat{\beta}) = \|\beta-\hat{\beta}\|_2 \le \delta$ and $\hat{\beta}=1$ in the bi-level formulation (5). Let μ denote the mean under the Laplacian approximation of Dirichlet. If $\nabla_\mu^2 \tilde{\mathcal{L}}_{val}(w^*,\mu)$ is Positive Semi-definite, the upper-level objective can be approximated bounded by:*

$$E_{q(\theta|\beta)}(\mathcal{L}_{val}(w,\theta)) \gtrsim \tilde{\mathcal{L}}_{val}(w^*,\mu) + \frac{1}{2}\left(\frac{1}{1+\delta}\left(1-\frac{2}{|\mathcal{O}|}\right) + \frac{1}{|\mathcal{O}|}\frac{1}{1+\delta}\right)tr\left(\nabla_\mu^2 \tilde{\mathcal{L}}_{val}(w^*,\mu)\right) \tag{6}$$

with:

$$\tilde{\mathcal{L}}_{val}(w^*,\mu) = \mathcal{L}_{val}(w^*, Softmax(\mu)), \ \ \mu_o = \log\beta_o - \frac{1}{|\mathcal{O}|}\sum_{o'}\log\beta_{o'}, \ \ o = 1,\dots,|\mathcal{O}|.$$

- 公式(6)的hessian矩阵迹就是特征值之和，被验证集loss的期望bound

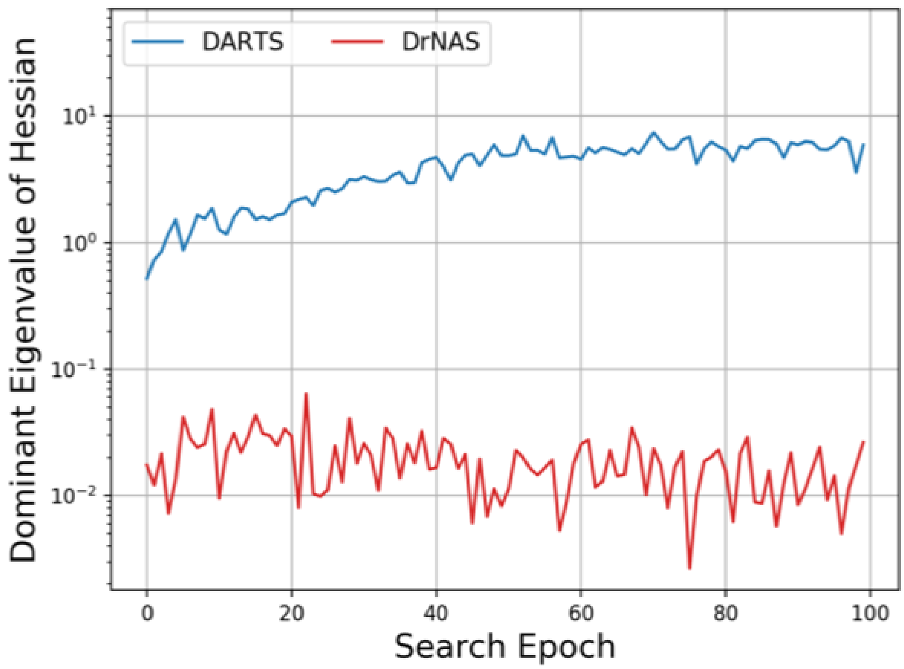


Figure 4: Trajectory of the Hessian norm on NAS-Bench-201 when searching with CIFAR-10 (best viewed in color).

渐进式架构学习

- GPU memory会随着DARTs的候选op增加而线性增长。
  - 常会使用一些proxy task（用更小的数据集、更少的网络层、channel数）
  - PC-DARTs提出partial channel，训练中随机使用1/k channel，剩余channel shotcut。这样会损失信息、带来随机性，直接结合分布学习加大不稳定

CIFAR-10		
K	**Test Accuracy (%)**	**GPU Memory (MB)**
1	94.36 ± 0.00	2437
2	93.49 ± 0.28	1583
4	92.85 ± 0.35	1159
8	91.06 ± 0.00	949
Ours	94.36 ± 0.00	949
CIFAR-100		
K	**Test Accuracy (%)**	**GPU Memory (MB)**
1	73.51 ± 0.00	2439
2	68.48 ± 0.41	1583
4	66.68 ± 3.22	1161
8	55.11 ± 13.78	949
Ours	73.51 ± 0.00	949

- 提出逐渐增加使用channel的比例，同时逐渐剪枝op(根据分布)
- 每次在一定搜索步后，增加partial channel比例，超网会逐渐变宽。调整卷积权重：

$$g(j) = \begin{cases} j & j \leq n \\ \text{random sample from } \{1, 2, \ldots, n\} & j > n \end{cases} \tag{7}$$

$$\mathbf{U}^{(l)}_{o,i,h,w} = \mathbf{W}^{(l)}_{g(o),g(i),h,w}, \tag{8}$$

- 加宽超网后，根据dirichlet分布参数 β ，剪掉不重要的op，保持GPU memory不变

实验结果

Darts space

CIFAR-10

- CIFAR-10搜索中堆20cell, initial channel=36

Table 2: Comparison with state-of-the-art image classifiers on CIFAR-10.

Architecture	Test Error (%)	Params (M)	Search Cost (GPU days)	Search Method
DenseNet-BC (Huang et al., 2017)*	3.46	25.6	-	manual
NASNet-A (Zoph et al., 2018)	2.65	3.3	2000	RL
AmoebaNet-A (Real et al., 2019)	3.34 ± 0.06	3.2	3150	evolution
AmoebaNet-B (Real et al., 2019)	2.55 ± 0.05	2.8	3150	evolution
PNAS (Liu et al., 2018)*	3.41 ± 0.09	3.2	225	SMBO
ENAS (Pham et al., 2018)	2.89	4.6	0.5	RL
DARTS (1st) (Liu et al., 2019)	3.00 ± 0.14	3.3	0.4	gradient
DARTS (2nd) (Liu et al., 2019)	2.76 ± 0.09	3.3	1.0	gradient
SNAS (moderate) (Xie et al., 2019)	2.85 ± 0.02	2.8	1.5	gradient
GDAS (Dong & Yang, 2019)	2.93	3.4	0.3	gradient
BayesNAS (Zhou et al., 2019)	2.81 ± 0.04	3.4	0.2	gradient
ProxylessNAS (Cai et al., 2019)†	2.08	5.7	4.0	gradient
PARSEC (Casale et al., 2019)	2.81 ± 0.03	3.7	1	gradient
P-DARTS (Chen et al., 2019)	2.50	3.4	0.3	gradient
PC-DARTS (Xu et al., 2020)	2.57 ± 0.07	3.6	0.1	gradient
SDARTS-ADV (Chen & Hsieh, 2020)	2.61 ± 0.02	3.3	1.3	gradient
GAEA + PC-DARTS (Li et al., 2020)	2.50 ± 0.06	3.7	0.1	gradient
DrNAS (without progressive learning)	2.54 ± 0.03	4.0	0.4‡	gradient
DrNAS	2.46 ± 0.03	4.1	0.6‡	gradient

* Obtained without cutout augmentation.
† Obtained on a different space with PyramidNet (Han et al., 2017) as the backbone.
‡ Recorded on a single GTX 1080Ti GPU.

Imagenet

- 堆叠14cells，inital channel=48

Table 3: Comparison with state-of-the-art image classifiers on ImageNet in the mobile settin

Architecture	Test Error(%) top-1	Test Error(%) top-5	Params (M)	Search Cost (GPU days)	Search Method
Inception-v1 (Szegedy et al., 2015)	30.1	10.1	6.6	-	manual
MobileNet (Howard et al., 2017)	29.4	10.5	4.2	-	manual
ShuffleNet 2× (v1) (Zhang et al., 2018)	26.4	10.2	~ 5	-	manual
ShuffleNet 2× (v2) (Ma et al., 2018)	25.1	-	~ 5	-	manual
NASNet-A (Zoph et al., 2018)	26.0	8.4	5.3	2000	RL
AmoebaNet-C (Real et al., 2019)	24.3	7.6	6.4	3150	evolution
PNAS (Liu et al., 2018)	25.8	8.1	5.1	225	SMBO
MnasNet-92 (Tan et al., 2019)	25.2	8.0	4.4	-	RL
DARTS (2nd) (Liu et al., 2019)	26.7	8.7	4.7	1.0	gradient
SNAS (mild) (Xie et al., 2019)	27.3	9.2	4.3	1.5	gradient
GDAS (Dong & Yang, 2019)	26.0	8.5	5.3	0.3	gradient
BayesNAS (Zhou et al., 2019)	26.5	8.9	3.9	0.2	gradient
DSNAS (Hu et al., 2020)†	25.7	8.1	-	-	gradient
ProxylessNAS (GPU) (Cai et al., 2019)†	24.9	7.5	7.1	8.3	gradient
PARSEC (Casale et al., 2019)	26.0	8.4	5.6	1	gradient
P-DARTS (CIFAR-10) (Chen et al., 2019)	24.4	7.4	4.9	0.3	gradient
P-DARTS (CIFAR-100) (Chen et al., 2019)	24.7	7.5	5.1	0.3	gradient
PC-DARTS (CIFAR-10) (Xu et al., 2020)	25.1	7.8	5.3	0.1	gradient
PC-DARTS (ImageNet) (Xu et al., 2020)†	24.2	7.3	5.3	3.8	gradient
GAEA + PC-DARTS (Li et al., 2020)†	24.0	7.3	5.6	3.8	gradient
DrNAS (without progressive learning)†	24.2	7.3	5.2	3.9	gradient
DrNAS†	23.7	7.1	5.7	4.6	gradient

† The architecture is searched on ImageNet, otherwise it is searched on CIFAR-10 or CIFAR-100.

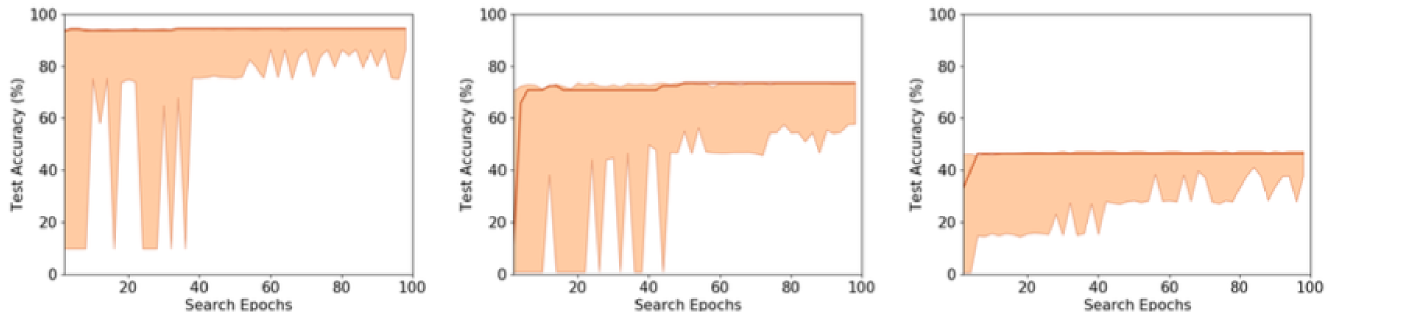
NAS-bench-201

- 4个不同随机种子实验

Table 4: Comparison with state-of-the-art NAS methods on NAS-Bench-201.

Method	CIFAR-10 validation	CIFAR-10 test	CIFAR-100 validation	CIFAR-100 test	ImageNet-16-120 validation	ImageNet-16-120 test
ResNet (He et al., 2016)	90.83	93.97	70.42	70.86	44.53	43.63
Random (baseline)	90.93 ± 0.36	93.70 ± 0.36	70.60 ± 1.37	70.65 ± 1.38	42.92 ± 2.00	42.96 ± 2.15
RSPS (Li & Talwalkar, 2019)	84.16 ± 1.69	87.66 ± 1.69	45.78 ± 6.33	46.60 ± 6.57	31.09 ± 5.65	30.78 ± 6.12
Reinforce (Zoph et al., 2018)	91.09 ± 0.37	93.85 ± 0.37	70.05 ± 1.67	70.17 ± 1.61	43.04 ± 2.18	43.16 ± 2.28
ENAS (Pham et al., 2018)	39.77 ± 0.00	54.30 ± 0.00	10.23 ± 0.12	10.62 ± 0.27	16.43 ± 0.00	16.32 ± 0.00
DARTS (1st) (Liu et al., 2019)	39.77 ± 0.00	54.30 ± 0.00	38.57 ± 0.00	38.97 ± 0.00	18.87 ± 0.00	18.41 ± 0.00
DARTS (2nd) (Liu et al., 2019)	39.77 ± 0.00	54.30 ± 0.00	38.57 ± 0.00	38.97 ± 0.00	18.87 ± 0.00	18.41 ± 0.00
GDAS (Dong & Yang, 2019)	90.01 ± 0.46	93.23 ± 0.23	24.05 ± 8.12	24.20 ± 8.08	40.66 ± 0.00	41.02 ± 0.00
SNAS (Xie et al., 2019)	90.10 ± 1.04	92.77 ± 0.83	69.69 ± 2.39	69.34 ± 1.98	42.84 ± 1.79	43.16 ± 2.64
DSNAS (Hu et al., 2020)	89.66 ± 0.29	93.08 ± 0.13	30.87 ± 16.40	31.01 ± 16.38	40.61 ± 0.09	41.07 ± 0.09
PC-DARTS (Xu et al., 2020)	89.96 ± 0.15	93.41 ± 0.30	67.12 ± 0.39	67.48 ± 0.89	40.83 ± 0.08	41.31 ± 0.22
DrNAS	$\mathbf{91.55 \pm 0.00}$	$\mathbf{94.36 \pm 0.00}$	$\mathbf{73.49 \pm 0.00}$	$\mathbf{73.51 \pm 0.00}$	$\mathbf{46.37 \pm 0.00}$	$\mathbf{46.34 \pm 0.00}$
optimal	91.61	94.37	73.49	73.51	46.77	47.31

Exploration and exploitation



(a) CIFAR-10 (b) CIFAR-100 (c) ImageNet16-120

Figure 1: Accuracy range (min-max) of the 100 sampled architectures. Note that the solid line is our derived architecture according to the Dirichlet mean as described in Section 2.2.

- 在搜索过程中每个epoch依照分布采样100个 θ 。再对这100个 θ 分别argmax获得子网
- 早期采样的架构准确率分布很散，代表探索。
- 晚期采样的架构准确率范围变窄，代表开发

消融实验

Table 5: Test accuracy of the searched architecture with different λs on NAS-Bench-201 (CIFAR-10). $\lambda = 1e^{-3}$ is what we used for all of our experiments.

λ	0	$5e^{-4}$	$1e^{-3}$	$5e^{-3}$	$1e^{-2}$	$1e^{-1}$	1
Accuracy	93.78	94.01	94.36	94.36	94.36	93.76	93.76

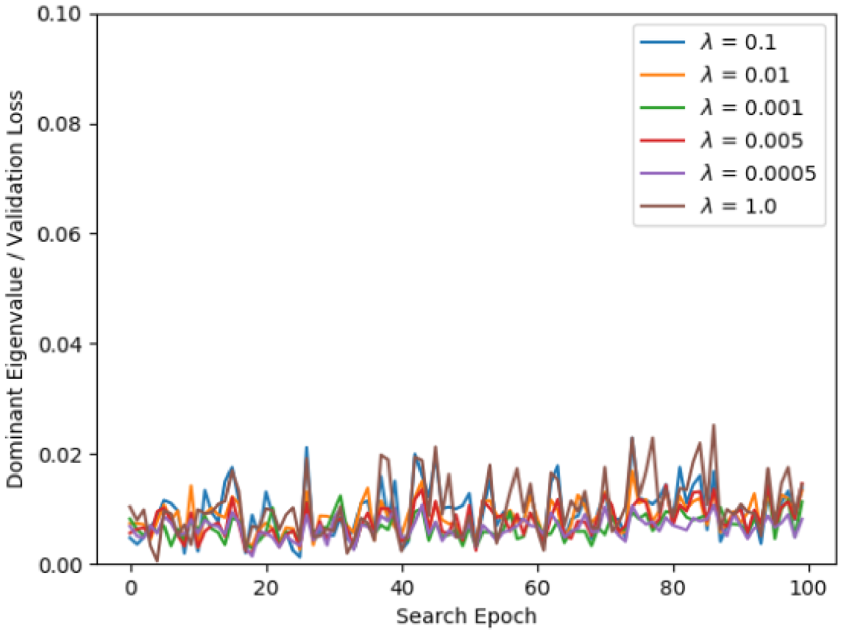


Figure 5: Trajectory of the Hessian norm under various λs on NAS-Bench-201 when searching with CIFAR-10 (best viewed in color).