DARTS for Inverse Problems: a Study on Hyperparameter Sensitivity

Jonas Geiping, Jovita Lukasik, Margret Keuper, Michael Moeller

Motivation/Conclusion

Neural Architecture Search (NAS) is the automatic search of novel architectures. To avoid computational costs during the search, differentiable architecture search [1], **DARTS**, proposes a continuous relaxation of the search problem. The increasing usage of DARTS for NAS is based on its effectiveness via its weight-sharing one shot paradigm resulting in promising results for image classification.

We apply DARTS to **inverse problems** in a systematic case study which allows us to analyze these potential benefits in a controlled manner.
We show that:

- DARTS can be extended to from classification to reconstruction
- **But**: Fundamental difficulty in the evaluation:
  The estimated shared weights network performance is not well correlated with the performance of the final architectures.
  ↦ The DARTS relaxation is too loose

We conclude the necessity to

- report the results of any DARTS-based methods from several runs along with its underlying performance statistics and
- show the correlation between the training and final architecture performance.

Basic on DARTS

Our sequential architecture from fig.1 consists of N nodes $x^{(j)}$, where $x^{(0)}$ represents the inputs data and the results $x^{(j+1)}$ of any layer is computed by applying some operation $o^{(j)}$ to the predecessor node $x^{(j)}$.

$$x^{(j+1)} = o^{(j)}(x^{(j)}, \theta^{(j)}),$$

$\theta^{(j)}$ are the learnable parameters of operation $o^{(j)}$. To determine which operation is most suitable one searches over the continuous relaxation

$$o^{(j)} = \sum_{t=1}^{T} \beta_{o_t}^{(j)} o_t, \qquad \beta_{o_t}^{(j)} = \frac{\exp(\alpha_{o_t}^{(j)})}{\sum_{t'=1}^{T} \exp(\alpha_{o_{t'}}^{(j)})}$$

where $\alpha = (\alpha_{o_t}^{(j)})$ are the architecture parameters. The optimization is relaxed to the soft-max of alpha.
DARTS formulates this search as a bi-level optimization problem in which the network parameters θ and the architecte parameters α are jointly optimized on the training and validation set:

$$\begin{aligned} &\min_{\alpha} \mathcal{L}_{val}(\theta(\alpha), \alpha) \\ &\text{s.t. } \theta(\alpha) \in \arg\min_{\theta} \mathcal{L}_{train}(\theta, \alpha), \end{aligned} \tag{4}$$

The optimization is done by approximating (4) by one (or zero) iterations of gradient descent.
The discrete architecture is obtained by choosing $\hat{o}^{(j)} = \text{argmax}_{o_t} \alpha_{o_t}^{(j)}$ for each node. The final architecture is retrained from scratch:
Takeaway: The fundamental assumption of darts is that the performance of the final network is highly correlated with the performance of the relaxed DARTS approach.

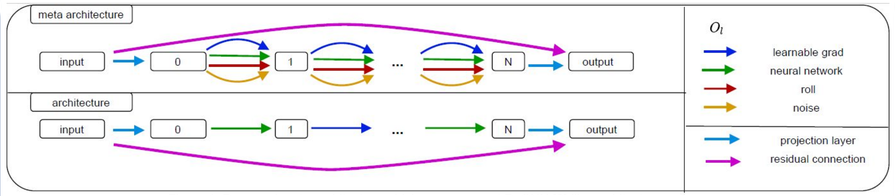

Figure 1: Proposed meta-architecture which can represent DnCNN [2]-like architectures.

Setup

- We investigate the meta-architecture shown in fig. 1
- The search for the optimal architecture that can be defined as a sequence of operations selected from a predefined set
  O_l = {learnable gradient descent, 2-layer-CNN, roll, noise}
  This set contains two beneficial layers, and two harmful ones.

Goal: A good differentiable architecture search algorithm should reliably find the optimal operations, even when presented with sub-optimal choices.

Dataset: *One-dimensional regression* of cosine waves of varying magnitude, amplitude and offset, distorted by Gaussian noise, blurring and subsampling.

↦ This setup allows us to mirror higher-dimensional datasets in an environment where extensive evaluations are possible.

Discussion

DARTS Takeaway:
For maximal performance DARTS can be used as a component in a large search that proposes trial architectures.
For average performance, and immediate performance, with a single run, we should maximize the correlation.

Optimal Architecture Takeaway: The optimal architectures found a hybrid version that mix learnable grad- and net- operations. The optimal mixed architectures have a remarkable advantage over the best plain architecture.

Evaluations

Method	Data Formation	Architecture Validation (PSNR)		
		Max.	Mean	Med.
DARTS (good ops.)	Blur	23.46	21.56	21.60
DARTS (all ops.)	Blur	22.86	15.64	18.57
Random (all ops.)	Blur	20.86	9.45	8.10
Learnable Grad. only	Blur	17.45	16.36	16.49
Nets only	Blur	21.63	19.45	20.71
DARTS (good ops.)	Downsampling	18.03	16.36	16.66
DARTS (all ops.)	Downsampling	18.01	15.39	16.12
Random (all ops.)	Downsampling	13.78	5.08	4.31
Learnable Grad. only	Downsampling	14.35	13.24	13.55
Nets only	Downsampling	16.92	13.13	14.05

Table 1: Architecture validation PSNR values found for 1D inverse problems. Shown is the maximal, mean and median PSNR over 100 trials.

Evaluating Robustness

1) Plainly, DARTS does work as shown in the tab. 1 (*Max*)
2) If we turn to towards *Mean*: DARTS with only “good” operations is stable, default DARTS with all operations suffers
-> DARTS performs on *average* worse than the *maximum* over 100 random architectures
Running DARTS for as many trials as possible achieves best benchmark results. BUT this is not the original goal, which is to find a good architecture within a single optimization of the weight-sharing DARTS problem,

Correlation of Architecture and DARTS Performance

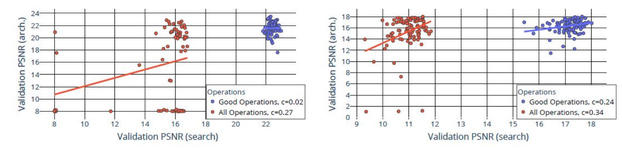

Figure 2: Scatter plot showing architecture PSNR (y-axis) plotted against 1-shot validation PSNR (i.e. the validation performance on the DARTS objective). Left: Blur. Right: Downsampling.

Fig. 2 takes a closer look on the results in tab. 1. The correlation between the validation performance of the one-shot architecture and the “true” architecture is a fundamental assumption of DARTS. Whereas these plots show that these two measurements are highly independent.
We even observe three direct problems:

1) DARTS fails directly
2) DARTS works but does not predict a useful architecture
3) DARTS does predict a useful architecture, but is unrelated to its search performance

References

[1] Hanxiao Liu, Karen Simonyan, and Yiming Yang. “DARTS: Differentiable Architecture Search” ICLR, 2019.

[2] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising”. IEEE Transactions on Image Processing, 2017.