Differentiable Architecture Search

Alexander Becker

Seminar: Recent Trends in Automated Machine Learning

Technical University of Munich

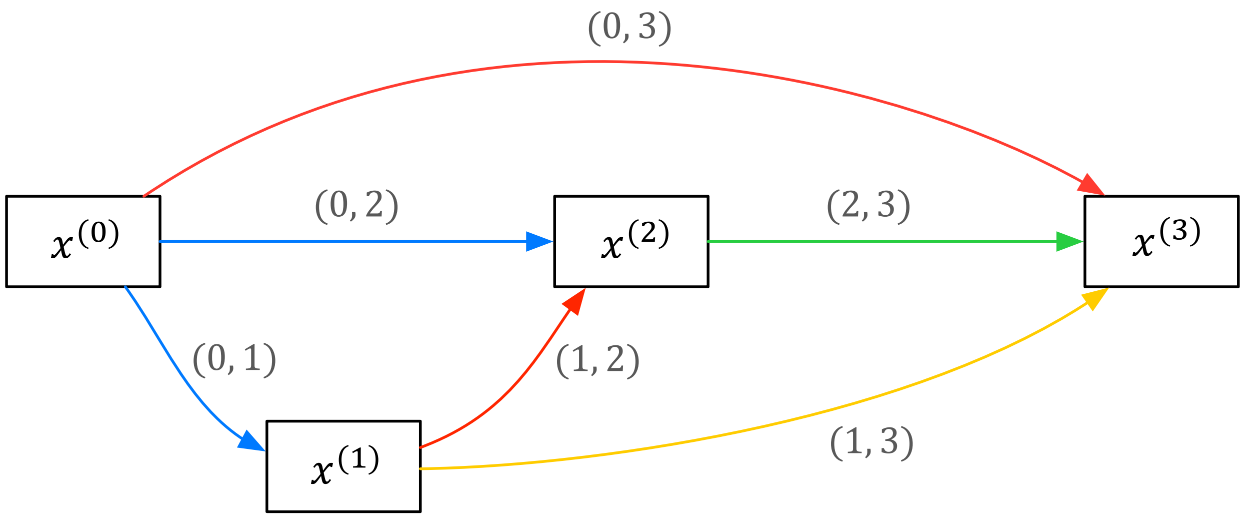
July 4, 2019

Overview

- Increasing interest in automatic architecture discovery
- Most approaches are computationally expensive, e.g. on ImageNet/CIFAR-10:
  - 2000 GPU days of reinforcement learning by Zoph et al. (2017)
  - 3150 GPU days of evolution by Real et al. (2018)
- Problem: Optimization over discrete domain, requiring many evaluations
- Liu et al. (2018) propose continuous relaxation of the search space

Search Space

- Search for building blocks instead of entire network architecture
- Building blocks ("cells") can then be stacked/connected recurrently
- Cell is represented as a directed acyclic graph with latent representations as nodes:



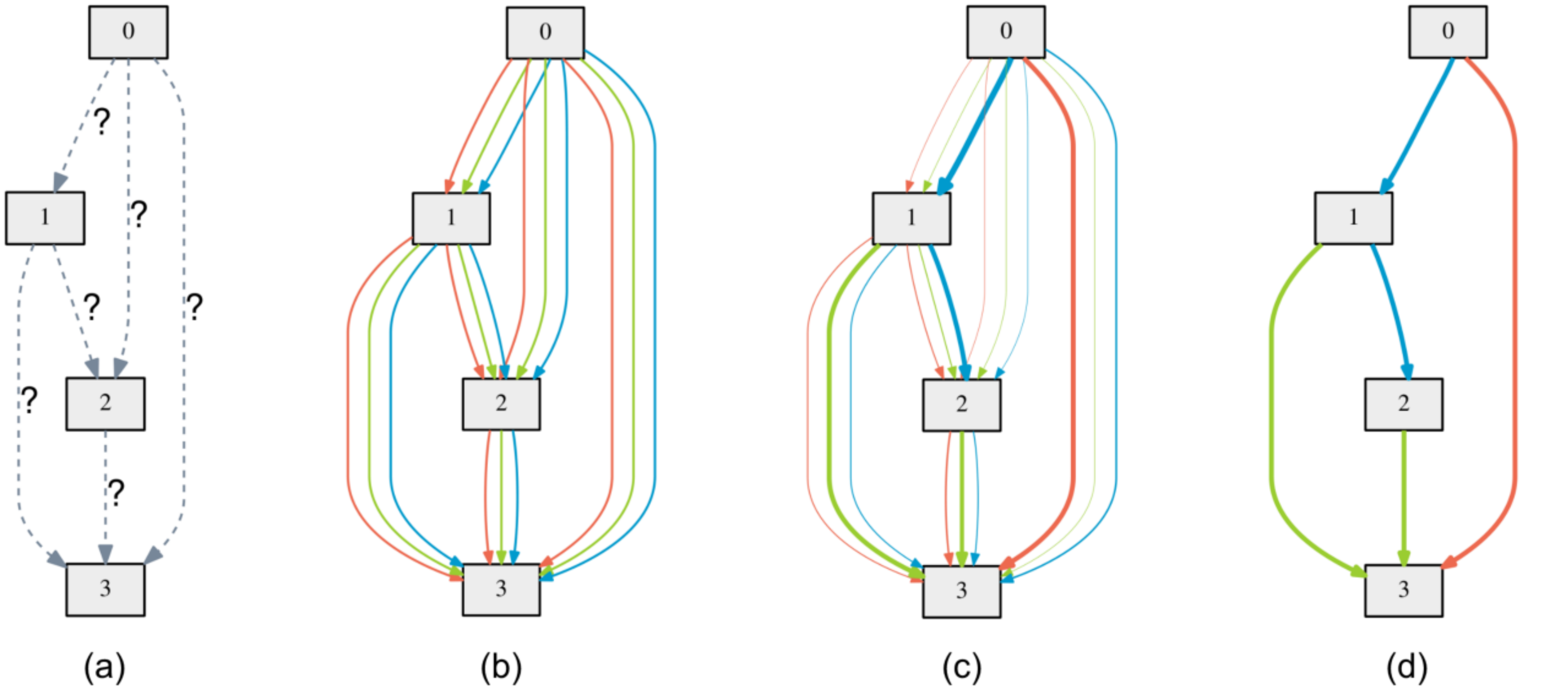
Continuous Relaxation

- Let O be a set of possible operations
- Relax each edge to a softmax weighted mixture of operations from O

$$\bar{o}^{(i,j)}(x) = \sum_{o \in \mathcal{O}} \frac{\exp(\alpha_o^{(i,j)})}{\sum_{o' \in \mathcal{O}} \exp(\alpha_{o'}^{(i,j)})} o(x)$$

- Parametrization by $\alpha = \{\alpha^{(i,j)}\}$
- After the search, $\bar{o}^{(i,j)}$ can be discretized by $o^{(i,j)} = \mathrm{argmax}_{o \in \mathcal{O}} \ \alpha_o^{(i,j)}$

Continuous Relaxation



Optimization

- Joint optimization of model weights and architecture parameters
- Optimize validation loss using gradient descent
- Bi-level optimization problem:

$$\begin{aligned} \min_{\alpha} \quad & \mathcal{L}_{val}(w^*(\alpha), \alpha) \\ \text{s.t.} \quad & w^*(\alpha) = \text{argmin}_w \ \mathcal{L}_{train}(w, \alpha) \end{aligned}$$

- Gradient can be expressed as

$$\begin{aligned} & \nabla_\alpha \mathcal{L}_{val}(w^*(\alpha), \alpha) \\ \approx & \nabla_\alpha \mathcal{L}_{val}(w - \xi \nabla_w \mathcal{L}_{train}(w, \alpha), \alpha) \end{aligned}$$

Optimization

Algorithm 1: DARTS – Differentiable Architecture Search

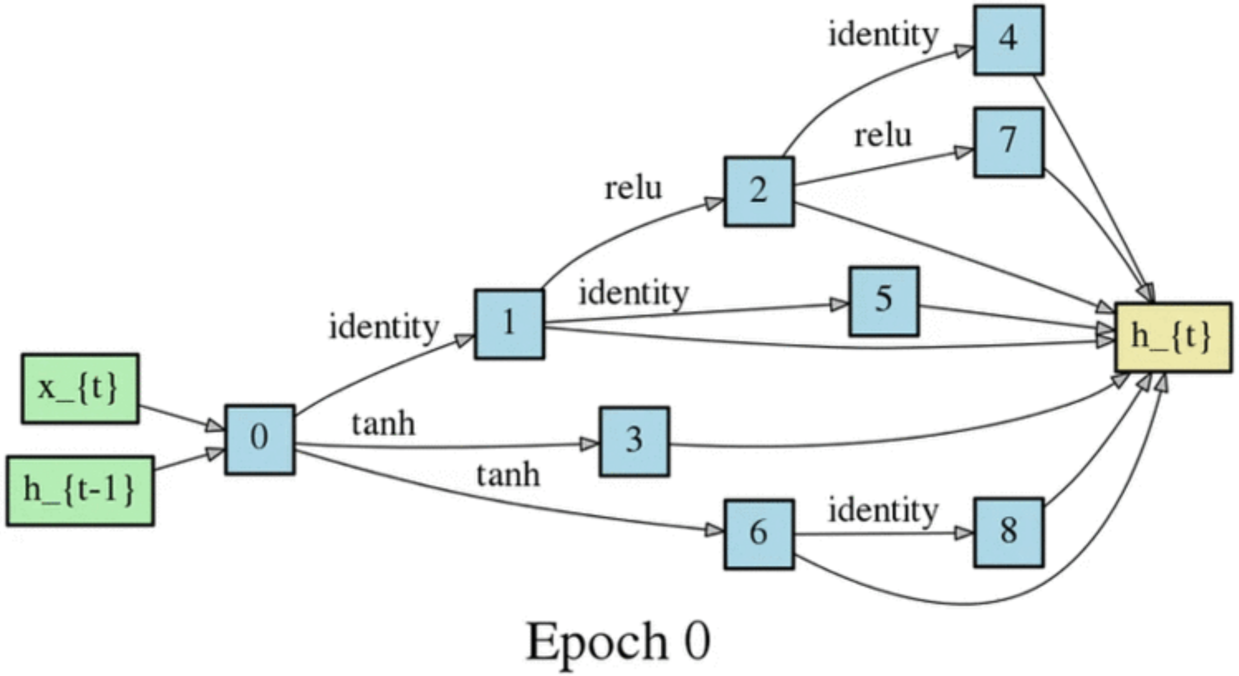
Create a mixed operation $\bar{o}^{(i,j)}$ parametrized by $\alpha^{(i,j)}$ for each edge (i, j)

while *not converged* **do**

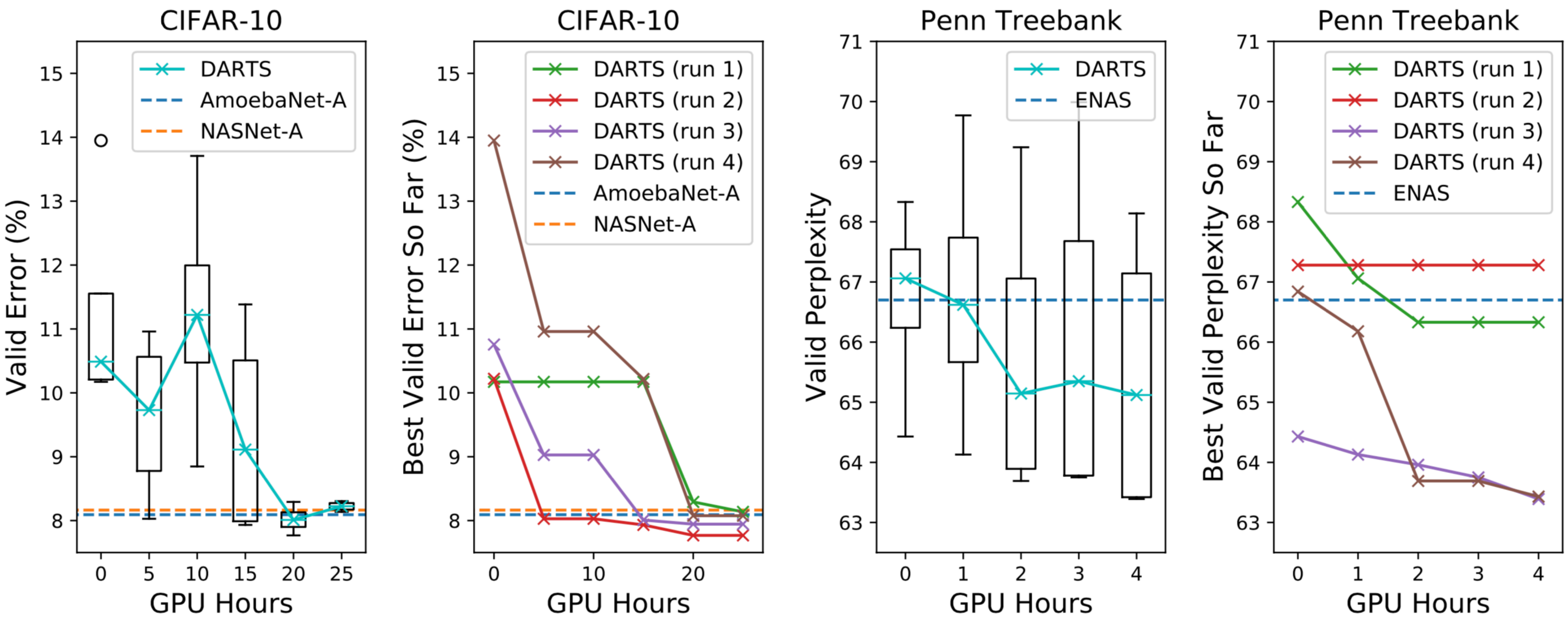
1. Update architecture α by descending $\nabla_\alpha \mathcal{L}_{val}(w - \xi \nabla_w \mathcal{L}_{train}(w, \alpha), \alpha)$
   ($\xi = 0$ if using first-order approximation)
2. Update weights w by descending $\nabla_w \mathcal{L}_{train}(w, \alpha)$

Derive the final architecture based on the learned α.

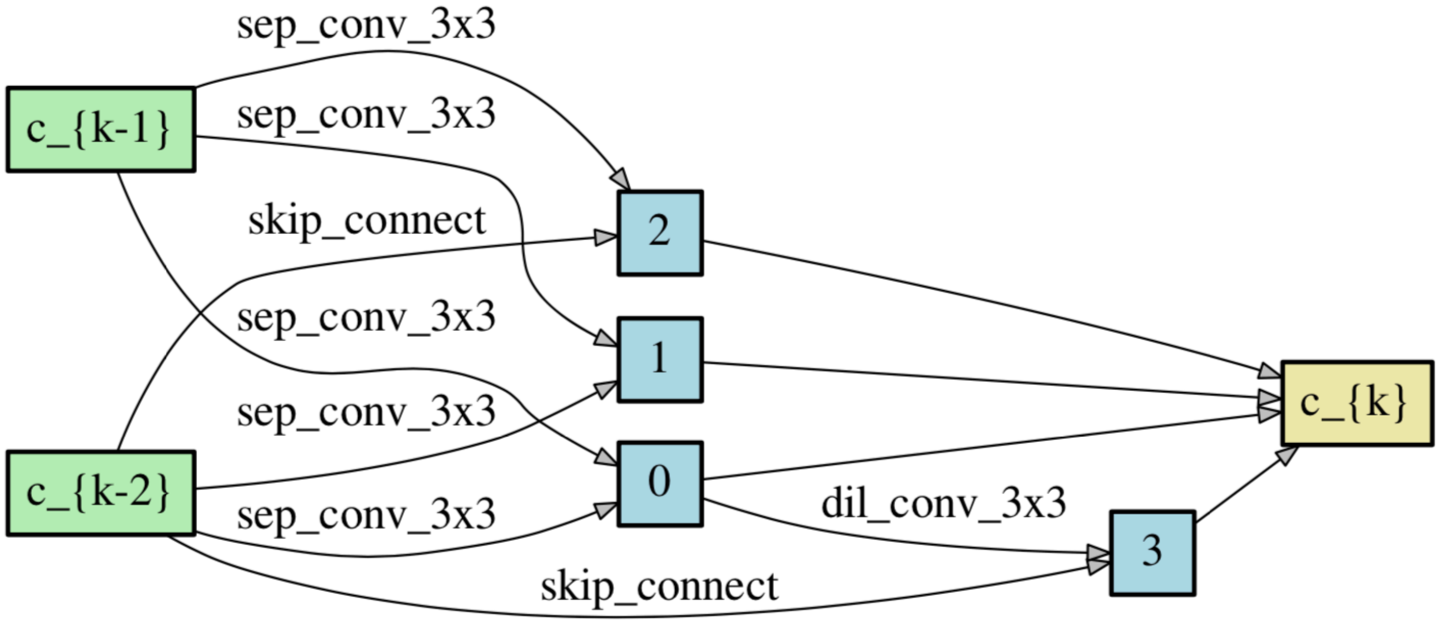
Optimization



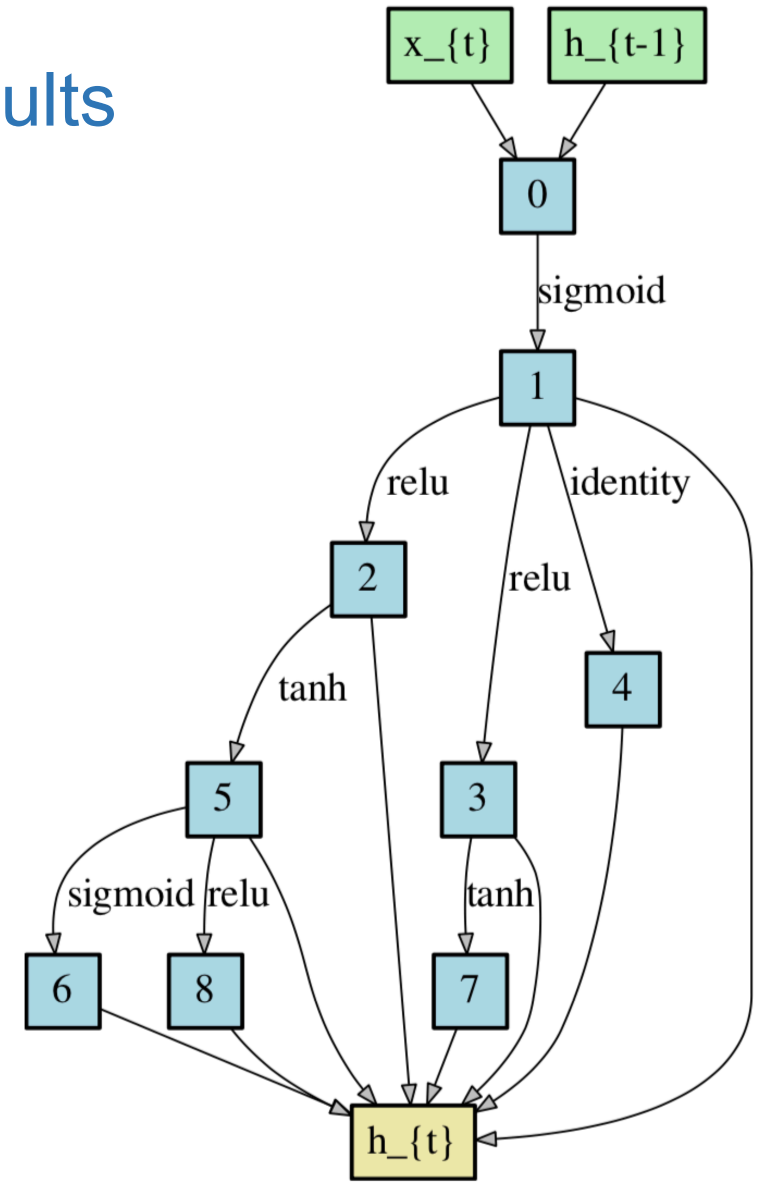
Experiments & Results



Experiments & Results



Experiments & Results



Experiments & Results

Architecture	Test Error (%)	Params (M)	Search Cost (GPU days)	#ops	Search Method
DenseNet-BC (Huang et al., 2017)	3.46	25.6	–	–	manual
NASNet-A + cutout (Zoph et al., 2018)	2.65	3.3	2000	13	RL
NASNet-A + cutout (Zoph et al., 2018)†	2.83	3.1	2000	13	RL
BlockQNN (Zhong et al., 2018)	3.54	39.8	96	8	RL
AmoebaNet-A (Real et al., 2018)	3.34 ± 0.06	3.2	3150	19	evolution
AmoebaNet-A + cutout (Real et al., 2018)†	3.12	3.1	3150	19	evolution
AmoebaNet-B + cutout (Real et al., 2018)	2.55 ± 0.05	2.8	3150	19	evolution
Hierarchical evolution (Liu et al., 2018b)	3.75 ± 0.12	15.7	300	6	evolution
PNAS (Liu et al., 2018a)	3.41 ± 0.09	3.2	225	8	SMBO
ENAS + cutout (Pham et al., 2018b)	2.89	4.6	0.5	6	RL
ENAS + cutout (Pham et al., 2018b)*	2.91	4.2	4	6	RL
Random search baseline‡ + cutout	3.29 ± 0.15	3.2	4	7	random
DARTS (first order) + cutout	3.00 ± 0.14	3.3	1.5	7	gradient-based
DARTS (second order) + cutout	2.76 ± 0.09	3.3	4	7	gradient-based

Experiments & Results

Architecture	Perplexity valid	Perplexity test	Params (M)	Search Cost (GPU days)	#ops	Search Method
Variational RHN (Zilly et al., 2016)	67.9	65.4	23	–	–	manual
LSTM (Merity et al., 2018)	60.7	58.8	24	–	–	manual
LSTM + skip connections (Melis et al., 2018)	60.9	58.3	24	–	–	manual
LSTM + 15 softmax experts (Yang et al., 2018)	58.1	56.0	22	–	–	manual
NAS (Zoph & Le, 2017)	–	64.0	25	1e4 CPU days	4	RL
ENAS (Pham et al., 2018b)*	68.3	63.1	24	0.5	4	RL
ENAS (Pham et al., 2018b)†	60.8	58.6	24	0.5	4	RL
Random search baseline‡	61.8	59.4	23	2	4	random
DARTS (first order)	60.2	57.6	23	0.5	4	gradient-based
DARTS (second order)	58.1	55.7	23	1	4	gradient-based

Conclusion

- DARTS proves feasibility of architecture search using gradient descent
- More efficient than non-differentiable approaches and reaches similar performance
- Simple and powerful
- Part of the one-shot family of algorithm search
- Possibly large gap between continuous solution and derived discrete architecture
- Does not find novel architectures in a broad sense
- Bi-level solving algorithm is not mathematically derived but rather a heuristic

Questions?