AutoML: Neural Architecture Search (NAS)

Speedup Techniques

Bernd Bischl Frank Hutter Lars Kotthoff
Marius Lindauer Joaquin Vanschoren

Overview of NAS Speedup Methods

- Multi-fidelity optimization
- Learning curve prediction
- Meta-learning across datasets
- Network morphisms & weight inheritance
- Weight sharing & the one-shot model

NAS Speedup Technique 1: Multi-fidelity optimization

- Analogous to multi-fidelity optimization in HPO
  - Many evaluations for cheaper fidelities (less epochs, smaller datasets, down-sampled images, shallower networks, etc)
  - Fewer evaluations necessary for more expensive fidelities

NAS Speedup Technique 1: Multi-fidelity optimization

- Analogous to multi-fidelity optimization in HPO
  - Many evaluations for cheaper fidelities (less epochs, smaller datasets, down-sampled images, shallower networks, etc)
  - Fewer evaluations necessary for more expensive fidelities

- Compatible with any blackbox optimization method
  - Using random search: ASHA [Li and Talwalkar. 2019]
  - Using Bayesian optimization: BOHB [Zela et al. 2018]
  - Using differential evolution: DEHB [Awad et al. under review]
  - Using regularized evolution: progressive dynamic hurdles [So et al. 2019]

NAS Speedup Technique 1: Multi-fidelity optimization

- Analogous to multi-fidelity optimization in HPO
  - Many evaluations for cheaper fidelities (less epochs, smaller datasets, down-sampled images, shallower networks, etc)
  - Fewer evaluations necessary for more expensive fidelities

- Compatible with any blackbox optimization method
  - Using random search: ASHA [Li and Talwalkar. 2019]
  - Using Bayesian optimization: BOHB [Zela et al. 2018]
  - Using differential evolution: DEHB [Awad et al. under review]
  - Using regularized evolution: progressive dynamic hurdles [So et al. 2019]

- Often used for joint optimization of architecture & hyperparameters
  - Auto-Pytorch [Mendoza et al. 2019; Zimmer et al. 2020]
  - "Auto-RL" [Runge et al. 2019]

NAS Speedup Technique 2: Learning Curve Prediction

- Analogous to learning curve prediction in HPO
  - Observe initial learning curve and predict performance at the end
  - Can use features of the architecture as input (just like hyperparameters as inputs)

NAS Speedup Technique 2: Learning Curve Prediction

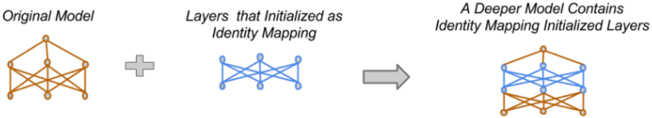
- Analogous to learning curve prediction in HPO
  - Observe initial learning curve and predict performance at the end
  - Can use features of the architecture as input (just like hyperparameters as inputs)
- Often used for joint optimization of architecture & hyperparameters
- Compatible with any blackbox optimization method
  - Using random search and Bayesian optimization: [Domhan et al. 2015]
  - Using reinforcement learning: [Baker et al. 2018]

NAS Speedup Technique 3: Meta-Learning

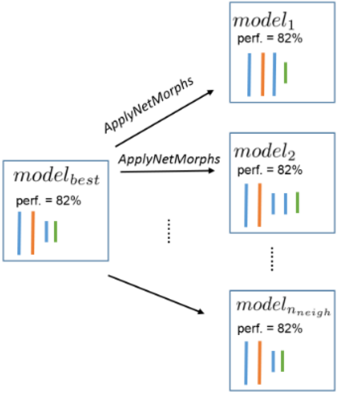
- Lots of work on meta-learning for HPO
- Only little work on meta-learning for NAS
  - Find a set of good architectures to initialize BOHB in Auto-Pytorch [Zimmer et al. 2020]
  - Learn RL agent's policy network on previous datasets [Wong et al. 2018]
  - Learn neural architecture that can be quickly adapted [Lian et al. 2019; Elsken et al. 2019]

NAS Speedup Technique 4: Network Morphisms

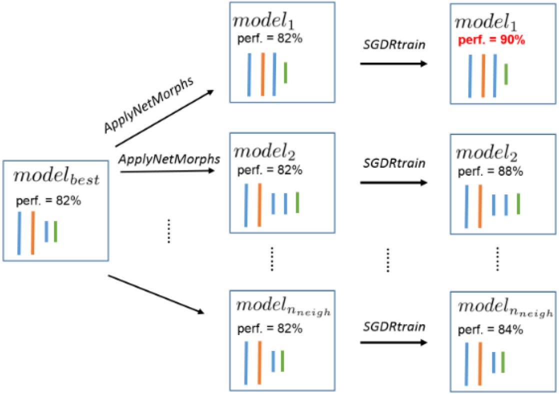
- Network Morphisms [Chen et al. 2016; Wei et al. 2016; Cai et al. 2017]
  - Change the network structure, but not the modelled function
  - I.e., for every input the network yields the same output as before applying the network morphisms operations
  - Examples: “Net2DeeperNet”, “Net2WiderNet”, etc.


Original Model
Layers that Initialized as Identity Mapping
A Deeper Model Contains Identity Mapping Initialized Layers

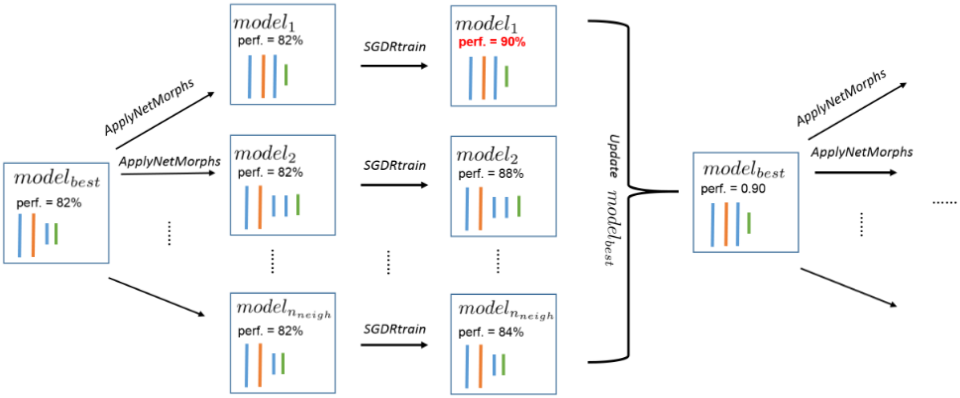
Network Morphisms Allow Efficient Moves in Architecture Space



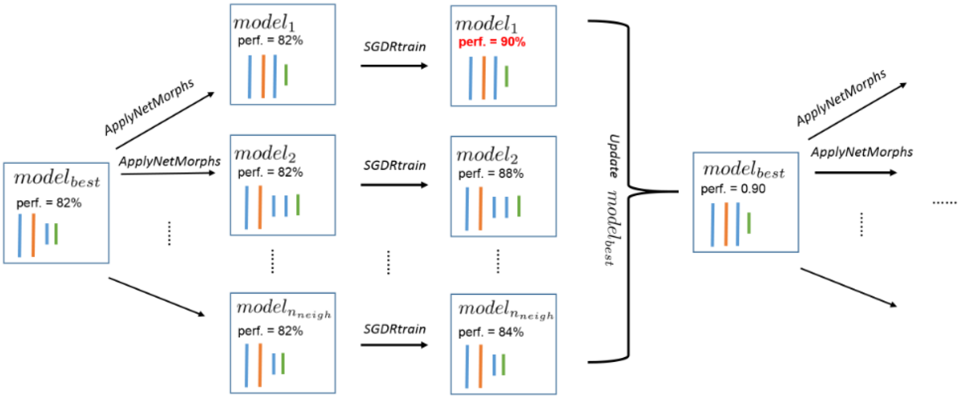
Network Morphisms Allow Efficient Moves in Architecture Space



Network Morphisms Allow Efficient Moves in Architecture Space



Network Morphisms Allow Efficient Moves in Architecture Space

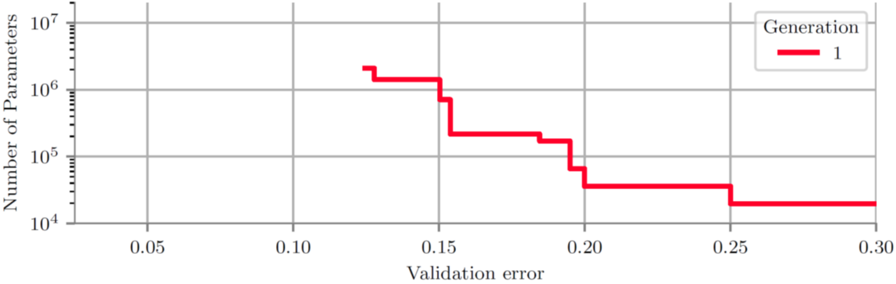


Weight inheritance avoids expensive retraining from scratch

[Real et al. 2017; Cai et al. 2018; Elsken et al. 2017; Cortes et al. 2017; Cai et al. 2018; Elsken et al. 2019]

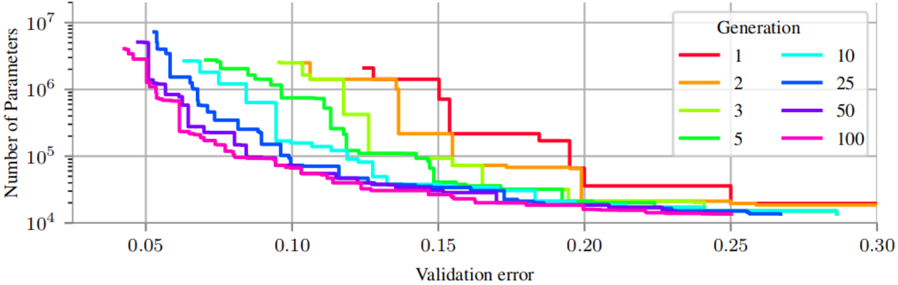
Network Morphisms for Multi-objective NAS [Elsken et al. 2019]

- To trade off error vs. resource consumption (e.g, #parameters):
  - Maintain a Pareto front of the two objectives
  - Evolve a population of Pareto-optimal architectures over time



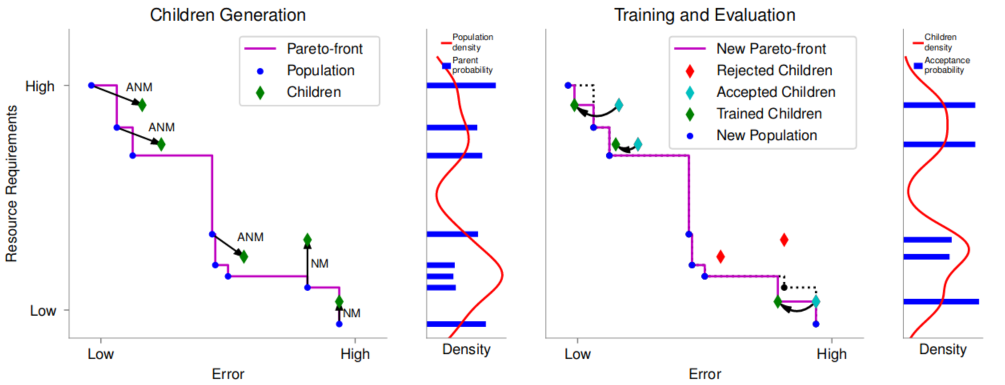
Network Morphisms for Multi-objective NAS [Elsken et al. 2019]

- To trade off error vs. resource consumption (e.g, #parameters):
  - Maintain a Pareto front of the two objectives
  - Evolve a population of Pareto-optimal architectures over time



Network Morphisms for Multi-objective NAS [Elsken et al. 2019]

- LEMONADE: Lamarckian Evolution for Multi-Objective Neural Architecture Design
- Weight inheritance through approximate morphisms (ANMs)
  - Dropping layers, dropping units within a layer, etc (function not preserved perfectly)



NAS Speedup Technique 5: Weight Sharing and One-shot Models

[Pham et al. 2018; Bender et al. 2018]

- All possible architectures are subgraphs of a large supergraph: the one-shot model

NAS Speedup Technique 5: Weight Sharing and One-shot Models

[Pham et al. 2018; Bender et al. 2018]

- All possible architectures are subgraphs of a large supergraph: the one-shot model
- Weights are shared between different architectures with common edges in the supergraph

NAS Speedup Technique 5: Weight Sharing and One-shot Models

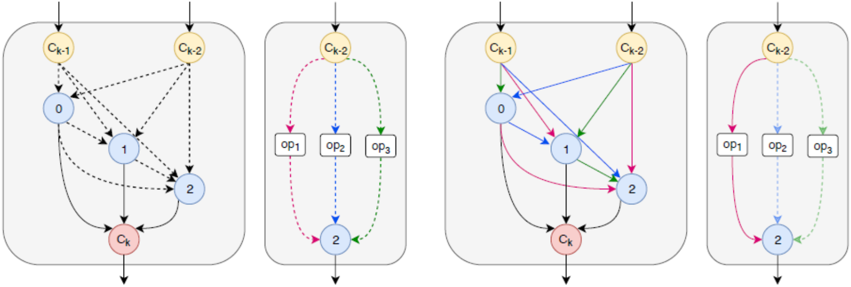
[Pham et al. 2018; Bender et al. 2018]

- All possible architectures are subgraphs of a large supergraph: the one-shot model
- Weights are shared between different architectures with common edges in the supergraph
- Search costs are reduced drastically since one only has to train a single model (the one-shot model).

NAS Speedup Technique 5: Weight Sharing and One-shot Models

[Pham et al. 2018; Bender et al. 2018]

- The one-shot model can be seen as a directed acyclic multigraph
  - ⇒ Nodes - latent representations.
  - ⇒ Edges (dashed) - operations.



(a) One-shot search **(b)** Final evaluation

- Architecture optimization problem: Find optimal path from the input to the output

Questions to Answer for Yourself / Discuss with Friends

- Repetition:
  List five methods to speed up NAS over blackbox approaches
- Repetition:
  Which speedup techniques directly carry over from HPO to NAS?
- Discussion:
  Why do network morphisms and the one-shot model only apply to NAS, and not to HPO?