NAS evaluation is frustratingly hard

Antoine Yang, Pedro M. Esperança, Fabio M. Carlucci

Huawei Noah's Ark Lab, London, UK

ICLR2020 Poster Session

Paper: https://arxiv.org/abs/1912.12522
Code: https://github.com/antoyang/NAS-Benchmark

Background

- **Neural Architecture Search (NAS):**

  Automated design of a neural architecture for a given task
- **3 main components:**

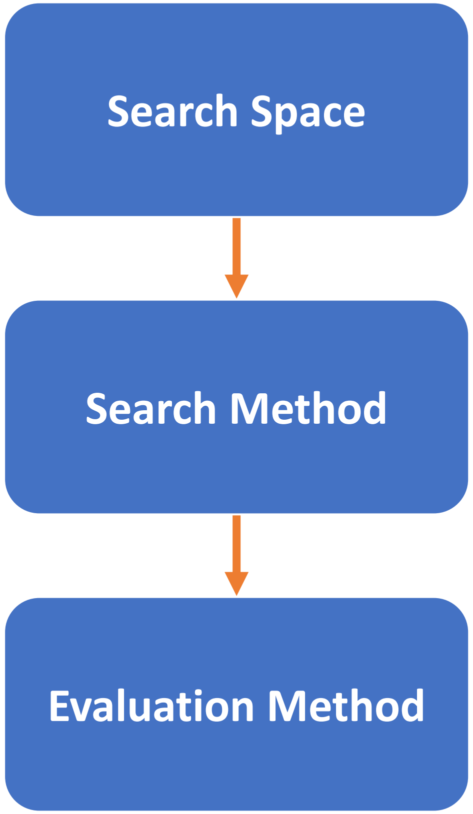
- A search space: set of architectures that can be found
- A search strategy: Random Search, Evolution, RL, Bayesian, Gradient-based …
- A training protocol: way we evaluate architectures

- **Issues related to the evaluation of search strategies:**

- Nowadays, most NAS methods fail to compare against an adequate baseline
- Unclarity about the contribution of each component to the final result

- **Our main contributions:**

- A benchmark of 8 NAS methods on 5 datasets with Random Sampling Baseline
- A study of the contribution of each component



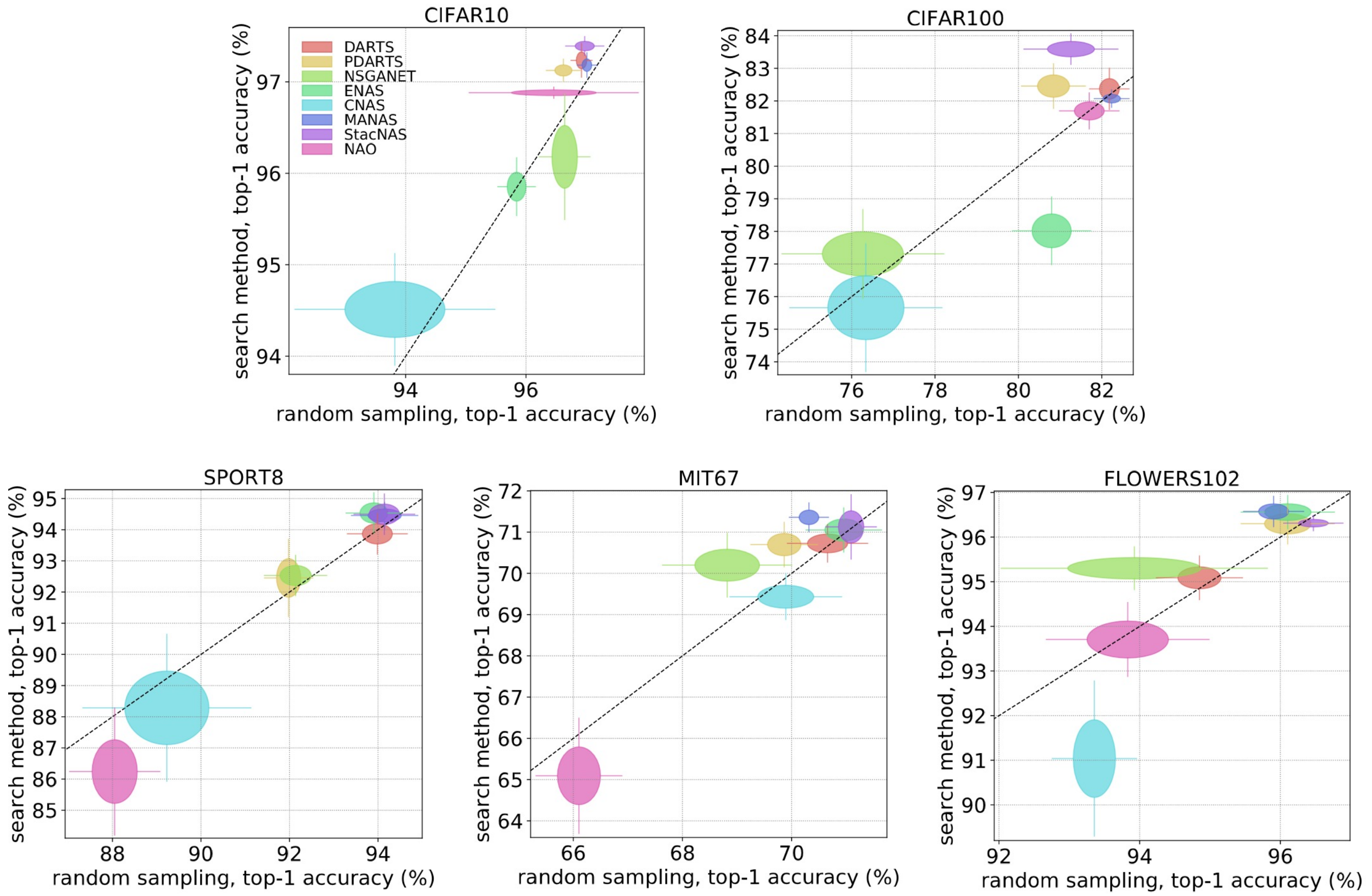
NAS Benchmark

- **Method selection:**
  8 *fast* open-source NAS methods
- **Random Sampling Baseline:**
  Randomly sample architectures from the method's search space (no search) and train them with the method's training protocol
- **Consistency, Generalization:**

- Average results over 8 runs
- Use a variety of 5 CV datasets

- **Results:**

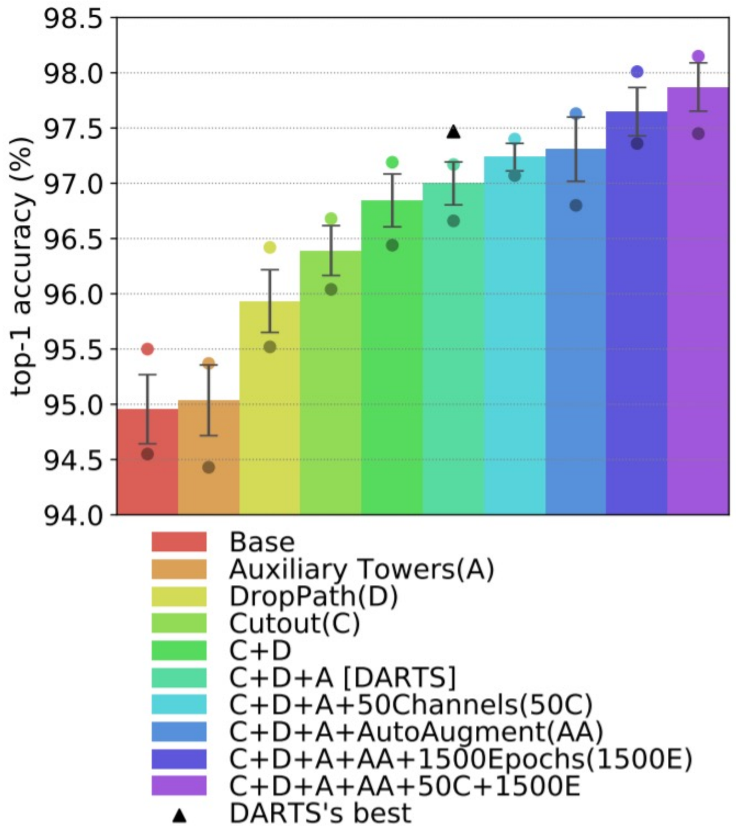
- The NAS methods barely beat this trivial baseline
- Substantial differences between the different random samplings



Comparison of training protocols

- **Goal:**
  Evaluate the importance of the different components in the final test accuracy
- **Methodology:**
  Train the same 8 randomly sampled architectures from DARTS search space with diverse protocols and report averaged results on CIFAR10
- **Results:**

- Significant differences between the different protocols: 3% gap between the worst and the best
- The best out of 8 random architectures with best protocol achieves 98.15% test accuracy (0.25% below state-of-the-art*)

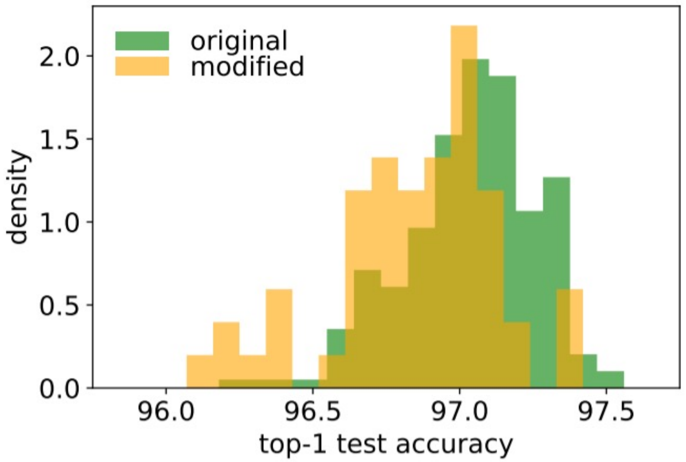


XNAS: Neural Architecture Search with Expert Advice, Niv Nayman et al, 2019

Study of DARTS' search space

- **Random Sampling Distribution:**

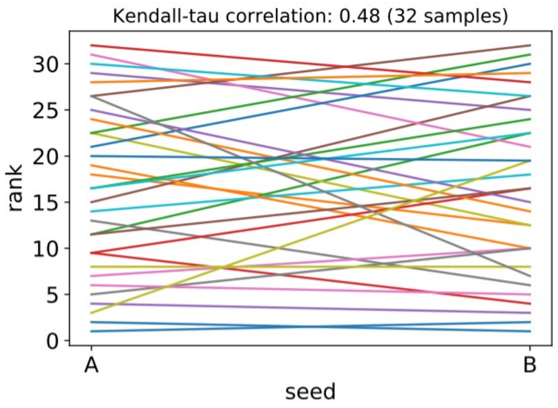
- Randomly sample 214 architectures in DARTS' search space and train them with DARTS' protocol
- Narrow accuracy range: average 97.03 ± 0.23, min 96.18, max 97.56



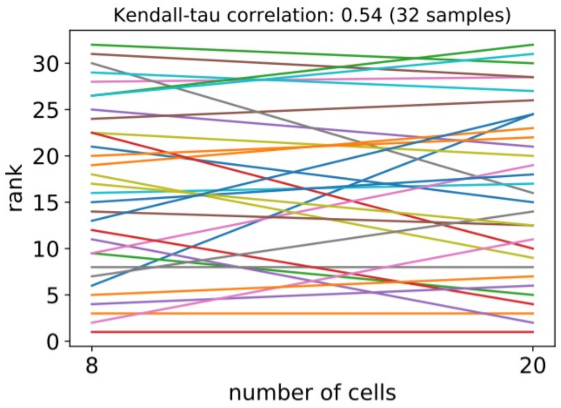
- **Importance of the Micro-Structure:**
  Similar study and observations with 56 architectures sampled from a modified search space based on (inefficient) vanilla convolutions

- **Importance of the Training Seed:**

- Randomly sample 32 architectures and train them with 2 different seeds
- Architectures' ranking heavily changes: Kendall Tau 0.48



- **Importance of the Depth Gap:**
  Similar study and observations with 32 architectures and 2 different number of cells: Kendall Tau 0.54

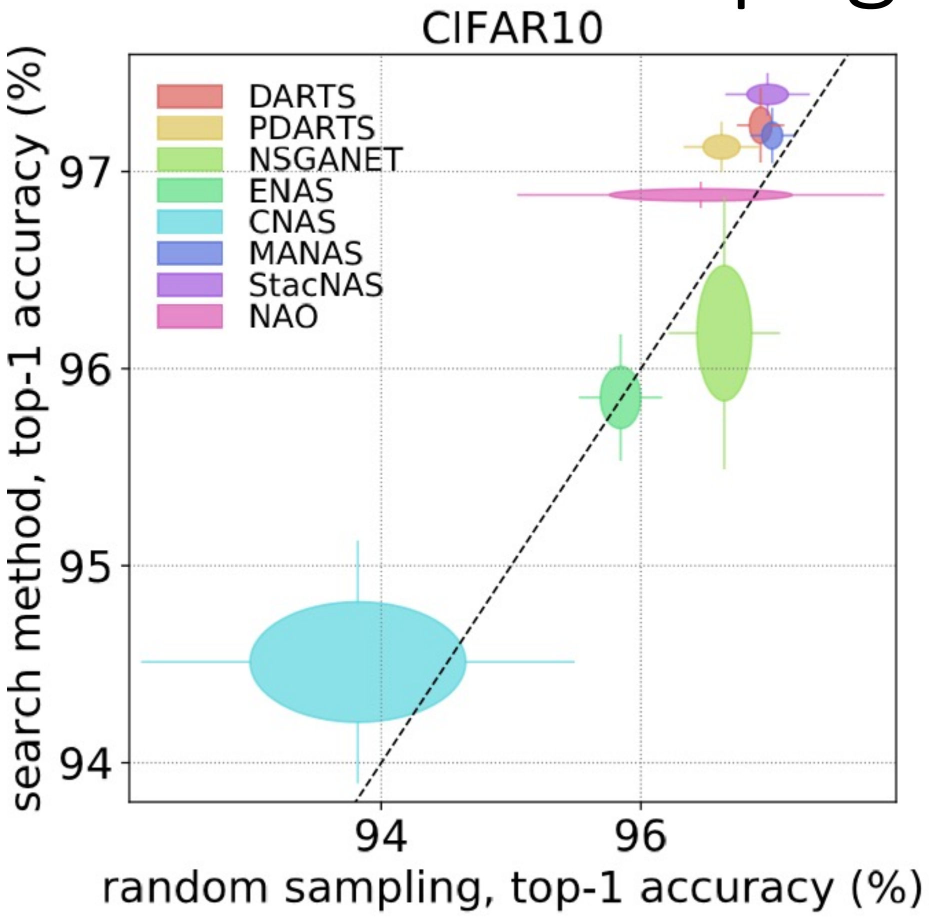


Discussion and Best Practices

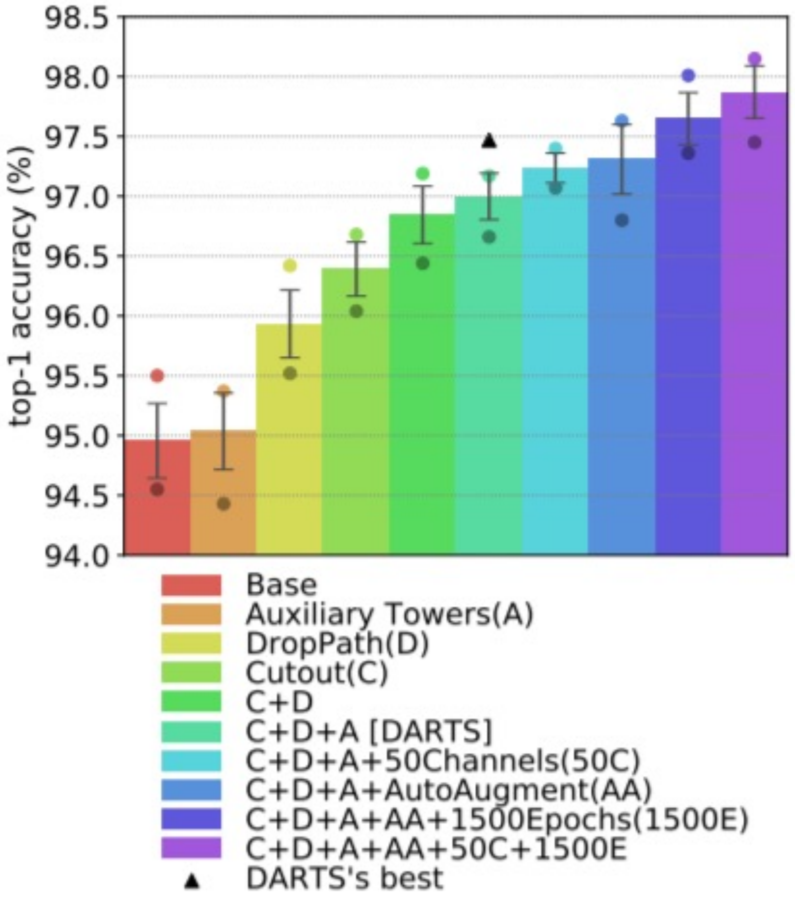
- **Comparing with baselines:**
- Either report a result with same training protocol / search space than previous works (e.g. NAS-Bench-101*)
- Either update the results of previous works with your new training protocol / search space
- Random Sampling is a simple, search-free and powerful baseline
- **Search Space Design:**
  If the goal of AutoML / NAS is to find the optimal architecture without human intervention, a wider search space (with a less constrained macro-structure) is a more interesting challenge than a narrow one.
- **Generability:**
  Evaluating on datasets with various sizes, image sizes, class granularity and learning task could avoid overfitting and highlight a costly hyperparameter tuning. This cost should be reported, if parameters have to be further tuned for other datasets / tasks.
- **Reproducibility:**
  Importance of providing all hyperparameters (including the seed) and open-sourcing the code

**NAS-Bench-101: Towards Reproducible Neural Architecture Search, Chris Ying et al., 2019*

ICLR webpage thumbnail



Comparison of search methods
with Random Sampling on CIFAR10



Comparison of different
training protocols