NAS-Bench-101: Towards Reproducible Architecture Search

Chris Ying*[1], Aaron Klein*[2], Esteban Real[1], Eric Christiansen[1], Kevin Murphy[1], Frank Hutter[2]

[1]Google Brain, [2]University of Freiburg

contact@chrisying.net, kleinaa@cs.uni-freiburg.de, ereal@google.com

Motivation

Neural architecture search (NAS) methods are notoriously difficult to reproduce and compare:

1. **Different search spaces** have different implicit biases
2. **Compute cost** limits number of trials and makes methods inaccessible to most researchers
3. Different methods use **different training procedures**

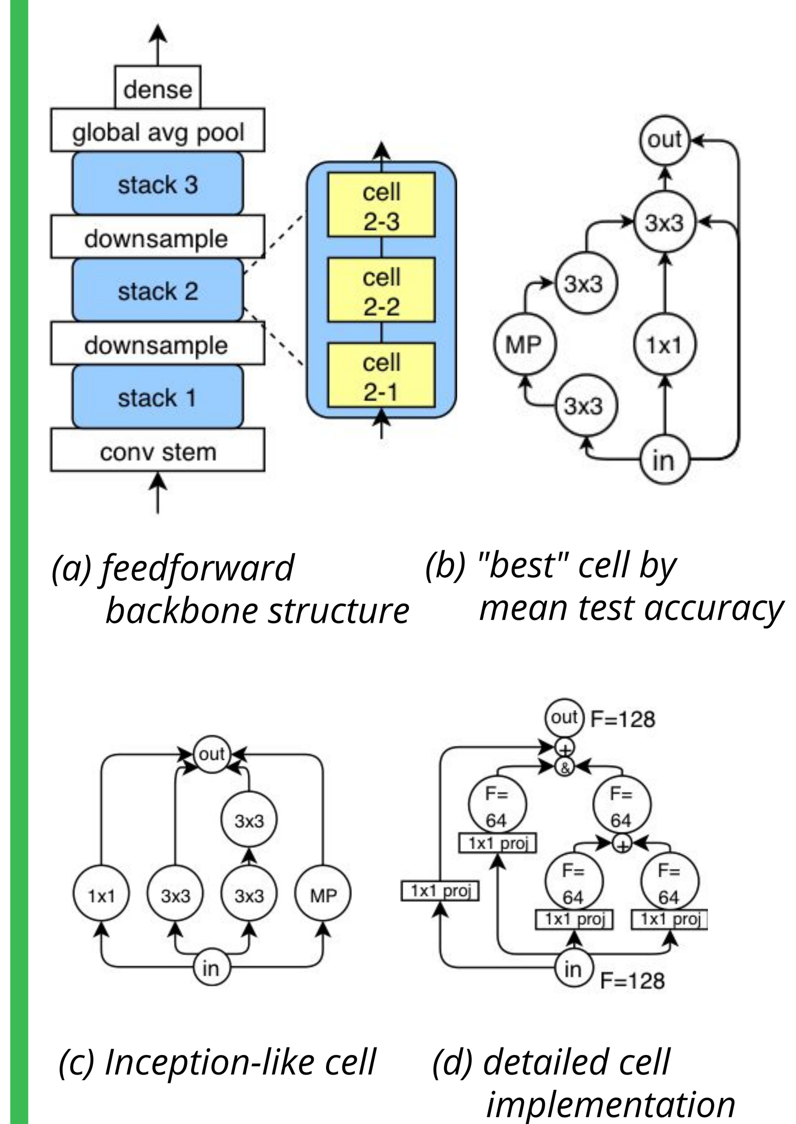
Our contribution: exhaustively evaluate all networks within a general search space → **NAS-Bench-101**

Enables:

1. Analysis of search space landscape as a whole
2. Cheap benchmarking of various NAS algorithms by querying the tabular dataset.

Dataset and code available at:
https://github.com/google-research/nasbench

Dataset



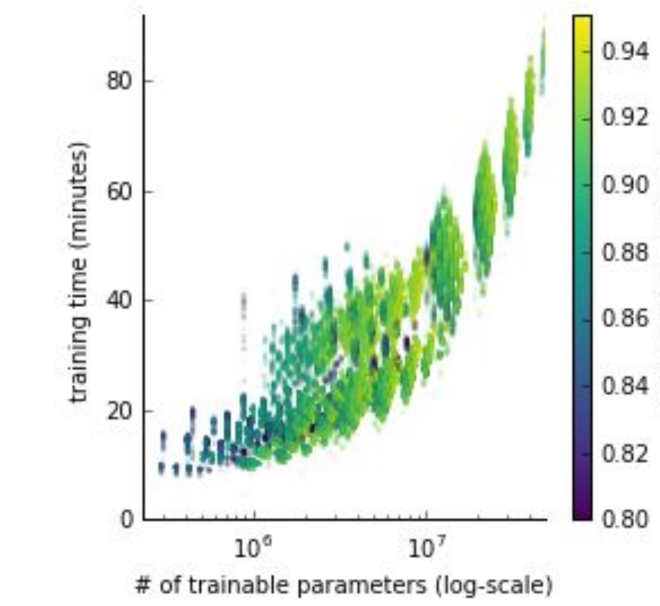
(a) feedforward backbone structure
(b) "best" cell by mean test accuracy
(c) Inception-like cell
(d) detailed cell implementation

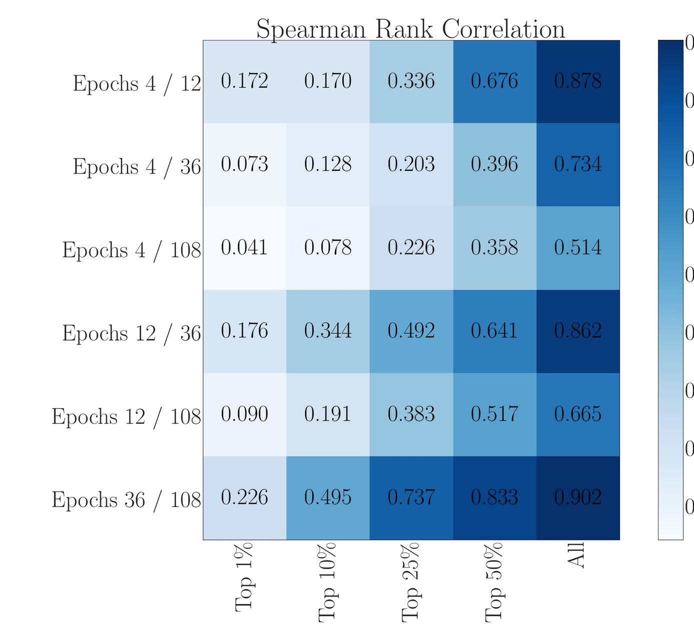
Architecture:

- Feedforward backbone with searched cells
- Cells are directed acyclic graphs with 3 operations:
  - 3x3 convolution
  - 1x1 convolution
  - 3x3 max-pool
- Limits to number of vertices & edges to keep dataset tractable
- Includes ResNet-like and Inception-like cells

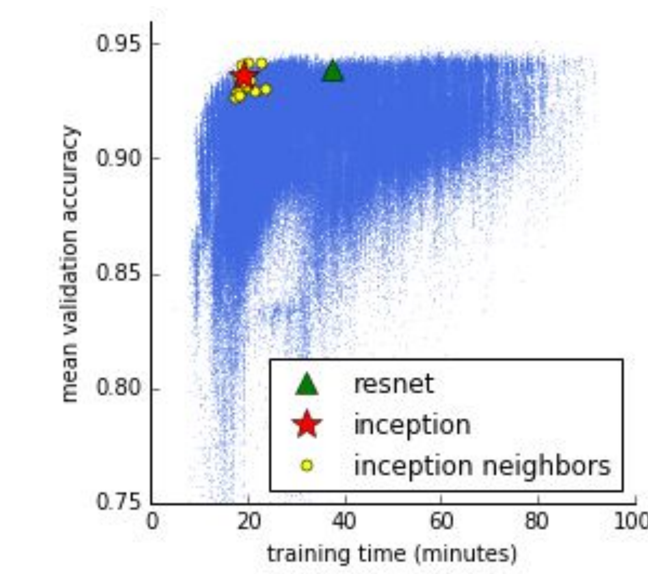
~423K <u>unique</u> cells
* 4 epoch budgets
* 3 repeats
= **~5M <u>total</u> models** trained

- Best cell is not most computationally expensive; ResNet & Inception are near Pareto frontier
- Correlation between small budget and large budget is low between top models

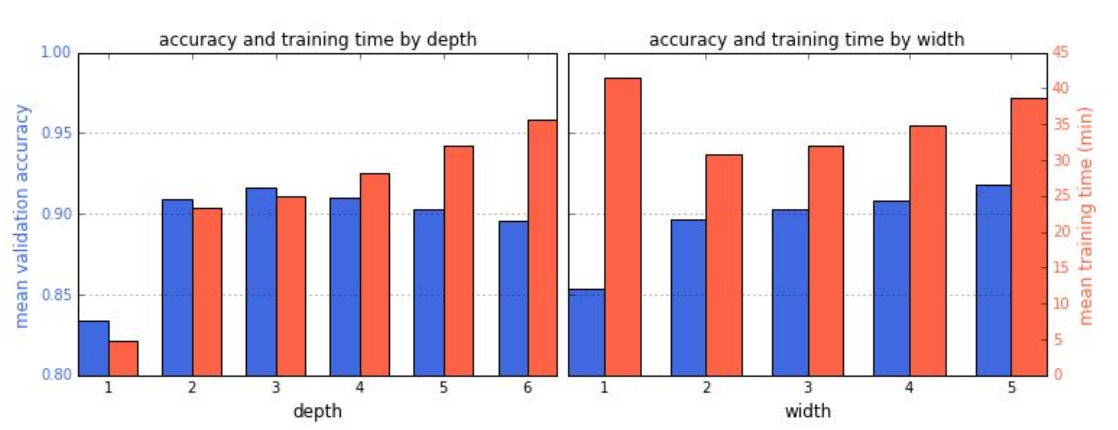




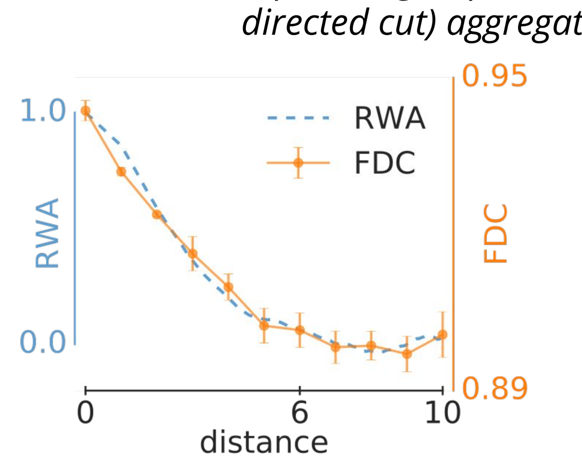
	Top 1%	Top 10%	Top 25%	Top 50%	All
Epochs 4 / 12	0.172	0.170	0.336	0.676	0.875
Epochs 4 / 36	0.073	0.128	0.203	0.396	0.734
Epochs 4 / 108	0.041	0.078	0.226	0.358	0.514
Epochs 12 / 36	0.176	0.344	0.492	0.641	0.892
Epochs 12 / 108	0.090	0.191	0.383	0.517	0.665
Epochs 36 / 108	0.226	0.495	0.737	0.833	0.902



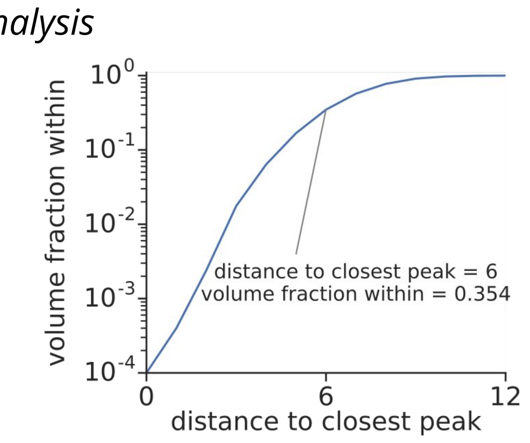
Analysis



(a) depth (longest path) and width (maximum directed cut) aggregate analysis



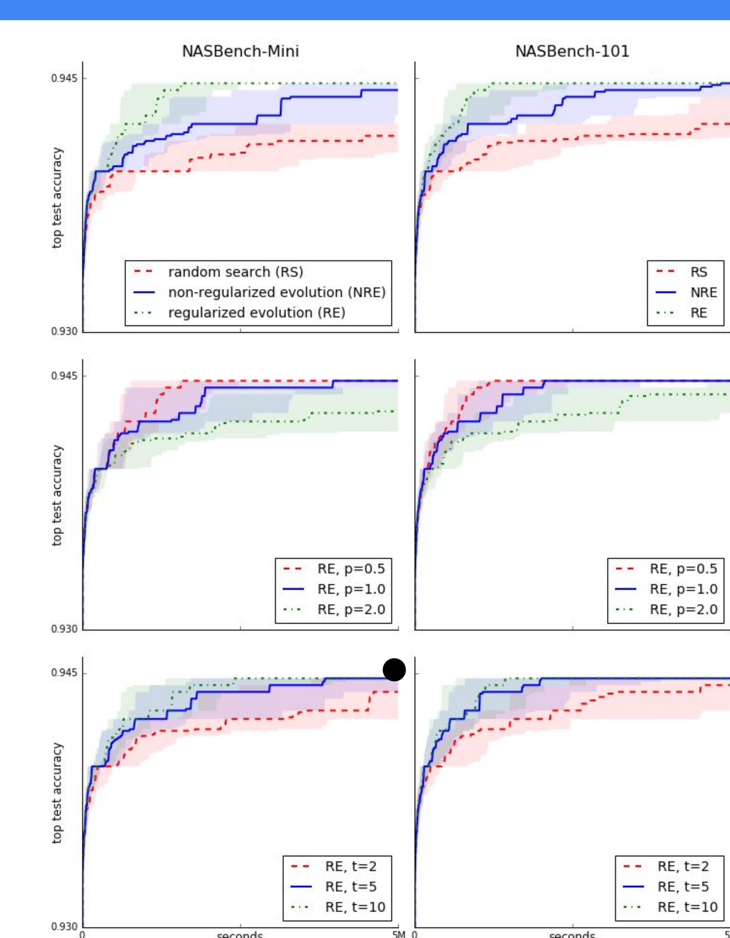
(b) Random walk autocorrelation (RWA) and fitness-distance correlation (FDC)



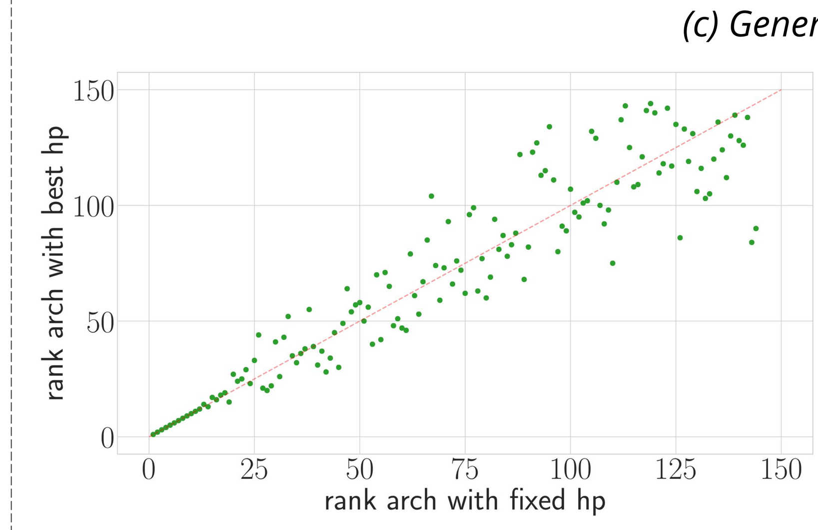
(c) Volume of search space that lies within edit distance from closest peak

- NAS-Bench-101 exhibits *locality*: similar architectures often have similar performance
- Randomly selecting one of the top models is extremely unlikely, but many models within short edit-distance from one of the top
- Locality-based search may be good choice

- Generalization bootstrap by comparing search trajectories of similar algorithms on subset vs. whole dataset
- Suggests that results from NAS-Bench-101 may generalize to larger search spaces



(c) Generalization bootstrap experiments

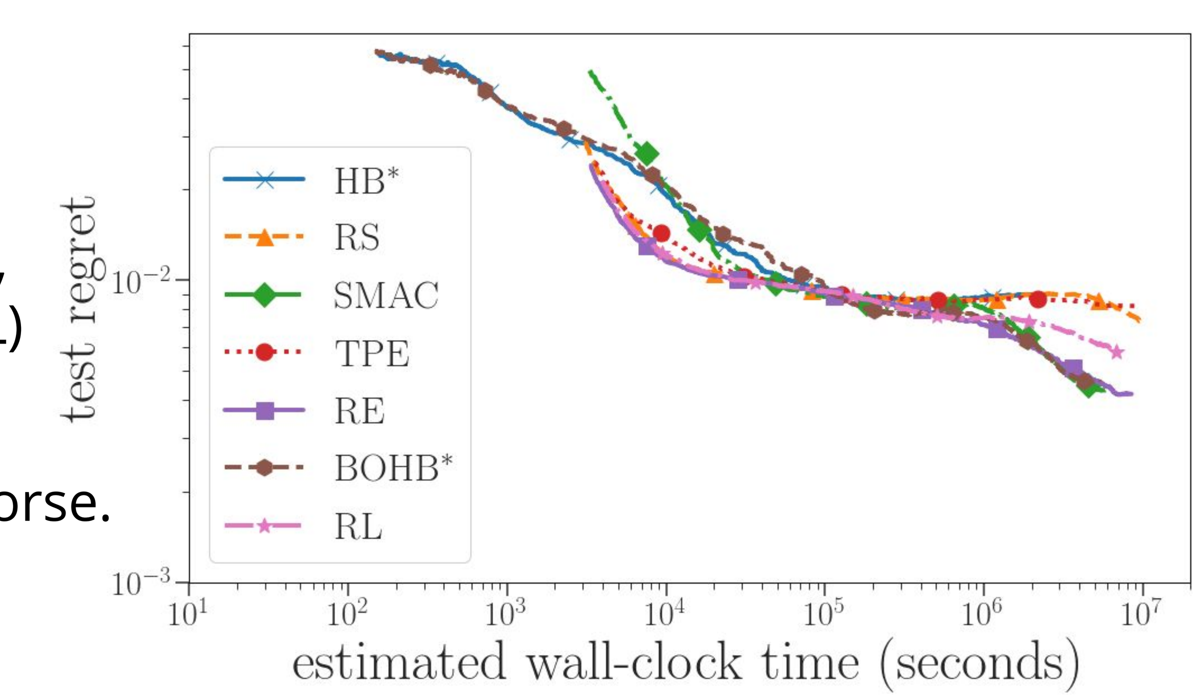


(c) Rank correlation with different hyperparameter selection on NAS-HPO dataset

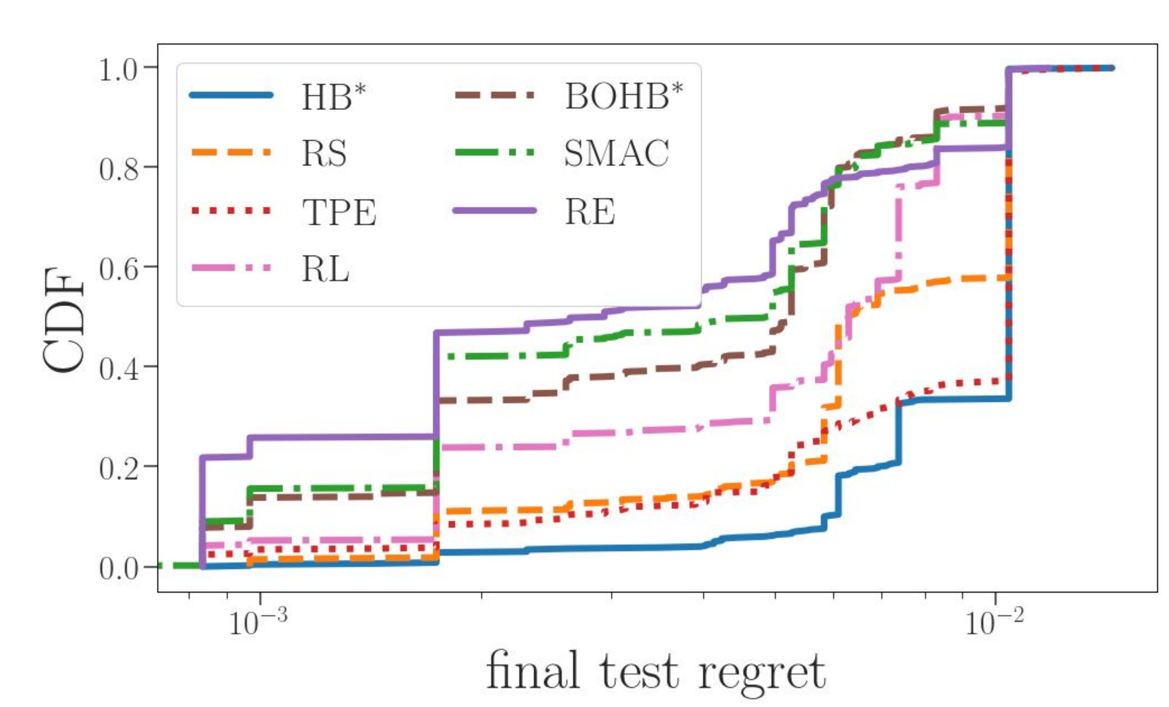
- Using fixed hyperparameters for all models is a reasonable choice (particularly for top-ranked models)

Benchmarking

- **Right**: comparison of various optimizers from the literature: Hyperband (HB), Random search (RS), SMAC, TPE, Regularized evolution (RE), BOHB and Reinforcement learning (RL)
- BOHB / RE / SMAC work equally well and achieve the lowest regret, RL is worse. TPE and HB do not work better than RS in this case



- Details: x-axis is estimated wall-clock time it *would* have taken to run on the original benchmark (but using the tabular benchmark, <u>evaluation only takes a few seconds</u>); y-axis is the average distance to the best average test error (i.e., simple regret)



- **Left**: to investigate the robustness of optimizers we show the cumulative distribution of the final regret after 10^7 seconds over all 500 independent runs of each optimizer
- None of the optimizers consistently converges to the same final regret and even the best methods only achieve a final regret of 10^{-3} in 50% of the cases