BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search

Galina Boeva

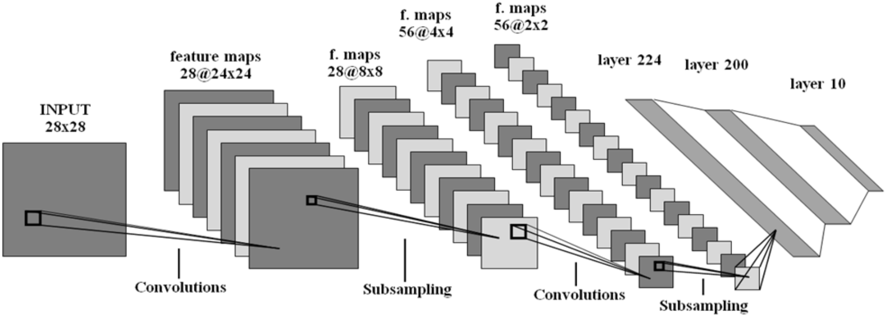
MIPT, 2023

February 12, 2024

Motivation

Main idea

NAS algorithms are now common. But the analysis in the papers often focuses on a full-fledged NAS, so it is difficult to say which individual components of the framework provide the best performance. Therefore, it is interesting to consider the individual components of the framework.

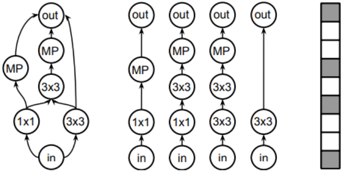


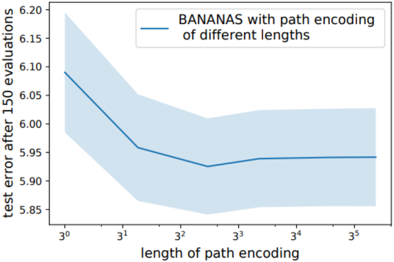
BO + Neural Predictor Framework

Consists of 4 components:

- Architecture encodings
- Neural predictors
- Uncertainty calibration
- Acquisition functions and optimization

Analysis of the Framework: path encoding





BANANAS

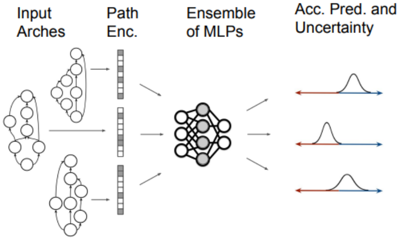
Algorithm 1 BANANAS

Input: Search space A, dataset D, parameters t_0, T, M, c, x, acquisition function ϕ, function $f(a)$ returning validation error of a after training.

1. Draw t_0 architectures $a_0, \ldots, a_{t_0}$ uniformly at random from A and train them on D.
2. For t from t_0 to T,
   i. Train an ensemble of neural predictors on $\{(a_0, f(a_0)), \ldots, (a_t, f(a_t))\}$ using the path encoding to represent each architecture.
   ii. Generate a set of c candidate architectures from A by randomly mutating the x architectures a from $\{a_0, \ldots, a_t\}$ that have the lowest value of $f(a)$.
   iii. For each candidate architecture a, evaluate the acquisition function $\phi(a)$.
   iv. Denote a_{t+1} as the candidate architecture with minimum $\phi(a)$, and evaluate $f(a_{t+1})$.

Output: $a^* = \text{argmin}_{t=0,\ldots,T} f(a_t)$.

Experiments of BANANAS



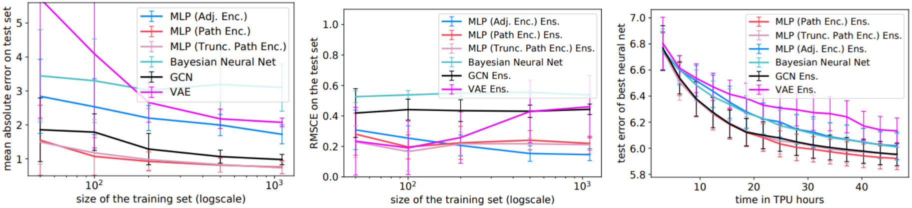


Figure: Performance of neural predictors.

Experiments of BANANAS

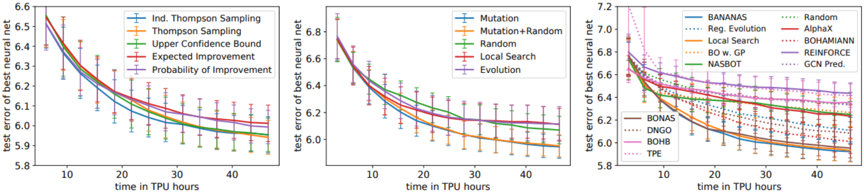


Figure: Performance of different acquisition functions, different acquisition function optimization strategies, other NAS algorithms.

NAS Algorithm	Source	Avg. Test error	Runtime	Method
Random search	(Liu, Simonyan, and Yang 2018)	3.29	4	Random
Local search	(White, Nolen, and Savani 2020)	3.49	11.8	Local search
DARTS	(Liu, Simonyan, and Yang 2018)	2.76	5	Gradient-based
ASHA	(Li and Talwalkar 2019)	3.03	9	Successive halving
Random search WS	(Li and Talwalkar 2019)	2.85	9.7	Random
DARTS	Ours	2.68	5	Gradient-based
ASHA	Ours	3.08	9	Successive halving
BANANAS	Ours	**2.64**	11.8	BO + neural predictor

Figure: Comparison of NAS algorithms on the DARTS search space.

Literature

1. **Main article** BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search.