Poster: Empirical Evaluation of AutoML Algorithms for Motor Health Prediction

Tanmay Goyal
ABB Research Center
Switzerland
tanmay.goyal@ch.abb.com

Pengcheng Huang
ABB Research Center
Switzerland
pengcheng.huang@ch.abb.com

Felix Sutton
ABB Research Center
Switzerland
felix.sutton@ch.abb.com

Balz Maag
ABB Research Center
Switzerland
balz.maag@ch.abb.com

Philipp Sommer
ABB Research Center
Switzerland
philipp.sommer@ch.abb.com

Abstract

The past few years have witnessed a growing interest in edge analytics across different industries. In this new paradigm, data is processed at the edge close to where it is generated. Therefore, in comparison to cloud analytics, edge analytics bring benefits of reduced data transmissions, enhanced data security, and improved responsiveness of in-field devices. Due to limited resources available to edge devices, one of the main challenges for edge analytics is to reduce the footprints of machine learning models in terms of timing, memory, and energy. This work evaluates and compares several common AutoML algorithms in optimizing an industrial edge analytics use case for motor health monitoring. We reveal the capabilities of existing algorithms in getting both accurate and small machine learning models. Based on our evaluations, future research directions are outlined.

1 Introduction

Bringing data analytics to industrial applications poses a multitude of new challenges. Many industrial computing systems on which new analytics enabled services will be installed, are severely limited in terms of their computing, and memory resources, *e.g.*, program logic controllers (PLCs), sensors, real-time control platforms, and gateways. This new paradigm is often termed as edge analytics; in contrast to the cloud counterpart, edge analytics are deployed on "small" computing platforms close to where data is generated.

To satisfy various constraints such as memory, energy, and latency, as induced by resource-limited edge analytics, users need to explore a huge search space which includes model parameters such as kernel sizes, number of channels etc., sensing parameters such as frequency and window, and hyperparameters such as epochs, learning rate, etc. This can be time-consuming if done manually. To tackle this, we need to automatically navigate through the search space in an intelligent manner in order to optimize for various constraints.

Conventionally, AutoML [6] techniques have been proposed, mainly focusing on optimizing analytics accuracy alone. One main area where AutoML algorithms are proposed is Neural Architecture Search (NAS). The NAS methods can be broadly divided into three categories. *(i)* Multi-trial algorithms sample different configurations, train them independently and then sample new configurations based on previous results. Typical methods used for sampling include Bayesian Optimization methods (TPE [1] and BOHB [4]), Anneal [6], and reinforcement learning [10]. Such methods also include SMAC [7], which uses Sequential Model-Based Optimization (SMBO). *(ii)* One-shot algorithms train a single supernet with weight sharing; results of different sampled configurations from this supernet are used to guide the selection of better models. Some well-known examples are DARTS [8], ENAS [9], and ProxylessNAS [2]. *(ii)* Multi-objective optimization algorithms such as the genetic algorithm NSGA-II [3], which searches for the best model configuration by optimizing several objectives simultaneously. The latest addition here is the *SMiLe* [5] framework, which uses hardware-in-the-loop feedback to perform optimization of an entire processing chain, including both sensing and machine learning.

To compare the performance of different AutoML algorithms for edge analytics, we benchmark in this work such algorithms against an industrial motor health prediction use case while making multi-objective optimization extensions when needed. Our empirical results highlight the effectiveness of the *SMiLe* framework and point out important future research directions in this area.

2 System Overview

We use a real-world use case, *i.e.*, motor health prediction for this study. We built a motor testbed featuring three ABB M3AA asynchronous 3-phase motors. The bearings of the motors were damaged to different degrees by adding metallic dust into the bearing cases, *i.e.*, 0 mg for a healthy, 250 mg

Table 1: Comparison of different AutoML algorithms (with *SMiLe* having hardware-in-the-loop) - each algorithm is allowed to run for 5 hours on NVIDIA GeForce RTX 2080 Ti GPU

Algorithm	# Parameters	Accuracy	Sensing Frequency (Hz)	Sensing window	Sensing Energy (mJ)	Inference Energy (mJ)	Sensing Duration (ms)	Inference Duration (ms)
Anneal	573	1	104	700	6.229	0.054	6.594	0.0188
TPE	671	0.999	1660	200	0.227	0.038	0.118	0.0112
BOHB	1878	0.9999	1660	300	0.343	0.066	0.176	0.019
SMAC	391	0.999	1660	700	0.778	0.063	0.412	0.021
NSGA-II	432	0.9999	1660	200	0.228	0.018	0.117	0.0062
SMiLe	339	1	1660	100	0.113	0.0073	0.058	0.0027

Table 2: Comparison of different AutoML algorithms - each algorithm runs for 5 hours on NVIDIA GeForce RTX 2080 Ti GPU with constant hyperparameters sensing window=200, sensing frequency=1660, batch=512, epochs=5, lr=0.01

Algorithm	# Parameters	Accuracy
Anneal	483	0.9999
TPE	447	0.9996
BOHB	528	0.9999
SMAC	273	0.999
NSGA-II	135	0.994
SMiLe	113	0.9916

for a lightly damaged, and 1000 mg for a heavily damaged bearing. The target of different AutoML algorithms is to build a machine learning model for motor health prediction with optimized accuracy and number of parameters. Furthermore we also compare the energy and latency of the optimized models. For this purpose, we ran the optimized models on a Nordic nRF5340 SoC and measured latency and energy consumption using our testbed.

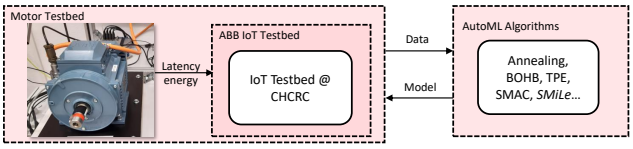

Figure 1: System overview - motor testbed and AutoML based design space exploration

3 Experiments and Results

Experiments. We benchmarked different types of AutoML algorithms - Anneal, TPE, BOHB, SMAC, NSGA-II, and *SMiLe*. We experimented with the algorithms to optimize *(i)* model + sensing + hyper- parameters and *(ii)* model parameters only; multi-objective optimization is performed with a linear combination of different objectives. We then compare the energy and latency measurements for optimized machine learning models generated by different algorithms (note that *SMiLe* can optimize out-of-the-box energy and latency with hardware-in-the-loop while other algorithms don't).

Results. From Table 1 and Table 2, we can see that *SMiLe* is able to find smaller models with equivalent accuracies while outperforming other state-of-the-art AutoML algorithms in terms of model parameters, latency, and energy. From this, we conclude that *SMiLe* is a powerful tool for edge analytics optimization due to its built-in capabilities to optimize final latency and energy characteristics of both sensing and machine learning with the help of hardware-in-the-loop.

Outlook. Most existing AutoML algorithms focus only on analytics accuracy and cannot yet cover a large generic search space which includes sensing configurations, latency, and energy as supported by *SMiLe*. This is especially the case for one-shot algorithms, which have very fixed search space and are not included in our evaluation in this work. Furthermore, existing algorithms either don't support multi-objective optimization or support it in a minimal way. It is important to make those extensions and continue with more extensive studies here.

4 References

[1] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

[2] H. Cai, L. Zhu, and S. Han. Proxylessnas: Direct neural architecture search on target task and hardware, 2018.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[4] S. Falkner, A. Klein, and F. Hutter. Bohb: Robust and efficient hyperparameter optimization at scale, 2018.

[5] T. Goyal, P. Huang, F. Sutton, B. Maag, and P. Sommer. Smile - automated end-to-end sensing and machine learning co-design. In *International Conference on Embedded Wireless Systems and Networks (EWSN)*, 2022.

[6] X. He, K. Zhao, and X. Chu. AutoML: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622, jan 2021.

[7] F. Hutter, H. H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Proceedings of the 5th International Conference on Learning and Intelligent Optimization*, LION'05, pages 507–523, Berlin, Heidelberg, 2011. Springer-Verlag.

[8] H. Liu, K. Simonyan, and Y. Yang. Darts: Differentiable architecture search, 2018.

[9] H. Pham, M. Y. Guan, B. Zoph, Q. V. Le, and J. Dean. Efficient neural architecture search via parameter sharing, 2018.

[10] B. Zoph and Q. V. Le. Neural architecture search with reinforcement learning, 2016.