BOMP-NAS: Bayesian Optimization Mixed Precision NAS

David van Son[1], Floran de Putter[1], Sebastian Vogel[2], and Henk Corporaal[1]

[1]Eindhoven University of Technology, [2]NXP Semiconductors

***Abstract*—Bayesian Optimization Mixed-Precision Neural Architecture Search (BOMP-NAS) is a method to quantization-aware neural architecture search that leverages both Bayesian optimization and mixed-precision quantization to efficiently search for compact, high performance deep neural networks. It is able to find neural networks that achieve state of the art accuracy with less search time. Compared to the closest related work, BOMP-NAS can find these neural networks in 6× less search time.**

I. Introduction

Deep learning models have greatly improved image processing tasks, but designing them is difficult, especially when they must be deployed on devices with limited resources, such as mobile phones and car control units, while still maintaining high accuracy and low latency [1] [2].

Neural architecture search (NAS) methods automate the tedious design of efficient and effective neural network architectures and typically outperforms human-designed networks. Additionally, model compression techniques such as pruning and quantization can further optimize neural networks for deployment on devices with limited resources, while maintaining performance.

To design highly accurate and efficient models in a limited amount of time we present a new approach for neural architecture search (NAS), called BOMP-NAS, which integrates mixed-precision quantization and Bayesian Optimization (BO). The contributions of this paper are:

1) A new sampling-based quantization-aware NAS methodology: Bayesian Optimization Mixed-Precision NAS (BOMP-NAS).
2) Integration of fine-grained mixed-precision quantization and quantization-aware fine-tuning in the NAS methodology with limited overhead.
3) BOMP-NAS finds more performant models with similar memory budgets at 6× shorter search time compared to state-of-the-art (SotA).

II. Related work

Studies have shown that jointly optimizing DNN architecture and model compression leads to better results than optimizing them separately; e.g., the best architecture for a neural network using floating point may not be the best for a quantized network [3], [4].

In μNAS [5] aging evolution combined with 8-bit post-training quantization was used to find networks suitable for micro-controllers. [3] extends this by considering mixed-precision quantization. However, a limitation of this method is that the search may get stuck in a bad local minimum. As BOMP-NAS uses Bayesian Optimization (BO) as a search strategy, it is less prone to getting stuck in local minima. Moreover, using BO instead of an evolutionary algorithm, the search space is traversed more efficiently.

Additionally, BOMP-NAS employs quantization-aware fine-tuning to learn to compensate for the quantization noise, whereas [5] and [3] rely on trained architectures that can be easily quantized without much accuracy loss.

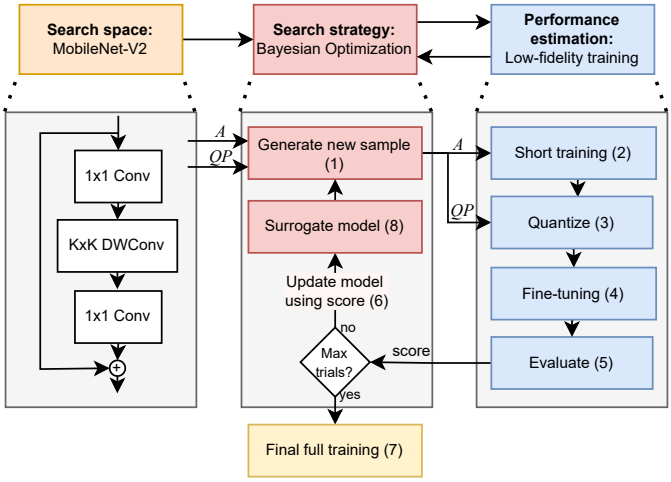

Fig. 1. Workflow of BOMP-NAS. Network architectures A and Quantization Policies QP are selected (1) from the Search space using the surrogate model (8). The network is shortly trained in full precision (2), then quantized according to QP (3). This quantized network is then fine-tuned quantization-aware (4). Next, the network is evaluated (5) and its results are scalarized into a score (Eq. 1). The score is used to update the surrogate model (6), which is then used to sample the next candidate network (1). If the maximum trials have exceeded, final Pareto optimal models are fully trained (7).

III. BOMP-NAS Methodology

In this study, we propose a new NAS method that utilizes BO to efficiently explore the search space. Our method, called BOMP-NAS, is illustrated in Fig. 1. BO relies on a probabilistic surrogate model (8) and an acquisition function (upper confidence bound, (1)) that uses the surrogate model to generate the next sample (network A with quantization policy QP) that should be evaluated. The network is shortly trained for 20 epochs (2), then quantized according to the mixed-

978-3-9819263-7-8/DATE23/© 2023 EDAA

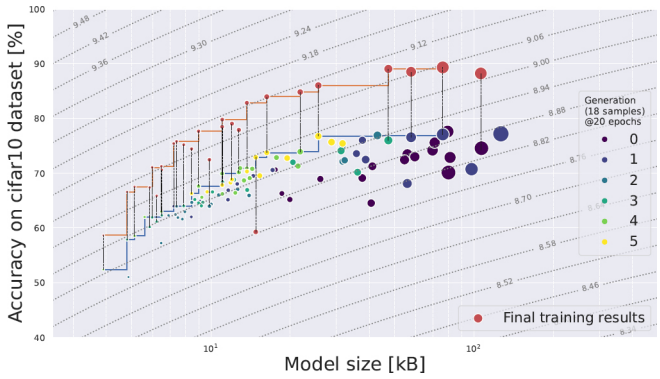

Fig. 2. The outcome of one single search of BOMP-NAS on CIFAR-10 ($ref\_acc = 0.8$, $ref\_model\_size = 8$). It compares the model size and accuracy of the candidate neural networks. The networks are colored based on when they were sampled, with earlier models being darker than later models. The graph shows that the networks sampled by BO improve over time as the surrogate model gets more information with each new sample. The final trained Pareto optimal models are shown in red and are connected to their respective candidate network. The dotted lines represent the score (Eq. 1).

TABLE I
PARETO OPTIMAL ARCHITECTURES FOUND BY A SINGLE SEARCH OF BOMP-NAS COMPARED TO SOTA. THE SHOWN NETWORKS ARE THE BEST PERFORMING NETWORKS THAT ARE OF SIMILAR SIZE AS THE RESPECTIVE SOTA NETWORK. BOMP-NAS FINDS NETWORKS THAT OUTPERFORM SOTA IN A SINGLE SEARCH IN A BROAD MODEL SIZE RANGE.

Dataset	Method	Accuracy [%]	Model size [kB]	Search cost [h]
CIFAR-10	JASQ (repr.)	65.97	4.47	72
	BOMP-NAS	**67.36**	**4.57**	**12**
	JASQ [3]	**97.03**	900.00	72
	BOMP-NAS	88.67	**76.08**	**12**
	μNAS [5]	**86.49**	**11.40**	552
	BOMP-NAS	83.96	16.30	**12**
CIFAR-100	DFQ [7]	**77.30**	11200.00	n.a.
	GZSQ [8]	75.95	5600.00	n.a.
	BOMP-NAS	75.84	**4199.00**	**30**
	LIE [9]	73.34	1800.00	n.a.
	BOMP-NAS	**74.00**	**1773.00**	**30**
	Mix&Match [10]	71.50	1700.00	n.a.
	LIE [9]	71.24	**1010.00**	n.a.
	BOMP-NAS	**72.36**	1047.00	**30**
	APoT [11]	66.42	**90.00**	n.a.
	BOMP-NAS	**68.18**	353.00	**30**

precision quantization policy. This quantized DNN is then fine-tuned quantization-aware (4). Next, the DNN is evaluated (5). These results are then scalarized into a score (5):

$$score = \frac{accuracy\ [\%]}{ref\_accuracy} + \frac{ref\_model\_size}{log_{10}\left(model\ size\ [bits]\right)}. \quad (1)$$

This score is used to update the surrogate model (6), which is then used to generate a new sample (1). Lastly, the resulting Pareto optimal models are fully trained (7) for 200 epochs, followed by 5 epochs of quantization-aware fine-tuning.

The search space of BOMP-NAS is build upon MobileNet-V2 [6]. For each inverted bottleneck block, the kernel size, width multiplier, expansion factor, number of repetitions and bitwidth (4,5,6,7,8-bits) was searchable. In total, the search space contains $4.73 \cdot 10^{39}$ mixed-precision models.

IV. RESULTS

Fig. 1 illustrates the outcome of one single search of BOMP-NAS on the CIFAR-10 dataset. In Table I, we compare the performance of BOMP-NAS to other SotA works on both the CIFAR-10 and CIFAR-100 dataset. For CIFAR-10 it shows BOMP-NAS outperforms a reproduced version of JASQ by more than 1pp, while having a similar model size. However, when compared to μNAS, BOMP-NAS performs 2.5pp worse, but has a search cost that is more than 40 times lower. For CIFAR-100, BOMP-NAS can outperform multiple SotA works in a **single** search, but not all. This is because the range of model sizes is large and the amount of trials per search is limited. Our expectation is that BOMP-NAS is able to find more performant networks when considering a certain size regime, or by raising the amount of trials.

V. CONCLUSION

Bayesian Optimization Mixed Precision (BOMP)-NAS is an approach to quantization-aware NAS. It utilizes both Bayesian Optimization (BO) and Mixed Precision (MP) to efficiently search for compact, high-performance networks. BOMP-NAS is capable of finding networks that achieve state-of-the-art accuracy on CIFAR-10. For example, networks designed by BOMP-NAS outperform JASQ [3] by 1.4pp with a memory budget of 4.5 kB. Furthermore, BOMP-NAS finds these state-of-the-art models at much lower design cost by employing BO as a search strategy. Compared to the closest related work, JASQ [3], BOMP-NAS finds better performing models with similar memory budgets at a 6$\times$ shorter search time.

REFERENCES

[1] K. He *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[2] J. Long *et al.*, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[3] Y. Chen *et al.*, "Joint neural architecture search and quantization," *CoRR*, vol. abs/1811.09426, 2018.

[4] H. Bai *et al.*, "Batchquant: Quantized-for-all architecture search with robust quantizer," *CoRR*, vol. abs/2105.08952, 2021.

[5] E. Liberis *et al.*, "μnas: Constrained neural architecture search for microcontrollers," *CoRR*, vol. abs/2010.14246, 2020.

[6] M. Sandler *et al.*, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[7] Y. Choi *et al.*, "Data-free network quantization with adversarial knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[8] X. He *et al.*, "Generative zero-shot network quantization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2021, pp. 3000–3011.

[9] H. Liu *et al.*, "Layer importance estimation with imprinting for neural network quantization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2021, pp. 2408–2417.

[10] S. Chang *et al.*, "Mix and match: A novel fpga-centric deep neural network quantization framework," *CoRR*, vol. abs/2012.04240, 2020.

[11] Y. Li *et al.*, "Additive powers-of-two quantization: A non-uniform discretization for neural networks," *CoRR*, vol. abs/1909.13144, 2019.