

GA-based Training-Free NAS Algorithm with Hybrid Score Function

Hsieh Cheng-Han

emiliastruelove@gmail.com

Department of Computer Science and Engineering,
National Sun Yat-sen University
Kaohsiung, Taiwan

Chun-Wei Tsai

cwtsai@mail.cse.nsysu.edu.tw

Department of Computer Science and Engineering,
National Sun Yat-sen University
Kaohsiung, Taiwan

ABSTRACT

Most neural architecture searches (NASs) are time-consuming caused by the fact that, during the searching, a candidate architecture must be trained to evaluate how good of this architecture. This is why some of training-free NAS algorithms have been proposed in recent years.

Although the training-free NASs are typically faster than training-based NAS method, however, the correlation between score value and the result of an architecture is not well enough in most cases.

To address this problem, we propose a genetic-based training-free NAS algorithm with hybrid training-free score function, which combines three highly heterogeneous training-free score functions to evaluate an architecture. In this method, the genetic algorithm plays a role to guide the searches of NAS algorithm while the hybrid training-free score function plays the role to evaluate a new candidate architecture during the convergence process of GA. More precisely, the first score function is noise immunity for neural architecture search without search (NINASWOT), as an evaluation of pattern recognition ability, second one is maximum-entropy detection (MAE-DET), as an evaluation of the entropy of an architecture and the third one is condition number of neural tangent kernel (NTK), as an evaluation of the speed of converge.

To evaluate the performance of the proposed algorithm, we compared it with several NAS algorithms, including weight-sharing methods, non-weight-sharing methods, and neural architecture search without training (NASWOT). We expect develop a faster and more accurate training-free NAS algorithm.

ACM Reference Format:

Hsieh Cheng-Han and Chun-Wei Tsai. 2023. GA-based Training-Free NAS Algorithm with Hybrid Score Function. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

1 INTRODUCTION

Neural architecture search (NAS) has recently drawn a big amount of attention, since the ability to automatically design a "good" neural architecture. By leveraging machine learning algorithms [1], NAS

algorithms can explore a search space, which is comprised of numerous potential architectures, to find out a good architectures that outperform those designed by human experts. In recent years, the use of NAS is widespread, from object detection [2], image recognition [3] and speech recognition [4] [5] to natural language processing. [6] [7] [8]

Despite the promising results of NAS, there are still many challenges to conquer. One major problem is the extremely high computational cost to search for an optimal architecture, which can make NAS impractical for real-world applications, particularly on resource-constrained platforms like embedded system. The reason why NAS is costly is that during the searching, a candidate architecture must be trained to evaluate how good of this architecture.

To overcome this challenge, recent works developed and proposed lots of method which is so called training-free NAS. For example, Mellor et al. [3] proposed the measurement of the correlation between the binary activation patterns, induced by the untrained network at two inputs, named as neural architecture search without training (NASWOT). The correlation is defined as the logarithmic determinant of the kernel matrix, as follows:

$$s = \log|K_H| \quad (1)$$

where K_H is the kernel matrix. Later, Wu et al. [9] found a high score obtained by such a function may not correspond to a high-performance model. Thus, they additionally applied noisy immunity method on NASWOT, named as noisy immunity for NASWOT (NINASWOT). For another example, Chen et al. proposed to compute the condition number of neural tangent kernel (NTK) [10] [11] [12], which is defined as

$$\mathcal{K}_N = \frac{\lambda_{\max}(\hat{\Theta})}{\lambda_{\min}(\hat{\Theta})} \quad (2)$$

and used as the score to estimate the trainability of an architecture.

However, most of score functions suffer from low correlation between score value and the result of an architecture, leading to a predicament that no matter how good the search method is used, we can hardly find an optimal architecture. The major problem causes the low correlation is that a single score function can only evaluate one perspective/characteristic of an architecture.

To address the problem, we propose cooperating three heterogeneous score functions with genetic-based search method, we shall evaluate an architecture from different aspects. More specifically, the first score function is NINASWOT [9] as an indicator of the ability to distinguish two images. The second one is the condition number of NTK [10] as score to estimate the trainability of an architecture. The last one is (TBD).

The major contribution of this paper can be summarized as follow:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, July 2017, Washington, DC, USA
© 2023 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/Y/Y/MM...\$15.00
<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

- Develop a genetic-based neural architecture search method based on three hybrid score functions.
- Cooperating three score functions to get a higher correlation between score and accuracy.

The remainder of this paper is organized as follows: Section 2 provides the detail about the three heterogeneous score functions. Section 3 gives a detailed description about the proposed method. Section 4 provide the simulation results in different search space. The conclusion and further prospect are given in Section 5.

2 RELATED WORKS

In these days, the studies of training-free neural architecture search (NAS) go viral, since the ability to accelerate the procedure for designing a neural network architecture used on specific application, while additionally, training-free NAS cleverly avoid the drawback of long-time training by using various training-free score functions.

In [3], Mellor et al. proposed a score function without the requirement for training. Here we call it neural architecture search without training (NASWOT) score function. Consider a mini-batch of data $X = \{x_i\}_{i=1}^N$ passing through a neural network architecture as $f(x_i)$. The activated ReLU units in every layer of the architecture form a binary code c_i that define the linear region. The correlation between binary codes for the whole mini-batch can be examined by computing the kernel matrix

$$K_H = \begin{pmatrix} N_A - d_H(c_1, c_2) & \cdots & N_A - d_H(c_1, c_N) \\ \vdots & \ddots & \vdots \\ N_A - d_H(c_N, c_1) & \cdots & N_A - d_H(c_N, c_N) \end{pmatrix} \quad (3)$$

where N_A is the number of ReLU units and $d_H(c_i, c_j)$ is the hamming distance between the binary code c_i and c_j . With the kernel matrix, the score of an architecture can be evaluate as follow:

$$s = \log|K_H| \quad (4)$$

The rationale behind is to see how dissimilar the linear region activated by the ReLU units between two inputs. An architecture shall learn better when inputs are well separated.

Based on the work of Mellor et al., Wu et al. [9] found, in some case, an architecture with high score may put different input data, which are originally in the same class, to different classes. To fix this defect, Wu et al. proposed using noise immunity to additionally evaluate an architecture. The score function picks a mini-batch of data, denoted X , and then applies Gaussian noise on it, denoted X' and defined by $X' = X + z$ where z is the Gaussian noise. By passing X and X' through the untrained architecture, the sum of the square differences between outputs $O = o_1, o_2, \dots, o_C$ and $O' = o'_1, o'_2, \dots, o'_C$ can be calculated as follows:

$$n = - \sum_{i=1}^C (o_i - o'_i)^2 \quad (5)$$

where C is the number of classes determined by the input data. According to the experiment outcome of Wu et al., noise immunity can separate those are unable to distinguish X and X' from architectures with high NASWOT score, which just complements the NASWOT score function.

On the other hand, instead of using the correlation of binary activation patterns as score. Chen et al. proposed using the condition

number of NTK as an evaluation of the trainability, defined as

$$\mathcal{K}_N = \frac{\lambda_{\max}(\hat{\Theta})}{\lambda_{\min}(\hat{\Theta})} \quad (6)$$

where $\lambda_{\max}(\hat{\Theta})$ and $\lambda_{\min}(\hat{\Theta})$ are the maximum and minimum eigenvalues of NTK ($\hat{\Theta}$) respectively. The rationale is based on the work of Lee et al. [13] and Xiao et al. [14], which concluded briefly here, the training dynamics of a wide neural network can be written in terms of the spectrum of the NTK:

$$\mu_t(\mathbf{X}_{\text{train}})_i = (\mathbf{I} - e^{-\eta\lambda_i t})\mathbf{Y}_{\text{train},i} \quad (7)$$

where λ_i are the eigenvalues of $\hat{\Theta}_{\text{train}}$ and they're ordered as $\lambda_0 \geq \dots \geq \lambda_m$. According to the hypothesis proposed by Lee et al. [13], the maximum feasible learning rate scale as $\eta \sim 2/\lambda_0$. Plug this scaling for η in Eq. 7, we see that the λ_m will converge exponentially at a rate given by $1/K_N$ where $K_N = \frac{\lambda_0}{\lambda_m}$. Then it can be concluded that if K_N is lower, the corresponding architecture is more trainable.

Another important component of training-free NAS is search algorithm. There're various search algorithms that applied on training-free NAS. The search algorithm that gained much attention in recent years is evolutionary algorithms (EAs) [15]. In [9], Wu et al. used genetic algorithm (GA), a popular evolutionary algorithm which has great potential to search a vast search space. The NINASWOT score is used as fitness of a chromosome in genetic algorithm. And the tournament selection is used to select an architecture with good NINASWOT score. Cavagnero et al. [16] used a custom tournament selection with ageing (REA). Instead of discarding less fitting individuals, REA discards the elder individuals, and thus explore more search space.

Though the search algorithm is very powerful to search a vast space, the key to the holy grail of training-free NAS is still the training-free score functions. Our work studies the right way to combining the score functions which complements each others, and explore the potential of rank-based GA.

3 THE PROPOSED ALGORITHM

In our work, however, we use rank-based GA. Rather than use single fitness to select, the method preserve the top 1 in NINASWOT score and also the top 1 in the condition number of NTK. After the evolution,

REFERENCES

- [1] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," 2016. [Online]. Available: <https://arxiv.org/abs/1611.01578>
- [2] Z. Sun, M. Lin, X. Sun, Z. Tan, H. Li, and R. Jin, "Mae-det: Revisiting maximum entropy principle in zero-shot nas for efficient object detection," 2021. [Online]. Available: <https://arxiv.org/abs/2111.13336>
- [3] J. Mellor, J. Turner, A. Storkey, and E. J. Crowley, "Neural architecture search without training," 2020. [Online]. Available: <https://arxiv.org/abs/2006.04647>
- [4] H. Zheng, K. An, and Z. Ou, "Efficient neural architecture search for end-to-end speech recognition via straight-through gradients," 2020. [Online]. Available: <https://arxiv.org/abs/2011.05649>
- [5] A. Mehrotra, A. G. C. P. Ramos, S. Bhattacharya, L. Dudziak, R. Vipera, T. Chau, M. S. Abdelfattah, S. Ishtiaq, and N. D. Lane, "{NAS}-bench-{asr}: Reproducible neural architecture search for speech recognition," in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=CU0APx9LMaL>
- [6] Y. Jiang, C. Hu, T. Xiao, C. Zhang, and J. Zhu, "Improved differentiable architecture search for language modeling and named entity recognition," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association

- for Computational Linguistics, Nov. 2019, pp. 3585–3590. [Online]. Available: <https://aclanthology.org/D19-1367>
- [7] N. Klyuchnikov, I. Trofimov, E. Artemova, M. Salnikov, M. Fedorov, and E. Burnaev, “Nas-bench-nlp: Neural architecture search benchmark for natural language processing,” 2020. [Online]. Available: <https://arxiv.org/abs/2006.07116>
- [8] H. Wang, Z. Wu, Z. Liu, H. Cai, L. Zhu, C. Gan, and S. Han, “Hat: Hardware-aware transformers for efficient natural language processing,” 2020. [Online]. Available: <https://arxiv.org/abs/2005.14187>
- [9] M.-T. Wu, H.-I. Lin, and C.-W. Tsai, “A training-free genetic neural architecture search,” in *Proceedings of the 2021 ACM International Conference on Intelligent Computing and Its Emerging Applications*, ser. ACM ICEA '21. New York, NY, USA: Association for Computing Machinery, 2022, p. 65–70. [Online]. Available: <https://doi.org/10.1145/3491396.3506510>
- [10] W. Chen, X. Gong, and Z. Wang, “Neural architecture search on imagenet in four gpu hours: A theoretically inspired perspective,” 2021. [Online]. Available: <https://arxiv.org/abs/2102.11535>
- [11] H. Wang, Y. Wang, R. Sun, and B. Li, “Global convergence of maml and theory-inspired neural architecture search for few-shot learning,” 2022. [Online]. Available: <https://arxiv.org/abs/2203.09137>
- [12] Y. Shu, S. Cai, Z. Dai, B. C. Ooi, and B. K. H. Low, “Nasi: Label- and data-agnostic neural architecture search at initialization,” 2021. [Online]. Available: <https://arxiv.org/abs/2109.00817>
- [13] J. Lee, L. Xiao, S. S. Schoenholz, Y. Bahri, R. Novak, J. Sohl-Dickstein, and J. Pennington, “Wide neural networks of any depth evolve as linear models under gradient descent,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2020, no. 12, p. 124002, dec 2020. [Online]. Available: <https://doi.org/10.1088%2F1742-5468%2Fabc62b>
- [14] L. Xiao, J. Pennington, and S. S. Schoenholz, “Disentangling trainability and generalization in deep neural networks,” 2019. [Online]. Available: <https://arxiv.org/abs/1912.13053>
- [15] Y. Liu, Y. Sun, B. Xue, M. Zhang, G. G. Yen, and K. C. Tan, “A survey on evolutionary neural architecture search,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 2, pp. 550–570, feb 2023. [Online]. Available: <https://doi.org/10.1109%2Ftnnls.2021.3100554>
- [16] N. Cavagnero, L. Robbiano, B. Caputo, and G. Averta, “Freerea: Training-free evolution-based architecture search,” 2022. [Online]. Available: <https://arxiv.org/abs/2207.05135>