Supplementary of "DU-DARTS: Decreasing the Uncertainty of Differentiable Architecture Search"

Shun Lu[1,2,3]
lushun19s@ict.ac.cn
Yu Hu*[1,2,3]
huyu@ict.ac.cn
Longxing Yang[1,2,3]
yanglongxing20b@ict.ac.cn
Zihao Sun[1,2,3]
sunzihao18z@ict.ac.cn
Jilin Mei[1,2,3]
meijilin@ict.ac.cn
Yiming Zeng[4]
zyms5244@gmail.com
Xiaowei Li[2,3]
lxw@ict.ac.cn

[1] Research Center for Intelligent Computing Systems, Institute of Computing Technology, Chinese Academy of Sciences
[2] State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences
[3] School of Computer Science and Technology, University of Chinese Academy of Sciences.
[4] Tecent ADlab

A Methodology

A.1 Cell Structure

Fig. 4 shows the structure of the searched cell. Edge indices are marked in the figure. These indices correspond to edges in Fig.2(a) in ascending order. Each cell has $\mathcal{M}$ nodes and $\mathcal{N}$ edges. We adopt $\mathcal{M} = 7$ nodes, which consists of 2 input nodes, 4 middle nodes, and 1 output node, thus $\mathcal{N} = 14$ edges.

A.2 Candidate Operations

Same as DARTS [2], each edge in Fig. 4 is a mixed operation. In our search space, we apply 7 candidate operations for each mixed operation. As shown in Tab. 4, we use abbreviations to denote them in the figures to save space.

*Corresponding Author

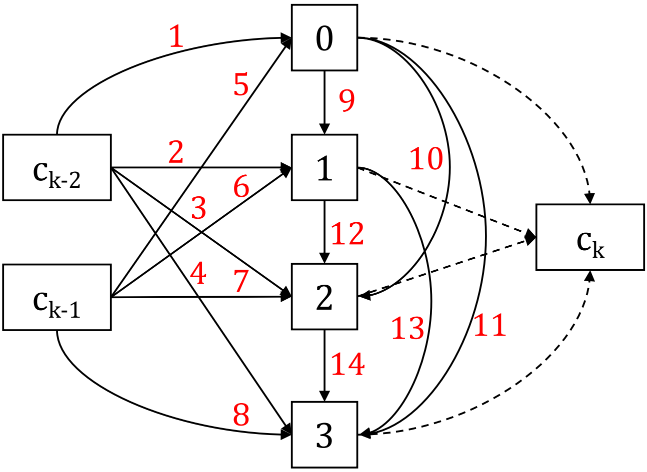


Figure 4: Cell structure. Each box denotes a node and every solid line represents an edge i.e., the mixed operation.

Operation	**Abbreviation**
max_pool_3x3	max3
avg_pool_3x3	avg3
skip_connect	skip
sep_conv_3x3	sep3
sep_conv_5x5	sep5
dil_conv_3x3	dil3
dil_conv_5x5	dil5

Table 4: Candidate operations.

A.3 Distance Loss

In this work, we propose to calculate the distance loss between architecture parameters and their one-hot counterpart. There are four types of distance loss investigated in our method: MAE, MSE, RMSE, and AoI.

The loss of mean absolute error (MAE) is calculated as follows:

$$\mathcal{L}_{MAE} = \frac{1}{\mathcal{N}} * \sum_{i}^{\mathcal{M}*\mathcal{N}} |\alpha_i - \hat{\alpha}_i| \tag{8}$$

The loss of mean squared error (MSE) is calculated as follows:

$$\mathcal{L}_{MSE} = \frac{1}{\mathcal{N}} * \sum_{i}^{\mathcal{M}*\mathcal{N}} (\alpha_i - \hat{\alpha}_i)^2 \tag{9}$$

The loss of root mean squared error (RMSE) is calculated as follows:

$$\mathcal{L}_{RMSE} = \sqrt{\frac{1}{\mathcal{N}} * \sum_{i}^{\mathcal{M}*\mathcal{N}} (\alpha_i - \hat{\alpha}_i)^2} \tag{10}$$

The loss of amount of information (AoI) is given by:

$$\mathcal{L}_{AoI} = \frac{1}{\mathcal{N}} * \sum \log_2(\alpha \cdot \hat{\alpha}) \tag{11}$$

B Experiments

B.1 More Implementation Details

When we search for the architectures on CIFAR-10 and CIFAR-100, we normalize the distance loss per edge to get a similar magnitude with the original loss value. As for the information entropy loss, we utilize its maximum to normalize it to the range of $[0,1]$. We initialize the $\beta = 1.95$ and thus $Sigmoid(\beta) \approx 0.875$, which makes the $Softmax(\alpha) \approx 0.125$ for each candidate operation and keeps the same with DARTS [2]. Other hyper-parameters and settings are also the same as DARTS.

When retraining the searched architecture on ImageNet, we use 8 NVIDIA Tesla V100 GPUs and spend about 3.5 days. Basic image processing skills are adopted and we also apply the auxiliary loss to help training. All settings and hyper-parameters are kept the same as SGAS [1].

B.2 Ablation Study

Entropy Loss	Gate Switch	Normal Depth	Reduction Depth	Test Error(%)
×	×	3.0	3.0	3.00±0.14†
×	√	4.0	4.0	2.84±0.16
√	×	2.0	3.0	2.62±0.05
√	√	4.0	3.0	**2.38±0.06**

Table 5: Analysis of the role of each component in our method. †: The result is from DARTS [2].

To analyze the role of each component in our method, we conduct the ablation study in Tab. 5 with 3 independent runs. It is obvious that DARTS has the highest test error rate. Either replacing the *zero* operation with a gate switch or adding the entropy loss for architecture parameters improves the performance and the latter plays a more critical role, showing that decreasing the uncertainty of NAS is very necessary. Moreover, with the two proposed techniques together, we can achieve the best performance than individually deploying each technique.

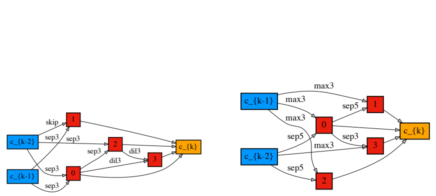
Additionally, it is worth noting that we also calculate the depth of the searched cell following [3]. With the single entropy loss, the searched cell has the smallest depth due to the dominance of the *zero* operation and the gate switch can greatly improve the cell depth, which is consistent with our analysis in Sec.3.4. Furthermore, by combining the two proposed techniques, DU-DARTS can search out architectures with moderate cell depth and the lowest test error rate, which demonstrates the superiority of our method.

B.3 Searched Cells

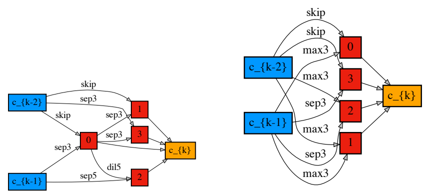
We visualize our searched cells on CIFAR-10 in Fig. 5 and all the searched cells on CIFAR-100 are shown in Fig. 6.

B.4 Evolution of Architecture Parameters

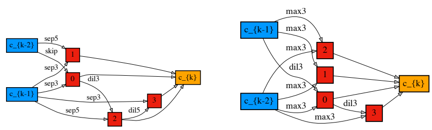
During the search process, we record architecture parameters of all edges and plot these data in Fig. 7 and Fig. 8, where each sub-figure corresponds to an evolution of architecture parameters of one edge. It is obvious that in most cases only one architecture parameter eventually approaches the maximum with clear superiority, which is then considered as the truly important operation in this work.



(a) Seed=0

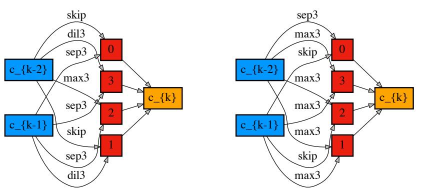


(b) Seed=1

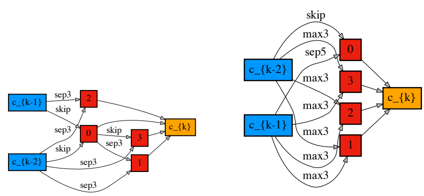


(c) Seed=2

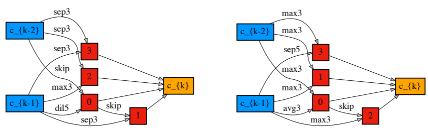
Figure 5: Our searched normal cells and reduction cells with different random seeds on CIFAR-10.



(a) Seed=0

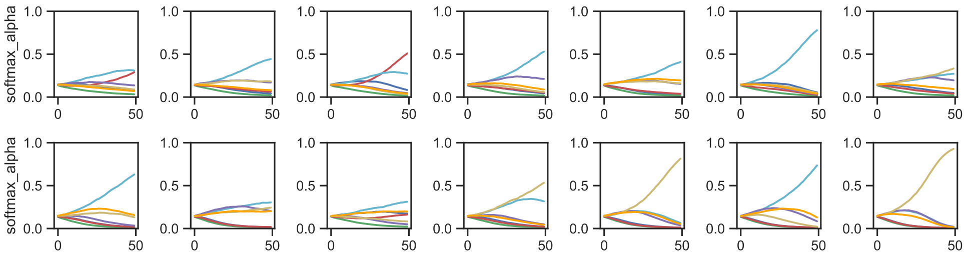


(b) Seed=1

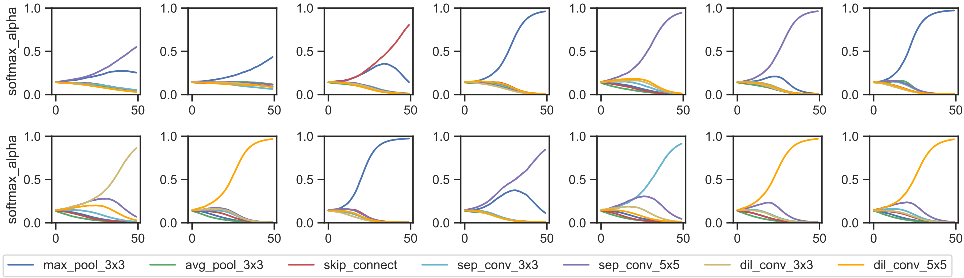


(c) Seed=2

Figure 6: Our searched normal cells and reduction cells with different random seeds on CIFAR-100.

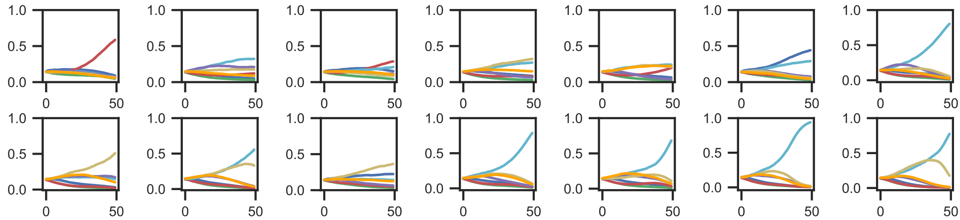


(a) Normal Cell

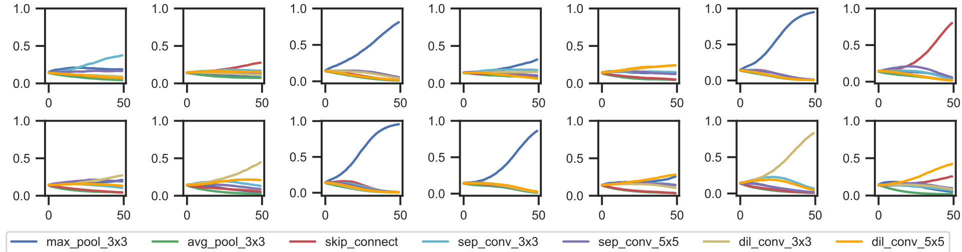


(b) Reduction Cell

Figure 7: Evolution of architecture parameters during search on CIFAR-10 with seed=0. X-axis: epoch. Y-axis: softmax alpha.



(a) Normal Cell



(b) Reduction Cell

Figure 8: Evolution of architecture parameters during search on CIFAR-100 with seed=0. X-axis: epoch. Y-axis: softmax alpha.

References

[1] Guohao Li, Guocheng Qian, Itzel C Delgadillo, Matthias Muller, Ali Thabet, and Bernard Ghanem. Sgas: Sequential greedy architecture search. In *CVPR*, 2020.

[2] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable architecture search. In *ICLR*, 2019.

[3] Yao Shu, Wei Wang, and Shaofeng Cai. Understanding architectures learnt by cell-based neural architecture search. In *ICLR*, 2019.