One-Shot Neural Ensemble Architecture Search by Diversity-Guided Search Space Shrinking
—— Supplementary Material ——

Minghao Chen[1]*, Jianlong Fu[2], Haibin Ling[1]

[1]Stony Brook University [2]Microsoft Research Asia

{minghao.chen,haibin.ling}@stonybrook.edu, jianf@microsoft.com

Appendix A

In this appendix, we include: (I) proof of the property stated in Section 3.2, (II) the detailed supernet structure and search space.

A-I: Proof of Diversity Score Property

In this section, we show a more detailed formula of the property stated in Section 3.2 and the proof of the property.

Property: Assume that $h_m := (o_{1,m}, \cdots, o_{j,m}, \cdots, o_{K,m})$ and $h_m^{'} := (o_{1,m}, \cdots, o_{j,m}^{'}, \cdots, o_{K,m})$ are different only by j_{th} operator. Denote the indexes of operators in h_m and $h_m^{'}$ as $\sigma_1, \sigma_2, \cdots, \sigma_K$ and $\sigma_1^{'}, \sigma_2^{'}, \cdots, \sigma_K^{'}$. If $S_{i,k}^m < S_{i^{'},k}^m$ for $k = 1, 2, \cdots, K$ and $r_i^m > r_{i^{'}}^m$, then we have:

$$Score(h_m) > Score(h_m^{'}), \tag{1}$$

where σ_j and $\sigma_j^{'}$ equal to i and $i^{'}$.

Proof: Given the property of matrix determinant and definition of L_m^y, the diversity score of h_m could be expressed as:

$$Score(h_m) = \prod_{i=1}^{K} r_{\sigma_i}^2 \cdot \det(S_m^y). \tag{2}$$

where S_m^y are the corresponding submatrixs of h_m in S_m.

According to the assumption, we know that $\prod_{i=1}^{K} r_{\sigma_i}^2 > \prod_{i=1}^{K} r_{\sigma_i^{'}}^2$. Now, if $\det(S_m^y)$ is greater than $\det(S_m^{y^{'}})$ then the property holds easily. Because h_m and $h_m^{'}$ are only different by the j_{th} operator and S_m is a symmetry matrix, the number of total different entries between S_m^y and $S_m^{y^{'}}$ is less than $2K$. We could construct a series of matrixs $B_i \in \mathbb{R}^{K \times K}, i = 0, 1, 2, \cdots, K$ as following:

$$B_i(k,l) = \begin{cases} S_m^y(k,l), & k < i, l = j, \\ S_m^y(k,l), & l < i, k = j, \\ S_m^{y^{'}}(k,l), & Otherwise, \end{cases} \tag{3}$$

where $B_i(k,l)$ is the entry in row k column l. We then prove the following inequality by induction:

$$\det(B_i) \leq \det(B_{i+1}), i = 0, 1, 2, \cdots K-1. \tag{4}$$

*This work is done when Minghao is an intern at Microsoft.

For $i = 0$, consider matrix A defined as follow:

$$A(k,l) = \begin{cases} \dfrac{S_m^y(1,j)}{S_m^{y'}(1,j)}, & k = 1, l = j, \\ \dfrac{S_m^y(j,1)}{S_m^{y'}(j,1)}, & l = 1, k = j, \\ 1, & Otherwise. \end{cases} \tag{5}$$

Given the assumption that $S_{i,k}^m < S_{i',k}^m$ for $k = 1, 2, \cdots, K$, we have $S_m^y(1,j) < S_m^{y'}(1,j)$. Then we could get A is a positive define matrix easily using the definition of positive define matrixs. Regarding A and B_0 are both semi-positive define matrix, we have following statement using **Oppenheim's inequality**:

$$\det(A \circ B_0) = \det(B_1) \geq \det(B_0) \prod_i^K A(i,i) = \det(B_0), \tag{6}$$

where $A \circ B_0$ is the **Hadamard product** (element-wise product) of A and B_0. Besides, B_1 is also a semi-positive define matrix according to **Schur product theorem**.

For $i = 1, 2, \cdots, K-1$, it is easy to construct A with similar definition like above and get the statement that $\det(B_i) \leq \det(B_{i+1})$. Now, combining the chain of inequality, we have:

$$\det(S_m^y) = \det(B_{K-1}) \geq \det(B_0) = \det(S_m^{y'}). \tag{7}$$

Using Eq. (2)(7), the property holds easily.

A-II: Supernet Structure and Search Space

In this section we give the detailed supernet structer and space of the new dimension *Splint Point*.

Input Shape	Operators	Channels	Repeat	Stride
$224^2 \times 3$	3×3 Conv	16	1	2
$112^2 \times 16$	3×3 Depthwise Separable Conv	16	1	2
$56^2 \times 16$	MBConv / SkipConnect	24	4	2
$28^2 \times 24$	MBConv / SkipConnect	40	4	2
$14^2 \times 40$	MBConv / SkipConnect	80	4	1
$14^2 \times 80$	MBConv / SkipConnect	112	4	2
$7^2 \times 112$	MBConv / SkipConnect	160	4	1
$7^2 \times 160$	1×1 Conv	960	1	1
$7^2 \times 960$	Global Avg. Pooling	960	1	-
960	1×1 Conv	1,280	1	1
$1,280$	Fully Connect	1,000	1	-
Split Point		(9, 20, 1)		

Table 1. The structure of the supernet. The "MBConv" contains 6 inverted bottleneck residual block MBConv [1] (kernel sizes of {3,5,7}) with the squeeze and excitation module (expansion rates {4,6}). The "Repeat" represents the maximum number of repeated blocks in a group. The "Stride" indicates the convolutional stride of the first block in each repeated group. (9, 20, 1) means space starts from 9 to 20 with a step of 1.

Appendix B

In appendix B, we show the detailed evolution algorithm, with the detailed algorithm of K-path evolution search below. Specific steps of $\mathrm{Crossover}, \mathrm{Mutation}$ are presented in Section 3.4.

Algorithm 1 K-Path Evolution Search

Input:
Shrunk search space $\tilde{\mathcal{S}}$, weights $W_{\tilde{\mathcal{S}}}$, population size P, resources constraints C, number of generation iteration $\mathcal{T}$, validation dataset D_{val}, training dataset D_{train}, Mutation probability of split point P_s, Mutation probability of layer combination P_m.

Output: The most promising ensemble architecture $\mathcal{A}^*$.

$G_{(0)} :=$ Random sample P ensemble architectures $\{\mathcal{A}_1, \mathcal{A}_2, \cdots \mathcal{A}_P\}$ from $\tilde{\mathcal{S}}$ with constrain C;
while search step $t \in (0, \mathcal{T})$ **do**
  while $\mathcal{A}_i \in G_{(t)}$ **do**
    Recalculate the statistics of BN on D_{train};
    Obtain the accuracy of $\Phi(\cdot; \mathcal{A}_i, W_{\tilde{\mathcal{S}}})$ on D_{val}.
  end while
  $G_{\mathrm{topk}} :=$ the Top K candidates by accuracy order;
  $G_{\mathrm{crossover}} := \mathrm{Crossover}(G_{\mathrm{topk}}, \tilde{\mathcal{S}}, C)$;
  $G_{\mathrm{mutation}} := \mathrm{Mutation}(G_{\mathrm{topk}}, P_s, P_m, \tilde{\mathcal{S}}, C)$;
  $G_{(t+1)} = G_{\mathrm{crossover}} \cup G_{\mathrm{mutation}}$
end while

References

[1] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018.