Deep Active Learning with a Neural Architecture Search

Yonatan Geifman
Technion – Israel Institute of Technology
yonatan.g@cs.technion.ac.il

Ran El-Yaniv
Technion – Israel Institute of Technology
rani@cs.technion.ac.il

目录

Problem setting

- AL usually **depends on the model output**, while many real tasks does not have a off-the-shelf working model architecture.
- **Expressive architecture** will tend to **overfit** when trained over a small sample (early stage in AL).

The model architecture should be optimized along with the AL process.

Motivation

- Complex models require more training data
- When the training data is limited, we should use a simple model to avoid over fitting.

The capacity of the architectural should:

- Start small.
- Monotonically increasing along the active learning process.

Pipeline

Algorithm 2 Deep Active Learning with iNAS

active-iNAS(U,A_0, $\mathcal{A}$, Q, b, k)
$t \leftarrow 1$
$S_t \leftarrow$ **Sample** k **points from** U **at random**
$U_0 \leftarrow U \backslash S_1$
while true do
    $A_t \leftarrow$ **iNAS**(S, A_{t-1}, $\mathcal{A}$)
    train $f_\theta \in A_t$ **using** S
    if budget exhausted or $U_t = \emptyset$ **then**
        Return f_θ
    end if
    $S' \leftarrow Q(f_\theta, S_t, U_t, b)$
    $S_{t+1} \leftarrow S_t \cup S'$
    $U_{t+1} \leftarrow U_t \backslash S'$
    $t \leftarrow t + 1$
end while

- Search model
- Query based on the current model

Methodology

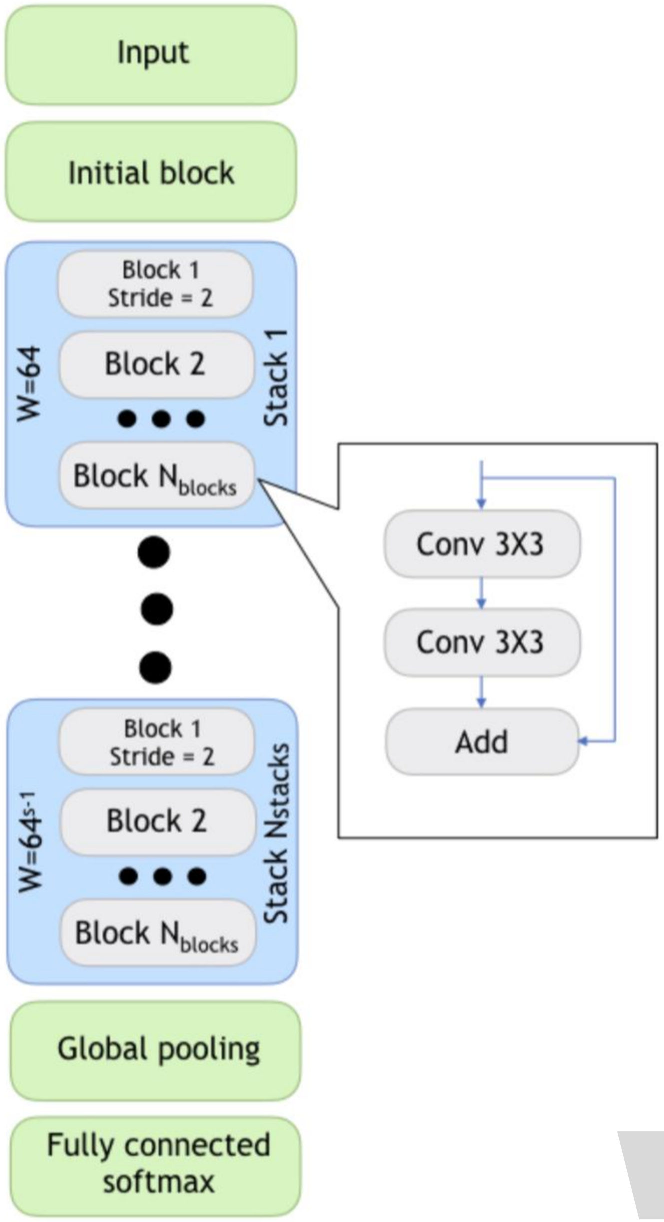
How to estimate the model complexity?

- Use the **depth** of the network.

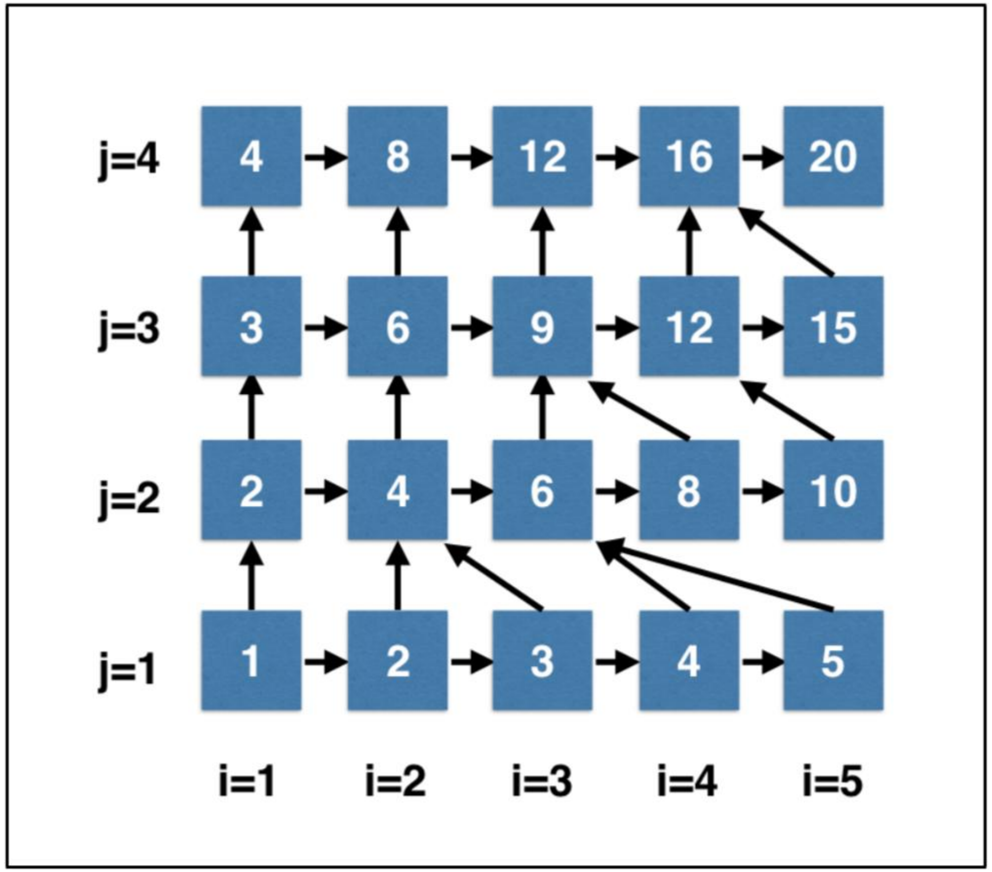
“homogenous” model architectures

- architectures composed of **a single block type** and with **each stack containing the same number of blocks.**
- Denoted as $A\,(B,\ \underline{N_{blocks}},\ \underline{N_{stacks}})$

Search



Search Space as an Acyclic Directed Graph (DAG)



A search space up to A(B, 5, 4) plotted on a grid.

Algorithm 1 iNAS

iNAS(S,$A(B, i_0, j_0)$, $\mathcal{A}$, T_{iNAS})

Let S', V' be an train-test random split of S

for t=1:T_{iNAS} **do**:

  $i \leftarrow i_{t-1}; j \leftarrow j_{t-1}$

  $$\mathcal{A}' = \{ \quad A(B, i, j), \quad A(B, \lfloor \tfrac{ij}{j+1} \rfloor + 1, j+1), \quad A(B, i+1, j)\}$$

  $\mathcal{A}' = \mathcal{A}' \cap \mathcal{A}$

  $$A(B, i_t, j_t) = \; = \operatorname{argmin}_{A \in \mathcal{A}'} \hat{r}_{V'}(\operatorname{argmin}_{f_\theta \in A} \hat{r}_{S'}(f_\theta))$$

  if $A(B, i_t, j_t) = A(B, i_{t-1}, j_{t-1})$ **then**

    break

  end if

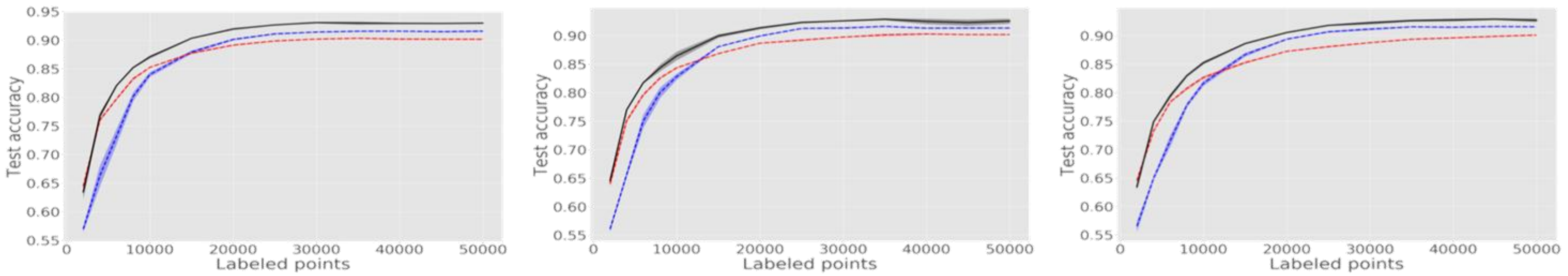
end for

Return $A(B, i_t, j_t)$

Experiments

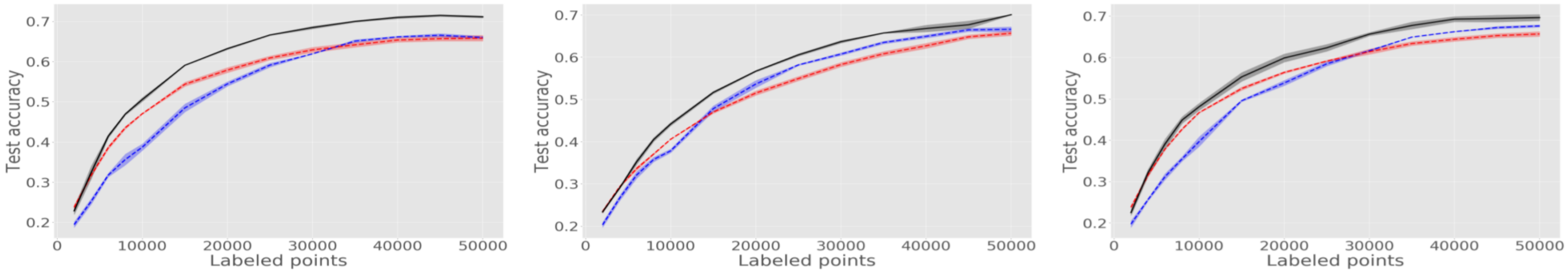
- Resnet-18 $(A(B_r, 2, 4))$
- $A(B_r, 1, 2)$
- Active-iNAS $(A(B_r, 1, 1) \rightarrow A(B_r, 12, 5))$

- CIFAR-10
- CIFAR-100
- SVNH



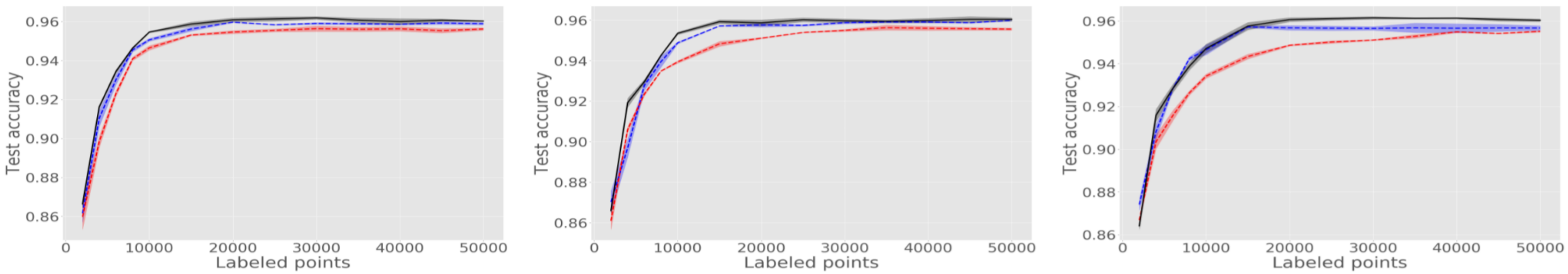
(a) Softmax response (b) MC-dropout (c) Coreset

Figure 2: Active learning curves for CIFAR-10 dataset using various query functions, (a) softmax response, (b) MC-dopout, (c) coreset. In black (solid) – Active-iNAS (ours), blue (dashed) – Resnet-18 fixed architecture, and red (dashed) – $A(B_r, 1, 2)$ fixed.



(a) Softmax response (b) MC-dropout (c) Coreset

Figure 3: Active learning curves for CIFAR-100 dataset using various query functions, (a) softmax response, (b) MC-dopout, (c) coreset. In black (solid) – Active-iNAS (ours), blue (dashed) – Resnet-18 fixed architecture, and red (dashed) – $A(B_r, 1, 2)$ fixed.



(a) Softmax response (b) MC-dropout (c) Coreset

Figure 4: Active learning curves for SVHN dataset using various query functions, (a) softmax response, (b) MC-dopout, (c) coreset. In black (solid) – Active-iNAS (ours), blue (dashed) – Resnet-18 fixed architecture, and red (dashed) – $A(B_r, 1, 2)$ fixed.

Comparison of active-iNAS for various query functions

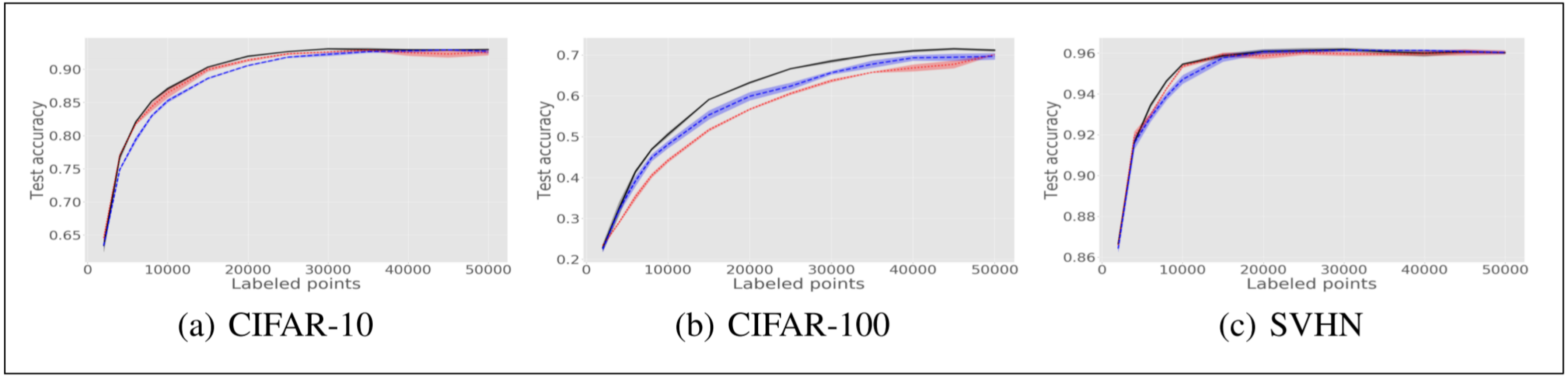
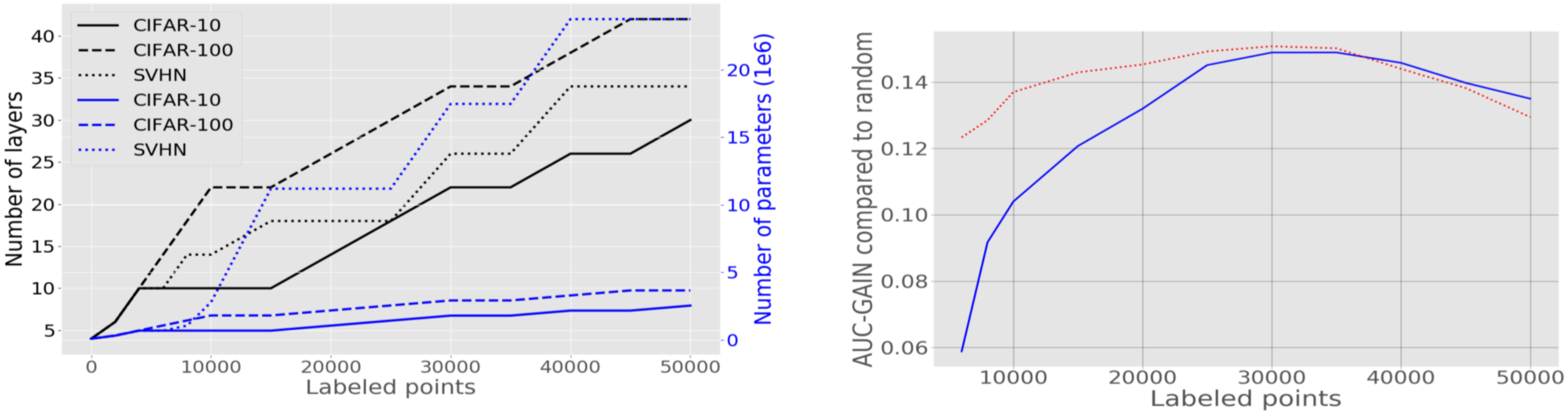


Figure 5: Comparison of active-iNAS for various query functions across three datasets, (a) CIFAR10, (b) CIFAR-100, (c) SVHN. In black (solid) – softmax response, red (dashed) – MC-dropout, and blue (dashed) – coreset.

Model complexity



(a) (b)

(b) Comparison of AUC-GAIN for softmax response active learning over CIFAR-10 for two architectures. In red (solid), the small architecture (A(Br, 1, 2) and in blue (dashed)
the Resnet-18 architecture (A(Br, 2, 4)).

感谢聆听！