Late Breaking Abstract

Self-Organized CNNs using MetaNCA with Attention

Meet Barot

Mythos Scientific, New York, USA

Objectives

In prior work [1], we introduced Meta-Neural Cellular Automata (MetaNCA) as a way to learn the rules of self-organization for artificial neural networks. The local rules are given by a neural network that learns to update weights of a "task neural network" to perform a classification task. Once trained, the local rule network allows us to efficiently sample many task networks of arbitrary architectures without backpropagation. In this work, we present a novel Weight Transformer architecture for the local rule network, which uses linear attention to aggregate signals from neighboring weights and hidden states. We show that the Weight Transformer can grow the weights of unseen convolutional architectures of 2 million parameters to reach 97% accuracy on the MNIST dataset.

Method Description

- Dataset $(x, y) \in D$ for a task $\mathcal{T}$.
- Task neural network that makes predictions for this task $\hat{y} = T(\theta_T; x)$ with weights $w \in \theta_T$.
- Each weight w has a hidden state vector $\vec{h} \in H_T$.
- Let w's forward and backward neighborhood be denoted as $\mathcal{N}_f(w)$ and $\mathcal{N}_b(w)$.
- Updates to $\theta_T^{(t)}$ and $H_T^{(t)}$ are given by a separate local rule network R, parametrized by θ_R.
- R is trained by updating sampled weights from θ_T, for several update steps, and calculating the loss. The gradient is calculated using backpropagation through time.

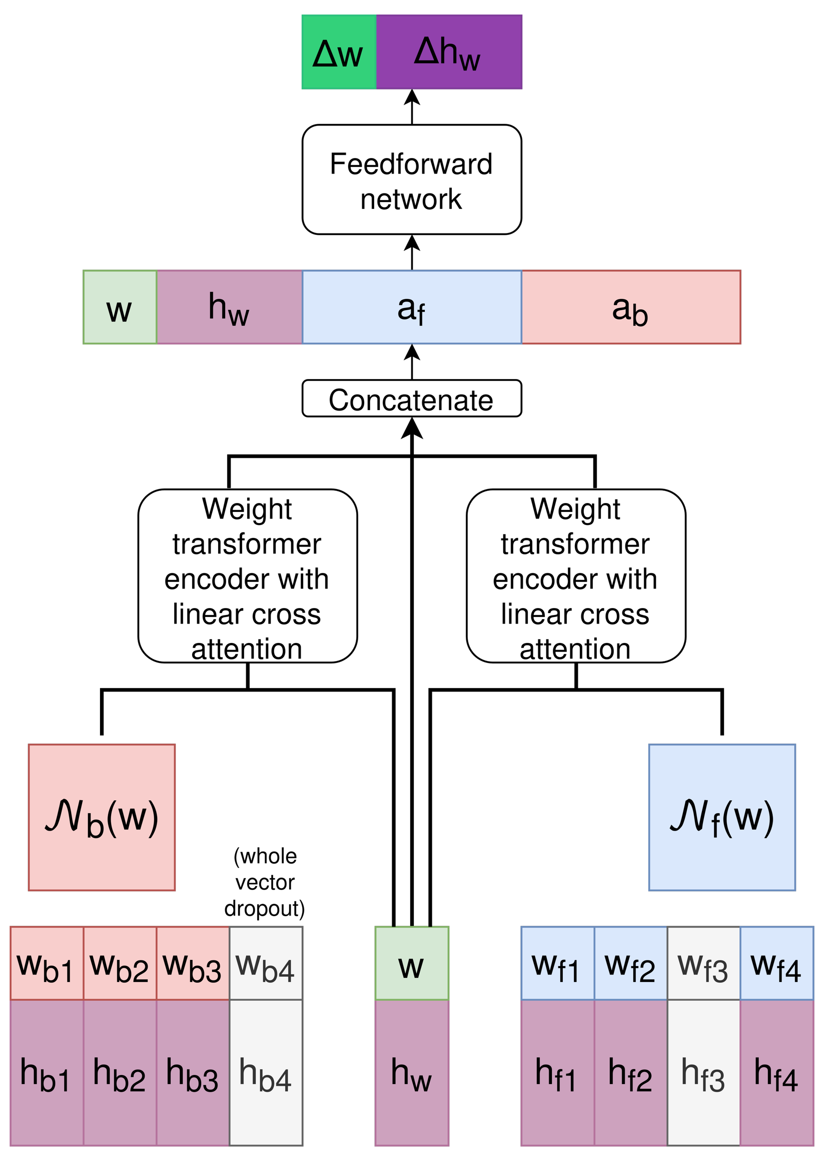

Figure 1:The Weight Transformer architecture which aggregates information from forward and backward neighborhoods with linear attention. Rotational Positional Embedding (RoPE)[2] is used to include relative position information between weights, corresponding to several directionalities: layer, forward neuron, backward neuron, horizontal and vertical kernel position.

Architecture generalization increases with more training architectures

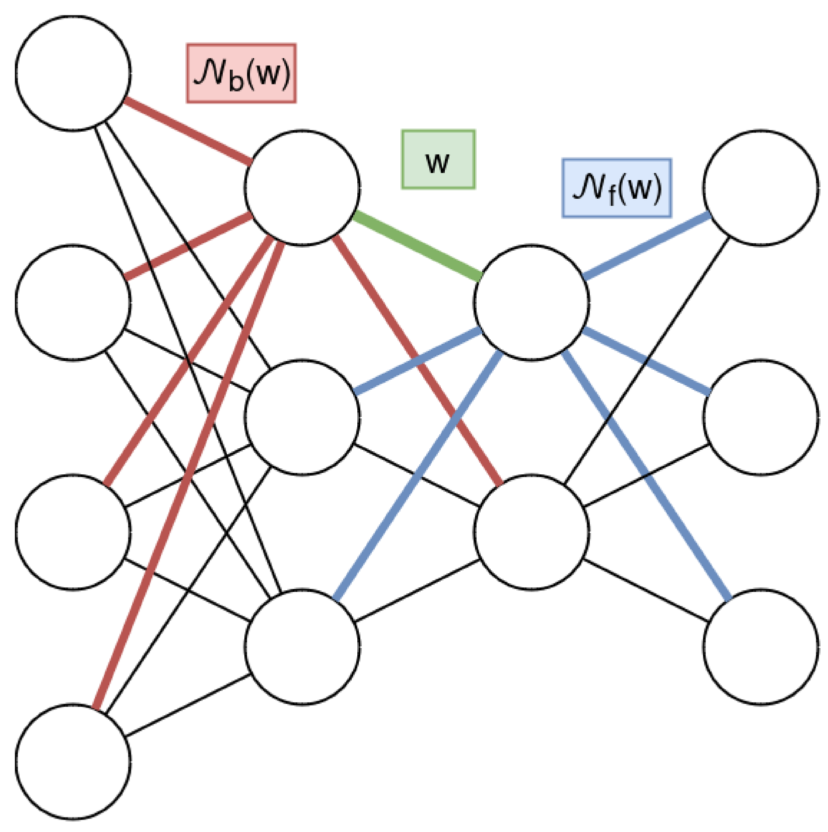

Figure 2:Illustration of the forward and backward neighborhoods $\mathcal{N}_f$ and $\mathcal{N}_b$ of an example weight w.

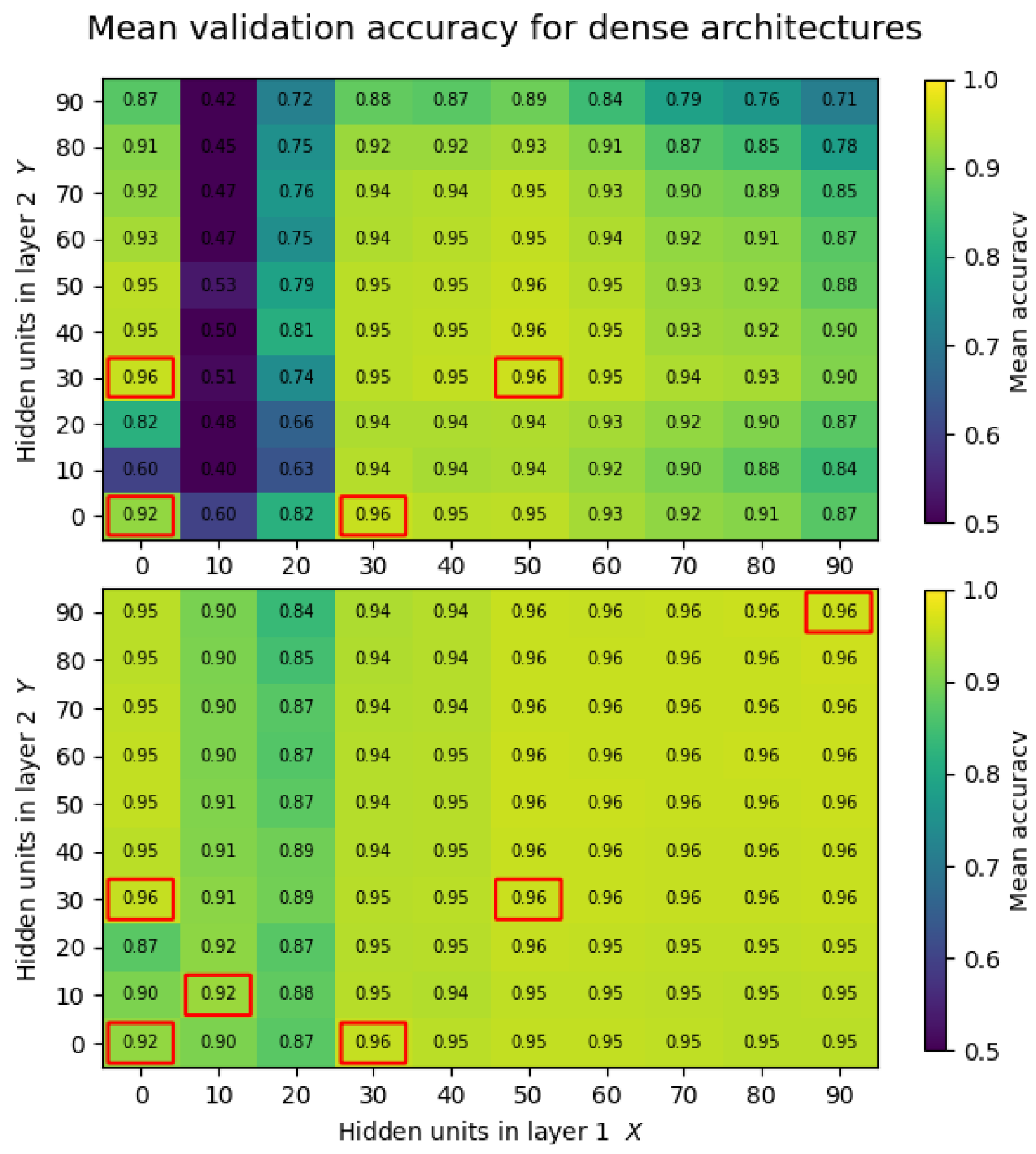

Figure 3:Heatmaps of grown dense networks' MNIST performances, with the training architectures in red rectangles.

Self-organized Convolutional Neural Networks

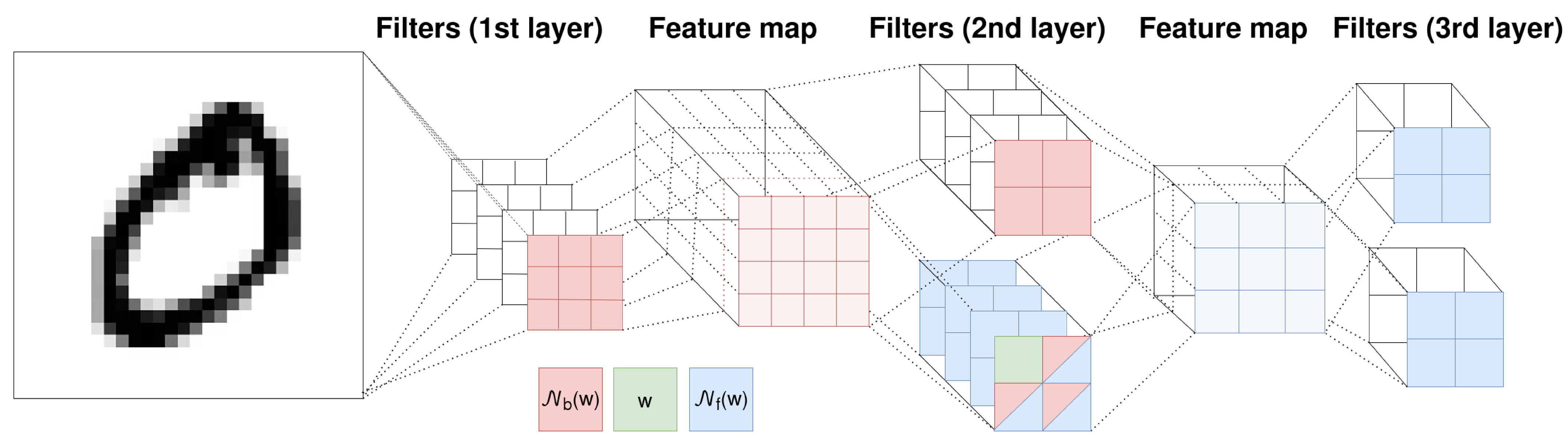

Figure 4:Diagram showing a weight w and its neighborhood in an example CNN's first 3 layers.

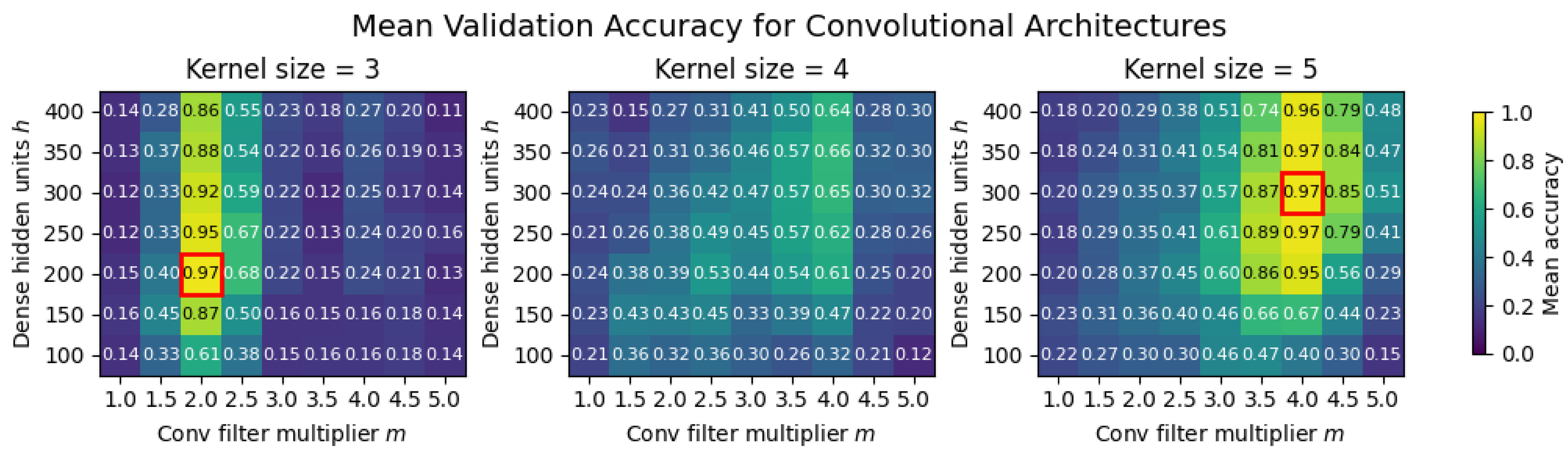

Figure 5:Heatmaps of MNIST performances of grown CNNs, with training architectures indicated in red rectangles. m is the multiplier for the number of filters for a 3-layer CNN with 32*m, 64*m, and 128*m filters. h is the layer dimension after the flattened outputs after the last convolutional layer, before the output layer.

Contact Information

- Web: https://www.meetbarot.com
- Web: https://www.mythos.science
- Email: meet@mythos.science

References

[1] Meet Barot.
Meta-neural cellular automata, 2024.
Late-breaking abstract poster presented at ALIFE2024 in Copenhagen. Available at https://meetbarot.com/alife_2024_poster.pdf.

[2] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu.
Roformer: Enhanced transformer with rotary position embedding.
Neurocomputing, 568:127063, 2024.