Improving transfer using augmented feedback in Progressive Neural Networks

Deepika Bablani
Carnegie Mellon University
dbablani@andrew.cmu.edu

Parth Chadha
Carnegie Mellon University
pchadha@andrew.cmu.edu

Abstract

Learning faster on a task by utilizing learned representations from previous similar tasks is an active area of research in reinforcement learning. Recently proposed progressive neural networks demonstrate this effectively. We use motivations from reciprocal feedback connections in the visual cortex to augment lateral connections in the progressive neural network architecture. We evaluate our modified architecture on Pong-v0 and its variants and show that it improves transfer over the progressive baseline.

1 Introduction

Progressive neural networks [1] utilize learned knowledge from one task to improve learning and convergence speed on another task. Learning on every new task combines previously learned features by learning lateral connections between hidden layers of the new network and those of previous tasks. This creates a columnar architecture that progressively learns a richer representation combining learned features with the feature hierarchy of the new network.

Progressive networks start with a single network such as an A3C [2] network which has L layers with hidden activations h_i^n, for layer i of task n. The network is parameterized by θ^n and has n_i units in layer i. Whenever a new task is learned, the parameters $\theta^{0:n-1}$ of all previous tasks are frozen. The layer h_i^n receives inputs from previous layer of the earlier tasks $h_{i-1}^{0:n-1}$ and the current task h_{i-1}^n. The vector of features that needs to be mapped to the input of hidden layer i is $h_{i-1}^{(<n)} = [h_{i-1}^{(1)} \ldots h_{i-1}^{(j)} \ldots h_{i-1}^{(n-1)}]$. As mentioned in [1], we use non-linear lateral connections termed $adapters$. Each element in $h_{i-1}^{(<n)}$ is scaled by a learned scalar to adjust for different scales of different inputs. The resultant hidden activation of layer i and column k is :

$$h_i^k = f(W_i^k h_{i-1}^k + U_i^{k:j} \sigma(V_i^{(k:j)} \alpha_{i-1}^{(<k)} h_{i-1}^{(<k)})) \tag{1}$$

where $W_i^k \in \mathbb{R}^{n_i \times n_{i-1}}$ is the weight matrix of layer i of task column k, $U_i^{(k:j)} \in \mathbb{R}^{n_i \times n_j}$ are lateral connections from layer $i-1$ of column j to layer i of column k and V is the projection matrix.

We propose architectural changes to progressive neural networks to achieve better transfer and faster convergence on new tasks. Our prime motivation comes from experimental results from neuroscience about connections between information processing layers of the visual cortex. Particularly, we drive inspiration from the observation that higher order concepts from the later layers of visual processing feed back to early visual areas [3]. We introduce additional lateral connections from higher layers to lower layers, which we hypothesize will help in transferring higher learned representation to efficiently learn low level features in the neural network.

2 Methods & Experiments

The traditional view on visual processing in the visual cortex emphasizes a hierarchical information processing pipeline, where the information about edges and bars is extracted from the visual input first, which is then combined into contours, and further combined into more complex forms in higher visual processing areas. This sequential pipeline is the widely accepted classical feed-forward modular view of how the visual cortex processes information. However, there is evidence from neuroscience experiments that higher order stimulus attributes and task experiences influence visual processing in early stages of the pipeline through reciprocal feedback connections from later stages to earlier stages. Specifically, recent neuroscience experiments demonstrate that higher order concepts influence information processing in the lower layers and the top down integration through feedback. The results point towards an incremental integration mechanism comprising of both feedforward and feedback transmission [3] [4] [5].

In our implementation of progressive networks, we added lateral connections from trained higher layers of previous columns to the lower layers of the new task column. The proposed architectural changes are shown in Figure 2.

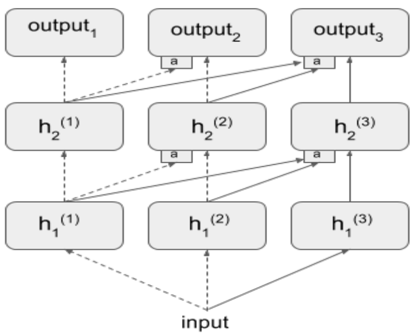

Figure 1: Progressive Neural Network

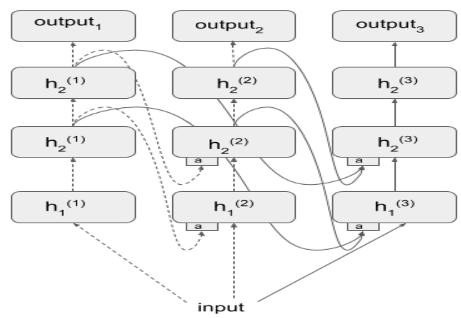

Figure 2: Proposed architecture changes

Since we are transferring from higher layer to a lower layer, we have to learn projection from a lower dimensional to a higher dimensional space. To implement this projection from the higher to the lower layer, we use deconvolution layers [6].

We focused our experiments Pong-v0. We first trained baseline models on this Atari game using A3C with 2 convolution layers followed by a fully connected layer before the policy and value output layers. We used this trained baseline to transfer learning on other tasks. We transferred to 2 variations of Pong-v0 (referred to as 'Pong soup' in the original paper): UDPong (received input state from environment flipped vertically) and LRPong (received state flipped horizontally).

We first trained a two column progressive network where the transfer was from the Pong-v0 baseline. Next, we trained a two column progressive network with only our modified lateral connections, i.e., which only had transfer from the higher to lower layers. We measure the speed of learning based on the final reward achieved by the networks after the same number of training iterations. We observed that using feedback from higher to lower layers, we were able to train the networks on an average 2.1 times faster than the LRPong and UDPong baselines (no transfer) and 1.15 times faster than the LRPong and UDPong baseline progressive network.

Significant improvement in training over the baseline (no transfer) and considerable improvement over baseline progressive neural networks suggest that transferring from a higher layer to a lower layer does indeed provide reliable transfer, i.e., transfer of learned higher level features enables faster learning of lower level representations.

References

[1] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

[2] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy P Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, 2016.

[3] Tai Sing Lee. The visual system's internal model of the world. *Proceedings of the IEEE*, 103(8):1359–1378, 2015.

[4] Minggui Chen, Yin Yan, Xiajing Gong, Charles D Gilbert, Hualou Liang, and Wu Li. Incremental integration of global contours through interplay between visual cortical areas. *Neuron*, 82(3):682–694, 2014.

[5] Tai Sing Lee, Cindy F Yang, Richard D Romero, and David Mumford. Neural activity in early visual cortex reflects behavioral experience and higher-order perceptual saliency. *Nature neuroscience*, 5(6):589–597, 2002.

[6] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2528–2535. IEEE, 2010.