Simulation and Animation of Learning and Development in Recurrent Neural Networks

Asohan Amarasingham
Avi Walsky

Introduction

We present methods for visualizing the activity patterns of a recurrent neural network for the purposes of studying the development of *synfire chains* [1, 2, 3], finely-structured reproducible patterns of spatiotemporal activity in neural networks. The method combines animation and color-coding with conventional visualization by "similarity matrices" to facilitate the rapid understanding of the structural relationships among developmental processes which occur across multiple time scales.

Methods

One of the great difficulties in computational approaches to the study of complex systems like neural networks is the sheer volume of data that the computer produces. Correspondingly, one of the great challenges of visualization is the compression of the sea of data into something the makes interesting, useful relationships visually discernible. We have a network consisting on the order of thousands of neurons, and simulation runs consisting on the order of thousands of time steps. Many times across a simulation run, an event E (a 'seed') occurs. We want to compare the behavior of the network's activity patterns following each occurrence of event E. Further, we want to track the development of this E-triggered behavior across the course of the run. Finally, we want to generalize the approach to handle multiple events, E_1 and E_2, allowing for comparisons across events as well as across time.

There are two ways to look at the nework's activity, one general, one specific, both of which involve finding the similarity between the current and previous time states of the network. The first method is to look at the network as a whole. The basic idea is simple. We construct a similarity matrix to compare the time periods following two occurences of an event E, $a(t_1, t_2) = \frac{\sum_{i=1}^{N} z_i(t_1) z_i(t_2)}{\sqrt{\sum_{i=1}^{N} z_i(t_1) \sum_{i=1}^{N} z_i(t_1)}}$, where $z_i(t_1)$ is a 0/1 binary variable indicating the state of neuron i and time t_1, and t_1 indexes time following the first occurence of E and t_2 indexes time following the second occurrence. We plot the grid by mapping the values of $a(t_1, t_2)$ onto a color map and drawing. We incorporate development by animating sequences of similarity matrices frame by frame while using "color trails" to preserve the visual memory of the last few similarity comparisons along with the current frame.

The second method is concerned with each individual neuron. Here, the activity of each specific neuron is compared directly with its activity at previous seed presentations. Similarities are computed by calculating the conditional expectation $E[z_{i,\text{seed } n}(t) z_{i,\text{seed } n-k}(t) | z_{i,\text{seed } n}(t) + z_{i,\text{seed } n-k}(t) \geq 1]$, which measures the correlation of neuron i's activity patterns following seed n with its activity patterns k seeds earlier. A color is assigned to each weighted similarity and each of the 2,000 neurons are displayed as a square. Successive seed presentations are layered with concentric boxes and animated so as to show the growth over time and local rate of growth simultaneously. Neurons are organized visually so that the neurons that fire together in the final state (last frame) lie next to each other. This method lacks the generality of the first, but allows for specific neuron patterns to be visualized.

Results & Discussion

A frame from each animation is shown in Figure 1. We tested the utility of this visualization approach by testing our ability to quickly disambiguate network behavior across a large range of the model space. The first method provides a precise measure of the stages and rate of learning globally across the network, while the second method allows more careful examination of the spatial patterns of activity over the course of learning and at convergence. The great advantage of the visualizations for the user is the resulting speed of interpretation. In tandem, the user can quickly gauge whether the network has learned, the rate of learning, and the spatiotemporal structure of the reproducible patterns that the network is producing. This is a big improvement over the standard static graph approach.

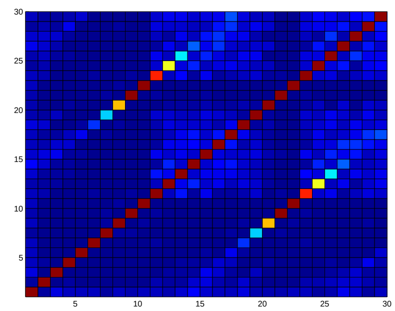

Figure 1: Method 1: Full network correlation

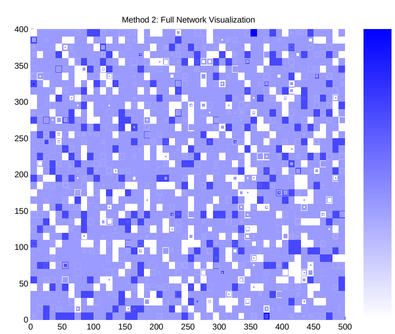

Figure 2: Method 2: Neuron-by-Neuron visualization

References

[1] Amarasingham, A., Levy, W. B., Neural Computation, 10, 25 – 57, 1998.

[2] Abeles, M. Corticonics: Neural Circuits of the Cerebral Cortex. 1991.

[3] Bienenstock, E. A model of neocortex. Network, 6, 179 –224, 1995.