Neural BEAGLE

An analogue to the BEAGLE algorithm using the Nengo framework

Eilene Tomkins-Flanagan

Supervised by Douglas Mewhort & Dorothea Blostein

Abstract

Holographic Reduced Representations (HRRs) are a growing topic of interest in Machine Learning. Originally developed by Tony Plate, HRRs have proven a highly effective tool for modeling the natural ways people interact with semantic information (knowledge about the meaning of things). BEAGLE is an HRR designed to be "semantically rich"; each BEAGLE vector represents the whole history of a word's use. CNRG lab at the University of Waterloo developed Nengo, a piece of software which implements the Neural Engineering Framework and Semantic Pointer Architecture, which together provide an interface between HRRs and biologically realistic neural networks. My thesis project implements an algorithm which reproduces the function of BEAGLE in the Nengo framework.

Background

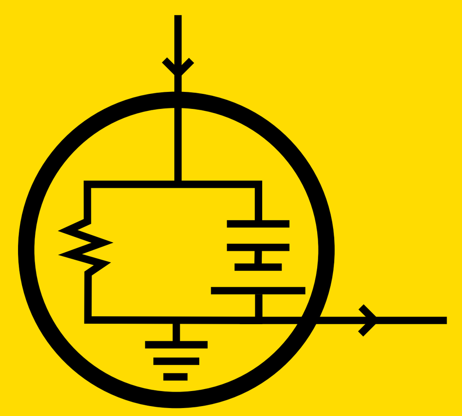
Eliasmith believes that there are useful properties to neurons over using HRRs alone, but are there? Is there use to Nengo besides showing that theorized mechanisms might possibly exist in the brain? I start by asking whether BEAGLE, an algorithm which fits *psychological data* very well, to show that it can exist in Eliasmith's software, and perhaps, begin to bridge the gap between neuroscience and psychology.

NEF

The NEF uses control theory to interface between vectors representing populations of Leaky Integrate-and-Fire neurons and the neural populations themselves. The presence of the neurons amounts to noise in the data, and the modeler can otherwise interact with vectors directly.

Holograms

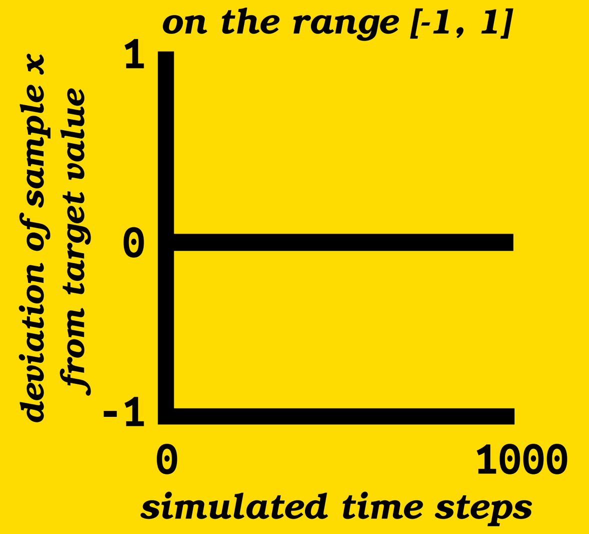
A hologram, superposition, or sum is the weighted sum of two or more vectors, usually on a large number of dimensions. You can reason about them simply using quantum logic.



Leaky Integrate-and-Fire neuron

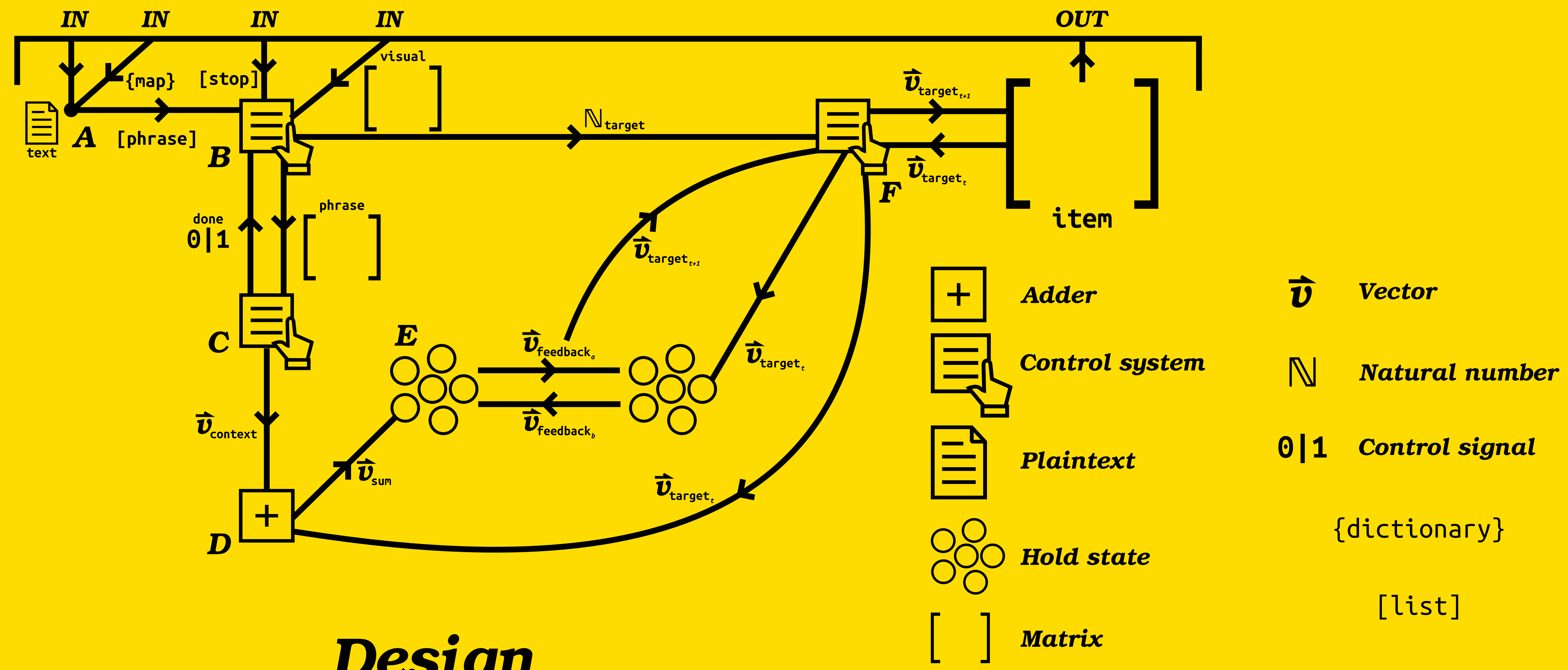
Feasibility & Technical Limitations

Representational instability of Nengo on the range [-1, 1]



Is Nengo up to the task of reproducing the BEAGLE algorithm? At first blush it performs well. Extensive statistical evaluations showed that it could perform the basic mathematical operations BEAGLE requires (addition, circular convolution, and random mapping), with variation on the order of 10^{-4} or less from the target value, and no recorded outliers.

However, other problems present themselves. Available control systems for Nengo haven't been implemented using Neurons, so any model is only partly "biologically realistic". Because Nengo linearly sums neuron tuning curves, any function it computes must be bounded on a finite range.



Design

BEAGLE is not bounded on a finite range; a small test using the original algorithm showed values quickly growing above reasonable maxima, and in principle, BEAGLE vectors tend to grow in magnitude over many training examples. However, magnitude is not usually taken to be important to meaning, what is important is that every training example should matter equally.

BEAGLE generates four vector types: visual, item, order, and lexical vectors. Visual vectors are random Gaussian vectors with σ = 1/√n and are generated once (they will be approximately orthonormal in expectation). BEAGLE generates its other vectors reading documents sentence-by-sentence. For each sentence, if two words x and y are both in it, the visual vector for x is summed into the item vector for y, and vice-versa. Order vectors use circular convolution to represent the order of words in sentences. By applying a different random mapping to each operand, convolution is made non-commutative. In each sentence where a vector x appears, it is temporarily replaced with a placeholder vector Φ so that only context is relevant. The visual vectors for every window of 2-5 words in which x appears is convolved, and all convolutions are summed. The item vector process is diagramatically reproduced using only tools available in Nengo above. Similar design strategies are employed in the order network.

A - a text file is read line-by-line and passed as sequences of words
B - the sentence is read by a control system and appropriate visual vectors are chosen to represent it, target words are selected one-by-one
C - a control system presents each vector to the adder one-by-one and signals when done
D - 0.5 * v[context] + 0.5 * v[target] are added and passed to the state module
E - State modules storing each of the updated and current targets feed into one another competitively until they reach an agreed upon state
F - Using the given target word, a control system presents the current value of the target and updates its item vector when it changes

Discussion

Implementing a functional analogue BEAGLE in Nengo, makes headway in bridging the gap between Psychological and Neuroscientific lines of evidence, and allows us to ask more interesting questions. What techniques can we borrow from neural models to improve upon our own models? What properties do biologically plausible neural networks have that were difficult to acheive using purely analytic methods?

Nengo hasn't yet achieved anything that can't be done with faster analytic methods, but possible extensions to its modeling ideas could push the boundaries of what has previously been possible using just analytic methods. For, how do you learn what a word is?

Eliasmith's new networks present promising new methods of encoding: in neuronal tuning curves. Could that put a holographic store of memory on the horizon?

References

Plate, T. A. (1995). Holographic reduced representations. IEEE Transactions on Neural networks, 6(3), 623-641.

Jones, M. N., & Mewhort, D. J. K. (2007). Representing word meaning and order information in a composite holographic lexicon. Psychological Review, 114(1), 1-37.

Eliasmith, C. (2013). How to build a brain: A neural architecture for biological cognition. Oxford University Press.