Project Report - Cognitive Robotics - Submission 31.January.2025

Agents in the Neural Engineering Framework

Written by Max Wassermann

Agent Implementation

The agent implemented is a neural network-based system designed to predict color sequences in a grid environment. Using the Neural Engineering Framework (NEF) and Semantic Pointer Architecture (SPA), the agent learns to associate sequences of colors as it moves through the grid and makes predictions about upcoming colors based on its learned experiences.

The agent has proximity sensors and basic movement patterns, allowing it to traverse the grid of colored squares (see Figure 1). It also has two sensory inputs for color: A close-range sensor that detects the color of the square the agent is currently on and a far-range sensor that detects the next color in the direction of the agent's movement. Both sensors receive RGB color values with added Gaussian noise ($\sigma = 0.1$) to simulate realistic sensor readings. These RGB values are encoded in a single node each and then transformed into semantic pointers representing discrete colors (WHITE, GREEN, RED, BLUE, MAGENTA, YELLOW) by matching them via the smallest Euclidean distance. The transformation occurs by feeding the nodes current_color and ahead_color into the semantic pointer states model.vision_close and model.vision_far, which are networks consisting of one or more neural ensembles to represent information in a D-dimensional space. Here, the dimensionality was chosen to be $D = 32$, which was large enough to consistently differentiate between the six colors. While transforming the signal into semantic pointers right away is less computationally efficient than working with the RGB signal, this discrete representation prevents similar colors from being confused, which is especially important in the presence of noise.

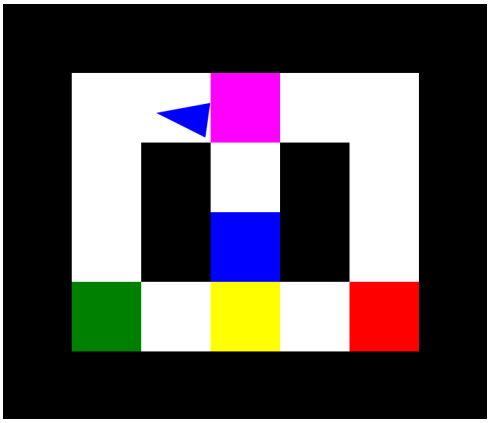

Figure 1 Grid environment the agent was tested in

These modules encode which color the agent is on and which one will come next, however, the goal is to build some sort of learning mechanism that registers which colors follow each other, in order to predict which comes next, without relying on the far-range sensor. This was achieved via a memory system, a learning and a prediction mechanism. The memory system consists of the semantic pointer states model.memory and model.previous, both employing feedback to maintain the signal from the color input even after the square. Additionally, two associative memory modules were used, namely model.cleanup and model.pred_cleanup. These consist of multiple neural ensembles, each tuned to respond to a specific semantic pointer from the input vocabulary, in order to learn patterns of that vocabulary. When an input and a known pattern are similar enough to exceed the specified threshold, the output is transformed into the corresponding "cleaned up" semantic pointer. While cleanup processes and cleans up bound color pairs during learning, model.pred_cleanup does so for the final prediction. A final semantic pointer state, buffer, serves as a binding buffer in the learning pathway. It temporarily holds the result of binding the current color with the next observed color, allowing these associations to be processed and cleaned up by the cleanup memory. In this way, working and cleanup memories work together to establish and maintain learned sequences of colors.

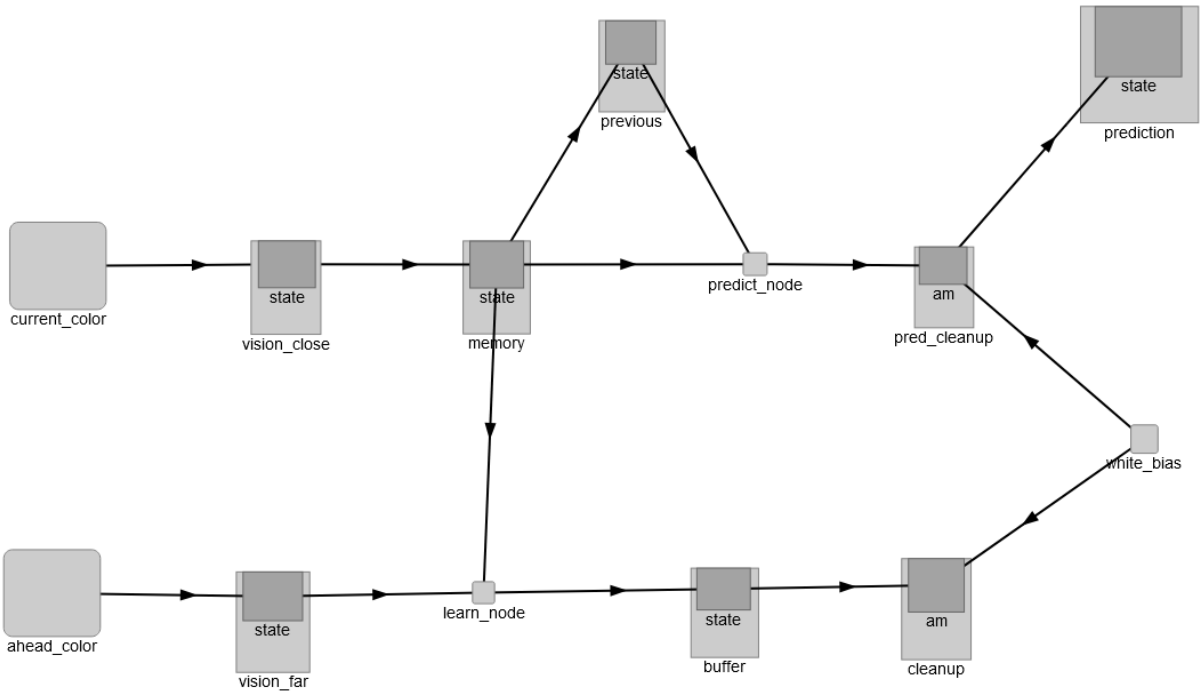

Figure 2 Architecture of the color prediction mechanism

The overall architecture (see Figure 2) is separated into two processes, the learning pathway and the prediction pathway. The former begins with the model.vision_close module, which feeds into the memory state. The current color from memory is then combined with input from model.vision_far in a binding node called learn_node. This node is a pure function without internal state that performs element-wise multiplication between its inputs, whose activity seems to resembles an error signal, at least it has been observed to be very active when a false color was predicted. The result goes through the working state and cleanup memory, establishing associations between sequential

colors. The prediction pathway operates independently of model.vision_far. It takes input from model.memory and model.previous states, combining them in model.predict_node, another pure function performing element-wise multiplication. The result passes through model.pred_cleanup to generate the final prediction. The separation of these pathways ensures predictions are based on learned patterns rather than signals leaking from the model.vision_far sensor. Both pathways use strong transformation values (1.5-2.0x) between components to maintain signal strength through multiple processing stages. Additionally, the system implements a bias against white representations to prevent the dominance of one color, achieved by scaling white representations to 0.3x their normal strength and through adaptive scaling in learn_node and predict_node (white 2.0x scaling, non-white 4.0x scaling). Since white is the background color, the agent spends the majority of its time on it. The anti-white bias was necessary to prevent the system from only ever predicting the next square to be white. As a result the agent is mostly limited to predicting non-white colors.

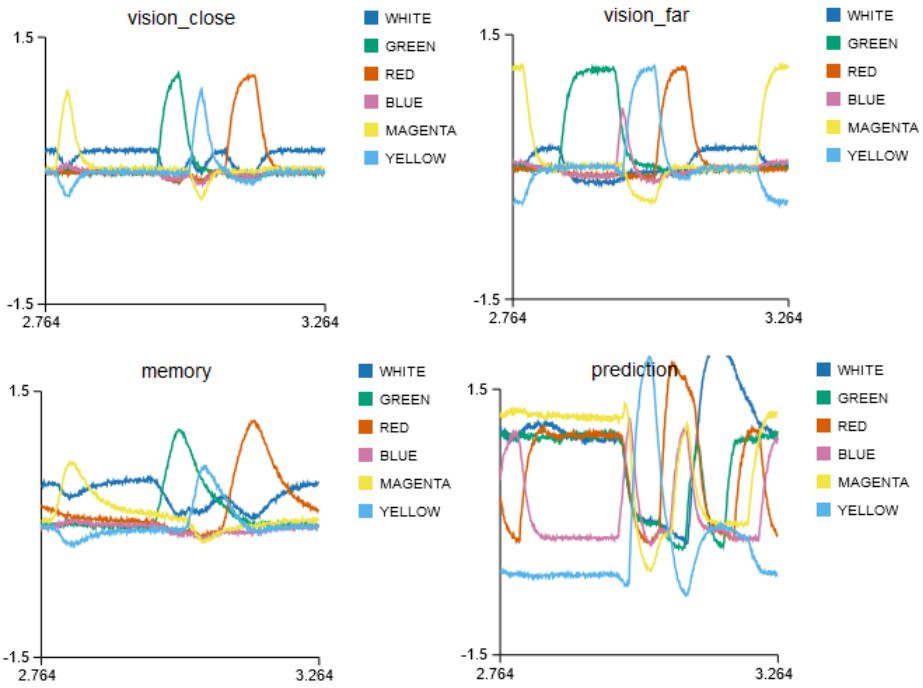

Figure 3 Performance of the color prediction (error while detecting green, rest correct)

The agent is able to predict the non-white colors fairly accurately and adapts to different environments. It can handle the sensor noise without issue and doesn't require much wind up time to adapt to a new environment. Due to the network working almost exclusively in semantic pointer space, the similarity between different colors will cause no mix ups, provided the dimensionality of the space is high enough to differentiate between all colors. The combination of NEF and SPA provides a neurally plausible implementation while maintaining computational tractability. The system's predictive ability is far from perfect, however, usually only correctly predicting three out of four of the non-white colors it encounters in the test environment (see Figure 1). In some cases the activation of the prediction state for the correct guess is only slightly higher than competing guesses (see Figure 3), which could cause issues in noisier environments. Finally, the prediction works well

in an environment as seen in Figure 1, where each color has only one unique color preceding and one unique color following it. An environment with colors appearing in sequence with various other colors would require a more sophisticated learning mechanism. The test environment also works well because there is enough time between non-white colors, while moving quickly through multiple colors would pose a challenge due to the semantic pointer states not being able to catch up. This last point can be seen as a weakness of SPA, as every state needs time to overwrite its old state, severely limiting the speed at which even simple building blocks, such as vision or memory modules, can detect and integrate changes in their inputs.

Reflection

The development of a basic color sequence learning agent using NEF and SPA provides an interesting case study for examining key questions in cognitive robotics. While the implementation is relatively simple, this agent demonstrates fundamental cognitive capabilities such as perception, learning, and prediction, which are essential building blocks for more complex cognitive architectures. Agents like this can serve as valuable minimal models for studying specific cognitive mechanisms in isolation. As argued by Stramandinoli et al. (2012), starting with basic sensorimotor grounding and building up to higher-order concepts through combination and association allows us to systematically investigate how different levels of cognitive capability might emerge. The color sequence agent, while limited, provides a concrete platform for exploring how predictive representations can be learned and grounded in sensorimotor experience, while also highlighting the difficulty of even the most basic functions such as remembering what color came last, when using biologically plausible approaches like NEF or spiking neurons in general. Key challenges for the NEF/SPA approach are scaling and stability – as evidenced by the unstable predictions in the color sequence task. This reflects a broader challenge identified by Eliasmith et al. (2012) in their work on large-scale brain models: Maintaining reliable computation across multiple interacting neural systems remains difficult. The "brittleness" we see in even simple prediction tasks points to the need for more robust neural architectures or a better understanding of producing stability in the interaction of high-dimensional dynamical systems.

While many cognitive architectures have been proposed, none has fully convinced the cognitive robotics research field to satisfy all requirements a model of cognition should have. What exactly these requirements are, is of course subject to debate, however, many of the proposed features a cognitive architecture needs follow similar themes:

1. Robust grounding in sensorimotor experience (Stramandinoli et al., 2012)
2. Systematic compositionality for building higher-order concepts (Cangelosi, 2010)
3. Flexible learning mechanisms that can extract patterns while maintaining stability (Edelman, 2015)
4. Scalability to handle multiple cognitive functions working together (Eliasmith et al., 2012)

No single architecture seems to excel at all four of these features. The emergent perspective, exemplified by deep learning systems, excels at extracting patterns from data and is highly scalable but often lacks the systematic compositionality needed for higher reasoning. Meanwhile, cognitivist approaches like ACT-R are much more structured, provide powerful reasoning capabilities but can be inflexible and difficult to ground in real-world interaction (Gratton, 2013). Purely cognitivist approaches have largely been overshadowed by the success of deep learning. At the same time, the lack of systematic compositionality in emergent approaches has been challenged by advances in large language models and the transformer architecture. Given these developments and the fundamental role of experience in learning for both humans and animals, I believe future research in cognitive robotics needs to utilize emergent approaches at least to some extent. While the reasoning capabilities of LLMs are undeniably impressive, deep learning still struggles to handle the unstructured, dynamic nature of real-world data. Moreover, it remains far from biologically plausible in how it represents and processes information.

A promising direction may be hybrid architectures that combine the strengths of both approaches. As demonstrated by the "Spaun" brain model (Eliasmith et al., 2012), structured neural architectures can potentially bridge between emergent sensorimotor processing and more systematic cognitive capabilities. The color sequence agent, in its small way, exemplifies this using structured neural representations (semantic pointers) while allowing for learned associations to emerge through experience. The NEF/SPA toolboxes offer a cognitivist interface with which symbolic-like operations can be defined in discrete steps. While the user can design a system in a cognitivist way, the implementation under the hood is much more emergent, driven by neural dynamics, feedback loops, and real-time interactions. These frameworks have made biologically plausible modeling more accessible, enabling researchers to test cognitive theories in ways that were previously impractical. As Edelman (2015) argues, progress in cognitive robotics may require reconsidering key assumptions about both emergent and cognitivist approaches, however, rather than waiting for entirely new frameworks, we may already possess the necessary tools. What is needed now is a better understanding of how to use the existing frameworks effectively and more bright minds developing innovative cognitive models. With continued research and interdisciplinary cooperation, hybrid architectures could provide a viable path toward more adaptive, scalable, and biologically inspired cognitive systems.

References

Cangelosi, A. (2010). Grounding language in action and perception: From cognitive agents to humanoid robots. *Physics of Life Reviews*, *7*(2), 139–151. https://doi.org/10.1016/j.plrev.2010.02.001

Edelman, S. (2015). The minority report: Some common assumptions to reconsider in the modelling of the brain and behaviour. *Journal of Experimental Theoretical Artificial Intelligence*, *28*(4), 751–776. https://doi.org/10.1080/0952813x.2015.1042534

Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., & Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science*, *338*(6111), 1202–1205. https://doi.org/10.1126/science.1225266

Gratton, M. J. (2013, January 1). *Cognitivist and emergent cognition - an alternative perspective*. Springer. https://doi.org/10.1007/978-3-642-39521-5_22

Stramandinoli, F., Marocco, D., & Cangelosi, A. (2012). The grounding of higher order concepts in action and language: A cognitive robotics model. *Neural Networks*, *32*, 165–173. https://doi.org/10.1016/j.neunet.2012.02.012