Symbol Based Self Guided Neural Architecture Search in a Spiking Neural Network

1[st] Tofara Moyo
Bulawayo , Zimbabwe
tofaramoyo@gmail.com, Mazusa AI

***Abstract*—A spiking neural networks neurons can viewed as feature detectors or alternatively instances of hieroglyphic symbols defined by the associated features they represent .The set of activations at any time step then represent a document written in this alphabet. If we feed this information from the previous time step back to the spiking neural network at each time step , the network will navigate its own space of internal representations and form a grounded language in which to analyze its own internal states and to guide their evolution. We describe this method and how it could be used by the algorithm to plan and design connections and critic its own thought processes if all of this increases the expected reward. We also show a simple method for an agent to learn levels of abstractions ordered by priority that ultimately increase the global expected reward. Each level is associated with a separate scalar output of the neural network at each time step t which is fed back to the agent as part of the state at time t+1. The agent then correlates them with features of the state initially randomly. It however learns the correct assignment by doing it in such a way that it increases the global reward. We describe an equation meant to order these scalar values and the global reward in order of priority and hence induce a heirachy of needs for the agent. This then forms the basis of goal formation for it**

I. Introduction

In this paper we describe a way to make a spiking neural network based agent discover the best firing patterns by jointly navigating the space of firing patterns and the state it is in.

Reinforcement learning process is typically modeled as Markov decision process.Markov decision processes may be visualized as a tuple (X, U, P, D, R). Where X is a finite set of states (here we consider the state as the environment that the agent is in) , U is a set of control inputs, P is a set of state transitions probabilities; D is the initial-state distribution from which the initial state x_0 was drawn; and $R : X => R$ is the reward function.

Our idea includes expanding the inputs to the agent (the state x_t) to include the graph connecting all the neurons involved in the activations at time t_{-1}. This graph will include the eligibility trace e of the activated neurons as the values associated with each node in this graph. The eligibility trace of a spiking neuron is defined with the following equation.

$$e = \partial ln(y|net)/\partial w$$

The eligibility trace contains information of the disposition of the neuron to fire in different states. This is being recorded and parsed to the network. Now instead of representing a collection of abstract data points , this graph now represents a collection of symbols and their associated relationships. This is because each neuron will represent a feature after training. And we can take the set of all neurons as a type of hieroglyphic alphabet.

Parsing this graph to the spiking network gives it the ability to navigate its own internal representations by arranging for them to increase the expected reward.

Ordinarily this is a blind process where the firing pattern being learnt depends on a weight update rule. Now the network has access to a set of instructions (as the graph will represent this if it increases expected reward) in a symbolic format that will guide it towards better and better processes. The graph will contain instructions as well as act as a critic of the activation process if these two qualities increase the expected reward.

This process of reflection on its own internal processes in order to improve them will be optimized as the model learns to refine its symbols and the structure of the syntactic graphs they are embedded on.

Additionally we could learn a procedure for when to connect new connections between neurons and break old ones by adding a it more to the idea. This procedure will be modulated by the neural network.

If we consider the set of neurons as a cyclic group then we could choose to always connect nodes that fire at two consecutive time steps that are related by being a step size of a group generator and weakening those that are not. Moyo et all describe such a process in their paper "An algorithm to optimize the routing of connections in a spiking neural network".

Now these connections are dependent on which neurons activated. If the algorithm has access to the graph of activated neurons it could learn to cause those particular neurons to fire that are separated by the generator to connect that increase the expected reward by using its graphical input.

That means the connection process will move in the direction of increasing the expected reward, which hopefully will correlate with learning and growth of internal components needed to effective solve the reward expectation maximization problem. In fact the algorithm will intentionally plan to place these components it as it communicates with itself.

II. FORMING A HEIRACHY OF NEEDS

Another feature that can be designed is to function to bring increased levels of complexity to the reward signal that the agent receives. Importantly, because of the form of the equation we will use ,this complexity will only increase if it is associated with an increase of the external reward.

In this modification the neural network will output a scalar value v along with its usual actions. This scalar value at actions t will be input into the agent as part of its state alongside the physical state at t_{+1} that it is in and the graph of its activations from time step t. The weight update rule will proportionate updates not by the external reward r on its own. But include the scalar value v in its updates.

The exact form of the update will involve:

$$\Delta w \propto rv + r$$

If you look at this equation, the form of it is designed to encourage the agent to only optimize v if it causes r to increase. Which is what we want from the complexity it introduces in the activations of the spiking neural agent.

Additionally, since we are inputting recurrently the value of v at t to the state at t_{+1} alongside activations graph at t, the algorithm can navigate the space of v outputs and correlate them with the state of activations that caused v.

This allows the algorithm to form theories on what would increase v only if forming those particular theories increases r and modulates v to increase r. These theories are essentially a heir-achy of needs that the agent will develop in its quest to jointly increase v and r.

We believe this will not stop at a one level heir-achy if we further include new scalar outputs v_2 and v_3 there will be a three step heir-achy formed when using an equation of the following form.

$$\Delta w \propto v_3(v_2(rv + r) + (rv + r)) + (v_2(rv + r) + (rv + r)) + ((rv + r) + r) + r$$

This will produce a heir-achy of needs that has three tiers. Importantly the agent will output three scalar values $v, v2$ and $v3$. And they will be parsed back to the algorithm as part of the state in the next time step.

Optimizing the top level will only happen if this optimizes the second and so on down to the global reward. Also the algorithm will learn what neural correlates optimize which tier and develop them in-such a way that they optimize the second equation. Involving a three stage heir-achy of needs that the agent uses to order its priorities when interacting with the physical state.

Increasing v, v_2 and v_3 outputs will all at first correlate with random features in the physical state. But as the algorithm learns to control them it will reassign them in order of importance to features in the real world that ultimately increase the expected reward r.

III. CONCLUSION

We have described a way for a spiking neural network to reflect on its own thought processes and to use this reflection to optimise it in a grounded meaningful manner using symbols. These symbols being based off of the features that the neurons detect. This could lead to efficient architecture search that is meditated upon by the algorithm and whose flaws are critiqued by its self through out the process.

We also show an update rule that will cause a spiking based agent to form theories on what would increase its expected reward. importantly those theories involve three abstractions of state visits that increase three dependent scalar values. Their dependency causing these abstractions to be ordered in a pyramid similar to Maslows heir-achy of needs.

At first the abstractions will be randomly assigned, but as it learns to control them such that they satisfy the update rule and increase the global reward ultimately, the abstractions start to take on meaning full attributes .

These abstractions being the basis for the formation of goals and planning in the agents life time.

REFERENCES

[1] An algorithm to optimise the routing of neurons in an artificial spiking network. -Tofara Moyo

[2] Representation learning for neural population activity with Neural Data Transformers Joel Ye, Chethan Pandarinath

[3] The Geometry of Information Coding in Correlated Neural Populations Rava Azeredo da Silveira, Fred Rieke

[4] Grandmother cells and localist representations: a review of current thinking Jeffrey S. Bowers[5] A. Punnappurat

[5] Mathematics and group theory in music Athanase Papadopoulos (IRMA)

[6] Unsolved Problems in Group Theory. The Kourovka Notebook E. I. Khukhro, V. D. Mazurov

[7] A Group-theoretic view of Music-Dimitris ChatzisDimitris Chatzis

[8] [1710.10686] Regularization for Deep Learning: A Taxonomy - Authors: Kukačka, Jan, Golkov, Vladimir, Cremers, Daniel
[9] [2208.04413] An automatic L1-based regularization method for the analysis of FFC dispersion profiles with quadrupolar peaks Authors: Landi, Germana, Spinelli, Giovanni V., Zama, Fabia…
[10] [2108.05623] Existence, Stability and Scalability of Orthogonal Convolutional Neural Networks
[11] Orthogonal-Nets: A Large Ensemble of 2D Neural Networks for 3D Brain Tumor Segmentation Authors:Kamlesh Pawar, Shenjun Zhong, Dilshan Sasanka Goonatillake, Gary Egan, Zhaolin Chen
[12] Heaps' Law and Vocabulary Richness in the History of Classical Music Harmony-Authors: Serra-Peralta, Marc, Serrà, Joan, Corral, Álvaro
[13] The Thousand Brains Theory -Jeff Hawkins ,Numenta
[14] [2210.12068] Grid cells and their potential application in AI- Toy, Jason