Microagent Convergence Testing using Stacked Holographic Reduced Representations

Author: Jackson Goble
Advisor: Dr. Joshua Phillips
MTSU Dept. Of Computer Science

ABSTRACT

Holographic reduced representation-based working memory models are biologically plausible models that show promise in their ability to solve temporal difference learning tasks. These working memory models suffer from large input sizes due to the random nature of holographic reduced representations (HRRs). This is because the current solution to overcome the randomness of HRRs is to choose conservatively large sizes for these vectors in order to decrease their linear dependence. This is fine for small models, but it scales poorly for larger or more complex models. It is unknown whether training multiple microagents each with independent and smaller HRRs is as effective as training a single agent with a larger HRR. We propose that training multiple agents this way, in parallel, and averaging the results may yield faster convergence while using fewer resources. We learn that while increasing the number of microagents does increase convergence, it does so at a slower rate than increasing the HRR length.

BACKGROUND

Q-learning is a machine learning method in which an agent is trained to solve a task by allowing it to learn and maintain an internal representation, called a Q-function, of the task it is solving. One of the benefits of Q-learning is that it a lot of training can happen even when the agent is only given sparse feedback about whether it was successful or unsuccessful in completing the task.

One of the shortcomings of Q-learning is its limited mechanisms for storing information long term, i.e. many time steps in the future. This is the problem that working memory models attempt to solve. One approach used by working memory models (Dubois and Phillips, 2017) is to encode the agents internal representations of it's environment using Holographic Reduced Representations (Plate, 1995).

The size of HRRs can be chosen conservatively large, e.g. 10 times the size of the state space, in order to increase the chance that the agent will be able to distinguish between them, and therefore be more likely to converge on the desired function. While this may be fine on small models, it could be a substantial waste of resources on larger or more complex models.

It is not currently known whether training multiple microagents with small HRRs or a single agent with a relatively large HRR is more likely to converge, which is what we explore in this research.

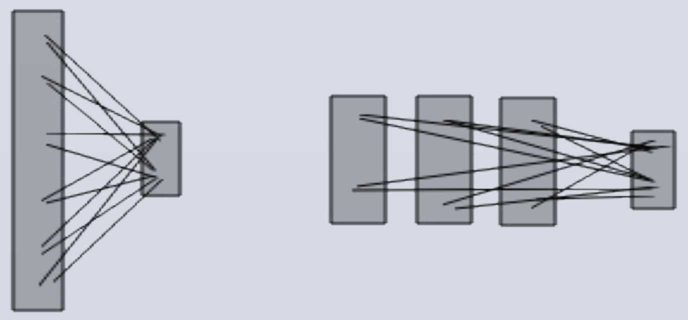

Fig. 01 single agent vs microagent architecture.

SPECIFIC AIM

The aim of this research is to compare the empirical convergence rates of multiple Q-Learning agents that use the traditional and proposed HRR encoding schemes as we decrease the HRR length.

METHODS

We use two metrics to understand the convergence of the models. First, we use the sum-squared difference from the known function that the agent should be learning. Second, we count the number of suboptimal steps made by the agents which provides a more intuitive measure of how far the agent travelled unecessarily.

We first created a framework that can generate and train agents with user-specified number of microagents and HRR-lengths. Each agent is comprised of an ensemble of identically-shaped microagents, and is given a single HRR per possible maze state, per microagent. Each microagent is a single, dense neural network layer with a linear activation function, whose weights are independent from each other microagent.

The Agents are trained for 100 episodes, in a simple 1-dimensional maze with 20 states, and a single static goal state. In each episode the agent is given a maximum of 20 moves to find the goal state. The suboptimal steps made in each episode are added together, and at the end of the 100 episodes we sum the squared difference from the learned q-function and the actual function at each state.

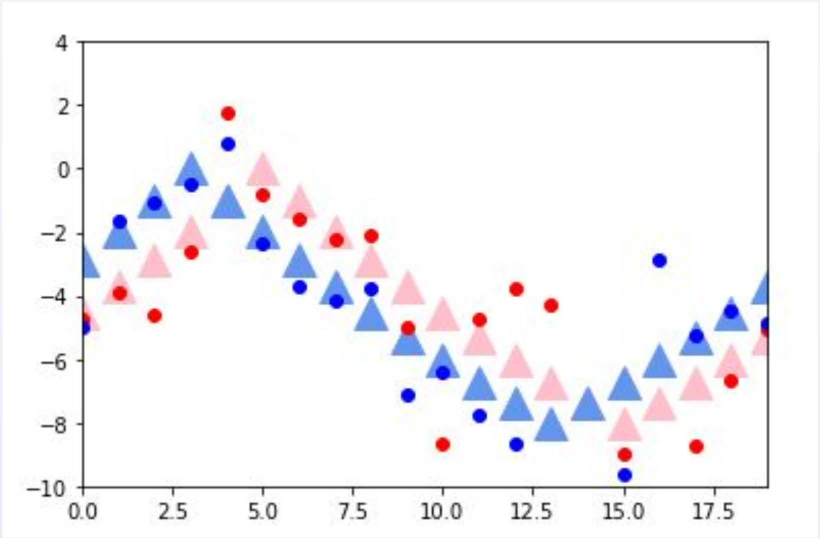

Fig. 02 Agent's Q-function (circles) as it attempts to learn the desired function (triangles). The red and blue represent moving left and right respectively.

For the first set of agents, we used a logarithmic training schedule of 1, 10, and 100 microagents with HRRs of length 2, 20, 200, and 2000 (.1x, 1x, 10x, and 100x the number of states in the maze.) We saved each agent's internal representation at every step and plotted this data over time into animations that allow us to watch as the agent learns. This training schedule and visualization confirmed our belief that we should focus on agents that consist of 1 to 6 microagents and 1x to 6x HRR length.

We then trained 3,600 agents, 100 each of the types specified above. i.e. 100 consisting of 1 microagent and 1x HRR length, 100 of 2 microagents and 1x HRR length, etc. And collected data on the number of suboptimal steps and sum-squared difference from the true function.

RESULTS

These figures show the effect of varying the HRR length and number of microagents on the performance of the models. The lighter cells represent models with fewer suboptimal steps (left) or lower sum-squared difference (right), and should be considered better. In the left chart, we can see that the model improves in performance with both HRR length and the number of microagents. Both variables reduce the number of suboptimal steps made, which

In the right chart, we see that while the sum-squared difference improves with the HRR length, it does not seem to improve with the number of microagents, resulting in columns of the same color. We suspect that this could be for a couple of reasons. First, the act of averaging the microagents' output could be causing the model to learn a different function than expected. If this is the case, the learned function is at least similar to the expected function, and results in the agent making the correct choices, but would mean that we have calculated the sum-squared difference to the wrong function, which is therefore less informative.

The second reason for the column-like behavior could be that the multi-microagent model does not converge as quickly. We can see some definite improvement between the first and second row, but then it flattens and then begins to perform worse. This shape lends itself to the idea that more microagents may simply require longer to converge to the true function when averaged together.

Fig. 03 The effect of varying our parameters on convergence. Notice that the left graph gets lighter vertically, while the right graph does not.

In order to fairly compare the models, in observing these graphs, we are most interested in the cells whose "coordinates" have equal products. e.g. cell (2, 6) and cell (3, 4) because these models have the same number of trainable parameters. Comparing these cells, we can see that increasing the HRR length has a greater effect in increasing the convergence rates than increasing the number of microagents does.

CONCLUSION

From this research we have learned that while increasing the number of microagents and averaging their predictions has the effect of improving model performance, it does not cause the model to converge more quickly than increasing the HRR length does. This does not however discount the microagent model as inferior, as there is a certain utility that can be gained from having multiple models. For example, one could use the output from a three-microagent model to perform a type of error-checking to detect HRRs that are interfering with one another, and replace them.

There is much left to explore with these models. We are interested in the possibility that increasing the learning rate of the multi-agents models might increase performance and maintain stability better than a single agent model. We would also like to explore the idea that the averaged model approaches a different function than the expected one. Other microagent models have been shown to exhibit hyperbolic discounting, which could be happening here, effectively making the model decide that it doesn't care how long it takes to get to the reward if it will only be received too far in the future. If that is the case, then this type of architecture could be good for models that need to ignore distant rewards .

REFERENCES

1. Ivo Danihelka, Greg Wayne, Benigno Uria, Nal Kalchbrenner, and Alex Graves. Associative long short-term memory. CoRR, abs/1602.03032,2016.URL http://arxiv.org/abs/1602.03032.
2. Grayson M. Dubois and Joshua L. Phillips.Working memory concept encoding using holographic reduced representations. In John Licato, Aleshia T. Hayes, and Michael Glass, editors, Proceedings of the 28th Modern Artificial Intelligence and Cognitive Science Conference 2017, Fort Wayne, IN, USA, April 28-29, 2017, volume 1964 of CEUR Workshop Proceedings, pages 137–144. CEUR-WS.org, 2017. URL http://ceur-ws.org/Vol-1964/NN3.pdf.
3. Michael Jovanovich and Joshua Phillips. n-task learning: Solving multiple or unknown numbers of reinforcement learning problems. In Chuck Kalish, Martina A. Rau, Xiaojin (Jerry) Zhu, and Timothy T. Rogers, editors, Proceedings of the 40th Annual Meeting of the Cognitive Science Society, CogSci 2018,Madison, WI, USA, July 25-28, 2018. cognitivesciencesociety.org, 2018. URL https://mindmodeling.org/cogsci2018/papers/0126/index.html.
4. Joshua L. Phillips and David C. Noelle. A biologically inspired working memory framework for robots. In IEEE International Workshop on Robot and Human Interactive Communication, ROMAN 2005, Nashville, TN, USA, 13-15 August, 2005, pages 599–604. IEEE, 2005. doi: 10.1109/ROMAN.2005.1513845. URL https://doi.org/10.1109/ROMAN.2005.1513845.
5. Tony A. Plate. Holographic reduced representations. IEEE Trans. Neural Networks, 6(3):623–641, 1995. doi: 10.1109/72.377968. URL https://doi.org/10.1109/72.377968.
6. Yi Tay, Shuai Zhang, Anh Tuan Luu, Siu Cheung Hui, Lina Yao, and Tran Dang Quang Vinh. Holographic factorization machines for recommendation. In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, AAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 -February 1, 2019, pages 5143–5150. AAAI Press, 2019. doi: 10.1609/aaai.v33i01.33015143. URL https://doi.org/10.1609/aaai.v33i01.33015143.

CONTACT / ACKNOWLEDGEMENTS

Jackson Goble: jlg2av@mtmail.mtsu.edu
Dr. Joshua Phillips: joshua.phillips@mtsu.edu

Acknowledgements: Dr. Suk Seo
Dr. Joshua Phillips
MTSU Dept. of Computer Science