



Neurocoder: General-Purpose Computation Using Stored Neural Programs

Presented by Hung Le



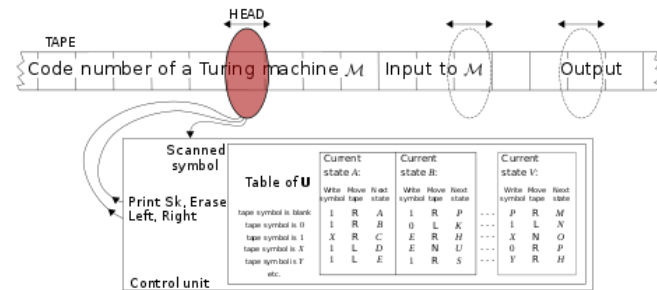
A²I²

APPLIED ARTIFICIAL
INTELLIGENCE INSTITUTE



Can neural network **efficiently** simulate Universal Turing Machine?

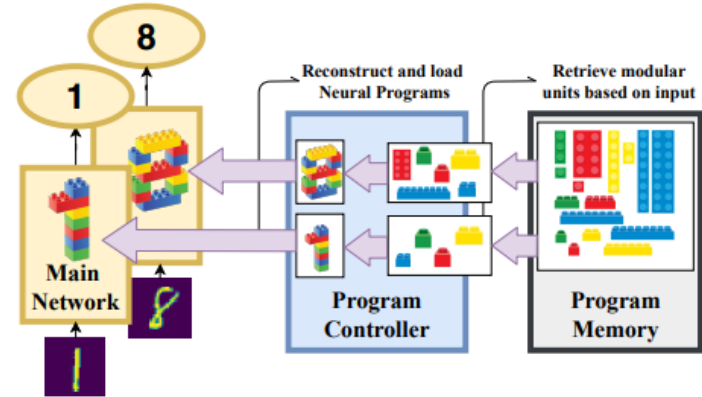
- Implement stored-program principle
- Store the weights of neural networks as programs in memory
- Use multiple programs for multiple tasks (adaptive and not forgetting)
 - E.g. NSM, MoE, ModuleNet
 - Their programs are big, the number of stored parameters are huge, requiring big data for training



Le, Hung, Truyen Tran, and Svetha Venkatesh. "Neural Stored-program Memory." In *International Conference on Learning Representations*. 2019.
Shazeer, Noam, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer." arXiv preprint arXiv:1701.06538 (2017).
Andreas, Jacob, Marcus Rohrbach, Trevor Darrell, and Dan Klein. "Neural module networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 39-48. 2016.

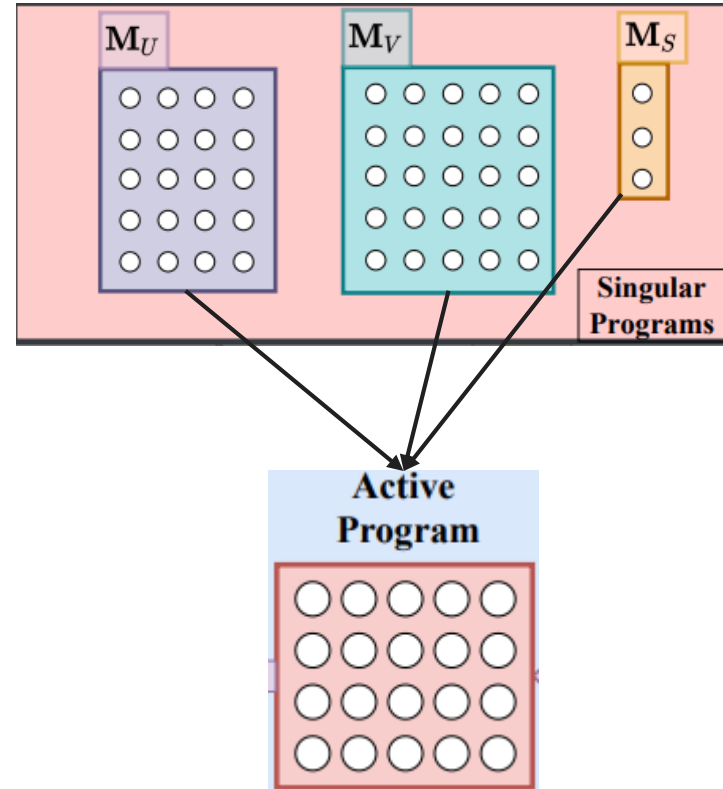
Our solution: Neurocoder

- A neural network stores **small blocks** (not the whole) of programs
- The small blocks compose the whole program just like **small, sharable programs** (e.g. libraries) are combined into a big program
- The composition is conditioned on the input data



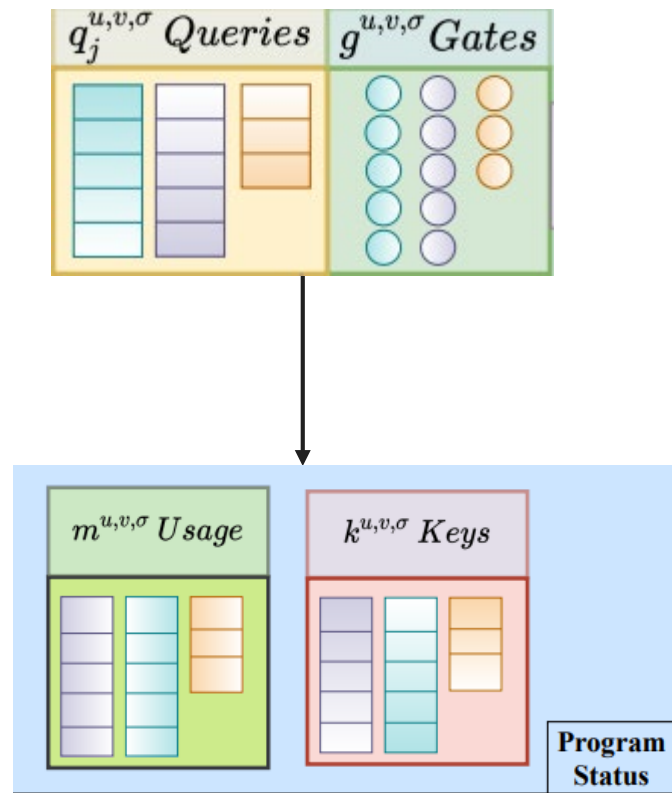
Neurocoder: how it **stores** program

- Instead of storing a weight matrix, it stores vectors and scalars representing the **singular vectors/values** of the matrix
- The weight matrix is composed by **low-rank approximation** from multiple singular vectors/values from the program memory



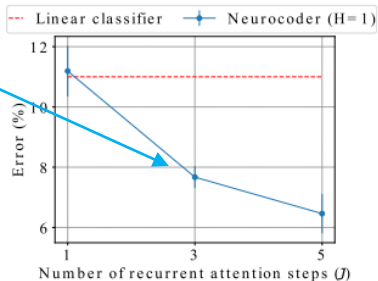
Neurocoder: how it **reads** program

- Given input data, the controller generates **queries**
- The program memory maintains **program status** variables (Current memory usage, the program name/address/key)
- The query is matched with the key via contented-based **attention** (call a **program by its name**)
- The final reading location is a mixture of **least-used and highly attended** slots



Experiment: behaviours of Neurocoder

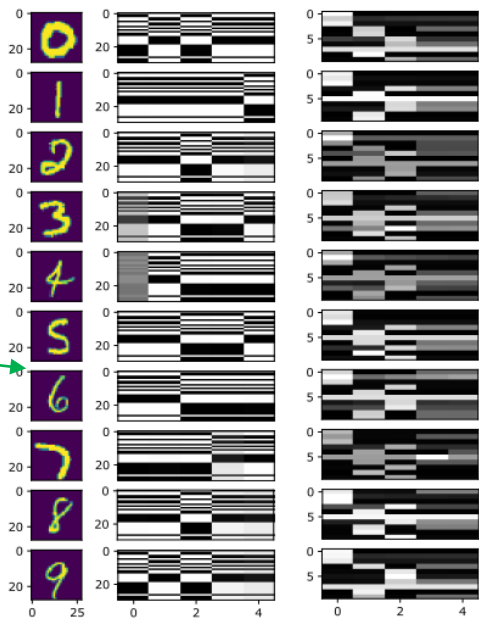
#parameters is fewer than perceptron's



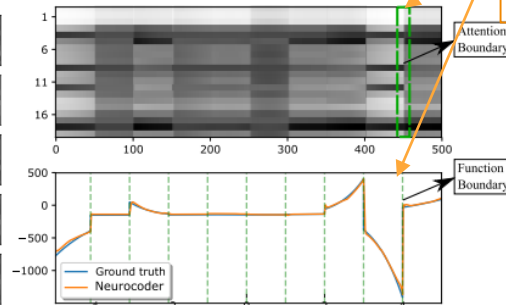
Attention hierarchy as a binary tree



(a)

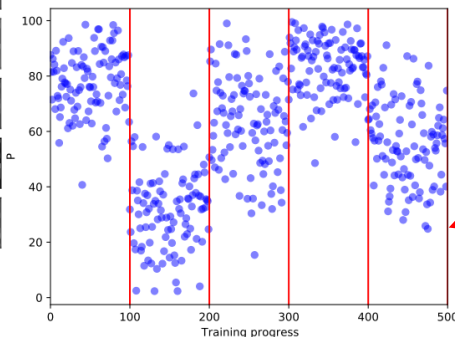


(b)



(d)

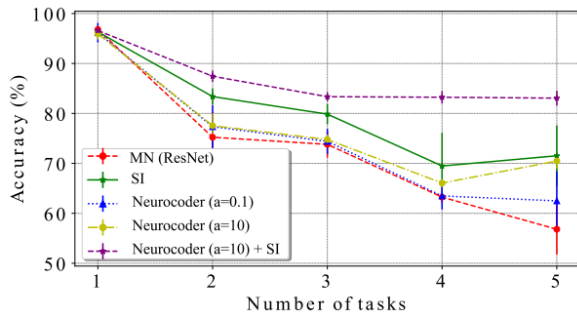
Auto-switch program when tasks change



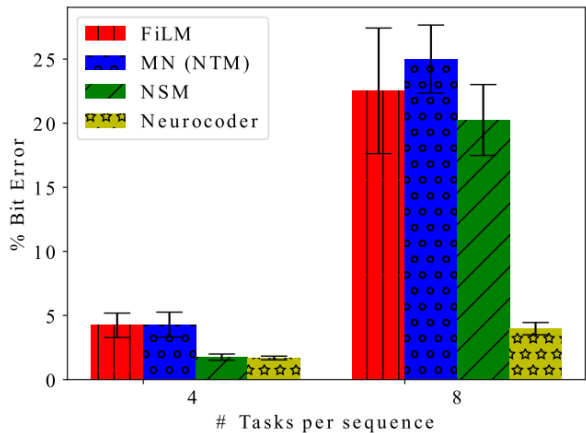
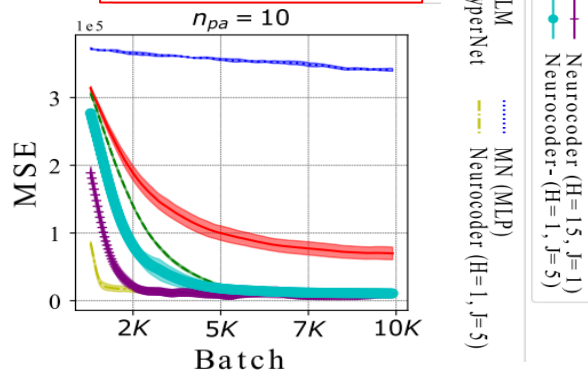
Continual learning: program attention

Experiment: promising performance

Continual learning

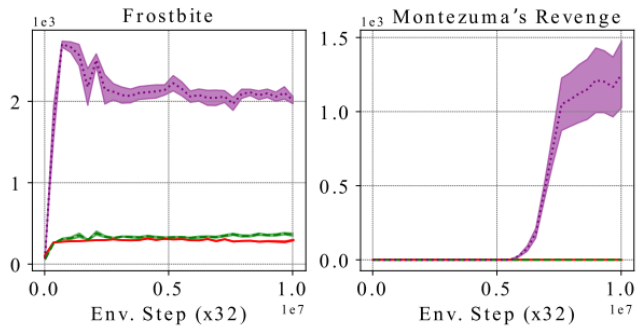


Multi-function fitting



Multi-algorithm reasoning

HyperNet MN (MLP) Neurocoder



Hard reinforcement learning

Thank you!

Meet the authors

Hung Le

Research Lecturer
Applied Artificial Intelligence Institute

Svetha Venkatesh

Co-Director, Applied Artificial
Intelligence Institute (A2I2)
Alfred Deakin Professor, Australian
Laureate Fellow

Hung Le

thai.le@deakin.edu.au

A²I²

Deakin University
Geelong Waurn Ponds
Campus, Geelong, VIC 3220

