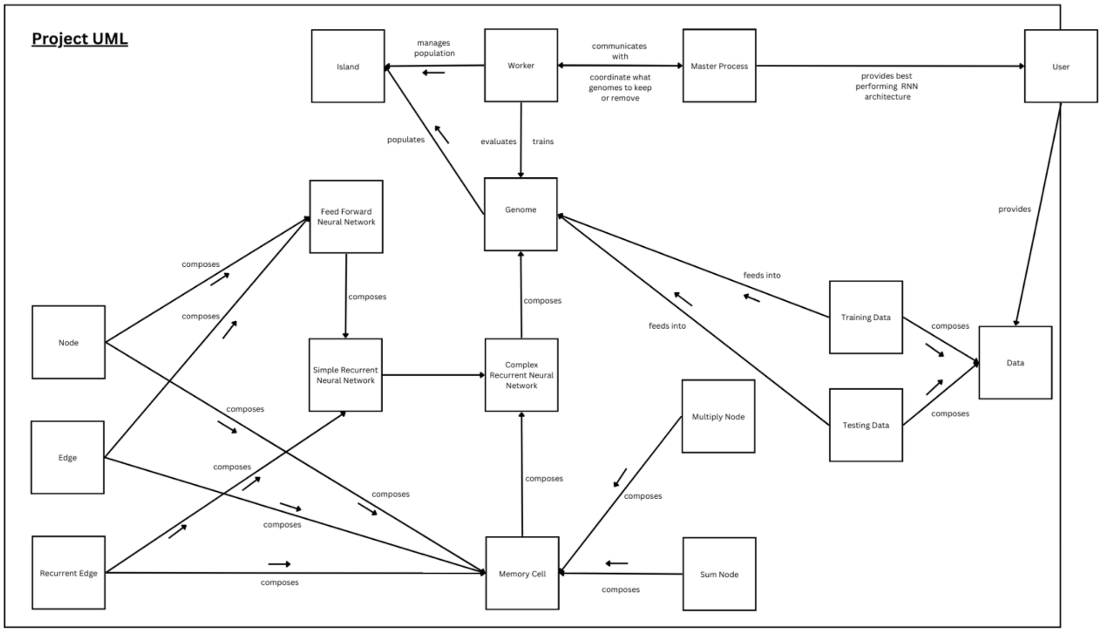
Approach Write-up: Discovering Novel Memory Cell Architectures for Recurrent Neural Networks Using Evolutionary Algorithms

Jared Murphy
Graduate Student
Department of Software Engineering
Rochester Institute of Technology
jmm2188@rit.edu

Travis Desell
Associate Professor
Department of Software Engineering
Rochester Institute of Technology
tjdvse@rit.edu

As was explained in the related works survey, a similar study to the one proposed here has been carried out by Rawal, Aditya, and Risto Miikkulainen. In their work titled "From nodes to networks: Evolving recurrent neural networks", the researchers break down LSTM cells into their constituent parts and use an evolutionary algorithm based on NEAT to evolve recurrent neural networks. The research project presented here differs from the previous study primarily in that the evolutionary algorithm used is different. Where the previous study basically used NEAT to evolve networks, here we use the EXAMM algorithm.

According to the paper titled "Investigating Recurrent Neural Network Memory Structures using Neuro-Evolution", "EXAMM differs from NEAT in that it … utilizes Lamarckian weight initialization along with backpropagation to evolve the weights as opposed to a simpler less efficient genetic strategy. Additionally, it has been developed with large-scale concurrency in mind, and utilizes an asynchronous steady state approach which has been shown to facilitate scalability to potentially millions of compute nodes." By using EXAMM to evolve RNN's, we can utilize backpropagation and concurrency to efficiently search the space of possible RNN architectures. In combining this approach with genetic programming, we will have come up with a novel method to evolve RNN's. Below is a UML diagram of the evolutionary process.



Two key observations are provided from this diagram. The first is that a Master Process coordinates subordinate processes called Workers to train Genomes. This allows for parallelization and concurrency. The second observation is that Memory Cells can clearly be broken down into their constituent parts by defining two new types of node structures, the Multiply Node and the Sum node. With the addition of these two new node types, which simply perform sum and multiply operations to their inputs, elements of genetic programming will be combined with the EXAMM algorithm to evolve complex RNN architectures from their most basic constituent parts. This will allow for the potential discovery of novel memory cell architectures through many iterations of this process.

How a dataset of RNN architectures will be best analyzed for new memory cells and the workflow involved is still being discussed by the researchers. Another UML diagram for that process will have to be created when discussions are complete. For now, the first stage of the overall project planning is complete. It is most important to achieve success in evolving networks using EXAMM with these new node types. This is where most of the technical work lies. After this result is achieved, a dataset of evolved RNN's can be created relatively easily, and standard data science methods can be applied for analysis.