**Final Year Project Proposal**

**The Working Problem**

A neural network topology is how neurons are connected to form a network. The topology of a neural network influences the functionality and performance. I want to investigate and provide a comparative analysis on how weight optimization with genetic algorithms differ between different neural network topologies for networks that solve reinforcement learning (RL) problems. The different neural network topologies I aim to compare are convolutional, recurrent and dense neural networks. When this bit is done successfully and if there is enough time I want to then hopefully discover a theory to give a reason to their differences.

**Why this is a Useful Problem to Try to Solve**

Research suggests that evolutionary algorithms are successful and proficient in training neural networks for reinforcement learning problems. It has been shown that they can outperform other more traditional reinforcement learning techniques. This proposes that evolutionary algorithms optimise neural networks in reinforcement learning problems. It would be useful and interesting to know how the evolutionary algorithm affects different structures of neural networks and be able to provide an explanation for this.

**Strategy**

## ***Prerequisites required:***

* Existing Understanding of Neural Networks and Machine Learning objectives
* Understanding of Reinforcement Learning
* Programming ability in Python.

As this is an area I haven’t tackled before, I will aim to read approximately 30 papers and articles of existing work that others have already done in this field, before my literature review and prior to developing the experiment.

I will spend some time researching design development and then I will attempt to develop the different neural network topologies which aim to solve two different reinforcement learning tasks, with the assistant of existing libraries in python.

Then I will train each problem with a genetic algorithm. I will the evaluate and compare the results I have obtained before concluding and specifying their differences.

Then if I have time I will try to develop a hypothesis as to why they different and will try to really explain their differences with a theory.

**Potential challenges**

* ***Training for a reinforcement learning problem***
* I have never trained a neural network to solve a reinforcement learning problem. Reinforcement learning usually requires a reward function. Trying to design a reward function that encourages learning behaviour, I imagine will be quite a challenge.
* ***Defining the parameters***
* This will be the first time I will develop a neural network. Thus consequently, comparative analysis on it will be challenging. As I do more research I’m sure my parameters will change, but neural networks can have so many parameters so selecting the parameters to compare with will be difficult.
* ***Testing the neural networks with a robust testing strategy.***
* How will I measure or know when the GA has evolved the neural network enough? How will I know when it is fully optimised and successful? If the testing of the neural networks method is weak, then this allows for an inaccurate basis for me to attempt to provide theoretical reasoning or explanation.