

Computer Games Development SE607

Technical Design Document

Year IV

|  |
| --- |
| [Leo Chen]  [C00250593]  [X/X/2023] |

[Declaration form to be attached]

Contents

[Technical Design 2](#_Toc54713656)

[References 2](#_Toc54713657)

Technical Design

# Architecture

Game Overview Diagram

# Artificial Intelligence

## Neural networks

A neural network, also known as artificial neural networks are a subset of machine learning which attempts to train itself through the input of data. With said data being used to make decisions on the type of actions it can conduct.(What are Neural Networks? | IBM n.d.)

Our neural network is meant to take a set amount of inputs in order to use one of three actions as its output:

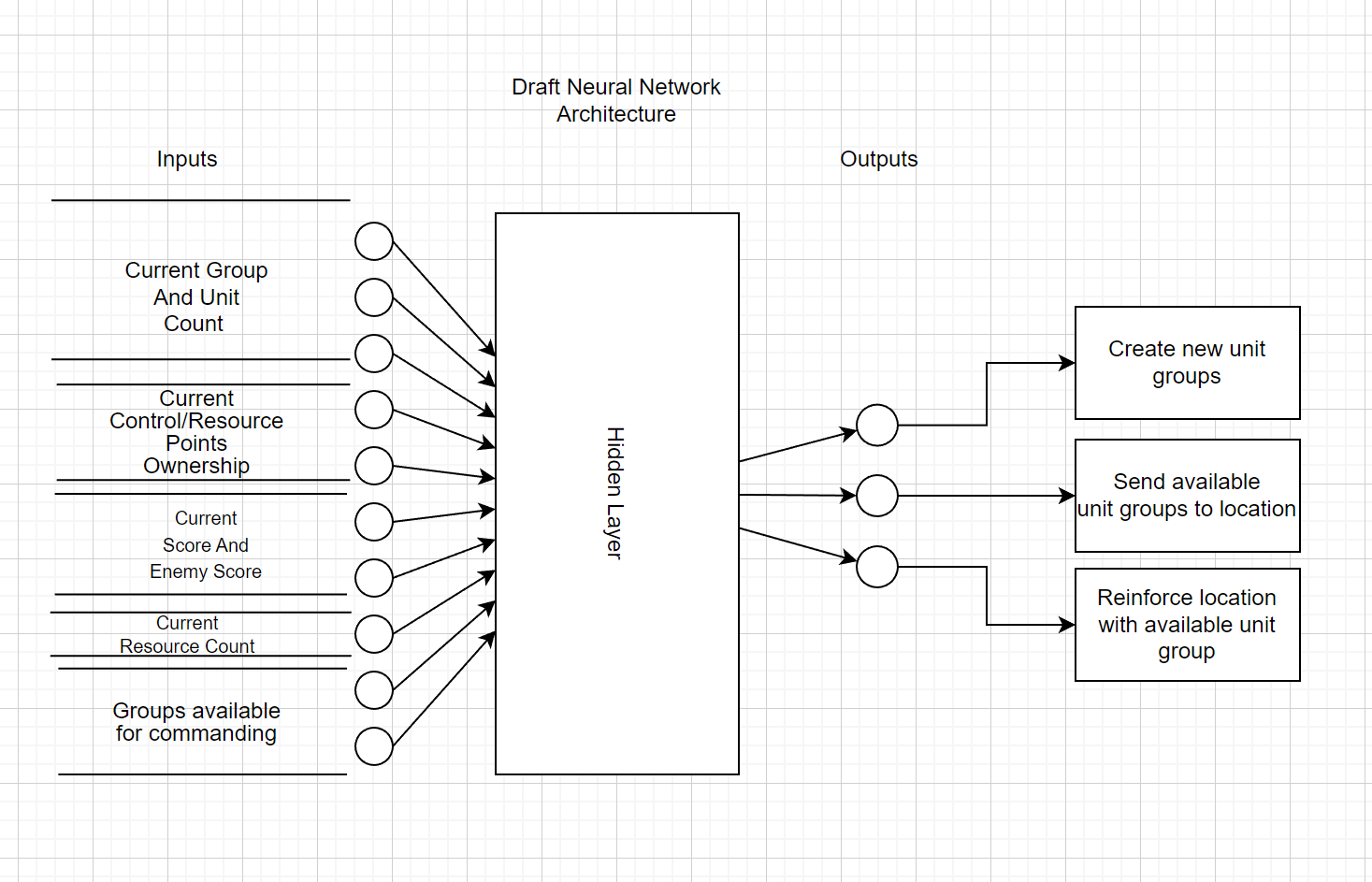


Fig.1 Proposed neural network structure

In this diagram, the inputs for the RTS consists of what the AI needs to know in order to produce an output. The AI requires the need to know what is available to it at all times when there is a need for deciding on what actions to take.

**Inputs:**

* Current group and unit count
  + The main purpose of this set of inputs is to tell the AI how many units and grouped units it has on the field.
* Current Control/Resource Points Ownership
  + Tracking the control/resource points of both itself and the opponent is needed for the tactical choice of moving groups to take or hold those objectives.
* Current score and current enemy score
  + Having both the scores tracked and fed into the neural network weighs on their decision to affect the former point. Should the AI priorities the resource points for more units to produce, or should it prefer control points that generate score?
* Current resource count
  + Similar to the above point, the resource count is meant to keep track of how much funding the player or AI has to produce units.

**Outputs**:

* Create new unit group
  + This action has the AI create a new set of units to immediately group up. The groups themselves are what the AI commands so it doesn’t need to manage individual units.
* Send available unit groups to a location
  + For this action as it implies, send an available group of units to a destination of the AI’s choosing. Usually towards a strategic objective.
* Reinforce location with available unit group
  + The action is meant for reinforcing a position with additional troops for the AI to cement its control over it.

## Unity’s Machine Learning Agents

The machine learning-agents toolkit (Unity ML-Agents Toolkit 2023) designed for both research and game development is one of the options for having it as an opponent for player in the game.

Similar to the actions listed in the neural network diagram, the AI requires several inputs in order to execute actions that can be utilised in game. For their observations they required a specific dataset to be utilised.

* Current friendly unit count.
* Current friendly group count.
* Location of the selector.
* AI Camera Location.
* Location of all control/resource points
* Team affiliation of said control/resource points
* Current resource count
* Current Score count

Their actions on the other hand will depend on the training data fed to it. Using imitation learning, I would need to modify the selection method on selecting units to be using a different

# References

*Unity ML-Agents Toolkit*. (2023). , 25 April 2023. Available from: https://github.com/Unity-Technologies/ml-agents [accessed 25 April 2023].

*What are Neural Networks? | IBM*. Available from: https://www.ibm.com/topics/neural-networks [accessed 25 April 2023].