FYP Spitballing

Sections:

Introduction (5 Pages)

Provide a clear context for project:

What is it about? Is it the right level (8)?

The purpose of this application is to train and reinforce the actions of particular agents who are on opposing teams, so that they work together to achieve a particular objective (i.e. scoring n amount of goals).

Is the scope correct?

The scope of this project incorporates many technologies, such as Unity, MLagents Toolkit, Neural Networks, Python and Jupyter Notebooks, etc

The reader will need to be informed thoroughly (and clearly!)

(Maybe a low-level breakdown of how the application works/is supposed to work?)

Why should the reader be interested? (Relevant to them, or a practical application)

This project intends to show the applications of reinforced learning algorithms in a neural network, and how they apply to this particular scenario. (Maybe touch on other practical applications?)

Set out objectives of project:

Will have to address these in Eval/Concl

The metrics by which success or failure is measured

Briefly list each chapter/ section and provide a description

(Plus Github link, and descriptions of elements within)

Methodology (3-5 pages)

Describe how you went about the project. Was your approach to the problem valid?

Agile/Incremental development process

Planning, storyboards? Requirements?

Initial concept had dogs/spiders and the agents. Felt that Unity would be an excellent environment as it had the capabilities for 3D models, as well as deeper logic behind.

Meetings, frequency, structure, etc etc

Meetings occurred twice weekly, on average, with Daniel Cregg and Gerard Harrison.

Validation and Testing

Github, and did you use other development tools?

How were problems solved?

Technology Review (10+ Pages)

The “Literature Review” of the dissertation

Describe each of the technologies you used at a conceptual level

Unity (C#)

<https://unity.com/>

MlAgents Toolkit (Acting as a connector)

<https://github.com/Unity-Technologies/ml-agents>

Jupyter Notebooks (And Python)

<https://jupyter.org/>

(Also mention coding standards we have/could have undertaken)

* Mention IronPython, and other tech we looked at but didn’t use

System Design (n...m pages)

The “How” of the project.

System that you’ve designed should be informed by the technology review, applying knowledge used during research.

How are the components coupled? How does the project “go together”?

Unity -> Ml-Agents -> Notebooks

Notebooks are used to “Stage” and prep the environment (Unity), and also pass through more high-level instructions/parameters (ML-Agents)

Use diagrams to improve the description of the architecture

Provide an overview of the different components and how they work together (DFD?)

UML, Class, Sequence and interaction diagrams

Screenshots of forms or other UI

System Evaluation (n…m pages)

Evaluate your project against the objective set out in the introduction.

Prove that the software is robust

Unit/Acceptance testing for robustness/behaviour

(How did we test the application?)

Stability metrics for structure

Tables/Graphs of results belong here

With a good bit of writing to supplement

Use performance benchmarks (Space/Time complexity)

Measure the outcomes/outputs of the system/software vs the objectives

Highlight any limitations or opportunities in your approach or technologies used

How our approach to the project reflected the outcome of goals hit/missed.

Conclusion (3-5 pages)

References and Appendices