Deliverable 1: Final Year Dissertation

Evolving a Learning Agent using Neuroevolution in the FightingICE Game Platform

Robert John Dunn
H00163867
BSc Honours in Computer Science
Heriot-Watt University

Supervisor:
Dr Patricia A. Vargas

Co-Supervisor:
Dr Fabrício Olivetti de França

Second Reader:
Dr Mohamed Abdelshafy

Declaration

I, Robert Dunn confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of other authors in any form (e.g., ideas, equations, figures, text, tables, programs) are properly acknowledged at any point of their use. A list of the references employed is included.

Signed:

Date:

Abstract

Neuroevolution is a popular technique for machine learning in which an artificial neural network is trained by an evolutionary algorithm. The technique takes inspiration from the evolution of the biological nervous system and is a popular approach for reinforcement learning problems. One way to demonstrate the effectiveness of neuroevolution is through artificial intelligence in games. This project aims to implement a learning agent in the FightingICE platform, a two-dimensional Java fighting game organised and maintained by Ritsumeikan University, Kyoto. The agent is designed to evolve through neuroevolution to improve its performance in the game, eventually becoming competitive versus a human opponent. By implementing a neuroevolution method in a simplistic environment, we hope to evaluate the effectiveness of neuroevolution as a method of machine learning and explore the potential of our agent's performance.

Contents

1 Introduction

2 Literature Review

2.1 Machine Learning

2.1.1 Learning Paradigms

Learning is an essential human function in which we modify our behaviour tendency according to experiences, to become better when a similar situation occurs. In the study of machine learning, algorithms, computer applications, and systems, utilise learning to improve their performance at certain tasks. There are two main entities in the machine learning model, the teacher and the learner. The teacher contains the knowledge to perform a given task while the learner has to learn the knowledge the teacher holds [book]. There are three types of machine learning[pat]:

1. *Supervised Learning*
   A teacher provides the learner with a set of input and desired output pairs. The learner can then use these examples to improve its performance at the task.

2. *Unsupervised/Self-organised Learning*
   There is no teacher, so the learner learns based only on the stimuli received. A common application is finding patterns or grouping within data, using methods such as cluster analysis.

3. *Reinforcement Learning*
   The agent uses goal-directed learning where a notion of reward is introduced and the agent attempts to maximise this reward. There is no teacher for the agent and no explicit model of the environment.

2.1.2 Incremental Evolution

When an agent is provided with a complex behaviour, it may have trouble evolving to perform at its potential. Using incremental evolution, the behaviour can be learnt incrementally with tasks gradually increasing in difficulty. [inc]

2.1.3 Co-evolution

2.2 Neuroevolution

2.2.1 Fundamentals

Neuroevolution is a popular biologically-inspired method of machine learning in which an artificial neural network is evolved using an evolutionary algorithm. The popularity of the method stems from the fact that artificial intelligence and control problems can be cast as optimisation problems,

and since the method is grounded in biological metaphor and evolutionary theory.[1]

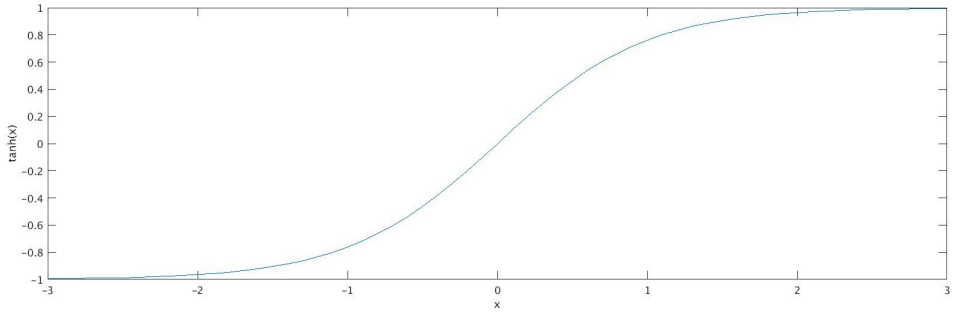
What is neuroevolution and how it can be used for optimisation. ANNs evolving with EAs.

Artificial neural networks Neural networks are about associative memory or content-addressable memory. Neural networks are also about parallel processing.[book] Used for pattern What a perceptron is

2.2.2 Artificial Neural Networks

The nervous system is responsible for Artificial neural networks model the way the brain solves problems with collections of neurons co.

More general nonlinear functions can be used. Usually required to saturate at -1 and +1. Tanhx (include image).



Another possibility is to have neurons operate probabilistically. State is again +1 or -1 but now only a probability is given for the state to be -1 or +1.

2.2.3 Genetic Algorithms

Multi-layer feed forward networks can learn by example, where the examples are inputs of which the outputs are known. This method of learning is called supervised learning. [book] Neural networks are In order to train an artificial neural network, weights between neurons need to be adjusted in order to map th

2.2.4 Neuroevolution in Games

How neuroevolution can be used to evolve learning agents in games, usually agent is optimising some value e.g. HP. How fitness in games is evaluated, giving some examples. How neuroevolution fares compared to other learning algorithms. Present examples of projects implementing neuroevolution to learn.

2.3 FightingICE

2.3.1 Game Platform

FightingICE is a Java based game platform organised and maintained by Intelligent Computer Entertainment Lab., Ritsumeikan University. The game is based in an arena where two fighters are competing versus each other, attempting to deplete the other's hit-points while preserving their own. The FightingICE platform was designed to allow easy development and evaluation of artificially agents in the game for research or hobby purposes. Once implemented, an agent receives information about the state of the game from the platform periodically, such as the opponent player's location and current energy levels. A delay is added to this game information in order to simulate the delay a human player would experience from reaction time.

2.3.2 Game Agents

Agents in the game have two main attributes- hit-points and energy. There are four characters available in the game: Zen, Garnet, Lud, and Kfm. Each of these characters is capable of moving, performing attacks, and combing these attacks into deadly combos. Present information on the agents in the FightingICE game. The capabilities of the agents including which attack moves / movement commands they can do. Touch on fitness evaluation (how to win the game: keep enemy HP low and your HP high).

2.3.3 Related Work

Discuss and reference relevant projects which have been completed using the FightingICE framework. Discuss what has been achieved in said projects

and the potential further research that can be undertaken.

3 Requirements

4 Evaluation Strategy

5 Project Management

6 References

[1] https://arxiv.org/pdf/1410.7326.pdf : Neuroevolution in Games: State of the Art and Open Challeng

[2] http://commerce3.derby.ac.uk/ojs/index.php/gb/article/view/3/1 : Improving AI for simulated cars using Neuroevolution

[3] http://commerce3.derby.ac.uk/ojs/index.php/gb/article/view/14/12 : Calculating Optimal Jungling Routes in DOTA2 Using Neural Networks and Genetic Algorithms

[4] https://www.cs.utexas.edu/ mhauskn/papers/atari.pdf : A Neuroevolution Approach to General Atari Game Playing

[5] https://arxiv.org/pdf/1512.01537v1.pdf : Reuse of Neural Modules for General Video Game Playing

http://www.cs.utexas.edu/users/nn/downloads/papers/gomez.ijcai99.pdf
https://arxiv.org/abs/1608.02971
https://arxiv.org/abs/1604.00644
https://arxiv.org/abs/1312.5355
https://arxiv.org/abs/1107.0037 http://nn.cs.utexas.edu/keyword?gomez:ab97

7 Appendices

References

[1] Risi, S & Togelius, J. (2015). *Neuroevolution in Games: State of the Art and Open Challenges*, [online], Available at: https://arxiv.org/pdf/1410.7326.pdf [Accessed 03 Nov. 2016]

[2] Pace, A. (2014). *Improving AI for simulated cars using Neuroevolution*, [online], Available at: http://commerce3.derby.ac.uk/ojs/index.php/gb/article/view/3/1 [Accessed 07 Nov. 2016]

[3] Lampropoulos, A (2005) *Machine Learning Paradigms.* Available at: *http://file.allitebooks.com/20150722/Machine%20Learning%20Paradigms-%20Applications%20in%20Recommender%20Systems.pdf*