Reinforcement Learning in Pokémon Red to explore complex multi-reward environments

by

LIAM O'DRISCOLL
URN: 6640106

January 27, 2024

Department of Computer Science
University of Surrey
Guildford, Surrey
England, United Kingdom
GU2 7XH

Project Supervisor: Sotiris Moschoyiannis

I declare that this dissertation is my own work and that the work of others is acknowledged and indicated by explicit references.

Liam O'Driscoll
January 27, 2024

Contents

1 Introduction **5**
1.1 What is Reinforcement Learning 5
1.2 Elements of Reinforcement Learning 5
1.3 Aims of the Project 6
1.4 Report Structure 6

2 Literature Review **7**
2.1 Introduction to Pokemon Red 7
2.2 Why Pokemon Red? 7
2.3 RL Algorithms and Methods 7

3 Research Problem **8**
3.1 Background 8
3.2 Analysis 8
3.3 Problem Approach 8

4 Analysis **9**

5 Problem Approach **10**

6 Implmentation & Design **11**

7 Analysis **12**

8 Conclusion **13**

9 References **14**

1 Introduction

The aim of this project is to develop a reinforcement learning (RL) agent to play Pokémon Red to compare the effectiveness of different RL algorithms and techniques to solving complex problem spaces. The pokemon red problem space is a videogame targetted at children but is complex due to its large action space and countless decisions which influnce future states.

1.1 What is Reinforcement Learning

Decision-making is a recurring activity that every individual faces in their everyday life, leading to short- or long-term consequences with differing levels of satisfaction. When making decisions in uncertain environments, humans have the ability to take previous experiences and apply them to new environments to make decisions grounded in knowledge. This is called sequential decision-making. In the context of machine learning, it refers to a decision-making agent in an observation and action loop, where after a series of actions, its performance can be measured [1]. Taking inspiration from biological learning systems, RL is the closest kind of machine learning that learns using observed data to influence the brain's reward system [2].

RL is different from common traditional machine learning techniques as it learns "how to map situations to actions so as to maximize a numerical reward signal [2]". Agents' requirement to learn through experience and actions performed on current states not only affect the present but also affect future states and actions are two characteristics that distinguish them from other forms of ML. It is also what makes Pokémon Red a suitable environment to apply this style of ML to. Initially, the agent has no understanding of the environment; however, through random action selection and the reward that it receives, it learns an understanding of the environment it is in. In RL, the agent is an AI that chooses actions by following a set of rules; this set of rules is called a policy. The action chosen by the policy is applied to the environment in which the agent is interacting. The place in which the decision-making agent performs actions and the entity that determines what is right and wrong are called the environment. The environment is everything the agent cannot control but is able to interact with. After the action has been applied to the environment, the change in the environment will be evaluated and return positive or negative feedback to the agent. This positive or negative feedback is called reward value. If the action chosen by the agent satisfied the goal of the environment, a positive reward would be returned. The agent's aim is to maximize the amount of positive reward it can receive. Therefore, over a long period of time, the agent will learn the perfect set of actions that lead to the highest accumulative reward.

RL is unique from other traditional AI because it is currently the closest form of natural intelligence [2]. Through understanding mistakes and trying to maximize correct actions, it never stops learning. The agent is incentivized to maximize its reward and will aim to find actions that will yield more reward. This constant state, action, and reward loop is what teaches the agent improve by making small adjustments to the policy after every cycle.

1.2 Elements of Reinforcement Learning

Other than the agent which makes decisions and the environment which the agents seeks to achieve a goal despite uncertainty [2], there are four other aspects which build up every RL system: a policy, reward signal, value function, and a model.

The policy is the set of rules which determines how an agent behaves at a given time. It is a mapping of experienced states to actions taken when in those experienced states, where the aim is to select the appropriate action to achieve its goal [3]. The policy of an agent can be a simple lookup table of states to actions and their corrosponding reward or involve a complex computation [2].

The reward signal of an environment defines the problem an agent aims to solve. After every action performed on a state, the agent recieves a numerical value called reward. The aim of the agent is to maximize this reward value in the long term, as reward is an indication by the environment of good and bad decisions. The reward recieved is associated to the action chosen for the experienced state, which is the basis for altering the policy to increase the probability of choosing better actions in the same or similar states [2].

While reward is a measure of how well an action is, in the given state, value is the long term reward of short-term actions. Value is the accumulative reward over a given time, which is more important when judging action choices. A set of actions that lead to a high amount of reward in the short-term is considered to be high in value. However, in the long-term where the chosen actions do no lead to increases in reward would have a low value. Despite of value in achieving the goal of the environment, value must be estimated by the agent over its lifetime while reward is given by the environment. Therefore, value is the most important and influental component of RL to making an optimal policy [2].

The last essential aspect of RL is the model of the environment. The model of the environment is an inferences of how the environment will behave. Given a state action pair, the model would be able to make a prediction of returned reward for the state-action pair and resulting next state given the action. Models are for forward planning and performing actions which will yield further high rewards, which are used in model-based methods of learning, contrasting from trial-and-error methods called model-free learning [2].

1.3 Aims of the Project

1.4 Report Structure

2 Literature Review

List of things to cover:

- justify research: -> what is pkmn and RPG games & why did I chose pkmn red

- up-to-date with relevant literature: -> the algorithms I plan on using and why they modern?

2.1 Introduction to Pokemon Red

2.2 Why Pokemon Red?

RL has only been applied to every Atari game, where it surpassed the human benchmark for every game [4]. However, these games lack long term randomness, where present actions influencing future states and future decisions. The game 'Pokémon Red' is an RPG game filled with various puzzles, a non-linear world, and a large amount of variance making each play through of the game unique while keeping consistent goals. In addition, the game has 2 states the players is constantly in, the player is either in an overworld where they control their movement on a map or they are in a battle with another individual, where they control the actions of their monster.

I chose to apply RL to this environment, Pokémon Red, because of the benefits it holds during trianing and applications of this research. This version of the game has the ability to speed-up the environment which allows for more timesteps to be completed so the agent can experience more states. Another reason is because its complexity. The end goal of Pokémon Red is to defeat all the gym leaders and become the champion. However, to reach this goal the player must complete a series of smaller tasks which are not explicitly specified in the reward function. An example of this would be navigation a 2-dimensional plane, solving puzzles and performing pokemon battles along the way. Getting the agent to learn smaller tasks while completing the main goal of the environment can be applied and extended to the real world. Compared to other forms of AI, RL never stops learning even when deployed, which makes it a very effective method to adapt to new environments outside of the simulation and constantly learn to improve itself.

Other similar projects which applies RL to find the optimal battling strategy by Kalose et al [5]. Their work focuses on one aspect of the game and does not have a large enough search space to justify application of RL techniques. Another similar work by Flaherty, Jimenez and Abbasi [6] applies RL algorithms A2C and DQN to play Pokémon Red. However, this piece of work does not go into enough detail about the comparison of different RL techniques to find the best method to train an agent to complete large complex environments with a large search space. I aim to extend their research in applying RL to Pokémon Red by applying more algorithms and various techniques RL that I will go into more detail in section 3.

2.3 RL Algorithms and Methods

3 Research Problem

3.1 Background

3.2 Analysis

3.3 Problem Approach

4 Analysis

5 Problem Approach

6 Implmentation & Design

7 Analysis

8 Conclusion

9 References

[1] O. Francon, S. Gonzalez, B. Hodjat, *et al.*, "Effective reinforcement learning through evolutionary surrogate-assisted prescription," in *Proceedings of the 2020 Genetic and evolutionary computation conference*, 2020, pp. 814–822.

[2] R. S. Sutton, F. Bach, and A. G. Barto, *Reinforcement learning: An introduction*, Second. MIT Press Ltd, 2018.

[3] G. De, "What is a policy in reinforcement learning?," 2023.

[4] G. Brockman, V. Cheung, L. Pettersson, *et al.*, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[5] A. Kalose, K. Kaya, and A. Kim, "Optimal battle strategy in pokemon using reinforcement learning," *Web: https://web. stanford. edu/class/aa228/reports/2018/final151. pdf*, 2018.

[6] J. Flaherty, A. Jimenez, B. Abbasi, B. Abbasi, J. Flahery, and A. Jimenez, "Playing pokemon red with reinforcement learning," 2021.