Project_Proposal_TeamPacman

May 4, 2024

1

The Pac

1.1 Names

- Edwin Ruiz
- Bradley Grace

1.2 Abstract

Our project aims to develop an intelligent agent that navigates the Atari game Pac-Man efficiently by making smart decisions using OpenAI Gym as the simulation platform. Our primary dataset will be the game configurations or states which includes all current in-game elements like Pac-Man's position, the ghosts, and the dots. The intelligent agent will leverage reinforcement learning algorithms to improve its decision-making and game strategies continuously. We will assess the agent's effectiveness by monitoring its game scores, level completion rates, and its efficiency in preserving lives during gameplay.

1.3 Background

As we know, Pac-Man is a widely known and it is on the top list of the most iconic arcade games. It is also particularly popular in the field of computational research like artificial intelligence. This is because the dynamic constraints of the environment make it an excellent research tool for evaluating reinforcement learning algorithms such as Deep Q-Networks (DQNs) and Q-learning. Both of which are good in navigating environments like Pac-Man states because they are able to balance the exploration of new strategies by making use of known paths, according to the environmental feedback received [1][2][3]. In this way this game is a great educational tool for students looking to explore and experiment with diffent AI methods.

1.4 Problem Statement

Our goal is to develop a reinforcement learning model that is able to efficiently navigate the complexity in decision-making to win in the game of Pac-Man. Furthermore, we want to be able to enhance the agent's ability to make real-time decisiions (not just learn to play the game). All of which is quantifiable by the game's scoring system, measurable through the agent's performance metrics, and reproducible through OpenAI Gym's simulation interface.

1.5 Data

Again, our data will be the live state generated by the OpenAI Gym environment for the Atari-Pac-Man game. The data source/environment can be found here https://gymnasium.farama.org/environments/atari/pacman/ . Each state will be captured as a pixel array that visually represents Pac-Man's current position, the ghosts, and the remaining dots. Key variables are the pixel data, the current game score, and the number of lives left. We anticipate some preprocessing of these pixel arrays to optimize them for model training.

1.6 Proposed Solution

We plan to use Deep Q-Networks (DQNs) to train the Pac-Man agent. The agent model will be trained iteratively by giving the agent rewards based on improvements of the game score and penalties for losing lives. Although, not sure about the specific code yet, we will implement the training process using Python and TensorFlow. In addition, we might consider using a simple Q-learning agent to evaluate the improvements done by the DQNs.

1.7 Evaluation Metrics

- **Average Score per Game for performance** : $\frac{\sum \text{Total Scores}}{\text{Number of Games Played}}$
- **Levels Completed for progress** : simple count (1, 2,....n)
- **Survival Time for strategy effectiveness** : Time measured in seconds (start to losing a life)

1.8 Ethics & Privacy

A. Data Collection - [x] A.1 Limit PII exposure: Our project involves no personal data as it utilizes publicly available simulations, via OpenAI Gym.

B. Data Storage - [x] B.1 Data security: We will securely store training data and models on GitHub, which will be made public after the project's completion.

C. Analysis - [x] C.1 Dataset bias: We will monitor for possible biases from the agent for discovering possible game flaws. - [x] C.2 Honest representation: We will present all results and performance metrics truthfully and accurately. - [x] C.3 Auditability: We will assure reproducibility through complete documentation.

D. Modeling - [x] D.1 Explainability: All strategic decisions by the agent will be thoroughly explained. - [x] D.2 Communicate limitations: We will clearly document any model limitations.

E. Deployment - [x] E.1 Monitoring and evaluation: The model will be regularly monitored and updated.

2 Team Expectations

Team Expectation 1: Regular communication weekly through virtual meetings or our group chat

Team Expectation 2: Conflicts will be addressed openly and constructively

Team Expectation 3: We will review each other's work to hold each other accountable

Team Expectation 4: Evenly split the responsibilities of the project to meet deadlines

3 Citations

1. Doe, J. (2020, April 5). How to train Ms. Pac-Man with reinforcement learning. Medium.
2. Mnih, V., et al. (2015). "Human-level control through deep reinforcement learning." Nature.
3. Silver, D., et al. (2016). "Mastering the game of Go with deep neural networks and tree search." Nature.