42186 Model-based Machine Learning

Elysia Livia Gao s222445
Daniela Bahneanu s184366
Diana Podoroghin s194768
Mathias Hovmark s173853

Technical University of Denmark
March 30, 2023

1 Project Outline

1.1 Dataset and Research Questions

Research Questions

This research investigates the use of Partially Observable Markov Decision Processes (POMDPs) with text-parsing to train computers to play complex video games such as Space Invaders. While Markov Decision Processes (MDPs) have been widely used in reinforcement learning research for game-playing agents such as AlphaGo, Pong, and Atari games, the incorporation of text-parsing and Large Language Models (LLMs) in POMDPs can provide even better training for game-playing agents. In-game text, tutorials, walkthroughs, or guides can offer valuable information to help agents perform actions and maximize rewards. This research aims to address the following research questions:

1. Can we build a PGM that incorporates both large language models and reinforcement learning to learn to play complex video games, and how does the complexity of the game environment affect the performance of the model? Can we improve performance of games that struggle with reinforcement learning alone ?
2. Can we use or investigate further transfer learning (using pre-trained LLMs to improve performance of RL agent, this would reduce training need for RL and improve performance) in the context of computer-trained video games ?
3. Can we improve upon or investigate further these areas of a Markov Decision Process or Partially Observable Markov Decision Process in RL ? (Incorporating uncertainity, handling high-dimensional state spaces, improving exploration, multi-objective RL etc.)

Dataset

We prepared and collected the following three datasets in order to investigate our research questions:

- Text data: This dataset was collected from web-scraping websites containing walkthroughs, in-game text, or forums revolving game strategy. We first web-scraped a series of websites from Google queries of "space invader walkthroughs", "space invader guides", "space invader manuals", and "space invader strategy". After collecting a list of websites, we web-scraped text from those websites. Next, we pre-processed text in the following way:

  - **1)** If the data read from the website is split into multiple cells, these are concatenated such that each row in the dataset only contains one word-vector.

  - **2)** A second version of the read text is then created by lowercasing, removing punctuation and stopwords, and lemmatizing the text. This process might be modified if it is found that it removes information that's important for the model.

  - **3)** Finally the read text is tokenized using GloVe - e.g. each word in a word-vector is given a number corresponding to the word's position in the GloVe vocabulary. It is then quickly checked that the GloVe vocabulary covers most of the words used in the dataset. A bag-of-words representation of each word-vector is also created.

  The data currently contains information from some websites that are unrelated to Space Invaders (e.g. a website which contains information about an episode of The Simpsons). We will leave this final cleaning of the data for later.

- Game environment dataset: This dataset would consist of information about the game environment, such as the game state, game mechanics, and rules of the game. This information could be gathered from game manuals, wikis, or game code. This dataset was prebuilt into OpenAI Gym's reinforcement learning API that had a list of actions, game mechanics, etc.
- RL training dataset: This dataset would consist of examples of game states and the optimal actions that the agent should take in those states to maximize the reward. This dataset was created by using OpenAI Gym's and writing code that self-played 100 episodes to test, choosing random actions and evaluating the reward values.

1.2 First Draft of Probabilistic Graphical Model & Generative Process

Probabilistic Graphical Model

Generative Process

1. Initialize the starting state $s_0 \sim \text{categorical}(\pi)$
2. For each time $t \in 0, ..., T$:
   (a) Draw an observation $o_t \sim \text{multinomial}(o_t|s_t)$
   (b) Draw an action $a_t \sim \text{multinomial}(a_t|o_t, o_{t-1}, ..., o_0)$ and possibly previous observations
   (c) Draw next state $s_{t+1} \sim \text{multinomial}(s_{t+1}|s_t, a_t)$
   (d) Calculate the reward r_t using reward function $R(s_t, a_t, s_{t+1})$

Note: for the text dataset, we will most likely use embeddings for the observations (o_t) which will be input to the multinomial distribution, thus, we do not need to update anything regarding the probabilities of the categories (we keep 2a as stated).

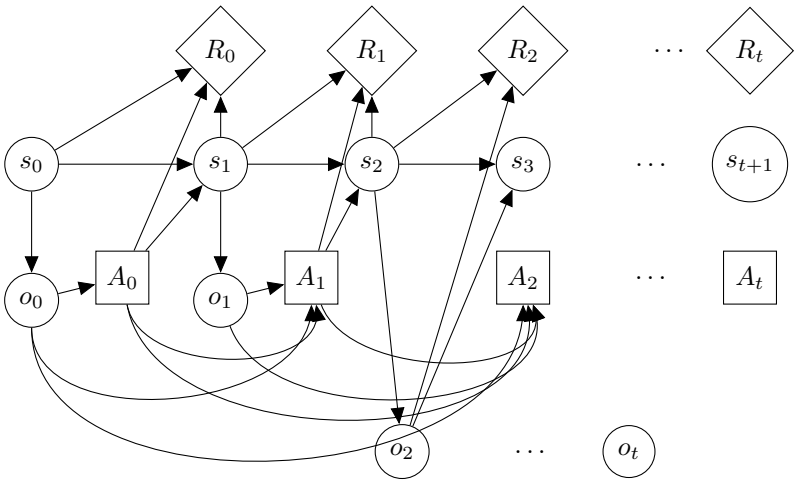

Figure 1: PGM