Rethinking Neural Networks in Reinforcement Learning

Anonymous Authors[1]

Abstract

Reinforcement learning (RL) has seen significant advancements by using neural network architectures. In this study, we systematically investigate the performance of several neural networks commonly employed in RL, including Long Short-Term Memory (LSTM), Multi-Layer Perceptron (MLP), Mamba, Transformer, Gated Recurrent Unit (GRU), and Kolmogorov-Arnold Networks (KAN). These architectures are evaluated across a variety of task settings, encompassing continuous control (e.g., MuJoCo environments), discrete decision-making (e.g., Atari games), and memory-based tasks (e.g., Minigrid environments). By analyzing their performance in these domains, we identify key strengths and limitations of each architecture and highlight their suitability for specific types of RL problems. Furthermore, we provide actionable insights into selecting neural network architectures based on task characteristics and performance requirements, offering practical guidance for researchers and practitioners in designing effective RL systems.

1. Introduction

Reinforcement learning (RL) has emerged as a powerful paradigm for decision-making tasks, with neural networks playing a crucial role in enabling agents to learn complex policies. Proximal Policy Optimization (PPO) (Schulman et al., 2017) is one of the most widely adopted RL algorithms due to its simplicity, robustness, and strong empirical performance. However, the choice of neural network architecture for PPO significantly impacts its effectiveness across diverse environments and hasn't been widely explored in the literature.

In this paper, we systematically evaluate the impact of various neural network architectures on PPO performance across a spectrum of tasks. These tasks include environments requiring memory, such as partially observable Markov decision processes (POMDPs), and environments focused on continuous control and discrete decision-making. By analyzing the strengths and weaknesses of architectures such as LSTM, GRU, Transformer, Mamba (Gu & Dao, 2024), and MLP, we aim to provide actionable insights into the design of RL systems.

Previous works have explored implementation details of PPO (Huang et al., 2022), what methods could be used to improve the agent performance in various environments (Andrychowicz et al., 2020), and studies like (Pleines et al., 2024) have demonstrated the efficacy of TransformerXL in episodic memory tasks. However, these studies often overlook comparisons with simpler architectures, such as PPO-LSTM, and emerging architectures, like PPO-Mamba. Furthermore, existing benchmarks, such as those conducted in Memory Gym (Pleines et al., 2024) and MiniGrid (Chevalier-Boisvert et al., 2023), highlight the need for memory in certain environments but lack comprehensive comparisons across a broader range of architectures and tasks.

Our contributions are threefold:

- We benchmark PPO implementations with a variety of neural network architectures, including traditional (MLP, LSTM), advanced (Transformer, GRU), and novel (Mamba) models.
- We evaluate these architectures in memory-intensive environments such as MiniGrid and Memory Gym, as well as in continuous and discrete control tasks like MuJoCo and Atari.
- We analyze the trade-offs between memory requirements, computational efficiency, and task performance, offering practical guidelines for selecting architectures based on task characteristics.

The rest of this paper is organized as follows: Section 2 provides an overview of related work. Section 3 describes the experimental setup, including the environments, architectures, and evaluation metrics. Section 4 presents the results and discussion, and Section 5 concludes with actionable insights and future research directions.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2. Related Work

2.1. Proximal Policy Optimization

PPO (Schulman et al., 2017) is a policy gradient method designed to balance exploration and exploitation by using a clipped surrogate objective. It has become a cornerstone in modern RL research due to its stability and ease of implementation. Despite its widespread adoption, the impact of neural network architectures on PPO's performance remains underexplored, motivating this study.

2.2. Memory in RL

Many RL tasks, particularly those modeled as POMDPs, require agents to maintain memory to succeed. Environments like MiniGrid (Chevalier-Boisvert et al., 2023) and Memory Gym (Pleines et al., 2024) serve as benchmarks for evaluating memory capabilities. While TransformerXL (Pleines et al., 2023) and GRU have shown promise in these settings, simpler architectures like LSTMs and novel architectures like Mamba have not been extensively compared.

2.3. Novel Architectures in RL

Recent advancements, such as Mamba (Gu & Dao, 2024), offer linear-time sequence modeling with selective state spaces, making them suitable for RL tasks requiring long-term dependencies. However, their performance in RL settings, particularly when integrated with PPO, is yet to be fully understood.

3. Experimental Setup

3.1. Environments

We consider a diverse set of environments to evaluate the architectures:

- **MiniGrid:** A collection of modular environments designed for goal-oriented tasks requiring memory (Chevalier-Boisvert et al., 2023). For our experiments, we use the **Door-Key** environment and the **Memory** environment:
  - **Door-Key:** The agent must pick up a key to unlock a door and reach the green goal square. This task involves sparse rewards and requires exploration strategies.
  - **Memory:** The agent starts in a room where it observes an object, navigates through a narrow hallway, and chooses between two objects at the end of the hallway. It must remember the initial object to succeed.
- **Memory Gym:** These environments are tailored to evaluate memory capabilities in RL (Pleines et al., 2023). We test on the **Endless Mystery Path**, where the agent observes only a segment of the environment at a time. This setting requires the agent to infer its position and direction by recalling previous observations. An example includes procedurally generated endless paths.
- **MuJoCo:** A physics-based continuous control suite for benchmarking sample efficiency and stability. We use the **Hopper**, **HalfCheetah**, and **Humanoid** tasks.
- **Atari:** Classic discrete decision-making environments with high-dimensional state spaces. We evaluate on **Breakout** and **Pong**.

Figures illustrating MiniGrid and Memory Gym environments can be seen below:

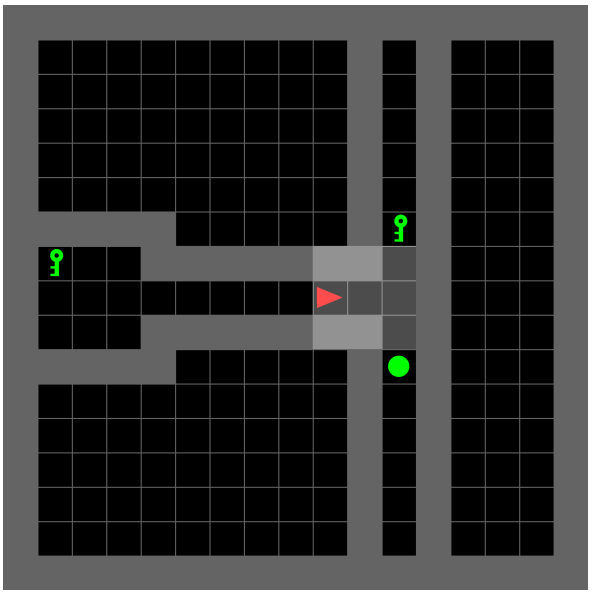

Figure 1. Example of MiniGrid environments: Memory.

3.2. Architectures

We integrate the following neural network architectures into PPO:

- **MLP:** A simple feedforward network serving as a baseline.
- **LSTM and GRU:** Recurrent networks for handling sequential dependencies.
- **Transformer and TransformerXL:** Advanced architectures with attention mechanisms for episodic memory tasks.
- **Mamba:** A novel architecture for efficient sequence modeling (Gu & Dao, 2024).

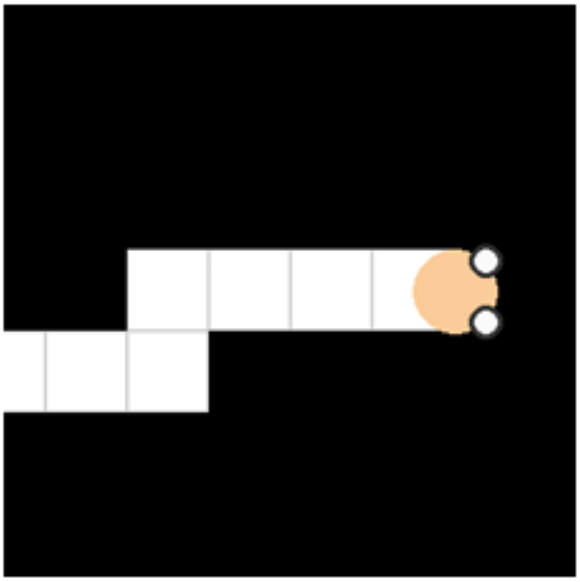

Figure 2. Memory Gym environment: Endless Mystery Path. The agent observes a segment of the environment at a time.

3.3. Metrics

The architectures are evaluated based on:

- **Average Return:** The cumulative reward achieved by the agent.
- **Sample Efficiency:** The amount of experience required to reach peak performance.
- **Computational Efficiency:** Training time and resource usage.
- **Stability:** Variance in performance across random seeds.

References

Andrychowicz, M., Raichuk, A., Stańczyk, P., Orsini, M., Girgin, S., Marinier, R., Hussenot, L., Geist, M., Pietquin, O., Michalski, M., Gelly, S., and Bachem, O. What matters in on-policy reinforcement learning? a large-scale empirical study, 2020. URL https://arxiv.org/abs/2006.05990.

Chevalier-Boisvert, M., Dai, B., Towers, M., de Lazcano, R., Willems, L., Lahlou, S., Pal, S., Castro, P. S., and Terry, J. Minigrid miniworld: Modular customizable reinforcement learning environments for goal-oriented tasks, 2023. URL https://arxiv.org/abs/2306.13831.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces, 2024. URL https://arxiv.org/abs/2312.00752.

Huang, S., Dossa, R. F. J., Raffin, A., Kanervisto, A., and Wang, W. The 37 implementation details of proximal policy optimization. In *ICLR Blog Track*, 2022. URL https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/. https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/.

Pleines, M., Pallasch, M., Zimmer, F., and Preuss, M. Transformerxl as episodic memory in proximal policy optimization. *Github Repository*, 2023. URL https://github.com/MarcoMeter/episodic-transformer-memory-ppo.

Pleines, M., Pallasch, M., Zimmer, F., and Preuss, M. Memory gym: Towards endless tasks to benchmark memory capabilities of agents, 2024. URL https://arxiv.org/abs/2309.17207.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

A. Appendix.

Appendix