GPT2PPO: Auto Regressive Proximal Policy Optimization*

*Note: Sub-titles are not captured in Xplore and should not be used

1st Andrei Cozma
Department of Electrical Engineering & Computer Science
University of Tennessee
Knoxville, United States
acozma@vols.utk.edu

2nd Hunter Price
Department of Electrical Engineering & Computer Science
University of Tennessee
Knoxville, United States
hprice7@vols.utk.edu

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—Reinforcement Learning, PPO, GPT2**

I. Introduction

In this project we explore the use of transformers in the context of Reinforcement Learning. The majority of theoretical works assume that problems follow a Markovian process, which is not always the case. Some problems need the contxt of previous states and actions to make an informed decision on the next decision. As a result, we propose an addition to the basic Proximal Policy Optimization (PPO) algorithm by using the Generative Pre-trained Transformer 2 (GPT2) model as the encoder for the critic network. This will allow the critic network to take into account the context of previous states and actions as well as apply attention to past states and actions that may be important. We will test this model on the LunarLander-v2 and Acrobot-v1 OpenAi Gym environments with discrete action spaces and compare it to the original PPO algorithm. Additionally we will test the model on BipedalWalker-v3 with continuous action spaces and compare it to the original PPO algorithm.

II. Previous Work

PPO citation - [1]

III. Background

[2]

A. Lunar Lander

The Lunar Lander environment is rocket trajectory optimization problem. The OpenAI Gym offers two versions of the environment: discrete or continuous. In this work we only use the discrete version. The state space is a 8-dimensional vector containing the position and velocity of the rocket, the angle and angular velocity of the rocket, and the position of the landing pad.

B. Acrobot

C. Bipedal Walker

IV. Methodology

A. Reinforcement Learning Methods

B. Code Design

C. main_train.py

D. main_eval.py

V. Results

VI. Conclusion

References

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.

[2] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, “Openai gym,” *arXiv preprint arXiv:1606.01540*, 2016.

Identify applicable funding agency here. If none, delete this.

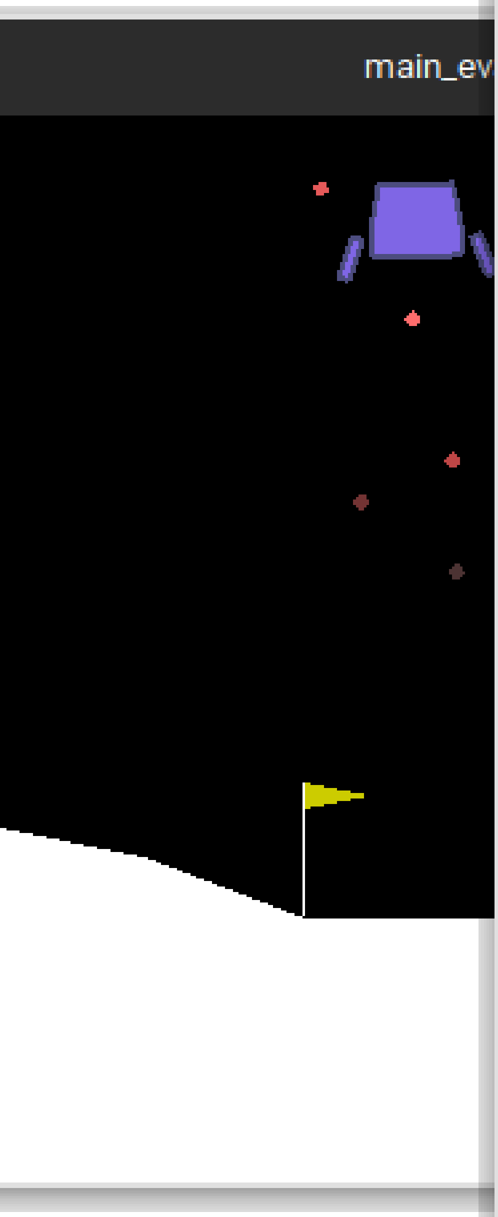

Fig. 1. The Lunar Lan

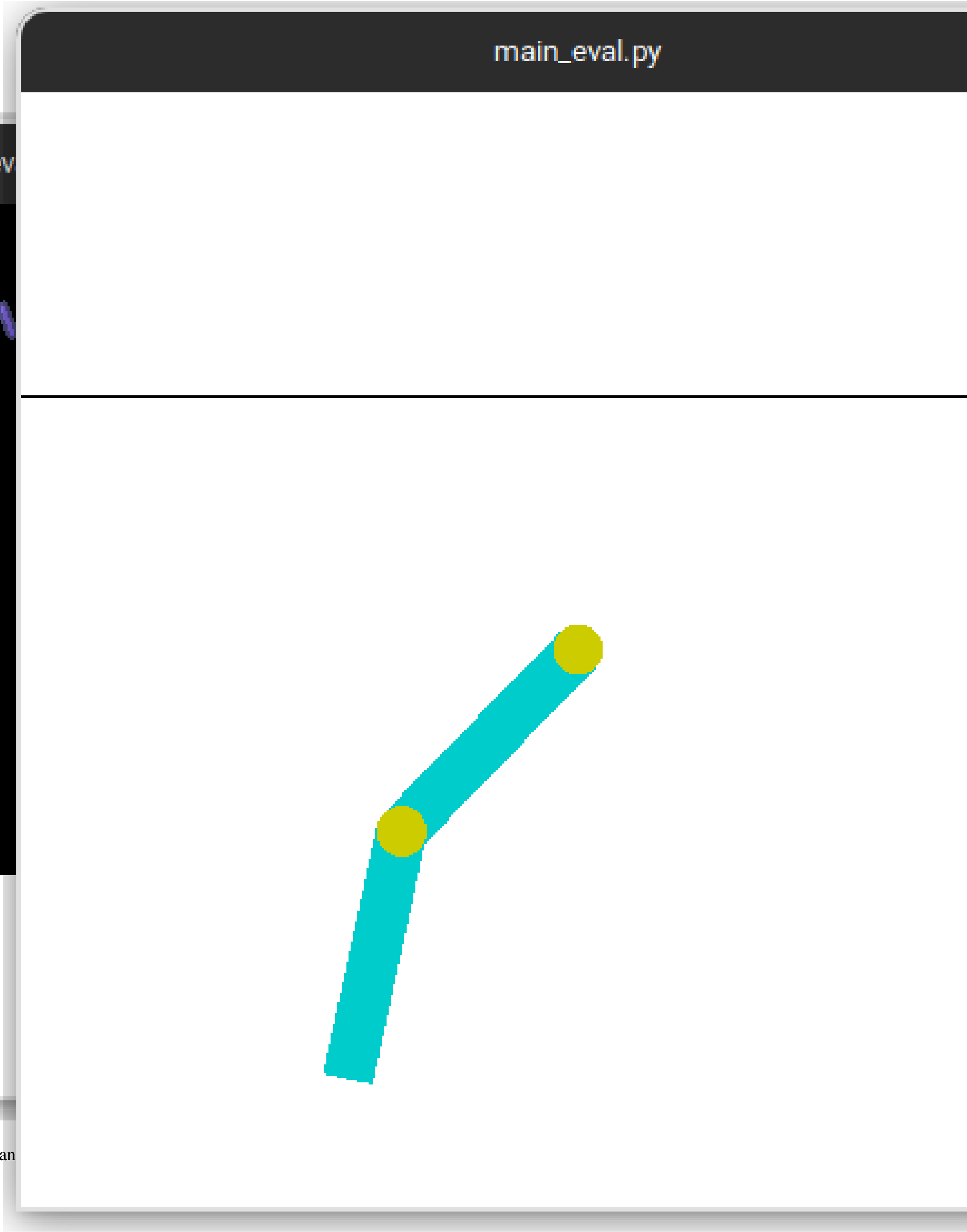

Fig. 2. The Acrobot environment.