

Uso de métodos de Deep Reinforcement Learning para tentar obter um modelo de IA capaz de concluir o jogo Frogger

Pedro Pinheiro Borges, Rafael Camargo e Moises Moreira Goncalves Feltrin Thimoteo

I. INTRODUÇÃO

O avanço acelerado das tecnologias de Inteligência Artificial e Aprendizado de Máquina tem revolucionado diversos campos da ciência e da indústria, incluindo o domínio dos jogos eletrônicos. Em particular, o campo do Aprendizado por Reforço Profundo (Deep Reinforcement Learning) tem se mostrado uma abordagem poderosa para resolver problemas complexos de tomada de decisão em ambientes dinâmicos e desafiadores, como os jogos de vídeo game.

Neste contexto, o objetivo do presente trabalho é aplicar técnicas de Deep Reinforcement Learning para resolver um dos desafios clássicos dos jogos Atari, o famoso "Frogger". Lançado pela Konami em 1981, Frogger é um jogo de arcade que desafia os jogadores a controlar um sapo através de um trânsito intenso, perigoso e repleto de obstáculos, com o objetivo de levá-lo em segurança até seu habitat no outro lado da estrada.

O desafio de resolver o jogo de Frogger reside na necessidade de tomar decisões rápidas e precisas para evitar colisões com veículos em movimento, enfrentar rios caudalosos e desviar de outros obstáculos, exigindo habilidades estratégicas e de coordenação sob pressão.

Com base na premissa de que o Deep Reinforcement Learning pode capturar padrões complexos e aprender estratégias eficientes para atingir metas específicas, exploraremos a utilização de algoritmos de aprendizado profundo, como o Deep Q-Network (DQN) e suas variantes, para treinar um agente virtual capaz de dominar o jogo de Frogger com maestria.

Ao longo deste relatório, apresentaremos a formulação do problema, a descrição das técnicas e arquiteturas utilizadas, os resultados obtidos, bem como uma análise crítica das limitações e possibilidades futuras para a aplicação do Deep Reinforcement Learning em jogos eletrônicos e em outros cenários de tomada de decisão.

II. IMPLEMENTAÇÃO

O ambiente do jogo Frogger possui um espaço de ações descrito pela tabela I. O espaço de estados do frogger consiste na posição do frogger no grid do mapa.

Usou-se como base o código do Laboratório 12 da matéria CT-213 ministrada no segundo semestre de 2023, no ITA. As principais mudanças foram na arquitetura da rede neural usada para o aprendizado e na função de recompensa.

Para a arquitetura da rede neural usou-se uma rede com 5 camadas, descritas na tabela II.

Value	Ação
0	Não fazer nada
1	UP
2	RIGHT
3	LEFT
4	DOWN

TABELA I
ESPAÇO DE AÇÕES DO FROGGER

Camada	Neurônios	Tamanho do Kernel	Strides	Função de Ativação
Conv2D	32	8	4	ReLU
Conv2D	64	4	2	ReLU
Conv2D	64	3	-	ReLU
Dense	512	-	-	ReLU
Dense	action.size	-	-	linear

TABELA II
ARQUITETURA DA REDE NEURAL

Para a função de recompensa, foi utilizado os critérios descritos nas tabelas III e IV. Na primeira, é descritos recompensas maiores, para objetivos mais gerais. Na segunda, é mostrado recompensas para cada ação específica, com o intuito de guiar melhor o treinamento.

Morte	-100
Atravessou a rua	100

TABELA III
RECOMPENSAS PARA ESTADOS GERAIS

Ação	Recompensa
0	-0.5
1	0.0
2	0.0
3	0.0
4	-1.0

TABELA IV
RECOMPENSA PARA CADA AÇÃO

O objetivo das recompensas menores é não incentivar o agente a ficar parado na mesma posição e nem a voltar para trás.

III. RESULTADOS E DISCUSSÕES

Infelizmente, o grupo não conseguiu rodar a otimização em uma GPU para vários episódios de treinamento, apenas usando CPU, fato que quase triplica a duração de um episódios de treinamento. Em média, um episódio de treinamento dura cerca de 1 minuto. Por causa desse problema, não conseguiu-se rodar muitos episódios para o treinamento e, conseqüentemente, não pode-se obter resultados tão satisfatórios.

Com isso, rodou-se apenas 60 episódios de treinamento, cujo os valores de Recompensa em cada episódio estão dispostos na figura 1

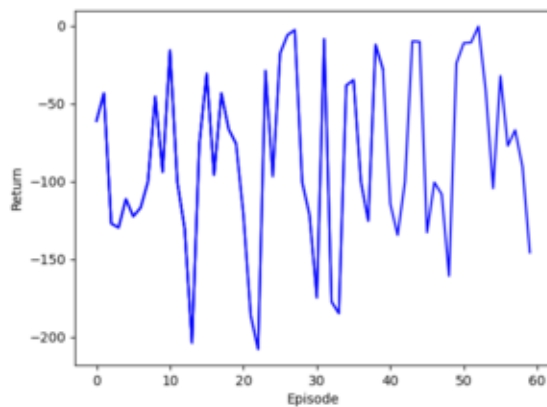


Fig. 1. Recompensa em cada episódio

Aparentemente, com os resultados obtidos, pode-se supor que o modelo é promissor para a resolução do problema proposto já que na maioria dos casos a recompensa total foi maior que -100, indicando que o agente buscou sobreviver nesses episódios de treinamento.

IV. CONCLUSÃO

Deste projeto, conclui-se que possivelmente obteve-se um modelo promissor para se obter um modelo de inteligência artificial capaz de vencer o jogo Frogger, no entanto é extremamente necessário um hardware minimamente poderoso, disposto de uma GPU, para conseguir rodar uma quantidade suficiente de episódios para se conseguir um modelo capaz de vencer o jogo.

REFERÊNCIAS

- [1] Laboratório 12 da disciplina CT-213
- [2] Slides das aula 13 da disciplina CT-213