

SprawozdanieMIO

Aleksander Siekliński, Wiktor Szewczyk, Jan Gorczyński

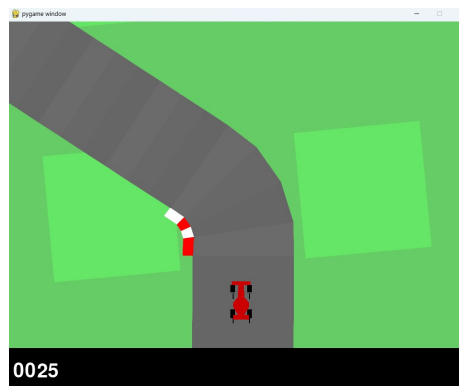
June 2024

1 Cel projektu

W naszym projekcie postanowiliśmy nauczyć sztuczną inteligencję grać w grę. Jako grę wybraliśmy 'CarRacing-v2' czyli jedną z gier dostępnych w bibliotece gymnasium. Jako algorytmów postanowiliśmy użyć algorytmów genetycznych oraz głębokiego uczenia ze wzmocnieniem.

2 Gra 'CarRacing-v2'

Gra 'CarRacing-v2' opiera się na pokonaniu toru przez samochód wyścigowy w jak najkrótszym czasie. W grze jako akcje przyjmujemy wektor trzech liczb, z których pierwsza odpowiada za skręt od -1 (w lewo) do 1 (w prawo), druga za gaz, a trzecia za hamulec. Po każdym wykonaniu akcji dostajemy z powrotem następny stan gry oraz nagrodę, która jest tym większa, im dalej i szybciej pojechaliśmy.



Rysunek 1: Gra 'CarRacing-v2'

3 Algorytm genetyczny

Tworząc algorytm genetyczny, skorzystaliśmy z biblioteki 'DEAP' w Pythonie. Jako metodę wybierania potomków użyliśmy 'selTournament' dostępnego w bibliotece 'DEAP'. Metoda ta polega na tworzeniu "konkursu" pomiędzy potomkami w celu wybrania najlepszego. Trenowaliśmy algorytm przez 40 pokoleń (około 8 godzin).

4 Uczenie przez wzmocnienie

4.1 Proximal Policy Optimization

W tym algorytmie sieć neuronowa uczy się wykonywać najbardziej optymalną akcję w danym stanie gry. Sieć konwolucyjna, dla wejścia, którym jest kilka kolejnych czarno-białych ramek z gry, zwraca najbardziej optymalne wyjście w postaci wektora trzech liczb. Skorzystaliśmy z biblioteki 'stable baselines3'.

4.2 Deep Q learning

W tym algorytmie staramy się przewidzieć wartość Q , czyli oczekiwaną nagrodę dla danego stanu gry. Potrzebna jest skończona liczba akcji. Wyjściem sieci jest n neuronów, pokazujących oczekiwaną nagrodę dla n stanów gry. Staraliśmy się zaimplementować ten algorytm od zera w PyTorchu, jednak nie udało nam się osiągnąć dobrych rezultatów.

5 Podsumowanie

Proces treningu tego typu algorytmów jest bardzo czasochłonny i trudny do zrównoleglenia. Udało nam się osiągnąć pewien stopień skuteczności, jednak nie wystarczyło nam czasu na udoskonalenie naszych algorytmów.