

### 1. Informacje ogólne

Tytuł projektu: Czy na MuZero można odpalić DOOM'a?

Podstawa projektu: Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model

<https://arxiv.org/pdf/1911.08265>

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, David Silver

Repozytorium:

[https://github.com/MarekBorzyszkowski/MuZero\\_redone](https://github.com/MarekBorzyszkowski/MuZero_redone)

### 2. Cel projektu

Próba powtórzenia eksperymentu z publikacji i rozszerzenie eksperymentu o nowe środowisko DOOM.

### 3. Przegląd artykułu

Model/Algorytm bazowy: MuZero

Streszczenie artykułu: Przedstawienie algorytmu MuZero i porównanie jego osiągnięć względem istniejących rozwiązań.

Kluczowe wnioski: MuZero radzi sobie zarówno z grami wymagającymi planowania (takimi jak szachy, czy Go) oraz posiadającymi skomplikowany interfejs graficzny. MuZero łączy planowanie z brakiem odtworzonego w pełni środowiska, posiadając tylko dań istotne informacje w warstwie ukrytej. Przez to sprawdza się w sytuacjach, gdzie nieznane jest całe środowisko, zachowując sprawność w momentach gdzie wymagane jest planowanie.

### 4. Przegląd literatury:

Przydatne repozytoria:

implementacje MuZero:

<https://github.com/johan-gras/MuZero>

<https://github.com/opensdilab/LightZero>

<https://github.com/werner-duvaud/muzero-general>

Zbiór gier atari:

<https://github.com/Farama-Foundation/Arcade-Learning-Environment>

<https://paperswithcode.com/dataset/dqn-replay-dataset>

Potencjalne inne środowiska do uczenia:

<https://github.com/clvrai/awesome-rl-envs>

<https://github.com/Farama-Foundation/ViZDoom>

### 5. Plan implementacji

Wybrane narzędzia i biblioteki: Python, torch

Wybrane środowisko gry: Atari 60 games

Zaproponowane modyfikacje: DOOM

Podział zadań: Marek – pisanie dokumentacji + trening/testy + integracja z grami

Paulina – implementacja modelu i pierwsze testy

### 6. Wstępne Wyniki:

Pierwsze obserwacje: Model może być ciekawy do testów, gdyż podobno sprawdza się w grach wymagających predykcji, jak i wymagających graficznie. Mimo stwierdzeń autorów o jakości modelu, da się znaleźć zestawienia ukazujące, że model niekoniecznie radzi sobie aż tak dobrze względem konkurencji.

Problemy i wyzwania: Sama implementacja może okazać się wyzwaniem, jak i nie znamy zapotrzebowania na moc obliczeniową. Dodatkowo pewną niepewnością może napawać niesprawdzone jeszcze środowisko DOOM.

### 7. Bibliografia:

<https://arxiv.org/pdf/1911.08265>

<https://github.com/johan-gras/MuZero>

<https://github.com/opensilab/LightZero>

<https://github.com/werner-duvaud/muzero-general>

<https://github.com/Farama-Foundation/Arcade-Learning-Environment>

<https://paperswithcode.com/dataset/dqn-replay-dataset>

<https://github.com/clvrai/awesome-rl-envs>

<https://github.com/Farama-Foundation/ViZDoom>

<https://paperswithcode.com/sota/atari-games-100k-on-atari-100k>