



Wydział Matematyki i Nauk Informatycznych

POLITECHNIKA WARSZAWSKA

Warsztaty z technik uczenia maszynowego

Przewidywanie zwycięzcy rundy w grze
Counter Strike: Global Offensive

Jakub Michalak, Damian Opoka, Piotr Kryczka, Adam Ryl

10 marca 2022

1 Temat projektu

Temat projektu wraz z danymi pochodzi ze [strony kaggle](#).

Celem projektu jest klasyfikacja zwycięzcy rundy w grze wieloosobowej Counter Strike: Global Offensive (w skrócie CS:GO). Rundę rozgrywają dwie drużyny - terroryści i antyterroryści. Gra jest bardzo taktyczna i zespołowa, jednak parametry takie jak wyposażenie, punkty zdrowia gracza czy liczba żyjących graczy mają w dużym stopniu, a niekiedy kluczowy wpływ na wynik danej rundy i przebieg całego meczu.

Na podstawie zgromadzonych danych wejściowych zostanie stworzony program, który będzie przewidywał zwycięzcę danej rundy.

2 Gra Counter Strike: Global Offensive

CS:GO to taktyczna gra typu FPS stworzona przez studio Valve, w której dwie drużyny (terroryści T i antyterroryści CT) rozgrywają mecz składający się z maksymalnie 30 rund, których podstawowy czas trwania 1 minuta i 55 sekund. W każdej drużynie jest po 5 graczy (w sumie 10), a grę wygrywa ta drużyna, która jako pierwsza wygra 16 rund.

Na początku jedna drużyna gra jako CT, a druga jako T. Po rozegraniu 15 rund drużyny zamieniają się stronami. Terroryści mogą wygrać rundę na dwa sposoby:

- eliminując drużynę przeciwnika,
- podkładając w strefie detonacji bombę i doprowadzając do jej wybuchu po 40 sekundach.

Natomiast antyterroryści wygrywają poprzez:

- eliminację drużyny terrorystów,
- rozbrojenie podłożonej bomby.

Istnieje 7 różnych map, na których można rozgrywać oficjalne mecze rankingowe. Rozgrywki są przeprowadzone na mapach, które w różnym stopniu dają przewagę danej drużynie - niektóre są bardziej przyjazne dla terrorystów, a inne dla antyterrorystów.

Każdy gracz posiada punkty zdrowia, broń, pancerz oraz różne rodzaje granatów. Odpowiedni ekwipunek gracza może w znaczącym stopniu ułatwić mu rozgrywkę podczas rundy.



Rysunek 1: Przykładowy zrzut ekranu rozgrywki CS:GO

3 Opis danych wejściowych

Zbiór danych składa się ze snapshotów rund z około 700 meczów z profesjonalnych turniejów rozgrywanych w 2019 i 2020 roku. Snapshoty, czyli zestawienie pewnych stanów kluczowych elementów rozgrywki, były rejestrowane podczas gry co 20 sekund aż do rozstrzygnięcia danej rundy. Łączna liczba zapisanych snapshotów wynosi 122411. Część tych rekordów będzie traktowana jako zbiór danych uczących, a pozostała część jako część danych testów. Każdy rekord traktowany jest jako pojedynczy, niezależny element do analizy danych.

Zmienna	Definicja	Przykład
time_left	czas pozostały do zakończenia rundy	
ct_score	bieżący wynik drużyny CT	
t_score	bieżący wynik drużyny T	
map	mapa, na której toczy się rozgrywka	np. de_dust2, de_inferno and de_overpass
bomb_planted	czy bomba została podłożona	False = nie, True = tak
ct_health	suma punktów życia drużyny CT	zakres 0-500
t_health	suma punktów życia drużyny T	zakres 0-500
ct_armor	suma punktów pancerza drużyny CT	
t_armor	suma punktów pancerza drużyny T	
ct_money	suma pieniędzy drużyny CT	wartość w USD
t_money	suma pieniędzy drużyny T	wartość w USD
ct_helmets	liczba graczy z hełmem w drużynie CT	
t_helmets	liczba graczy z hełmem w drużynie T	
ct_defuse_kits	liczba graczy z zestawem do rozbrajania	
ct_players_alive	Liczba żywych graczy w drużynie CT	zakres 0-5
t_players_alive	Liczba żywych graczy w drużynie T	zakres 0-5
ct_weapon_X	liczba broni X w drużynie CT	np. Ak47, Deagle i UMP45.
t_weapon_X	liczba broni X w drużynie T	np. Ak47, Deagle i UMP45.
ct_grenade_X	liczba granatów X w drużynie CT	np. HeGrenade, Flashbang.
t_grenade_X	liczba granatów X w drużynie T	np. HeGrenade, Flashbang.
round_winner	Zwycięzca rundy	CT = antyterrorysta, T = terrorysta

4 Podział pracy

Ogólny podział pracy:

1. Wybór niezbędnych bibliotek - Jakub Michalak,
2. Walidacja i wydzielenie danych treningowych - Damian Opoka,
3. Analiza i przygotowanie danych - Damian Opoka,
4. Definicja modelu - Adam Ryl,
5. Weryfikacja krzyżowa - Adam Ryl,
6. Dostosowanie hiperparametrów - Jakub Michalak,
7. Wygenerowanie wyników - Piotr Kryczka,
8. Analiza wyników - Piotr Kryczka.

5 Język programowania

Z uwagi na dużą liczbę przydatnych bibliotek dotyczących uczenia maszynowego, w projekcie zdecydowano się na użycie wysokopoziomowego języka Python w wersji 3.