

Sztuczna inteligencja

Ćwiczenia 4

(tydzień rozpoczynający się 22 maja)

Każde zadanie warte jest 1 punkt, chyba, że w treści napisane jest inaczej.

Zadanie 1. Czym jest *null move heuristic*? Jakie uzasadnienie ma ta heurystyka, jakie wiążą się z nią problemy? (wystarczy artykuł na angielskiej Wikipedii, ale oczywiście mile widziane rozszerzenia).

Zadanie 2. 1-3p, ★ Ciekawe ujęcie problemu Sokoban-a przedstawia praca: https://ieee-cog.org/2020/papers/paper_44.pdf. Zapoznaj się z nią i bądź gotowy opowiedzieć reszcie grupy. Liczbę punktów (którą wpisujesz w deklaracji) tłumaczymy w następujący sposób (można wpisywać wartości niecałkowite, jeżeli uważasz, że lepiej oddają stan faktyczny):

- Przejrzałem dość uważnie całą pracę, rozumiem ogólną ideę, ale być może pewne szczegóły nie są dla mnie w 100% jasne. (1p)
- Przeczytałem uważnie całą pracę, sądzę, że umiałbym zaimplementować agenta FESS (2p)
- Jak wyżej, a dodatkowo rozumiem zasadniczo wszystko z tej pracy i mam pewne pomysły, co do innych zastosowań przedstawionych w pracy pomysłów, ewentualnie do kontynuacji badań z pracy (3p)

Zadanie 3. Gry karciane są dobrym przykładem sytuacji, w której nie tylko występuje element losowy (rozdawanie kart), ale również gracz musi podejmować decyzje w sytuacji, w której nie zna w pełni stanu gry (nie wie, co otrzymali inni gracze). Gdy programuje się takiego agenta, często używanym pomysłem jest wielokrotne losowanie możliwego stanu gry (czyli bieżącego układu kart), znajdowanie najlepszego ruchu w tym stanie (używając standardowych metod rozwiązywania gier z jawnym stanem) i ostateczny wybór ruchu, który był najlepszy w największej liczbie losowań.

Wyjaśnij następujące kwestie:

- a) Co może oznaczać: losowanie **możliwego** stanu?
- b) W jakich sytuacjach losowanie z poprzedniego punktu chcielibyśmy wykonywać przypisując stanom niejednakowe prawdopodobieństwa? Co można w ten sposób uzyskać i jakie to rodzi problemy? Uwaga dla osób mało grających w gry karciane: wiele gier ma element licytacji, w której gracze deklarują (często wielokrotnie) jaki wariant przyszłej rozgrywki wydaje im się odpowiedni.
- c) Jaki istotny aspekt gier karcianych jest pomijany w tym podejściu?

Zadanie 4. Rozważmy grę oszust (zob. `Cheat_(game)` w Wikipedii). Gramy standardową talią 52 kart (asy są najniższymi kartami). Mamy k graczy, którym rozdane są wszystkie karty. Zaczyna rozdający, po czym gracze na zmianę „zagrywają” od 1 do 4 kart (przy czym zagrywają je koszulkami do góry, tak że nie są widoczne dla innych graczy). Celem jest pozbycie się wszystkich kart. Gracz zagrywając kartę (karty) mówi o ich wartości (na przykład mówi: zagrywam dwie siódemki). Można deklarować jedynie karty o równej wielkości, dodatkowo deklaracja powinna być starsza (bądź równa) deklaracji poprzedniego gracza, czyli na „dwie siódemki” można rzucić „trzy dziewiątki”, ale nie odwrotnie¹. Gracze mogą kłamać odnośnie tego, co rzucili. Po każdej zagrywce jest faza sprawdzania, w której kolejni gracze mogą spasować lub sprawdzić zagrywającego (ta faza albo zawiera $k - 1$ pasów, albo pewną liczbę pasów i pierwsze sprawdzenie). Jeżeli zagrywający kłamał, to zabiera wszystkie karty². Jeżeli zagrywający powiedział prawdę, to karty bierze sprawdzający. W obu przypadkach kolejny gracz kontynuuje rozgrywkę (ponieważ stos jest pusty, może wyrzucić karty o dowolnej wysokości, oczywiście wyrzucając karty może skłamać o ich wartości).

Zaproponuj dwóch przykładowych prostych agentów, grających w tę grę (dozwolone są tylko takie idee, co do których masz przekonanie, że dadzą lepszą grę od agenta w pełni losowego).

Zadanie 5. Potestuj playground.tensorflow.org. Odpowiedz na pytania:

- a) Dla jakich zbiorów danych i jakich cech wystarcza 1 neuron do poprawnej klasyfikacji? (i dlaczego)

¹W starszeństwie nie ma znaczenia, ile kart rzucamy, jedynie to, jakie są ich wysokości

²Zasady gry tego nie prezycują, ale możemy przyjąć, że gracz zabiera karty zachowując ich kolejność i jeżeli ma odpowiednio dobrą pamięć, to może po fakcie sprawdzić, kto kłamał, a kto mówił prawdę w podczas budowy tego stosu.

- b) Co dzieje się, gdy dla bardziej złożonych sieci damy zbyt duży *Learning rate*?
- c) W którym zadaniu przydają się cechy \sin i \cos ?
- d) Dla każdego zbioru danych (oprócz spirali) powiedz, jaka najprostsza³ sieć neuronowa korzystająca tylko z cech x_1 i x_2 poprawnie klasyfikuje ten zbiór danych.

Zadanie 6. Rozważamy sieć neuronową z prostą funkcją schodkową w roli σ (równą 1 dla liczb dodatnich, 0 w przeciwnym przypadku). Wejściami do tej sieci będą zera i jedynki, zatem sieć będzie obliczała jakąś funkcję boolowską.

- a) Podaj sieci neuronowe (złożone z jednego neurona) obliczające $x \vee y$, $x \wedge y$, $\neg x$.
- b) Podaj sieć neuronową dla $x \text{ xor } y$
- c) Uzasadnij, że nie jest możliwa sieć z punktu b), która ma tylko 1 neuron

Zadanie 7. Rozważamy takie same sieci, jak w poprzednim zadaniu. Czy za pomocą sieci neuronowych można wyrazić dowolną funkcję boolowską? Jaka jest minimalna liczba warstw, która wystarcza (zakładamy, że neurony mogą mieć dowolną liczbę wejść).

Zadanie 8. Pokaż, że jeżeli graf MDP jest acykliczny, to (również dla $\gamma = 1$) algorytm *Value Iteration* (Bellmana) znajduje optymalną politykę. Pokaż, jak znaleźć tę politykę w krótszym czasie.

Zadanie 9. ★ Co to jest nadracjonalność (superrationality)? (znajdź odpowiednie informacje w Internecie)

Zadanie 10. ★ Podobnie jak w poprzednim zadaniu, powinieneś posilkować się samodzielnie znalezionymi informacjami. Wyjaśnij, co to jest punkt równowagi Nasha. Opowiedz, na czym polega gra w Dylemat więźnia i jaki jest dla niej punkt równowagi Nasha. Jak twórca agenta grającego w tę grę mógłby wykorzystać to, że prawdziwe są następujące fakty:

- i) Jest wielu graczy, każdy gra w tę grę wiele razy, w różnych parach.
- ii) Każdy gracz przedstawia się przed rozgrywką swoim unikalnym identyfikatorem
- iii) Liczy się sumaryczny wynik wielu rozgrywek.

³Mająca się najmniej warstw i (w drugiej kolejności) najmniej neuronów.