

Wprowadzenie

Wprowadzenie do sztucznej inteligencji - Wykład 1

Maciek Gębala

8 października 2021

Maciek Gębala Wprowadzenie

Literatura

- Stuart Russell, Peter Norvig,
Artificial Intelligence: A Modern Approach,
Pearson Education Limited 2022, (ISBN: 1-292-40113-3)

Maciek Gębala Wprowadzenie

Czym jest sztuczna inteligencja?

Sztuczna inteligencja to automatyzacja zdolności przypisanych
ludzkemu myśleniu, zdolności takich jak podejmowanie decyzji,
rozwiązywanie problemów, uczenie się ... [Bellman, 1978]

Sztuczna inteligencja to badania prowadzone w kierunku stworzenia
komputerów, które myślą ... maszyn posiadających umysł.
[Haugeland, 1985]

Sztuczna inteligencja to badanie zdolności umysłowych za pomocą
modeli obliczeniowych. [Charniak & McDermontt, 1985]

Sztuczna inteligencja to sztuka tworzenia maszyn zdolnych do
wykonywania działań, wymagających od człowieka zaangażowania
inteligencji. [Kurzweil, 1990]

Maciek Gębala Wprowadzenie

Czym jest sztuczna inteligencja?

Sztuczna inteligencja to badania mające na celu stworzenie
komputerów posiadających umiejętności, w których człowiek jest
obecnie lepszy. [Rich & Knight, 1991]

Sztuczna inteligencja to badanie mające na celu opis i symulację
inteligentnego zachowania w kategoriach procesów obliczeniowych.
[Schalkoff, 1990]

Sztuczna inteligencja to studia nad modelami obliczeniowymi, które
umożliwiają percepcję, wnioskowanie i działanie. [Winston, 1992]

Sztuczna inteligencja jest dziedziną informatyki, zajmującą się
automatyzacją inteligentnego zachowania. [Luger & Stubblefield,
1993]

Maciek Gębala Wprowadzenie

Notatki

Notatki

Notatki

Notatki

Cztery kategorie definicji sztucznej inteligencji

Ludzkie myślenie

[Kurzweil, 1990] [Rich & Knight, 1991]

Myślenie racjonalne

[Charniak & McDermott, 1985] [Winston, 1992]

Ludzkie zachowanie

[Bellman, 1978] [Haugeland, 1985]

Racjonalne zachowanie

[Schalkoff, 1990] [Luger & Stubblefield, 1993]

Idealistyczna koncepcja racjonalności

System zachowuje się racjonalnie, jeśli wykonuje właściwie przypisane czynności, w sposób gwarantujący osiągnięcie celu, przy przyjętych wstępnie założeniach.

Maciek Gębala

Wprowadzenie

Notatki

Działać jak człowiek: Test Turinga

Charakterystyka

- Badanie porównawcze w kontekście pewnego zbioru pytań [standard odwołujący się do pewnego wzorca].
- Ignorowanie wewnętrznych procesów maszyny [nie rozstrzygamy świadomości podejmowanych decyzji].
- Ograniczenie kontaktu do form zdalnych [eliminacja preferowania organizmów żywych].
- Ograniczenie badań zachowania jedynie do zadań o charakterze symbolicznym [nie badamy postrzegania zmysłowego czy umiejętności manualnych].
- Ograniczenie pojęcia inteligencji jedynie do ludzi [ocenia człowiek na swoje podobieństwo].

Maciek Gębala

Wprowadzenie

Notatki

Test Turinga

Mechanizmy niezbędne w teście:

- zdolność przetwarzania języka naturalnego,
- zdolność prezentowania wiedzy,
- zdolność automatycznego wnioskowania,
- zdolność uczenia.

Dodatkowe mechanizmy niezbędne w pełnym teście:

- zdolność rozpoznawania obrazów,
- zdolności manualne (poruszanie się i przemieszczanie obiektów).

Maciek Gębala

Wprowadzenie

Notatki

Myśleć jak człowiek: kognitywistyka

Opis i modelowanie ludzkiego sposobu myślenia, jego procesu poznania i inteligentnego zachowania.

Metody badań

- Introspekcja - obserwacja i analiza własnych subiektywnych stanów psychicznych, myśli i przymysłów.
- Eksperymenty psychologiczne.
- Eksperymenty neurologiczne.

Maciek Gębala

Wprowadzenie

Notatki

Myślenie racjonalne: logika

Języki logiki formalnej

Precyzyjna notacja i sposób wyrażania obiektów i związków między nimi.

Zalety

Zgodnie z twierdzeniem G odla możliwe jest zbudowanie formuły która opisuje problem i znajduje jego rozwiązanie (o ile istnieje) [szukanie rozwiązania może się nie skończyć].

Wady

Trudność wyrażania dużej części wiedzy w języku logiki, szczególnie niepełnej czy nieprecyzyjnej. Cząso- i pamięcio-łłonnoř algorytmów rozwiązujących problemy logiki.

Maciek Gbala

Wprowadzenie

Racjonalne zachowanie/działanie

Racjonalne działanie to metoda prowadząca właściwą drogą do osiągnięcia celu, poprawne działanie przy przyjętych założeniach i postawionym celu.

Racjonalne myślenie prowadzi do racjonalnego zachowania, ale nie jest konieczne.

Maciek Gbala

Wprowadzenie

Racjonalny agent

Z uwagi na skomplikowanie często tworzymy system skupiony tylko na ograniczonych zadaniach (określona gra, rozpoznawanie określonych klas rzeczy, ...) i o ograniczonych zasobach, stąd pojawia się pojęcie racjonalnego agenta - rozwiązującego tylko jeden problem.

Maciek Gbala

Wprowadzenie

Najważniejsze obszary zastosowań

- Gry (strategie wygrywające w grach)
- Automatyczne wnioskowanie i dowodzenie twierdzeń
- Rozpoznawanie obrazów
- Systemy eksperckie
- Robotyka (planowanie działań)
- Automatyczne (maszynowe) uczenie się
- Przetwarzanie języków naturalnych/automatyczne tłumaczenia

Maciek Gbala

Wprowadzenie

Notatki

Notatki

Notatki

Notatki

- Reprezentacja wiedzy: przestrzeń stanów gry
- Prosta struktura formalna problemów, proste zasady działania
- Metody rozwiązywania: przeszukiwanie grafów przestrzeni stanów (A^* , alfa-beta-cięcia, więzy), przeszukiwanie baz danych (znane rozgrywki)
- Problemy z dobrymi funkcjami/heurystykami wyznaczającymi drogę do celu
- Historycznie najstarsze problemy sztucznej inteligencji w odniesieniu do szachów czy warcabów

Przedmioty na których pojawiają się te zagadnienia

I stopień: Programowanie w logice

II stopień: Technologia więzów

Automatyczne wnioskowanie i dowodzenie twierdzeń

- Reprezentacja wiedzy: rachunek predykatów, rachunek zdań, formuły logiczne
- Metody wnioskowania: reguła odrywania (modus ponens), unifikacja (uzgadnianie), rezolucja (strategie rezolucyjne)
- Zastosowanie: projektowanie i weryfikacja układów cyfrowych, weryfikacja poprawności programów komputerowych, sterowanie złożonymi systemami

Przedmioty na których pojawiają się te zagadnienia

I stopień: Programowanie w Logice, Niezawodne Systemy Informatyczne

II stopień: Automatyczna weryfikacja

Rozpoznawanie obrazów

- Reprezentacja wiedzy: rastrowa (dyskretyzacja)
- Metody: wygładzanie, segmentacja, triangulacja, ...
- Zastosowania: automatyczne kierowanie robotami, diagnostyka medyczna, rozpoznawanie produktów, ...

Systemy eksperckie

- Reprezentacja wiedzy: reguły, dane z atrybutami
- Metody: pytania-odpowiedzi (drzewa decyzyjne), dopasowywanie wzorców, ...
- Zastosowanie: nauka, medycyna, wojskowość

Robotyka - planowanie działań

- Reprezentacja wiedzy: różna
- Metody: podejmowanie często decyzji na podstawie niepełnych danych (algorytmy on-line)
- Zastosowania: rozpoznawanie terenu, planowanie działań (przydział zasobów), ...
- Problemy: niepełność danych, niepewność danych. konflikty z innymi ruchomymi obiektami

Przedmioty na których pojawiają się te zagadnienia

I stopień: Algorytmy optymalizacji dyskretniej

II stopień: Algorytmy on-line, Metody Optymalizacji

Wprowadzenie

Uczenie automatyczne (maszynowe)

- Reprezentacje wiedzy: drzewa decyzyjne, reguły decyzyjne, sieci neuronowe
- Metody: uczenie bez nadzoru (automatyczny podział danych na klasy), uczenie z nadzorem, uczenie ze wzmocnieniem
- Zastosowanie: systemy oparte na bazach danych, eksploracja i odkrywanie danych w dużych strumieniach
- Podział ze względu na sposób wykorzystania wiedzy: klasyfikacja, aproksymacja, podejmowanie decyzji

Przedmioty na których pojawiają się te i podobne zagadnienia

II stopień: Data Mining, Big Data

Wprowadzenie

Przetwarzanie języków naturalnych

- Reprezentacja wiedzy: sieci semantyczne, korpusy językowe
- Metody: przeszukiwanie, skojarzenia (dopasowania)
- Zastosowania: rozpoznawanie mowy (częściowo, korekta błędów), translacja

Wprowadzenie

Notatki

[illegible]

Notatki

[illegible]

Notatki

[illegible]

Notatki

[illegible]

Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Wprowadzenie do sztucznej inteligencji - Wykład 2

Maciek Gębala

15 października 2021

Maciek Gębala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Przestrzeń stanów

Przestrzeń stanów to uporządkowana czwórka (V, E, S, F) , gdzie

- V jest zbiorem wierzchołków reprezentujących stany powstałe w trakcie rozwiązywania problemów;
- E jest zbiorem krawędzi odpowiadającym przejściom między stanami zgodnymi z regułami rozwiązywanego problemu ($E \subseteq V \times V$);
- S jest niepustym podzbiorem V zawierającym stany początkowe problemu;
- F jest niepustym podzbiorem V zawierającym stany docelowe problemu (mogą być zdefiniowane wprost lub przez własności które chcemy osiągnąć).

Rozwiązanie

Rozwiązaniem będziemy nazywać ścieżkę prowadzącą przez przestrzeń stanów od stanu początkowego do stanu docelowego.

Maciek Gębala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Podstawowe problemy przy przeszukiwaniu przestrzeni stanów

- Czy algorytm przeszukujący gwarantuje znalezienie rozwiązania?
- Czy algorytm zawsze się kończy (nie zapętla się)? Jak wykryć zapętlenie?
- Czy algorytm znajduje rozwiązanie optymalne?
- Jaka jest złożoność pamięciowa i czasowa algorytmu? Czy jest do zaakceptowania?
- Jak można poprawić złożoność pamięciową lub czasową?

Maciek Gębala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Sformułowanie zadania dla algorytmów przeszukiwania

Przestrzeń stanów

- Precyzyjna definicja przestrzeni stanów.
- Określenie stanu początkowego.
- Określenie reguł przejść między stanami (operatory akcji/funkcja następnika).
- Zbiór stanów docelowych lub funkcja weryfikacji osiągnięcia celu.
- Funkcja kosztu ścieżki (liczba akcji/koszt akcji).

Kierunki przeszukiwania

- W przód (od stanu początkowego do celu).
- W tył (od celu do stanu początkowego).
- Przeszukiwanie dwukierunkowe.

Maciek Gębala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Notatki

Notatki

Notatki

Notatki

Przeszukiwanie w przód

Kiedy stosować

- Znamy dokładnie tylko początkowe stany.
- Jest dużo dopuszczalnych stanów docelowych.
- Cel jest określony przez własności i nie znamy jego postaci.

Przeszukiwanie w tył

Kiedy stosować

- Stan docelowy jest jeden i jest łatwy do zdefiniowania.
- Liczba reguł przejść w przód jest duża i odwrócenie kierunku może zmniejszyć rozgałęzienia przeszukiwań (zmniejszyć liczbę odwiedzanych rozgałęzień).
- Kiedy dane o problemie nie są znane dokładnie i znamy dokładnie cel (reverse engineering, diagnostyka medyczna).

Przeszukiwanie dwukierunkowe

Stosuje się głównie dla ograniczenia złożoności czasowej. Szukamy wspólnego stanu osiągalnego od stanu początkowego i stanu końcowego (wstecz).

Problemem jest zapewnienie, że ścieżki się spotkają a nie miną.

Strategie przeszukiwania

Przeszukiwanie w głąb

Zalety: łatwość implementacji, małe wymagania pamięciowe.
Wady: możliwość zapętlenia, znalezione rozwiązanie nie musi być optymalne.

Przeszukiwanie wszerz

Zalety: znalezione rozwiązanie jest optymalne ze względu na najmniejszą liczbę kroków, brak zapętleń.
Wady: duża złożoność pamięciowa.

Best-First-Search

Rozszerzenie przeszukiwania wszerz po dodaniu funkcji kosztu przejścia (krawędzi) - np. algorytm Dijkstry.

Notatki

Notatki

Notatki

Notatki

Przykład 1 - Mapa połączeń drogowych

Problemy

- Przeszukiwanie w głąb może dać zapętlenie.
- Algorytm Dijkstry szukając najkrótszej drogi z Wrocławia do Warszawy (355km) najpierw znajdziemy Pragę (322) czy Berlin (341).
- Algorytm Dijkstry musi pamiętać wszystkie wierzchołki grafu połączeń które odwiedził aby uniknąć zapętleń.

Algorytm Dijkstry ma swoją metodologię - szybkość znalezienia rozwiązania można próbować zmienić przez zastosowanie odpowiedniej heurystyki. Jakiej?

Możliwe rozwiązanie

Kierować się zawsze w stronę Warszawy (starać się nie zwiększać w trakcie szukania odległości w linii prostej) - funkcja oceny heurystycznej.

Maciek Gbala

Przeszukiwanie przestrzeni stanów - problemy jednoosobowe

Notatki

Przeszukiwanie heurystyczne

Rola heurystyki w przeszukiwaniu

Pozwala w sposób naturalny wykorzystać informacje niepewne lub nieprecyzyjne do szacowania kosztów i dzięki temu poprawiać efektywność przez wskazywanie najlepszych kierunków przeszukiwań lub pośrednio przez eliminację najmniej obiecujących kierunków.

Gdzie się je najczęściej stosuje

- Problemy jednoosobowe (np. łamigłówki).
- Problemy optymalizacyjne (szukanie najkrótszej drogi, planowanie ruchów robota, szeregowanie zadań).
- Gry dwuosobowe.
- Systemy dowodzenia twierdzeń.

Maciek Gbala

Przeszukiwanie przestrzeni stanów - problemy jednoosobowe

Notatki

Funkcja oceny heurystycznej

Funkcją oceny heurystycznej nazywamy funkcję kosztów określoną na stanach postaci:

$$f(v) = g(v) + h(v).$$

gdzie $g(v)$ jest aktualną odległością (kosztem) od stanu początkowego do stanu v , a $h(v)$ jest heurystycznym oszacowaniem odległości (kosztu) od stanu v do celu.

Przykład 1 - cd

Dla połączeń drogowych jako funkcję h możemy przyjąć odległość w linii prostej między aktualną pozycją (stanem) a celem.

Best-First-Search a funkcja oceny heurystycznej

Jeżeli z algorytmie Best-First-Search zastosujemy w funkcji oceny heurystycznej tylko drugi składnik ($g(v) = 0$ dla każdego stanu) to otrzymamy algorytm przeszukiwania zachłannego (który jednak nie gwarantuje znalezienia optymalnego rozwiązania).

Maciek Gbala

Przeszukiwanie przestrzeni stanów - problemy jednoosobowe

Notatki

Algorytm A - definicja

Weźmy funkcję oceny heurystycznej

$$f(v) = g(v) + h(v)$$

gdzie dla każdego stanu v

$g(v) \geq 0$ jest kosztem osiągnięcia stanu v ze stanu początkowego,

$h(v) \geq 0$ jest heurystycznym oszacowaniem kosztu przejścia ze stanu v do stanu końcowego.

Zastosowanie algorytmu Best-First-Search z tak zdefiniowaną funkcją oceny heurystycznej nazywamy algorytmem A.

Maciek Gbala

Przeszukiwanie przestrzeni stanów - problemy jednoosobowe

Notatki

Przykład 2 - Problem hetmanów

Przykład dla 4 hetmanów wykonany na tablicy.



Stan początkowy

Wszyscy hetmani są w jednym (górnym wierszu).

Przejście między stanami

Przesunięcie któregoś hetmana na jedno z czterech sąsiednich pól (jeśli istnieje i nie jest zajęte).

Funkcja oceny heurystycznej

$g(v)$ definiujemy jako liczbę przejść.

$h(v)$ definiujemy jako liczbę par które się biją wzajemnie (przy ocenie ignorujemy pozostałe hetmany).

Maciek Gębala

Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Przykład 2 - Problem hetmanów

Cel

Nie znamy postaci celu, ale wiemy że funkcja h dla celu ma wartość 0.

Ponieważ bardziej szukamy rozwiązania niż liczby ruchów możemy też zastosować algorytm zachłanny (nie liczyć ruchów, czyli zawsze $g(v) = 0$).

Modyfikacja startu

Dowolne rozstawienie hetmanów.

Rozwiązania

Istnieją tylko dwa rozwiązania dla 4 hetmanów (symetryczne).

Maciek Gębala

Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Własności heurystyk

Dopuszczalność heurystyki (admissibility)

Czy algorytm znajduje rozwiązanie o najniższym koszcie?

Monotoniczność heurystyki (monotonicity)

Czy każdy stan jest osiągalny przy najmniejszym koszcie?

Informacyjność heurystyki (informedness)

Czy można szybciej znaleźć rozwiązanie przy pomocy innej (lepszey) heurystyki?

Maciek Gębala

Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Idealna funkcja oceny heurystycznej

Przyjmijmy następujące oznaczenia:

$g^*(v)$ - optymalny koszt ścieżki ze stanu początkowego do v ,

$h^*(v)$ - rzeczywisty optymalny koszt z v do celu,

$f^*(v)$ - rzeczywisty optymalny koszt ze stanu początkowego do celu przechodzącej przez stan v
($f^*(v) = g^*(v) + h^*(v)$)

W trakcie przeszukiwania powinno być $g(v) \geq g^*(v)$, dopóki nie odkryjemy najkrótszej ścieżki do v - wówczas zajdzie równość.

Przed zakończeniem przeszukiwania nie można podać $h^*(v)$ więc stosujemy heurystykę $h(v)$ - jej oszacowanie.

Maciek Gębala

Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Notatki

Notatki

Notatki

Notatki

Algorytm A^*

Dopuszczalność algorytmu

Algorytm przeszukiwania jest dopuszczalny jeżeli zawsze znajduje optymalną ścieżkę do rozwiązania (o ile ona istnieje).

Przykład

Algorytm A^*

Jeżeli algorytm A wykorzystuje funkcję oceny heurystycznej taką, że dla każdego v zachodzi $h(v) \leq h^*(v)$, to otrzymany algorytm nazywamy algorytmem A^* .

Maciek Gebala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Algorytm A^*

Algorytm A^* jest dopuszczalny.

Dowód (Szkic)

Niedoszacowanie funkcji h powoduje, że zaniżamy rzeczywisty koszt ścieżki, czyli nie może zająć sytuacja, że ominiemy optymalny wierzchołek w drodze do celu. Gdyby oszacowanie było zawyżone to zanim rozpatrzylibyśmy ten wierzchołek przez inny moglibyśmy dotrzeć do celu.

Maciek Gebala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Monotoniczność

Funkcja oceny heurystycznej f jest monotoniczna gdy

- jeśli v' jest następnikiem v na ścieżce, to $h(v) - h(v') \leq \text{koszt}(v, v')$, gdzie $\text{koszt}(v, v')$ jest rzeczywistym kosztem przejścia ze stanu v do stanu v' , oraz
- oszacowanie heurystyczne stanu docelowego jest równe 0.

Monotoniczność to lokalna dopuszczalność - każdy stan, a nie tylko cel, jest osiągalny z optymalnym kosztem.

Zadanie na ćwiczenie

Pokazać, że jeśli heurystyka jest monotoniczna, to funkcja oceny heurystycznej dla każdego stanu nigdy nie maleje podczas przeszukiwania.

Wniosek

Heurystyka monotoniczna jest heurystyką dopuszczalną.

Maciek Gebala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Informacyjność

Dla dwóch dopuszczalnych heurystyk h_1 i h_2 , jeżeli dla dowolnego stanu v zachodzi $h_1(v) \leq h_2(v)$, to mówimy, że h_2 zawiera więcej informacji niż h_1 (jest lepiej poinformowana). Jeżeli h_2 jest lepiej poinformowana niż h_1 , to zbiór stanów odwiedzonych przez algorytm A^* z heurystyką h_2 jest podzbiorem stanów odwiedzonych z heurystyką h_1 .

Jeśli mamy kilka heurystyk powinniśmy wybrać najlepiej poinformowaną.

Maciek Gebala Przeszukiwanie przestrzeni stanów - problemy jednosobowe

Notatki

Notatki

Notatki

Notatki

Więzy

Notatki

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Wprowadzenie do sztucznej inteligencji - Wykład 3

Maciek Gębala

22 października 2021

Maciek Gębala Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Przypomnienie - heurystyczne przeszukiwanie

Role heurystyki w przeszukiwaniu

- Połączenie wiedzy o naturze rozwiązywanego problemu z metodami wyszukiwania w celu przyspieszenia znalezienia rozwiązania.
- Ograniczenie kombinatorycznej eksplozji stanów w drzewie przeszukiwań.
- Nie zawsze gwarantuje znalezienie optymalnego rozwiązania, chyba że spełnia określone własności.
- Pozwala wykorzystywać wiedzę nieformalną i niepewną.

Maciek Gębala Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Gry dwuosobowe

Cel

Rozpatrujemy gry nie kooperacyjne: szukamy strategii wygrywającej/maksymalizującej szanse wygranej dla gracza pierwszego (drugiego).

Gry kooperacyjne

Istnieją gry w której możliwa jest współpraca graczy w celu osiągnięcia wzajemnych korzyści - wykład wybieralny na drugim stopniu: *Algorytmiczna Teoria Gier*.

Algorytmy przeszukiwania heurystycznego dla gier dwuosobowych

- Algorytm Min-Max
- Algorytm α - β

Maciek Gębala Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Gry dwuosobowe

Dostępna wiedza

- gry z pełną informacją (mamy pełną wiedzę o sytuacji i celach przeciwnika)
- gry z niepełną informacją (braki w wiedzy o przeciwniku, losowe elementy gry)

Kryterium zwycięstwa: sumy wypłat

- gry o sumie zerowej (zyski i straty graczy bilansują się)
- gry o sumie niezerowej (wygrane i przegrane nie muszą się bilansować)

Maciek Gębala Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Notatki

Notatki

Notatki

Notatki

Ogólne zasady gry - z pełną informacją i o sumie zerowej

Mamy dwóch graczy: patrzymy na grę z punktu widzenia pierwszego MAX, który maksymalizuje swój rezultat końcowy, czyli stratę drugiego gracza. Analogicznie drugi gracz MIN minimalizuje wynik pierwszego gracza, czyli maksymalizuje swój zysk.

Co szukamy?


- Strategię wygrywającą pierwszego (drugiego) gracza - czy istnieją takie reguły ruchów, że gracz zawsze wygrywa. (Ogólnie problem jest PSPACE-zupełny.)
- Najlepszy kolejny ruch gracza, przy założeniu tylko ograniczonego przeszukiwania (ograniczony poziom zagłębienia lub ograniczona liczba rozpatrywanych stanów).

Przykład strategii wygrywającej

Gra (uproszczone NIM)

Pierwszy gracz mówi 100. W kolejnych ruchach gracze odejmują od aktualnej liczby od 1 do 10. Wygrywa ten, który powie 1.

Drzewo gry i strategia wygrywająca

Przykład na tablicy 

Strategia wyboru kolejnego ruchu

Heurystyczna ocena stanu gry

Funkcja pozwalająca oszacować szanse na wygraną gracza MAX w danym stanie gry.

Problem

Jakie kryteria zastosować do takiej oceny?

Algorytm Min-Max

Gracz MAX (MIN) rozpatruje wszystkie możliwości kolejnych k ruchów wybierając ten który prowadzi go do maksymalnej (minimalnej) wartości funkcji heurystycznej po tych ruchach. (Można też zamiast k ruchów rozpatrywać kryterium ograniczonej liczby stanów kierując się maksymalizacją (minimalizacją) wartości funkcji heurystycznej.) Stosujemy algorytm przeszukiwania w głąb i funkcję drzewa zgodną z zasadą min-max przekazywaną z dołu do góry.

Przykład - gra w kółko i krzyżyk

W grze nie ma strategii wygrywającej - jeśli obaj gracze grają poprawnie kończy się remisem.

Funkcja oceny heurystycznej

Różnica liczby możliwych układów wygrywających (możliwych jeszcze trzelementowych linii) gracza X (MAX) i gracza O (MIN).

Przykład rozgrywki na tablicy (dla głębokości 2 ruchów i z uwzględnieniem symetrii)



Notatki

[illegible]

Notatki

[illegible]

Notatki

[illegible]

Notatki

[illegible]

Poprawienie algorytmu: α - β -cięcia

Ograniczenie dolne - α

Dolne ograniczenie dla wierzchołków MAX (najwyższa wartość jaką dotychczas osiągnął MAX)

Ograniczenie górne - β

Górne ograniczenie dla wierzchołków MIN (najniższa wartość jaką dotychczas osiągnął MIN)

Cięcia

Cięcie α jest wykonywane w wierzchołku MIN a cięcie β w wierzchołku MAX

Wykonanie cięcia

Jeśli zachodzi warunek $\alpha \leq \beta$, to nie ma potrzeby analizowania następników danego stanu.

Maciek Gębala

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Przykład działania

Rysunek i analiza na tablicy.



Maciek Gębala

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Słabości algorytmu Min-Max (α - β)

Efekt horyzontu

Spadek wartości stanu może nastąpić tuż za wyznaczoną głębokością przeszukiwania.

Kolejność przeszukiwania a efektywność cięć

Kolejność wyboru następników rozpatrywanego stanu znacząco wpływa na liczbę pojawiających się cięć.

Maciek Gębala

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Rozszerzenia algorytmu

Doskonalenie funkcji oceny stanu (funkcji heurystycznej)

Modyfikacje sposobu przeszukiwania

- zastosowanie większej pamięci (połączenie przeszukiwania w głąb z przeszukiwaniem wszerek w celu wyboru potencjalnie korzystniejszych ścieżek)
- porządkowanie następników według funkcji oceny
- zmienna głębokość przeszukiwań

Urównoleglenie obliczeń - łatwe dla przeszukiwania drzew w głąb.

Maciek Gębala

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Notatki

Notatki

Notatki

Notatki



Notatki

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Notatki

Przeszukiwanie przestrzeni stanów - gry dwuosobowe

Notatki

Wstęp do uczenia maszynowego

Wprowadzenie do sztucznej inteligencji - Wykład 4

Maciek Gębala

29 października 2021

Maciek Gębala Wstęp do uczenia maszynowego

Co to jest uczenie maszynowe?

Uczenie dobierania wzorców najlepiej pasujących do danych bez pisania odpowiedniego algorytmu który by wykonał to zadanie.

Regresja - dla danych szukamy liczby rzeczywistej.

Klasyfikacja - dla danych szukamy kategorii do której należy.

Maciek Gębala Wstęp do uczenia maszynowego

Uczenie maszynowe

Uczenie z nadzorem (Supervised Learning)

- regresja liniowa
- klasyfikacja (regresja logistyczna)
- sieci neuronowe

Uczenie bez nadzoru (Unsupervised Learning)

- klasteryzacja
- redukcja wymiaru
- detekcja anomalii
- systemy rekomendacji

Maciek Gębala Wstęp do uczenia maszynowego

Uczenie z nadzorem

Mając zbiór przykładów X i etykiety Y staramy się stworzyć funkcję zwracającą odpowiednie wartości dla nowych przykładów.

Model

Wyrażenie które bierze przykłady i zwraca etykiety.

Formalnie szukamy funkcji h takiej, że $h(x; \theta) = y$, czyli staramy się znaleźć (nauczyć) parametry θ takie aby predykcja etykiet była jak najlepsza.

Maciek Gębala Wstęp do uczenia maszynowego

Notatki

Notatki

Notatki

Notatki

Funkcja kosztu

Funkcja która oblicza koszt pomyłek naszego modelu. Jeśli \hat{Y} jest ciągiem wartości funkcji h na ciągu przykładów X , to przykładową funkcją kosztu może być

$$L(Y, \hat{Y}) = \sum_i (y_i - \hat{y}_i)^2$$

$$L(Y, \hat{Y}) = \sum_i (y_i - \hat{y}_i)^2$$

Celem jest minimalizacja funkcji kosztu.

Notatki

Przykład regresji liniowej w pigułce

x_i	1	2	3	4	-1
y_i	2,01	3,98	6,10	?	?

x_i	1	2	3	4	-1
y_i	2,01	3,98	6,10	?	?

Jakie równanie opisuje te dane?

Jeśli przyjmujemy możliwość niewielkiego błędu to łatwo zauważyć, że $y_i \approx 2x_i$ (aproksymacja funkcją liniową).

Uczenie maszynowe możemy traktować jako znajdowanie wzorców w danych.

[illegible]

Uogólnienie

Ogólnie dla n -wymiarowych funkcji liniowych moglibyśmy problem sformułować następująco: Dla m podanych punktów znaleźć współczynniki β_1, \dots, β_n takie, że $r_i = \beta_1 x_{i1} + \dots + \beta_n x_{in}$ i $S = \sum_{i=1}^m (y_i - r_i)^2$ jest jak najmniejsze (błąd średniokwadratowy).

Notatki

Regressja liniowa

Minimalizację S moglibyśmy zrobić np. przez różniczkowanie cząstkowe po β_j

$$\frac{\delta S}{\delta \beta_j} = 2 \sum_{i=1}^m (y_i - r_i) \frac{\delta (y_i - r_i)}{\delta \beta_j} = 0$$

Stąd dostajemy równanie

$$2 \sum_{i=1}^m (y_i - \sum_{k=1}^n \beta_k x_{ik})(-x_{ij}) = 0$$

Co po uporządkowaniu daje dla każdego j

$$\sum_{i=1}^m \sum_{k=1}^n x_{ij} x_{ik} \beta_k = \sum_{i=1}^m y_i$$

Po uwzględnieniu czynnika stałego otrzymamy model zwany regresją liniową.

Notatki

Od regresji do klasyfikacji

Na wartość funkcji nakładamy próg, który mówi czy w zależności od wartości etykiety akceptujemy ją (1) czy nie (0) (nałożenie funkcji progowej).

Funkcja progowa nie jest najlepsza, gdyż nie jest różniczkowalna.

Regresja logistyczna

Wprowadzamy model z funkcją sigmoidalną $g(x) = \frac{1}{1+e^{-x}}$ postaci

$$h(x; \theta) = g(\theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n)$$

Funkcja sigmoidalna przyjmuje wartości z przedziału (0, 1) i można je traktować także jako prawdopodobieństwa.

Podobny model jest wykorzystywany w sieciach neuronowych.

Maciek Gębala

Wstęp do uczenia maszynowego

Notatki

Minimalizacja funkcji kosztu

Jak rozwiązuje się problem minimalizacji funkcji kosztu?

- Dla funkcji różniczkowalnej stosujemy metodę spadku gradientu.
- Idea jest start w pewnym punkcie i poruszanie się w kierunku największego spadku (gradient).
- Nie $F(x)$ będzie funkcją m zmiennych i będzie różniczkowalna w sąsiedztwie punktu x_0 , wtedy $F(x_0)$ maleje najszybciej jeśli idziemy z punktu x_0 w kierunku ujemnego gradientu F , czyli $-\nabla F(x_0)$ (jeśli $h: R^m \rightarrow R$ to $\nabla h = [\frac{\partial h}{\partial x_1}, \dots, \frac{\partial h}{\partial x_m}]$).
- Dostajemy ciąg $x_{n+1} = x_n - \eta_n \nabla F(x_n)$, gdzie η_n to współczynnik kroku (współczynnik uczenia).
- Ponieważ metoda spadku gradientu może mieć wolną zbieżność, wprowadza się różne modyfikacje.

Maciek Gębala

Wstęp do uczenia maszynowego

Notatki

Sieci neuronowe

Naśladowanie ludzkiego układu nerwowego. Sieć złożona z połączonych neuronów.

Zastosowanie

Głównie do czynności niesprawiających człowiekowi problemów a trudnych do opisanie algorytmicznie, np. rozpoznawanie obrazów.

Właściwości (mające odzwierciedlenie w biologicznym układzie nerwowym)

- Zdolność do generalizacji (uogólniania danych).
- Interpolacja i predykcja.
- Mała wrażliwość na błędy (szumy) w zbiorze danych.
- Zdolność do efektywnej pracy nawet przy częściowym uszkodzeniu sieci.
- Przetwarzanie równoległe i rozproszone (szybkość).

Maciek Gębala

Wstęp do uczenia maszynowego

Notatki

Podstawowe kierunki zastosowań sieci neuronowych

Predykcja

Nauczanie sieci na pewnym zbiorze danych w celu podawania prognoz dla nowo dostarczonych danych. Sieć nabiera zdolności przewidywania tylko na podstawie dostarczonych danych, nie podaje się żadnych wzorów czy teorii wiążących dane z wynikami.

Klasyfikacja i rozpoznawanie

Sieć na podstawie danych do uczenia rozpoznaje określone klasy danych i klasyfikuje nowe dane według nauczonych wcześniej kategorii.

Analiza danych

Znalezienie związków występujących w wejściowym zbiorze danych.

Maciek Gębala

Wstęp do uczenia maszynowego

Notatki

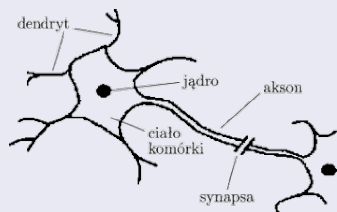
Filtracja sygnałów

Ze względu na odporność na szumy sieci nadają się do filtrowania sygnałów.

Optymalizacje

Nadają się do szukania rozwiązań optymalnych w wielu klasach problemów optymalizacyjnych (również kombinatorycznych).

Inspiracja biologiczna

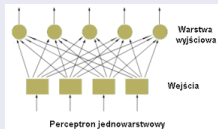


Poziom złożoności

Mózg człowieka ma około 10^{10} komórek nerwowych a liczba połączeń między nimi sięga 10^{15} . Szybkość przetwarzania jest oceniana na 10^{18} operacji na sekundę.

Sieci jednokierunkowe (perceptrony)

Perceptron prosty (jednowarstwowy)



Sieci wielowarstwowe



Perceptrony

- Warstwa wejściowa - nie dokonuje żadnych obliczeń, dostarcza dane wejściowe.
- Przed warstwą wyjściową możemy mieć warstwy pośrednie (ukryte, nie mają bezpośrednich połączeń ze światem zewnętrznym).
- Dla sieci jednokierunkowych nie ma połączeń w kierunku warstw poprzednich i między jednostkami tej samej warstwy.

Zacniemy od analizy jednowarstwowej sieci prostej o n wejściach.

Jednowarstwowy perceptron

Wejścia i wyjścia oznaczmy odpowiednio przez x_i i y_j . Wtedy obliczenia definiowane przez sieć są opisane wyrażeniem

$$y_j = g(\sum_i w_{ij}x_i + \theta_j),$$

gdzie g jest funkcją aktywacji (na ogół nieliniową, np. sigmoidalną),
 w_{ij} są wagami tzw. połączeń, a θ_j przesunięciem/

Niestety jednowarstwowe perceptrony są bardzo słabe, mogą rozwiązywać tylko problemy liniowo separowalne.

Na kolejnym wykładzie przejdziemy do sieci wielowarstwowych i podstawowego algorytmu ich uczenia.

Wstęp do uczenia maszynowego

Podsumowanie

Sieć neuronowa jest ciągiem regresji wykorzystywanym do modelowania neuronów w mózgu.

Każdy węzeł (neuron) ma próg aktywacji po przekroczeniu którego wysyła sygnał do swoich następników.

Warstwa wejściowa odbiera dane, warstwy ukryte odbierają dane od warstwy poprzedzającej i wysyłają wyniki do warstwy następnej, warstwa wyjściowa zwraca wyniki.

Każdy neuron wykonuje regresję liniową postaci $h(x_1, \dots, x_n) = g(\theta_1 x_1 + \dots + \theta_n x_n + \theta_{n+1})$, gdzie f jest funkcją aktywacji przez którą przepuszczamy wynik regresji.

Wstęp do uczenia maszynowego

Notatki

[illegible]

Notatki

[illegible]

Notatki

[illegible]

Notatki

[illegible]