Tema 1 – Q-Learning

Vlad Bogolin

Invatare automata 2017

Data afisarii: 22.03.2017
Data predarii: 06.04.2017

Se accepta maxim 4 zile de intarziere, cu penalizare (din 10 puncte) astfel:

- 0.5p pentru prima zi de intarziere
- 1p/zi pentru urmatoarele 3 zile

Tema se prezinta la laborator. Tema se va realiza in limbajul de programare Python.

1 Enunt

Intr-un castel se gasesc mai multe camere care comunica intre ele prin portale.

Fiecare camera este reprezentata printr-un grid bidimensional (matrice n x m, unde n reprezinta numarul de linii si m numarul de coloane). Camerele pot avea dimensiuni diferite. In fiecare camera pot exista:

- maxim un gardian
- comori (toate comorile au aceeasi valoare)
- spatii inaccesibile
- portale spre alte camere.

Un portal este specificat prin $((r_1, c_1), (r_2, c_2))$, unde (r_1, c_1) reprezinta linia si coloana din camera 1 unde se afla un capat al portalului, iar (r_2, c_2) linia si coloana din camera 2 unde se afla celalalt capat al portalului. Gardienii nu pot folosi portalele pentru a se deplasa dintr-o camera in alta, asadar ei raman in aceeasi camera pe tot parcursul jocului. In castel exista un portal de iesire care se afla intr-o anumita camera.

Gigel este captiv in castel si isi doreste sa gaseasca portalul de iesire din castel fara a fi prins de gardieni. El poate colecta in drumul lui comorile pe care le intalneste pentru a-si maximiza recompensa. Atat Gigel, cat si gardienii au o raza de vizibilitate care reprezinta distanta pana la care observa lucruri in jurul pozitiei lor curente. Gigel si gardienii se pot misca stanga, dreapta, sus, jos sau pot ramane pe loc. In momentul in care Gigel calca pe o celula in care se afla un portal, atunci el este transporat automat la celalalt capat al portalului.

Se considera ca un gardian il captureaza pe Gigel in momentul in care cei doi au aceeasi pozitie pe harta (in aceeasi camera si in aceeasi pozitie din camera). In momentul in care un gardian il vede pe Gigel, acesta se deplaseaza spre el. Jocul se termina in momentul in care Gigel ajunge la portalul de iesire sau este prins de un gardian. Se considera ca Gigel castiga jocul in momentul in care ajunge la portalul de iesire. Gigel doreste sa evadeze din castel cat mai repede, asadar se poate considera ca el pierde jocul daca nu evadeaza intr-un anumit numar de pasi determinat de dimensiunea castelului.

2 Cerinte

Se cere dezvoltarea unui sistem care sa il ajute pe Gigel sa reactioneze in mediu folosind algoritmul Q-Learning. In implementarea algoritmului, reprezentarea unei stari se face pe baza elementelor din campul

vizual al lui Gigel. De asemenea, se poate realiza o grupare a starilor pentru a minimiza numarul de stari din tabela de utilitati.

Se va experimenta cu 3 strategii de exploatare/explorare:

- MaxFirst: exploatare pura, actiunea cu utilitatea maxima va fi aleasa de fiecare data
- Random: ignora tabela de utilitati si alege aleator o actiune posibila
- ϵ-greedy: atat timp cat exista actiuni ce nu au fost explorate, se va alege aleator una dintre acestea. Altfel, cu o probabilitate ϵ se va alege o actiune aleatoare, iar cu probabilitate $1 - \epsilon$ cea mai buna actiune din starea respectiva

Jocul se ruleaza pana cand Gigel reuseste sa castige un procent satisfacator de jocuri.

Se cere:

- **[3p]** implementarea jocului
  - numarul de gardieni, dimensiunea si distributia camerelor sa poata varia. Pozitia de start a lui Gigel si pozitiile gardienilor sunt alese aleator. Generarea jocului trebuie sa garanteze ca exista cel putin un drum de acces de la pozitia initiala a lui Gigel pana la portalul de iesire.
  - Vizualizare desfasurarii jocului (in format text sau interfata grafica)
- [**7p**] implementarea sistemului
  - [**3p**] implementare Q-Learning:
    - [**1p**] parametrii variabili (rata de invatare, factor de discount)
    - [**0.5p**] Max-First
    - [**0.5p**] Random
    - [**1p**] ϵ-greedy
  - [**2p**] grafice:
    - [**0.5p**] influenta factorului de invatare pentru un anumit scenariu al jocului.
    - [**0.5p**] influenta factorului de discount
    - [**0.5p**] evolutia scorului in functie de numarul episodului de antrenament.
    - [**0.5p**] diferenta intre cele 3 strategii de explorare din punct de vedere al evolutiei scorului.
  - [**2p**] limitari ale algoritmului
    - [**1.5p**] Cum influenteaza numarul de camere (portalul de iesire se afla in alta camera decat camera de start a lui Gigel), raza de perceptie, dimensiunea camerelor performanta algoritmului? Se dorestea testarea pe minim 4 scenarii diferite care surprind diverse aspecte. (Discutie sau grafice)
    - [**0.5p**] Gaseste algoritmul intotdeauna drumul spre portalul de iesire? (Discutie cu exemple concrete in functie de raspuns)
- [**maxim 2p**] bonus:
  - [**2p**] Discutie pe baza unei re-implementari a algoritmului, insotita de grafice si exemple despre cum adaugarea unor indicatori care arata directia spre portalul final (de exemplu, Gigel are un detector care ii spune distanta pana la portal) influenteaza procesul de invatare si performanta algoritmului Q-Learning. Cum se schimba performanta algoritmului pentru scenariile testate anterior?
  - [**2p**] Implementarea unei interfete grafice care sa permita afisarea sugestiva a tuturor parametrilor jocului.
  - **Punctajele pentru bonus nu se cumuleaza.**

Graficele si discutiile se vor prezenta intr-un fisier .pdf.

3 Hint-uri pentru implementare

Spatiul de stari al problemei poate creste foarte repede, asadar pentru a nu avea probleme in implementarea temei este de preferat sa incepeti cu scenarii de joc mici (o camera, de dimensiuni relativ mici (8x8, 10x10), 3-4 comori, un gardian, o raza de perceptie mare - sa permita vizualizarea tuturor obiectelor din camera), iar apoi sa variati treptat parametrii pentru a putea raspunde la toate intrebarile.