Osnove podržanog učenja

Petar Ozretić

Sveučilište u Splitu

21. rujna 2022.

Motivacija

DeepMind's StarCraft-playing AI beats 99.8 per cent of human gamers

TECHNOLOGY 30 October 2019

By Donna Lu

AlphaStar in green dealing with flying units from the Zerg players with a combination of powerful anti-air units
AlphaStar

An artificial intelligence can now play the real-time strategy video game *StarCraft II* so well that it is better than 99.8 per cent of human players.

The AI, called AlphaStar, was developed by tech firm DeepMind, which is owned by the same parent company as Google.

AlphaStar played anonymously against human players in a series of online games on the official *StarCraft II* game server, Battle.net, and ended up ranked in the top 200 players for each of the leagues it competed in.

AI triumphs against the world's top pro team in strategy game Dota 2

It's the first time an AI has beat a world champion e-sports team.

By Kelsey Piper | Apr 13, 2019, 6:30pm EDT

SHARE

OpenAI demoed their Dota bot with live matches at an event in San Francisco. | Kelsey Piper/Vox

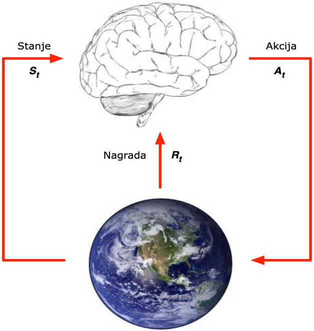
FUTURE PERFECT

Finding the best ways to do good.

Pro gamers, look out — for the first time ever, a world champion e-sports team has lost to an AI team.

In a series of live competitions between the **reigning Dota 2 world champion** team OG and the five-bot team OpenAI Five, the AI won two matches back-to-back, settling the best-of-three tournament. With 45,000 years of practice at Dota 2 under its belt, the system looked unstoppable — deftly navigating strategic decisions and racing to press

Agent i okolina



Agent sljedeći **strategiju** poduzima **akciju** i od **okoline** dobija **nagradu** te prelazi u novo **stanje**

MDP

Definition (MDP)

MDP je uređena trojka $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}_0)$ gdje je

$\mathcal{S}$ (prebrojiv) neprazan skup čije elemente nazivamo stanja;

$\mathcal{A}$ (prebrojiv) neprazan skup čije elemente nazivamo akcije;

$\mathcal{P}_0$ matrica prijelaza, koja svakom paru $(s, a) \in \mathcal{S} \times \mathcal{A}$ pridružuje vjerojatnosnu mjeru povrh $\mathcal{S} \times \mathbb{R}$ koju označavamo sa $\mathcal{P}_0(\cdot|s, a)$.

Dobit

- $G_t = R_{t+1} + R_{t+2} + R_{t+3} + ... + R_T$, gdje je T zadnji korak.

Dobit

- $G_t = R_{t+1} + R_{t+2} + R_{t+3} + ... + R_T$, gdje je T zadnji korak.
-

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{1}$$

parametar γ, $0 \leq \gamma \leq 1$, nazivamo *korekcijski faktor*.

Dobit

- $G_t = R_{t+1} + R_{t+2} + R_{t+3} + ... + R_T$, gdje je T zadnji korak.
- $$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2}$$

  parametar γ, $0 \leq \gamma \leq 1$, nazivamo *korekcijski faktor*.
- $$\begin{aligned} G_t &= R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} + ... \\ &= R_{t+1} + \gamma[R_{t+2} + \gamma^1 R_{t+3} + \gamma^2 R_{t+4} + ...] \\ &= R_{t+1} + \gamma G_{t+1} \end{aligned} \tag{3}$$

Funkcija vrijednosti

- Ako agent slijedi strategiju π u trenutku t, onda nam $\pi(a|s)$ kaže kolika je vjerojatnost da je $A_t = a$ ako je $S_t = s$, tj. $\pi(a|s) = \mathbb{P}[A_t = a|S_t = s]$

Funkcija vrijednosti

- Ako agent slijedi strategiju π u trenutku t, onda nam $\pi(a|s)$ kaže kolika je vjerojatnost da je $A_t = a$ ako je $S_t = s$, tj. $\pi(a|s) = \mathbb{P}[A_t = a | S_t = s]$
- $$v_\pi(s) = \mathbb{E}_\pi[G_t | S_t = s] = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s],$$

  za sve $s \in \mathcal{S}$. - vrijednosna funkcija stanja

Funkcija vrijednosti

- Ako agent slijedi strategiju π u trenutku t, onda nam $\pi(a|s)$ kaže kolika je vjerojatnost da je $A_t = a$ ako je $S_t = s$, tj. $\pi(a|s) = \mathbb{P}[A_t = a|S_t = s]$
- $$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s] = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s],$$

  za sve $s \in \mathcal{S}$. - vrijednosna funkcija stanja
- $$q_\pi(s,a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = s] = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s, A_t = a]$$

  - vrijednosna funkcija akcije

Bellmanova jednadžba za v_π

$$
\begin{aligned}
v_\pi(s) &= \mathbb{E}_\pi[G_t|S_t = s] \\
&= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s] \\
&= \sum_a \pi(a|s) \sum_{s'} \sum_r \mathcal{P}_0(s', r|s, a)[r + \gamma \mathbb{E}_\pi[G_{t+1}|S_{t+1} = s']] \\
&= \sum_a \pi(a|s) \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma v_\pi(s')], \forall s \in \mathcal{S}
\end{aligned} \tag{4}
$$

Optimalna strategija

Definition

Neka su za MDP $\mathcal{M}$ sa skupom stanja $\mathcal{S}$ dane strategije π i π' sa vrijednosnim funkcijama v_π i $v_{\pi'}$ redom. Za strategiju π kažemo da je *bolja ili jednaka* strategiji π', i pišemo $\pi \geq \pi'$, ako za svaki $s \in \mathcal{S}$ vrijedi $v_\pi(s) \geq v_{\pi'}(s)$.

Optimalna strategija

- Uvijek postoji barem jedna strategija koja je bolja ili jednaka od svih ostalih. Iako možda postoji više takvih strategija sve ih se označava sa π_* i nazivamo ih *optimalna strategija*.

Optimalna strategija

- Uvijek postoji barem jedna strategija koja je bolja ili jednaka od svih ostalih. Iako možda postoji više takvih strategija sve ih se označava sa π_* i nazivamo ih *optimalna strategija*.
- $$v_*(s) = \max_{\pi} v_\pi(s)$$

  za sve $s \in \mathcal{S}$. - optimalna vrijednosna funkcija stanja

Optimalna strategija

- Uvijek postoji barem jedna strategija koja je bolja ili jednaka od svih ostalih. Iako možda postoji više takvih strategija sve ih se označava sa π_* i nazivamo ih *optimalna strategija*.
- $$v_*(s) = \max_{\pi} v_{\pi}(s)$$

  za sve $s \in \mathcal{S}$. - optimalna vrijednosna funkcija stanja
- $$q_*(s,a) = \max_{\pi} q_{\pi}(s,a)$$

  za sve $s \in \mathcal{S}$ i sve $s \in \mathcal{A}$. - optimalna vrijednosna funkcija akcije

Bellmanova jednadžba optimalnosti

$$
\begin{aligned}
v_*(s) &= \max_{a \in \mathcal{A}(s)} q_{\pi *}(s, a) && (5)\\
&= \max_a \mathbb{E}_{\pi *}[G_t | S_t = s, A_t = a] \\
&= \max_a \mathbb{E}_{\pi *}[R_{t+1} + \gamma G_{t+1} | S_t = s, A_t = a] \\
&= \max_a \mathbb{E}[R_{t+1} + \gamma v_*(S_{t+1}) | S_t = s, A_t = a] && (6)\\
&= \max_a \sum_{s', r} \mathcal{P}_0(s', r | s, a)[r + \gamma v_*(s')], && (7)
\end{aligned}
$$

za sve $s \in \mathcal{S}$

Bellmanova jednadžba optimalnosti

- Iako rješavanje Bellmanove optimalne jednadžbe daje optimalnu strategiju za problem PU-a, ova metoda je sama po sebi rijetko kad korisna.
- Da bi ju se uopće razmotrilo trebaju biti ispunjena tri uvjeta : (1) potpuno poznavanje okoline, (2) dovoljan broj računalnih resursa za izračun; (3) sva stanja imaju Markovljevo svojstvo;
- U stvarnom svijetu rijetko je slučaj da su sva tri uvjeta ispunjena, a najčešće nije nijedan.
- Npr. za partiju igre Backgammon, isto analogno vrijedi i za Go ili Šah, iako vrijede uvjeti (1) i (3), uvjet (2) je nepremostiva prepreka: igra ima 10^{20} različitih stanja i najmoćnijim današnjim računalima bi trebale tisuće godina za rješavanje pripadnih Bellmanovih jednadžbi.

Dinamičko programiranje

Dinamičko programiranje

- Koristimo Bellmanove jednadžbe za v_π kao pravilo ažuriranja

Dinamičko programiranje

- Koristimo Bellmanove jednadžbe za v_π kao pravilo ažuriranja
-

$$\begin{aligned} v_{k+1}(s) &= \mathbb{E}_\pi[R_{t+1} + \gamma v_k(S_{t+1}) | S_t = s] \\ &= \sum_a \pi(a|s) \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma v_k(s')], \end{aligned} \tag{8}$$

, za sve $s \in \mathcal{S}$.

Dinamičko programiranje

- Koristimo Bellmanove jednadžbe za v_π kao pravilo ažuriranja
-

$$\begin{aligned} v_{k+1}(s) &= \mathbb{E}_\pi[R_{t+1} + \gamma v_k(S_{t+1}) | S_t = s] \\ &= \sum_a \pi(a|s) \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma v_k(s')], \end{aligned} \tag{9}$$

  , za sve $s \in \mathcal{S}$.
- Očito je $v_k = v_\pi$ fiksna točka ovog pravila ažuriranja jer Bellmanove jednadžbe za v_π osiguravaju jednakost u tom slučaju.

Dinamičko programiranje

- Koristimo Bellmanove jednadžbe za v_π kao pravilo ažuriranja
- $$\begin{aligned} v_{k+1}(s) &= \mathbb{E}_\pi[R_{t+1} + \gamma v_k(S_{t+1}) | S_t = s] \\ &= \sum_a \pi(a|s) \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma v_k(s')], \end{aligned} \tag{10}$$

  , za sve $s \in \mathcal{S}$.
- Očito je $v_k = v_\pi$ fiksna točka ovog pravila ažuriranja jer Bellmanove jednadžbe za v_π osiguravaju jednakost u tom slučaju.
- Može se pokazati da niz v_k konvergira ka v_π kada $k \to \infty$ (*Banachov teorem o fiksnoj točki*)

Iterativno vrednovanje strategija (predviđanje)

Algoritam 1: *Iterativno vrednovanje strategije*

Podatci: strategija π koju treba vrednovati

Odredi: mali $\epsilon > 0$ kojim određujemo željena preciznost

Za sve $s \in \mathcal{S}$ proizvoljno postavi $V(s)$ uz uvjet da je

$V(terminal) = 0$;

$\Delta \leftarrow \epsilon + 1$;

dok $\Delta > \epsilon$ **čini**

  $\Delta \leftarrow 0$;

  za $s \in \mathcal{S}$ **čini**

    $v \leftarrow V(s)$;

    $V(s) \leftarrow \sum_a \pi(a|s) \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma V(s')]$;

    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$;

  kraj

kraj

Iteracija strategija (kontrola)

Algoritam 2: *Iteracija strategija za procjenu $\pi \approx \pi_*$*

Inicijaliziraj: za sve $s \in \mathcal{S}$ (nasumično) postavi $V(s) \in \mathbf{R}$ i

$$\pi(s) \in \mathcal{A}(s),\ V(terminal) = 0$$

Vrednovanje strategije:

$\Delta \leftarrow \epsilon + 1$;

dok $\Delta > \epsilon$ **čini**

- $\Delta \leftarrow 0$;
- **za** $s \in \mathcal{S}$ **čini**
  - $v \leftarrow V(s)$;
  - $V(s) \leftarrow \sum_{s',r} \mathcal{P}_0(s', r | s, \pi(s))[r + \gamma V(s')]$;
  - $\Delta \leftarrow \max(\Delta, |v - V(s)|)$;
- **kraj**

kraj

Poboljšanje strategije:

strategija-stabilna $\leftarrow$ *true*;

za $s \in \mathcal{S}$ **čini**

- *stara-akcija* $\leftarrow \pi(s)$;
- $\pi(s) \leftarrow \text{argmax}_a \sum_{s',r} \mathcal{P}_0(s', r | s, a)[r + \gamma V(s')]$;
- **ako** *stara-akcija* $\neq \pi(s)$ **onda**
  - *strategija-stabilna* $\leftarrow false$;
- **kraj**

kraj

ako *strategija-stabilna* **onda**

- zaustavi algoritam i vrati $V \approx v_*$ i $\pi \approx \pi_*$;

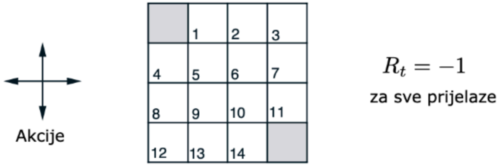
kraj

inače

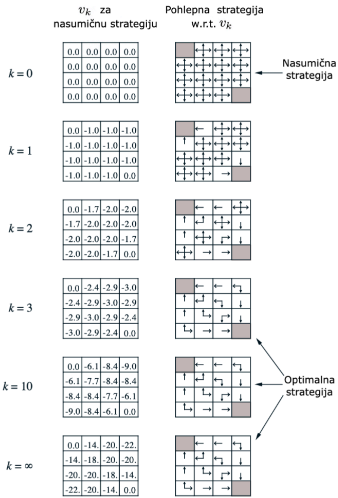
- idi na *Vrednovanje strategije*;

kraj

Mrežni svijet 4x4



Mrežni svijet 4x4



Iteracija vrijednosti

Algoritam 3: *Iteracija vrijednosti za procjenu $\pi \approx \pi_*$*

Inicijaliziraj: za sve $s \in \mathcal{S}$ (nasumično) postavi $V(s) \in \mathbf{R}$, $V(terminal) = 0$ te odaberi mali $\epsilon > 0$ kojim se određuje željenu preciznost

$\Delta \leftarrow \epsilon + 1$;

dok $\Delta > \epsilon$ **čini**

  $\Delta \leftarrow 0$;

  za $s \in \mathcal{S}$ **čini**

    $v \leftarrow V(s)$;

    $V(s) \leftarrow \max_a \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma V(s')]$;

    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$;

  kraj

kraj

Vrati determinističku strategiju $\pi \approx \pi_*$ za koju je

$\pi(s) = \operatorname{argmax}_a \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma V(s')]$;

Predviđanje bez modela

Predviđanje bez modela

- DP metode mogu koristiti samo ako je dinamika okoliša u potpunosti poznata.

Predviđanje bez modela

- DP metode mogu koristiti samo ako je dinamika okoliša u potpunosti poznata.
- *Monte-Carlo učenje* - metode koji prijeđu cijelu putanju agenta i procjenjuju vrijednost iz dobiti uzoraka;

Predviđanje bez modela

- DP metode mogu koristiti samo ako je dinamika okoliša u potpunosti poznata.
- *Monte-Carlo učenje* - metode koji prijeđu cijelu putanju agenta i procjenjuju vrijednost iz dobiti uzoraka;
- *Učenje s vremenskom razlikom*, eng. *temporal-difference learning* - metode koje gledaju jedan korak unaprijed i procjenjuju dobit nakon tog jednog koraka.

Monte Carlo evaluacija

Algoritam 4: *MC metoda prvog posjeta za procjenu vrijednosti* $V \approx v_\pi$

Zadana je strategija π koju treba procjeniti;

Inicijaliziraj: za sve $s \in \mathcal{S}$ (nasumično) postavi $V(s) \in \mathbf{R}$;

$Dobiti(s) \leftarrow$ prazna lista, za sve $s \in \mathcal{S}$;

dok *true* **čini**

  Generiraj epizodu sljedeći strategiju π:

  $S_0, A_0, R_1, S_1, A_1, R_2, \cdots, S_{T-1}, A_{T-1}, R_T$;

  $G \leftarrow 0$;

  za *svaki korak* $t \in [T-1, T-2, ..., 0]$ **čini**

    $G \leftarrow \gamma G + R_{t+1}$;

    ako *Stanje* S_t *se* ***ne*** *nalazi u nizu* $S_0, S_1, \cdots, S_{t-1}$ **onda**

      Dodaj G na kraj liste $Dobiti(S_t)$;

      $V(S_t) \leftarrow prosjek(Dobiti(S_t))$;

    kraj

  kraj

kraj

Temporal difference evaluacija

Algoritam 5: *TD(0) za procjenu vrijednosti* $V \approx v_\pi$

Dana je strategija π koju treba procijeniti

Inicijaliziraj: za sve $s \in \mathcal{S}$ (nasumično) postavi $V(s) \in \mathbf{R}$;

Odaberi parametar algoritma $\alpha \in \langle 0, 1math]$;

dok *true (za svaku epizodu)* **čini**

  Inicijaliziraj S:

  za *svaki korak epizode:* **čini**

    $A \leftarrow$ akcija dana od π za stanje S;

    poduzmi akciju A i promatraj R, S';

    $V(S) \leftarrow V(S) + \alpha[R + \gamma V(S') - V(S)]$;

    $S \leftarrow S'$;

    dok ne dođe do terminalnog S;

  kraj

kraj

MC vs TD

- Prednost TD metoda nad MC metodama je da su implementirane na inkrementalan način
- Kod MC metoda potrebno je dočekati kraj epizode, jer se tek onda saznaje kolika je dobit, dok se kod TD-a čeka samo jedan korak
- U brojnim primjenama epizode traju jako dugo, a kod kontinuiranih slučaj uopće nema epizoda
- Koja metoda brže uči? (otvoreno pitanje)
- U praksi se pokazalo da TD metode u pravilu konvergiraju brže od MC metoda na stohastičkim zadatcima

Kontrola bez modela

Kontrola bez modela

- Pohlepna strategija
- $$\pi'(s) = \underset{a}{\operatorname{argmax}} \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma v_\pi(s')]$$

Kontrola bez modela

- Pohlepna strategija
- $$\pi'(s) = \operatorname*{argmax}_a \sum_{s',r} \mathcal{P}_0(s', r|s, a)[r + \gamma v_\pi(s')]$$
- $$\pi'(s) = \operatorname*{argmax}_{a \in \mathcal{A}} Q(s, a)$$

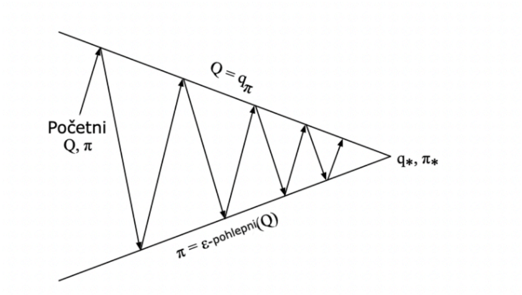
Kontrola bez modela

- Problem iskorištavanja i istraživanja (*exploration & exploration*)
- ako sljedimo pohlepnu strategiju (obzirom na trenutno stanje vrijednosne funkcije) može se dogoditi da neka stanja ne budu (dovoljno) istražena

Kontrola bez modela

- Problem iskorištavanja i istraživanja (*exploration & exploration*)
- ako slijedimo pohlepnu strategiju (obzirom na trenutno stanje vrijednosne funkcije) može se dogoditi da neka stanja ne budu (dovoljno) istražena
- ϵ-pohlepna istraživanje

MC iteracija strategija

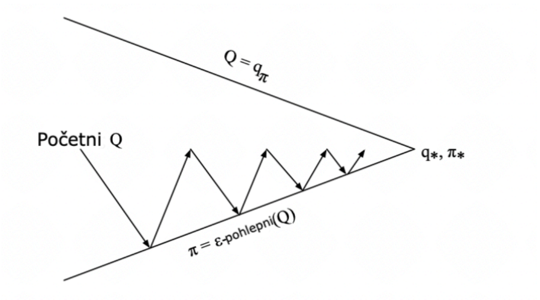


Slika 4.1: Monte-Carlo iteracija strategija [1]:

Vrednovanje strategije: MC vrednovanje, $Q = q_\pi$

Poboljšanje strategije: $\epsilon - pohlepno$

TD iteracija strategija



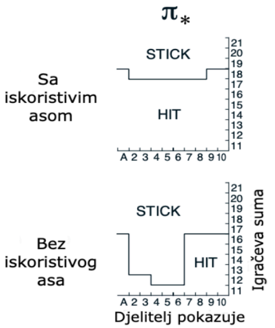
Slika 4.2: TD iteracija strategija (nakon **svakog** vremenskog koraka) [1]:

Vrednovanje strategije: SARSA Q_π

Poboljšanje strategije: $\epsilon - pohlepno$

Praktični primjer: Blackjack

Monte Carlo primjena



Sutton i Barto

```
s Asom:
      A    2    3    4    5    6    7    8    9   10
21 ["s","s","s","s","s","s","s","s","s","s"]
20 ["s","s","s","s","s","s","s","s","s","s"]
19 ["s","s","s","s","s","s","s","s","s","s"]
18 ["h","s","s","s","s","s","s","s","h","h"]
17 ["h","h","h","h","h","h","h","h","h","h"]
16 ["h","h","h","h","h","h","h","h","h","h"]
15 ["h","h","h","h","h","h","h","h","h","h"]
14 ["h","h","h","h","h","h","h","h","h","h"]
13 ["h","h","h","h","h","h","h","h","h","h"]
12 ["h","h","h","h","h","h","h","h","h","h"]
================================
bez Asa:
      A    2    3    4    5    6    7    8    9   10
21 ["s","s","s","s","s","s","s","s","s","s"]
20 ["s","s","s","s","s","s","s","s","s","s"]
19 ["s","s","s","s","s","s","s","s","s","s"]
18 ["s","s","s","s","s","s","s","s","s","s"]
17 ["s","s","s","s","s","s","s","s","s","s"]
16 ["h","s","s","s","s","s","h","h","h","s"]
15 ["h","s","s","s","s","s","h","h","h","h"]
14 ["h","s","s","s","s","s","h","h","h","h"]
13 ["h","s","s","s","s","s","h","h","h","h"]
12 ["h","s","s","s","s","s","h","h","h","h"]
petarozretic@MacBook-Pro-2 Diplomski %
```

Ozretić

Sutton i Barto

"This policy is the same as the "basic" strategy of Thorp (1966) with the sole exception of the leftmost notch in the policy for a usable ace, which is not present in Thorp's strategy. We are uncertain of the reason for this discrepancy, but confident that what is shown here is indeed the optimal policy for the version of blackjack we have described."

Hvala na pažnji

Hvala na pažnji