senzory

MDPI

Článok

Hranie Flappy Bird na základe rozpoznávania pohybu pomocou modelu transformátora a senzora LIDAR

Iveta DirgovááLuptáková,Martin Kubovčík * a JiríPospíchal *

Ústav počítačových technológií a informatiky, Fakulta prírodných vied,
Univerzita sv. Cyrila a Metoda, J. Herdu 2, 917 01 Trnava, Slovensko; iveta.dirgova.luptakova@ucm.sk
* Korešpondencia: martin.kubovcik@ucm.sk (MK); jiri.pospichal@ucm.sk (JP)

Abstrakt: Transformátorová neurónová sieť sa v tejto štúdii používa na predpovedanie hodnôt Q v simulovanom prostredí pomocou techník učenia sa. Cieľom je naučiť agenta navigovať a excelovať v hre Flappy Bird, ktorá sa stala obľúbeným modelom ovládania v prístupoch strojového učenia. Na rozdiel od väčšiny špičkových existujúcich prístupov, ktoré využívajú ako vstup vykreslený obraz hry, náš hlavný prínos spočíva v použití senzorického vstupu z LIDAR, ktorý je reprezentovaný metódou ray casting. Konkrétne sa zameriavame na pochopenie časového kontextu meraní z pohľadu vrhania lúčov a optimalizáciu potenciálne rizikového správania zvážením stupňa priblíženia sa k objektom identifikovaným ako prekážky. Agent sa naučil používať merania z vrhania lúčov, aby sa vyhol kolíziám s prekážkami. Náš model výrazne prevyšuje súvisiace prístupy. V budúcnosti sa snažíme tento prístup aplikovať v reálnych scenároch.

kľúčové slová:posilňovanie učenia; pohybové senzory; odlievanie lúčov; spracovanie signálu; spracovanie časových radov; model transformátora; robotika; hra Flappy Bird; kontrola agentov

Citácia: DirgováLuptáková,I.; Kubovčík, M.; Pospíchal, J. Hranie Flappy Bird na základe rozpoznávania pohybu pomocou modelu transformátora a senzora LIDAR.*Senzory* **2024**,*24*, 1905. https://doi.org/ 10.3390/s24061905

Akademickí redaktori: Arturo de la Escalera Hueso, Fernando Fernández-Martínez, Manuel Gil-Martín a Rubén San-Segundo

Prijaté: 22. decembra 2023
Upravené: 7. marca 2024
Prijaté: 14. marca 2024
Zverejnené: 16. marca 2024

1. Úvod

Autonómne riadenie robotov pomocou posilňovacieho učenia (RL) sa ukázalo ako jedna z dôležitých tém strojového učenia. Rozsiahle využitie technológie hlbokej neurónovej siete z nej urobilo najbežnejšiu voľbu na vytváranie riadiacich systémov, ktoré sa spoliehajú na informácie zozbierané z operačného prostredia robota. Tento článok sa zameriava na spracovanie zozbieraných údajov v časovom rámci a používanie informácií o pohybe na riadenie akcií robota. Použitá architektúra je model transformátora [1], ktorý dokáže efektívne spracovať dlhé časové rady [2].

Populárna počítačová hra Flappy Bird, ktorú vytvoril vietnamský programátor Dong Nguyen [3] tu funguje ako simulačné prostredie. Cieľom hráča, ktorý ovláda simulovaného robotického vtáka, je letieť nepretržite vpred bez kolízie. Vták narazí na rad párov rúr, ktoré mu bránia v ceste, a tie sú zavesené na vrchu a vyčnievajú zospodu herného prostredia. Medzi každým párom rúrok je udržiavaná konštantná vzdialenosť, ktorá vytvára medzeru, cez ktorú môže vták lietať. Vertikálna poloha tejto medzery je generovaná náhodne, čím sa do hry vnáša dynamický prvok. Neustále sa meniace prostredie vyžaduje, aby sa hráči prispôsobili a rýchlo reagovali. Gravitácia ťahá vtáka nadol, zatiaľ čo hráčove činy ho tlačia nahor. Horizontálna rýchlosť zostáva konštantná. Hra sa okamžite končí, ak sa vták zrazí buď s potrubím alebo so zemou.

Existuje niekoľko prístupov k trénovaniu hráčov vo Flappy Bird. Typickým prístupom je použitie obrazu generovaného hrou [4], s rôznymi úpravami. Ďalšia úprava zahŕňa zavedenie ďalšej negatívnej odmeny, keď sa agent zrazí s horným okrajom hernej obrazovky [5]. Ďalšie úpravy sú založené na vytvorení troch úrovní obtiažnosti tréningu, ľahké, stredné a ťažké [6], ktoré sa odlišujú šírkou medzery medzi rúrkami. Následná metóda zahŕňala počítačového experta, ktorý extrahoval kľúčové informácie z potrubí a agenta, ktoré sa potom používajú na predpovedanie akcií [7].

V tomto článku je hráč-vták vybavený senzorom simulovanej detekcie svetla a dosahu (LIDAR) reprezentovaným metódou vrhania lúčov na detekciu potrubí a zeme. Hráč môže využiť spracovanie signálov v časovom rade na manévrovanie okolo potrubí a vyhýbanie sa kolíziám. Cieľom je použiť údaje o pohybe na navigáciu cez prekážky, ako sú potrubia a zem. V tomto modeli sa na spracovanie signálov časových radov a lúčov využíva na mieru vytvorená hlboká neurónová sieť, tu nazývaná „transformátor pohybu".

Podobný prístup k spracovaniu časových komponentov používa [8–10]. Tieto práce sa však primárne zameriavajú na spracovanie údajov o ľudskej činnosti a zároveň zahŕňajú priestorové komponenty. Všetky priestorové merania sa interpretujú ako znaky. Model transformátora v tejto štúdii hľadá iba časové korelácie a nie korelácie medzi lúčmi snímača.

Transformátorový model už bol použitý na predikciu akcií, kde určuje ďalšiu akciu na základe aktuálneho stavu, predchádzajúcich akcií a odmien. Tento model však špecificky využíva kauzálny transformátor, ktorý obmedzuje spracovanie informácií iba jedným smerom z minulosti do budúcnosti [11]. Ďalšia aplikácia využíva model transformátora v RL ako náhradu konvolučných vrstiev na extrakciu prvkov. Toto je prípad modelu Swin Transformer používaného na spracovanie obrazu [12]. Líši sa od tohto dokumentu, ktorý nezahŕňa celú hernú obrazovku ako súčasť svojho vstupu. Ďalšia aplikácia modelu transformátora je v časovej doméne, ale zvažuje iba použitie posledného časového kroku na predikciu akcie, zatiaľ čo zostávajúce časové kroky sa používajú iba v procese učenia na výpočet chyby modelu. Používa tiež kauzálny transformátor [13].

Ďalšie stratégie na zlepšenie predikcie časových radov zahŕňajú využitie posledného časového kroku, spriemerovanie funkcií naprieč časovými krokmi a určenie maximálnej hodnoty medzi časovými krokmi.

Transformátor videnia [14] používa funkcie z posledného časového kroku na predpovedanie akcií, najmä prostredníctvom použitia tokenu triedy. V tomto článku predstavuje posledný časový krok konečný stav hry, čím sa eliminuje potreba ďalšieho tokenu triedy v časovom rade.

Priemer vlastností v jednotlivých časových krokoch sa používa v článku [15] a ich maximálna hodnota je uvedená v [16]. Posledná alternatíva zahŕňa zlúčenie prvkov v priebehu času, hoci tento prístup môže viesť k množeniu vstupov do nasledujúcej vrstvy a prispievať k nadmernému prispôsobeniu modelu [17].

Na rozdiel od predchádzajúceho výskumu Flappy Bird sa tento článok zameriava na využitie pochopenia časového kontextu z meraní vrhania lúčov. Transformátorový model sme použili na spracovanie historických meraní stavu, následne sme tieto údaje rozumným spôsobom agregovali, aby sme predpovedali aktuálnu činnosť agenta. Senzor simulovaný v našej štúdii má obmedzenejšie zorné pole v porovnaní s metódami používanými v predchádzajúcom výskume [18]. Preto je náš model navrhnutý tak, aby využil svoje predchádzajúce znalosti o prekážkach v prostredí na efektívnu navigáciu. Na rozdiel od modelov neurónových sietí, ktoré už boli aplikované na problém Flappy Bird, naším cieľom je navrhnúť metódu, ktorá dokáže efektívne kondenzovať informácie prenášané v priebehu času. To nám umožní vyjadriť vlastnosti akcií pre aktuálny stav agenta a jeho zodpovedajúcu reakciu. V dôsledku toho je náš model navrhnutý tak, aby predpovedal kategorickú distribúciu akcií na základe aktuálneho stavu agenta, berúc do úvahy predtým vyhodnotené stavy agenta. Tento prístup umožňuje modelu robiť informované rozhodnutia na základe súčasných aj minulých stavov agenta.

Hlavné príspevky tohto článku sú nasledovné:

- Vylepšený výkon: Náš model transformátora so snímačom vzdialenosti výrazne prekonal existujúce metódy (viac ako 50-násobné zvýšenie priemerného aj maximálneho skóre). To naznačuje, že skutočné roboty vybavené podobnými senzormi môžu potenciálne dosiahnuť výrazne vyššiu presnosť pri spracovaní dlhých sekvencií údajov senzorov.

- Učenie zamerané na senzory: Na rozdiel od predchádzajúcich prístupov sa náš agent spolieha výlučne na údaje senzorov (nie na celý obraz hry), aby sa poučil z minulých skúseností, identifikoval prekážky a navigoval v prostredí. To naznačuje, že zameranie sa na relevantné údaje zo senzorov môže byť efektívnou stratégiou na ovládanie robotov.
- Vizualizácia a sledovanie časovej podobnosti údajov zo senzorov: Tento výskum predstavuje techniku vizualizácie na sledovanie podobností v rámci sekvencií údajov zo senzorov počas tréningu modelu transformátora. Táto technika pomáha prispôsobiť model tak, aby sa zameral na kľúčové merania, ktoré majú vplyv na stratégiu hry a konečný výsledok, čím sa efektívne zbavia nekritických informácií. Tento prístup bol vyvinutý s cieľom skrátiť čas školenia a znížiť požiadavky na pamäť pre agenta.
- Použiteľnosť v reálnom svete: Naše zistenia majú potenciál byť aplikované na skutočné roboty pracujúce v nebezpečných prostrediach (porovnateľné so simuláciou Flappy Bird, kde môže agent havarovať). Začlenením koncepcie „súkromnej zóny“ a hĺbkového učenia sa roboty môžu potenciálne navigovať v zložitých úlohách a zároveň minimalizovať kolízie a predĺžiť ich prevádzkovú životnosť.

Tento dokument je usporiadaný nasledovne. oddiel2poskytuje prehľad základných algoritmických a výpočtových prístupov použitých v tejto práci. Medzi ne patrí architektúra duelovej siete pre Q-learning, architektúra pohybového transformátora, databázový server DeepMind Reverb používaný na strojové učenie, vrhanie lúčov na detekciu prekážok, epizodická pamäť integrovaná do vstupu transformátora a koncept súkromnej zóny, ktorý pomáha pri vyhýbaní sa prekážkam. . oddiel3podrobne popisuje proces optimalizácie pre zvolené metódy a ich hyperparametre. Táto časť skúma faktory, ako je počet použitých časových krokov, použité techniky redukcie funkcií a optimálna veľkosť súkromnej zóny. Končí sa analýzou zrážky s cieľom posúdiť potenciál na zlepšenie konečných výsledkov. oddiel4 rozoberá budúce aplikácie tejto metódy a skúma sľubné spôsoby, ako ju zlepšiť. Nakoniec, sekcia5sumarizuje kľúčové zistenia a závery prezentované v celom príspevku.

2. Materiály a metódy

Model transformátora je trénovaný prístupom hĺbkovej siete Q. Aby sme dosiahli efektívne učenie, musíme zbierať údaje o rôznych cestách skúmaných v rámci stavového priestoru a zdieľať aktualizované charakteristiky nášho výpočtového modelu. Túto úlohu uľahčuje špecializovaný databázový server DeepMind Reverb. Stavový priestor obsahuje iba merania z vrhania lúčov. Merania z odlievania lúčov si preto vyžadujú osobitnú expozíciu. Pretože je transformátor postavený na epizodickej pamäti, jeho použitie v probléme Flappy Bird je riešené nezávisle. Napokon, inovatívny prístup zahŕňa vytvorenie súkromnej zóny obklopujúcej agenta, aby sa zlepšila jeho schopnosť udržiavať bezpečnú vzdialenosť pri prechádzaní prekážkami. Zavedenie tohto konceptu výrazne zlepšuje výkon počas procesu učenia. Dôkladná analýza týchto metodík bude vykonaná v nasledujúcich častiach.

2.1. Duel Deep Q Network

Princípom architektúry duelovej siete je extrahovať vlastnosti zo stavového priestoru, ktoré sú relevantné pre predikciu hodnotovej funkcie a funkcie výhod. Hodnotová funkcia vyjadruje, aký výhodný je aktuálny stav agenta pre jeho politiku. Agent uprednostňuje prechádzajúce stavy, ktoré majú vyššie hodnoty. Táto stratégia zabezpečuje maximalizáciu celkovej hodnotovej funkcie. Aby bolo možné urobiť informovaný výber z množstva potenciálnych opatrení, je nevyhnutné zistiť prínos spojený s každým opatrením. To sa dosiahne využitím výhodnej funkcie [19]. V prípade diskrétneho akčného priestoru je potrebné vyjadriť pravdepodobnosti každej akcie vo forme

logits, ktoré sú predpovedané modelom hlbokej neurónovej siete [20]. Logity predstavujú Q-hodnoty, ktoré možno vypočítať podľa nasledujúceho vzťahu [21]:

$$Q(s,a) = V(s) + \left(A(s,a) - \frac{1}{A}\sum_{a'} A(s,a')\right) \quad (1)$$

$Q(s,a)$ vyjadruje funkciu kvality pre danú akciu a a v danom stave s. $V(s)$ vyjadruje hodnotovú funkciu pre daný stav s. $A(s,a)$ vyjadruje výhodnú funkciu pre danú akciu a v danom stave s. Priemer funkcie výhod naprieč akciami v danom stave s sa odpočítava od $A(s,a)$ funkciu. Preto výhodná akcia má nulový priemer [22].

Model je trénovaný pomocou funkcie logaritmickej hyperbolickej kosínusovej chyby (LogCosh), ktorá je menej citlivá na odľahlé hodnoty ako konvenčnejšia funkcia strednej štvorcovej chyby (MSE) [23]. Chybová funkcia modelu je vyjadrená takto:

$$L(\theta) = \mathrm{E}_{(s,a,r,s')\sim U(D)}\left[LogCosh\left(y_{DQN} - Q(s,a;\theta)\right)\right] \quad (2)$$

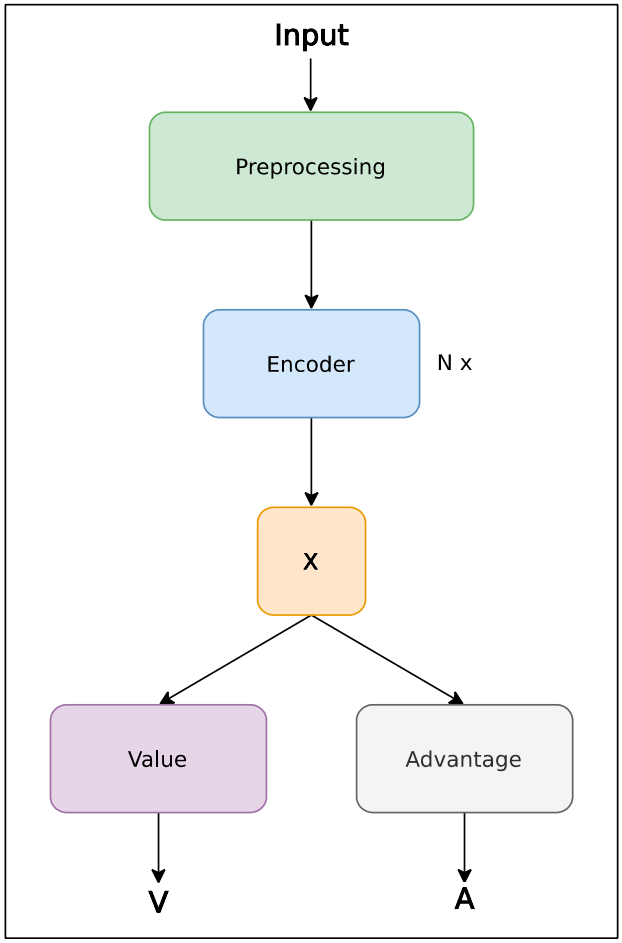
$$r_{DQN} = r + \gamma \max_{a'} Q\left(s', a'; \theta^{-}\right) \quad (3)$$

$U(D)$ predstavuje rovnomerné vzorkovanie z vyrovnávacej pamäte prehrávania D ktorý obsahuje trajektórie. $Q(s,a;\theta)$ vyjadruje Q-hodnotu predpovedanú modelom. Odmena je symbolizovaná r. a je ďalšia akcia vyjadrená maximálnou hodnotou Q v nasledujúcom stave s'. θ^{-} sú parametre modelu exponenciálneho kĺzavého priemeru (EMA) [24].

2.2. Pohybový transformátor

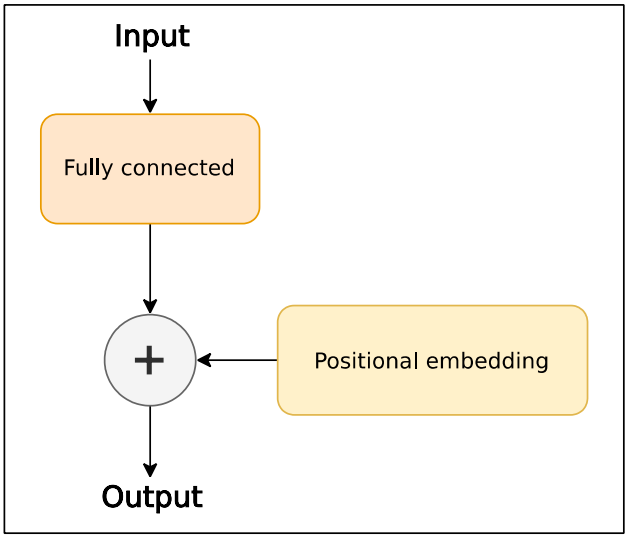
Architektúra pohybového transformátora je založená na bloku kódovača v modeli transformátora [25]. Účelom bloku kódovača je prejsť vstupným vektorom cez časovú os v oboch smeroch. Týmto spôsobom je možné hľadať vzťahy v historických údajoch z minulosti do budúcnosti alebo z budúcnosti do minulosti a prípadne ich vhodne priradiť k poslednému časovému kroku. Posledný časový krok predstavuje zdroj informácií v klasickom Markovovom rozhodovacom procese (MDP) [26]. Stavový vektor reprezentujúci lokálnu pamäť modelu sa privádza na vstup modelu. Úlohou procesu učenia modelu je potom optimalizovať globálnu pamäť (parametre) modelu tak, aby sa stavový priestor ideálne pretransformoval na akčný priestor. Výstup bloku kódovača však opäť predstavuje sekvenciu; tj pre každý časový krok predpovedá množinu extrahovaných vlastností zo vstupného vektora. Tu je uvedených niekoľko metód na extrakciu jednej konkrétnej distribúcie aktuálnej akcie a_t. Jedným z možných prístupov je použiť iba extrahované funkcie z posledného časového kroku na predpovedanie distribúcie akcií a_t, podobne ako token triedy [27]. Cieľom je použiť posledný časový krok s_t predvídať akciu a_t ako pri klasickom MDP. Ak sú potrebné nejaké historické prvky, sú vložené počas posledného časového kroku vďaka mechanizmu pozornosti. Ďalšou možnosťou je použiť priemer alebo maximum vo všetkých časových krokoch pre každý extrahovaný prvok samostatne.

Obrázok 1 ukazuje architektúru pohybového transformátora. Architektúra pozostáva z vrstvy predbežného spracovania, ktorá pridáva informácie o polohe do vstupného vektora v rámci časového radu. Potom nasleduje niekoľko blokov kódovača, ktoré extrahujú prvky pozdĺž časovej osi. Vrstva označená X predstavuje redukčnú vrstvu extrahovaných prvkov v časovom rade. Jeho typ sa počas experimentov menil. Posledné vrstvy sú hodnotou a výhodou, predstavujú plne prepojené výstupné vrstvy. Rovnica (1) sa potom aplikuje na výstup pohybového transformátora.



Fobrázok 1. Architektúra modelu pohybového transformátora.

Obrázok2zobrazuje architektúru vrstvy ayer, ktorého súčasťou je projekcia lpredspracovania vo forme plne prepojenej vrstvy s aktivačná funkcia. Jeho úlohou je tlineárnou transformáciou počtu vstupných prvkov na ber skrytých prvkov používaných v tpočet zvyšku modelu. Následne pozičné vloženie ding, ktorý predstavuje trénovateľné premenné, sa sčítava s výstupom plne prepojenej vrstvy. V tomto článku sú teda trénované vloženia pre časové rady spolu s modelom [28].



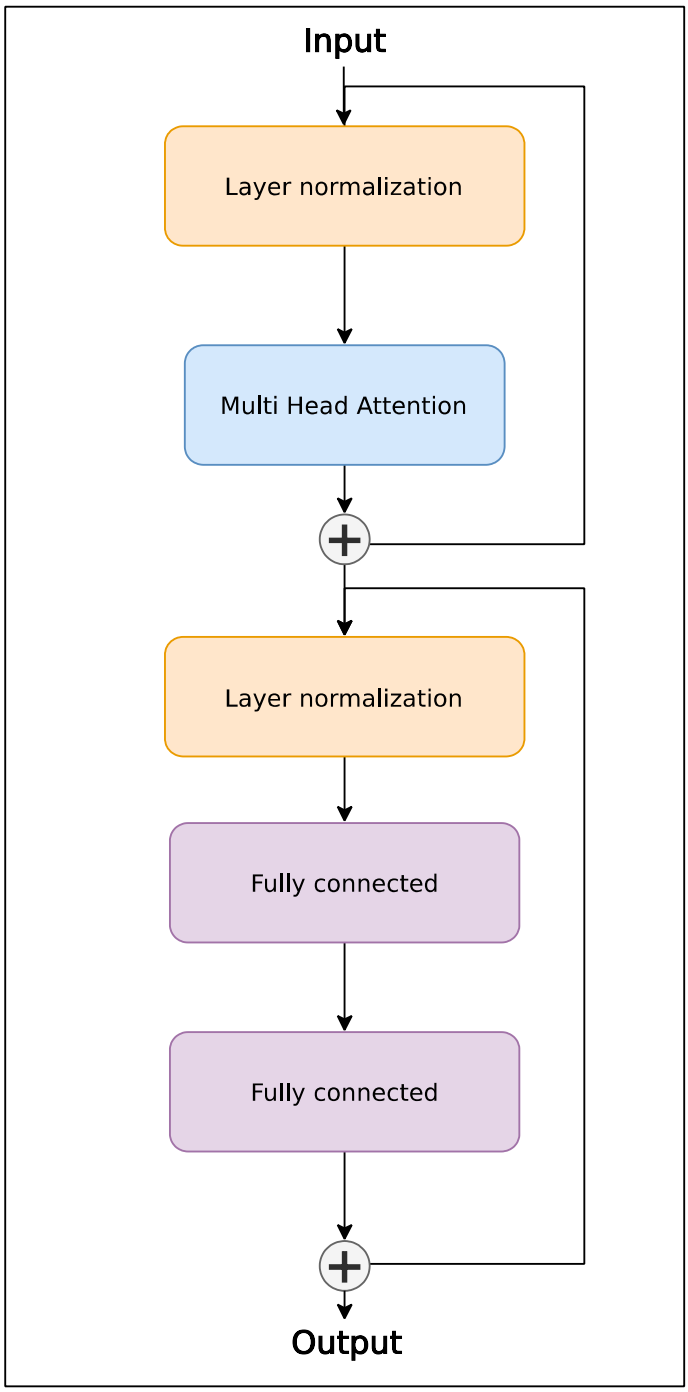
Fobrázok 2. Architektúra vrstvy predbežného spracovania.

Architektúra bloku kódovača je znázornená na obrázku al [3. Pozostáva z dvojice rezidencií, u29] čiastkové bloky. Prvým je pozornosť viacerých hláv [30], ktoré spracovávajú časové rady apodľa nasledujúcich vzťahov:

$$MultiHead(Q,K,V) = Concat(hlavu_1, \ldots, hlavu_n)W_O + b_O \quad (4)$$

$$hlavu_i = Pozor\left(QW_{Q\ i} + b_{Q\ i}, KW_{K\ i} + b_{K\ i}, VW_{V\ i} + b_{V\ i}\right) \quad (5)$$

$$Pozornost(Q,K,V) = softmax\left(\frac{QK_T}{\sqrt{d_k}}\right)V \quad (6)$$



Obrázok 3. Architektúra vrstvy kódovača.

W_o predstavuje váhy a b_o predstavuje odchýlky lineárnej transformácie po zlúčení hláv. Proces spájania hláv zahŕňa zreťazenie tenzorov pozdĺž rozmer hlavy (os). $W_{Q\ i}$, $W_{K\ i}$, a $W_{V\ i}$ predstavujú váhy a $b_{Q\ i}$, b_{Ki}, a $b_{V\ i}$ reprezentovať odchýlky lineárnej transformácie vstupného vektora vrstvy Q (dotaz), K (kľúč) a V (Hodnota) do priestoru, ktorý spravuje funkcia pozornosti. d_k predstavuje počet rozmerov K po lineárnej projekcii vektora vstupu vrstvy.

Druhým blokom je viacvrstvový perceptrón (MLP) [31], a jeho úlohou je nelineárne transformovať spracované časové rady. Použitá nelineárna aktivačná funkcia je Gaussovské chybové lineárne jednotky (GeLU) [32], nanáša sa po prvej úplne spojenej vrstve. Dá sa vyjadriť nasledujúcim vzťahom:

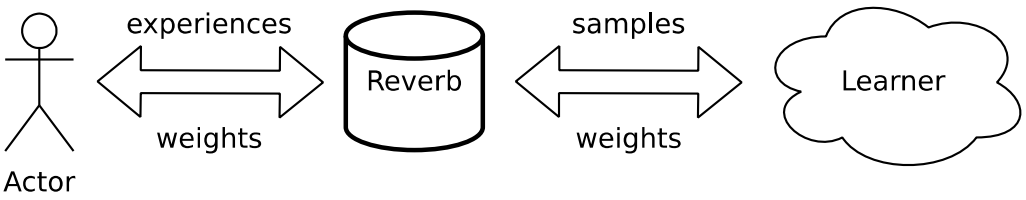
$$r = GeLU(xW_1 + b_1)W_2 + b_2 \quad (7)$$

W_1 a b_1 predstavujú parametre hmotnosti a odchýlky pre prvú plne pripojenú vrstvu, na ktorú sa následne aplikuje nelineárna transformácia. Parametre W_2 a b_2 predstavujú druhú vrstvu bloku. Táto vrstva je zodpovedná za vykonávanie lineárnej transformácie na výstupe odvodenom z predchádzajúcej vrstvy. Rozmer tohto transformovaného výstupu sa rovná rozmeru pôvodného vstupného vektora. Prvá vrstva bloku má zvyčajne 4-krát viac neurónov ako posledná vrstva bloku [33].

2.3. Databáza Reverb

Databázový server DeepMind Reverb sa používa na efektívne spravovanie zhromaždených trajektórií a distribúciu aktualizovaných parametrov modelu. Tento dedikovaný databázový server je prispôsobený pre algoritmy RL, kde funguje ako vyrovnávacia pamäť pre prehrávanie. Používatelia môžu ovládať stratégie výberu a odstraňovania prvkov z databázy a možnosti riadenia pomeru medzi vzorkovanými a vloženými prvkami. Databázový server môže obsahovať niekoľko tabuliek, kde sú uložené trajektórie alebo parametre modelu. Dôležitou vlastnosťou je kompresia uložených údajov, ktoré poskytuje databázový server. V prípade prekrývajúcich sa trajektórií je dôležité vyhnúť sa ukladaniu duplicitných trajektórií [34]. Stratégia používaná na vzorkovanie trajektórií je jednotný vzorkovač, ktorý vyberá trajektórie z tabuľky s rovnakou pravdepodobnosťou. Stratégiou odstraňovania trajektórií z tabuľky je metóda FIFO (first-in-first-out). Pomer medzi vzorkovanými a vloženými položkami je empiricky nastavený na 32 s toleranciou 10 %.

Použitá architektúra klient-server je znázornená na obrázku4. Server predstavuje úložisko databázy, kde sú uložené trajektórie. Aktér zastupuje klienta vo forme agenta, ktorý prostredníctvom interakcií s prostredím zbiera skúsenosti vo forme trajektórií a trajektórie ukladá na databázový server. Študent predstavuje klienta, ktorý získava trajektórie z databázového servera a používa ich na trénovanie modelu agenta. Po tréningovom procese dostane agent novo aktualizované parametre modelu cez databázový server. Podobný princíp sa používa v rámci Acme [35].



Obrázok 4. Školenie klient-server.

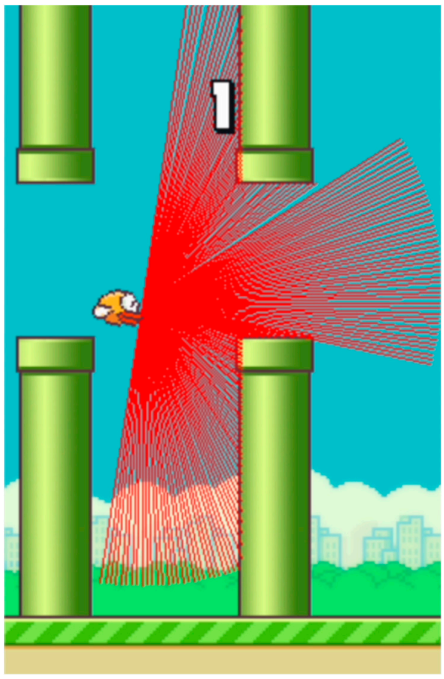
2.4. LIDAR

Metóda používaná na detekciu blízkych objektov a sledovanie pohybu agenta v hernom prostredí využíva techniku vrhania lúčov, pre jednoduchosť označovanú ako LIDAR. Skladá sa zo 180 lúčov, ktoré sú nasmerované z prednej strany agenta na

pravý okraj obrazovky (pozri obrázok5). Koncové body lúčov sú určené nasledujúce vzťahy:

$$x = d_{MAX} * \cos\left(\alpha - hráč_{\alpha} - \frac{\pi}{2}\right) + hráč_{x} \quad (8)$$

$$r = d_{MAX} * \text{hriech}\left(\alpha - hráč_{\alpha} - \frac{\pi}{2}\right) + hráč_{r} \quad (9)$$



Obrázok 5. Senzor LIDAR reprezentovaný vrhaním lúčov.

d_{MAX}predstavuje maximálnu dĺžku lúča. Uholαurčuje smer žiarenia lúčov a a$hráč_{\alpha}$ vyjadruje uhol sklonu hráča k rovine herného priestoru. Počiatočný bod lúča je vyjadrený súradnicami čela agentov $hráč_{x}$a$hráč_{r}$.

Keď je vták vytlačený nahor, otočí sa smerom k oblohe pod uhlom 45 stupňov. Pri absencii vstupu hráča sa vták pomaly otáča smerom k zemi, kým nedosiahne uhol −90 stupňov a potom spadne priamo dole.

Maximálna dĺžka lúča je vzdialenosť medzi prednou stranou agenta a pravým okrajom obrazovky. Kolmý lúč sa teda dotýka okraja obrazovky (ak nie sú žiadne prekážky), zatiaľ čo iné lúče pod väčším alebo menším uhlom zvyčajne nedosahujú okraj obrazovky. Toto správanie odráža skutočné šírenie ideálneho svetla a merania jeho odrazov od ideálne odrážajúcich objektov pod rôznymi uhlami. Naproti tomu, keď iné metódy využívajú obraz vygenerovaný hrou, systém ho vidí ako celok.

Tu sa lúče vyžarované vrhaním lúčov rozprestierajú v polkruhovom vzore a dokážu odhaliť prekážky v obmedzenej oblasti pred agentom. Navyše, ak vták nie je umiestnený v správnej výške a orientácii vzhľadom na rovinu herného prostredia, detekčné lúče ani nezaregistrujú zem. V dôsledku toho agent nemôže poznať svoju nadmorskú výšku počas každej epizódy. Senzor pracuje pri uhlovom rozlíšení 1 stupeň a má dosah obmedzený iba viditeľnou časťou prostredia pred hráčom. Zrážka s lúčom nastáva, keď lúč dopadne na povrch potrubia alebo na zem. Vzdialenosť k objektu sa meria ako euklidovská vzdialenosť medzi prednou stranou agenta,

odkiaľ lúč pochádza a bod kolízie. Tieto hodnoty však nie sú štatisticky optimálne pre vstup modelu; preto je vhodné ich normalizovať na rozsah [0, 1].

$$d_{\alpha}^{norma} = \frac{d_{\alpha}}{d_{MAX}} \quad (10)$$

d_{α}^{norma}repre posiela normalizovanú vzdialenosť k objektu pod uhlom α. d_{α}vyjadruje mzabezpečená hodnota vzdialenosť k objektu. d_{MAX}predstavuje maximálnu dĺžku lúča. ned takto: Ton d_{MAX}je defi

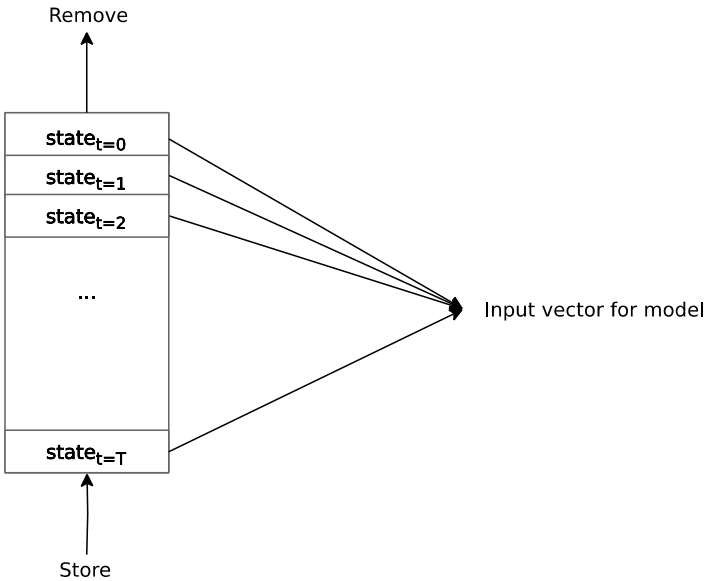
$$d_{MAX} = 0{,}8 * Obrazovka_w - Hráč_w \quad (11)$$

$Hráč_w$re predstavuje šírku agenta. Nasledujúce miery sú uvedené v t má šírku 34 pixels. Agen a výšku 24. $Obrazovka_w$predstavuje šírku een je viditeľná časť herného sobrazovka. Scr prostredia, ktorú možno vidieť, keď d. Šírka obrazovky je 288 a výška 512. game sa vykresľuje

2.5. Epizodická pamäť

Stavový priestor agenta pozostáva z okna s pevnou dĺžkou meraní z histórie časových krokov. Preto je potrebné vytvoriť pamäť na ukladanie týchto meraní počas hernej epizódy. Celý obsah tejto pamäte slúži ako vstupný vektor pre pohybový transformátor. Dátová štruktúra first-in-first-out (FIFO) zabezpečuje tok informácií jedným smerom, vyjadrujúci plynutie času v hernom prostredí. Keď sa počas epizódy získavajú nové merania, nahradia tie najstaršie v poradí. Na začiatku každej epizódy sa front inicializuje s počiatočným stavom prostredia. Zatiaľ čo podobnosti sú pozorované v zamýšľanom výsledku v porovnaní so skladacími rámami Atari na úrovni kanála [36], súčasný prístup zavádza novú dimenziu časového kroku vo vstupnom vektore. To umožňuje modelu využívať časové vzťahy medzi meraniami. Veľkosť pamäte určuje, ako ďaleko dozadu dokáže model efektívne analyzovať namerané stavy v miestnej histórii. Nedostatočná kapacita pamäte môže brániť dostupnosti informácií a brániť efektívnej predikcii akcií. Naopak príliš veľká pamäť zbytočne vyčerpáva výpočtové zdroje.

Obrázok6ilustruje princíp aplikovanej epizodickej pamäte. Nový stav prichádza na koniec radu zdola. Najstarší štát odchádza od začiatku frontu, tj horná časť. Vstup do pohybového transformátora predstavuje všetky položky, ktoré sú uložené vo fronte a sú zoradené tak, ako prichádzajú.



Obrázok 6. Architektúra epizodickej pamäte.

2.6. Súkromná zóna okolo agenta

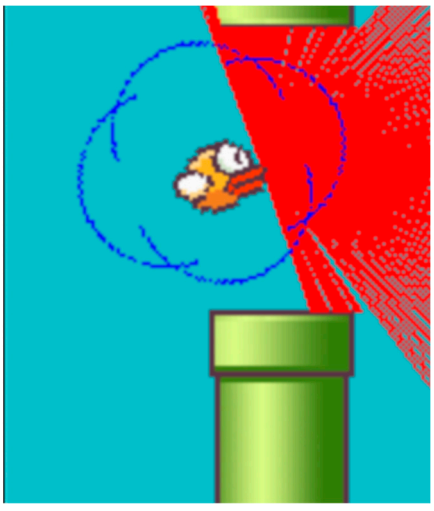
Pri pokusoch sa zistilo, že činidlo má tendenciu riskovať a pri prechode cez medzeru medzi rúrkami sa pohybuje príliš blízko k okrajom potrubia. V politike existuje možnosť penalizácie agenta za rizikové správanie. Preto bol zavedený trest za prekážku približujúcu sa dovnútra privátnej zóny agenta. S touto penalizáciou je agent motivovaný nájsť optimálne riešenie pre daný problém s ohľadom na jeho blízkosť k rozpoznaným prekážkam. V aplikácii v reálnom svete by rozpoznávanie objektov zvyčajne zahŕňalo vyhradenú hlbokú neurónovú sieť, ktorej cieľom je rozlišovať medzi prekážkami a požadovanými objektmi, ako sú jedlo alebo mince, v iných herných scenároch [37].

Agent si musí pri prechádzaní cez prekážky udržiavať bezpečnú vzdialenosť, aby zabezpečil, že jeho politika nebude riskantná. Táto vzdialenosť môže byť experimentálne určená nájdením optimálneho polomeru pre súkromnú zónu, ktorá je znázornená kruhom. Kruhový model je zvolený z dôvodu získavania údajov snímača vo formáte kruhovej polárnej mriežky. Keďže lúče sú vyžarované z povrchu agenta, stred kruhu súkromnej zóny sa musí zhodovať so stredom senzora. V prípade, že simulácia obsahuje prekážky a objekty, s ktorými môže agent musieť interagovať, ako napríklad zberateľské predmety, je potrebné dynamicky definovať túto súkromnú zónu. Proces klasifikácie prekážok vs. zberateľské predmety môže byť zložitý a zahŕňa sofistikované metódy [38,39] a udržiavanie identifikovaného objektu medzi po sebe nasledujúcimi meraniami môže vyžadovať špecializované metódy [40,41]. V súčasnom hernom prostredí, kde existujú iba prekážky, však stačí jednoduchá klasifikácia.

$$r = \frac{MAX(Hráč_w, Hráč_h) + x}{2} \quad (12)$$

Polomer kruhu súkromnej zóny je definovaný ako r, kde $Hráč_w$ predstavuje šírku agenta a $Hráč_h$ predstavuje výšku agenta. Hyperparameter x určuje veľkosť súkromnej zóny. Keďže lúče vyžarujú z okrajov činiteľa a nie jeho stredu, keď x je nastavený na 0, polomer súkromnej zóny sa rovná polovici maximálneho rozmeru agenta.

Obrázok 7 ilustruje privátnu zónu agenta, kde je parameter x je nastavená na 30. Medzera medzi rúrkami je pevne stanovená na veľkosť 100 jednotiek (tj pixelov) a šírka každej rúry meria 52 jednotiek. Ako je vidieť, vysoká hodnota x penalizuje agenta, ak sa pokúsi prejsť cez medzeru medzi potrubiami. Naopak, an x príliš nízka hodnota zmenšuje existenciu súkromnej zóny, čo vedie agenta k zvýšenému riziku.



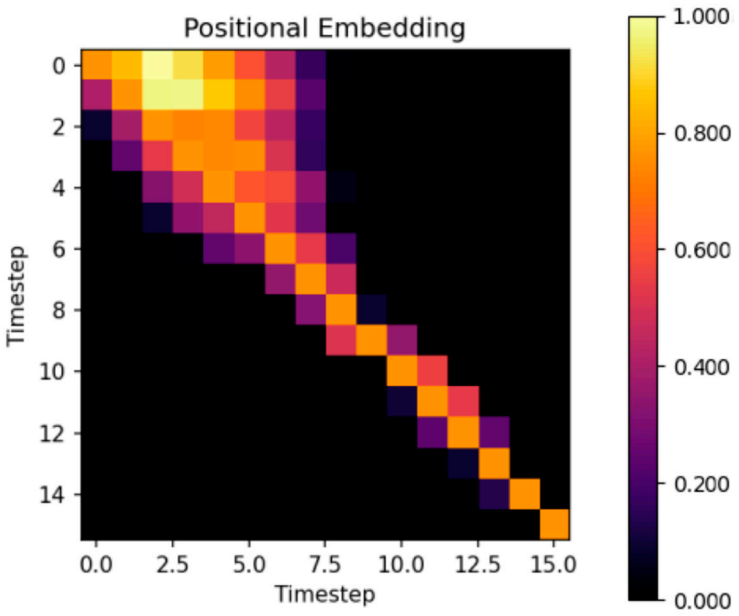
Obrázok 7. Súkromná zóna agenta.

3. Výsledky

Aby sa zlepšil výkon hlbokej neurónovej siete ovládajúcej vyhýbanie sa prekážkam Flappy Bird, rôzne techniky si vyžadovali jemné ladenie. To zahŕňalo výber správnej architektúry riadiaceho systému a algoritmických techník, ako aj výber vhodných hyperparametrov počas implementácie. Nasledujúca časť poskytuje podrobné vysvetlenie kľúčových aspektov tohto procesu.

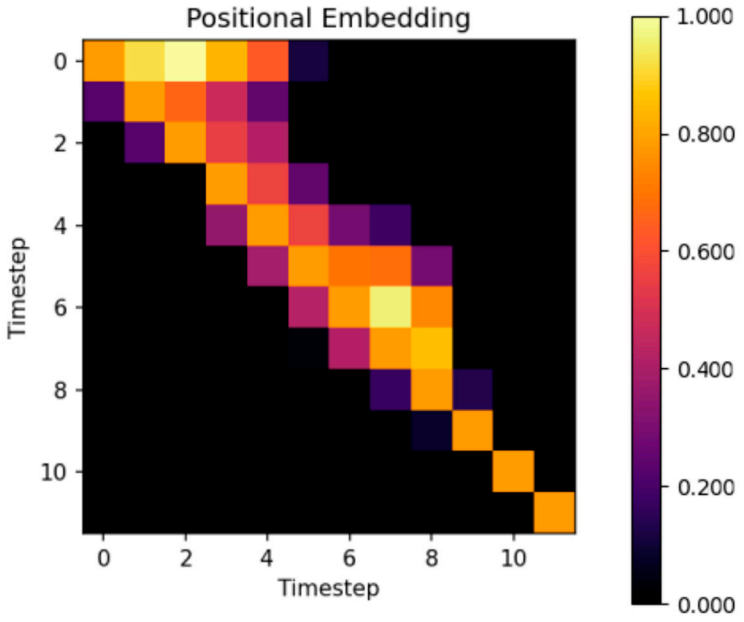
Jeden aspekt zahŕňal optimalizáciu počtu časových krokov uchovávaných v epizodickej pamäti. To určilo, do akej miery si agent vybaví krátkodobú históriu a využije ju vo svojich akčných predpovediach. Okrem toho sa štúdia zamerala na zdokonalenie architektúry modelu. Konkrétne sa skúmalo, či bolo efektívne použiť posledný časový krok výstupnej série akcií. Alternatívy zahŕňali globálny priemer alebo globálne maximálne združovanie. Tieto operácie zahŕňajú výpočet priemeru alebo maxima funkcií naprieč osou časového kroku. Tieto metódy sa bežne používajú pri redukčných úlohách, ako je vidieť v transformátoroch videnia a konvolučných neurónových sieťach. Nakoniec bola pozornosť zameraná na určenie optimálnej veľkosti súkromnej zóny s možnosťami nastavenými na 30, 15 a 0.

Prvá testovaná konfigurácia používala 16 časových krokov ako veľkosť epizodickej pamäte. Obrázok8 ukazuje kosínusovú podobnosť medzi vloženiami pre rôzne páry časových krokov. Najbližšie podobnosti sú v oblasti ľavého horného rohu tepelnej mapy. Preto sa nasledujúci experiment zameral na zníženie počtu časových krokov, aby sa znížila hustota podobností medzi časovými krokmi a aby sa spresnil optimálny počet časových krokov. Keďže medzi počiatočnými časovými krokmi existuje podobnosť, je možné obmedziť ich počet. Celkovo sa použilo 12 časových krokov, od ktorých sa očakávalo zníženie hustoty podobností, najmä v ľavom hornom rohu.



Obrázok 8. Podobnosť vložených časových krokov medzi 16 rôznymi časovými krokmi.

Druhý experiment mal použiť iba 12 časových krokov. Kosínusová podobnosť medzi časovými krokmi je znázornená na obrázku9. Na rozdiel od použitia 16 časových krokov sa hustota podobností vložených časových krokov v ľavom hornom rohu znížila, ale skóre agenta sa výrazne nezhoršilo. Obrázok9nie je len podmnožinou obrázku8; rozdiel v hustote podobností je zjavný. Okrem toho distribúcia podobnosti nie je dokonale symetrická pozdĺž uhlopriečky, pričom minulé kroky vykazujú väčšiu podobnosť s budúcimi krokmi, najmä pre vzdialené časové rámce. Tento trend sa však v posledných časových krokoch nepozoruje. Na základe týchto pozorovaní by v budúcich experimentoch mohlo byť prospešné preskúmať použitie menšieho počtu minulých časových krokov, pretože vykazujú podobnosti s budúcimi krokmi. Časové kroky menšie ako 12 alebo vyššie ako 16 neboli v tejto štúdii testované.



Obrázok 9. Podobnosť vložených časových krokov medzi 12 rôznymi časovými krokmi.

V našom skúmaní sme zistili, že so zvyšujúcim sa časovým krokom sa podobnosť vložení znižuje. Tento trend je zrejmý najmä v konečnom časovom kroku, bez ohľadu na to, či je nakonfigurovaných 16 alebo 12 časových krokov. Markovov rozhodovací proces sa zvyčajne aplikuje iba na aktuálny stavs_t. To znamená, že najunikátnejším časovým krokom musí byť posledný časový krok, čo bolo podporené aj meraním s konfiguráciou 16 aj 12 časových krokov. Konkrétne posledný časový krok vykazuje najvyššiu podobnosť iba vo vzťahu k sebe samému. V tomto článku je upravený typický Markovov rozhodovací proces. Historický stav a súčasný stav sa používajú súčasne$s_{t-N:t}$, s výnimkou prvého štátu$s_t$kvôli nedostatku existujúcej histórie. Niektoré historické stavy môžu mať pre agenta pravdepodobne podobný význam. Táto úprava prináša paralely s vkladaním slov, kde slová s podobným významom majú vyššiu pozitívnu kosínusovú podobnosť, ale na druhej strane slová s veľmi odlišným významom majú malú kosínusovú podobnosť blízko nule [42].

V nasledujúcich meraniach sa pri porovnaní posledného časového kroku, globálneho priemeru a globálneho maximálneho súhrnu použili údaje zozbierané z 500 epizód. Priemerné a maximálne skóre v rámci epizód sa meralo pre deterministického, vopred vyškoleného agenta. Skóre predstavuje počet potrubí, ktorými agent úspešne prešiel.

Tabuľka1ukazuje výsledky porovnaní medzi rôznymi redukčnými technikami. Priemer prvkov pozdĺž osi časového kroku je výrazne lepší ako pri iných prístupoch.

Tabuľka 1.Výsledky testovaných redukčných metód.

Architektúra	Časové kroky	Najvyššie skóre	Priemerné skóre
Globálne priemerné združovanie	16	**2970**	324,198
Posledný časový krok	16	2809	286,394
Globálne maximálne združovanie	16	1948	329,194
Globálne priemerné združovanie	12	2348	**380,284**
Posledný časový krok	12	1922	335,114
Globálne maximálne združovanie	12	1128	152,858

Najvyššie skóre je zvýraznené tučným písmom.

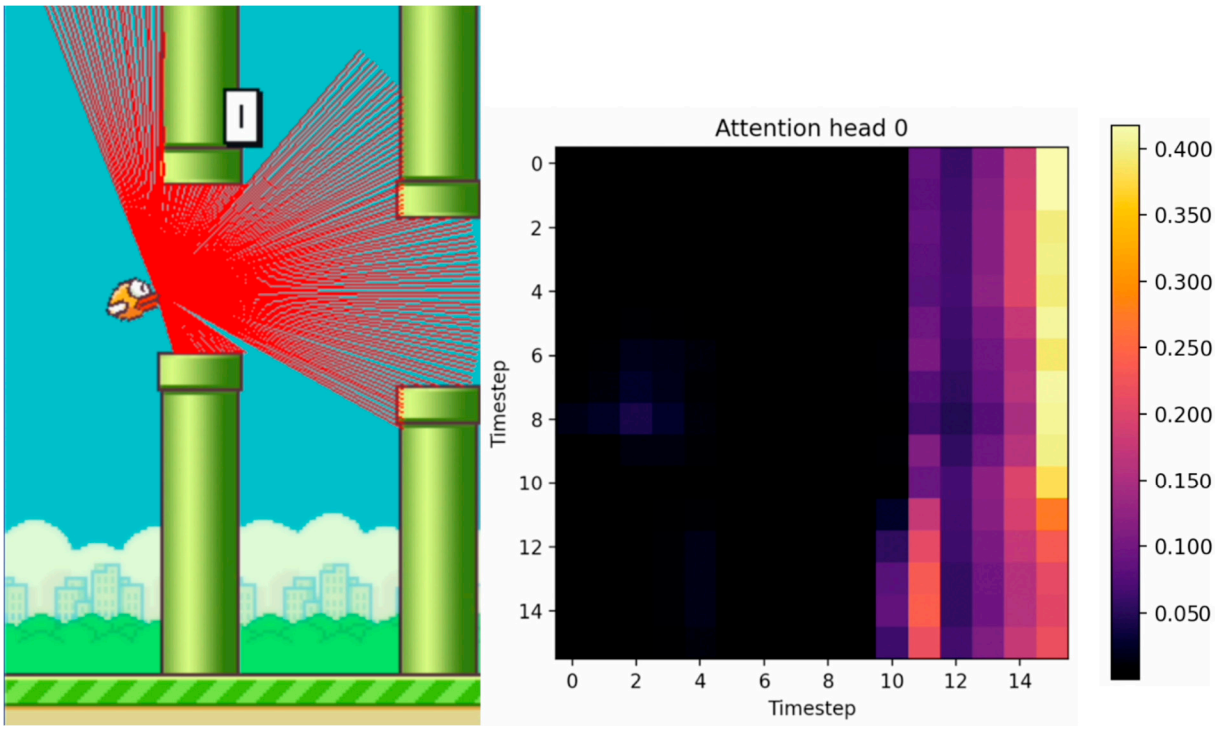
Tabuľka2uvádza porovnanie najlepších výsledkov dosiahnutých v najvyššom a priemernom skóre v tomto príspevku na rozdiel od iných prác. Tento dokument má výrazne lepšie skóre.

Tabuľka 2 Popis.

Papier	Najvyššie skóre	Priemerné skóre
[43]	15	3 300
[4]	80	16 400
[6]	215	82,200
[5]	-	102,170
[7]	1491	209,298
Tento papier bez súkromnej zóny	**2970**	**380,284**
Tento papier so súkromnou zónou	**74,755**	**13 156,590**

Skóre získané v tomto článku je zvýraznené tučným písmom.

Figúrky10–12zobraziť sledovanie posúvajúcich sa potrubí pozdĺž časovej osi. Agent používa históriu zo 16 časových krokov. Odkaz na video zobrazujúce animáciu meniacej sa matice pozornosti spolu s meniacim sa prostredím je uvedený v doplnkových materiáloch.

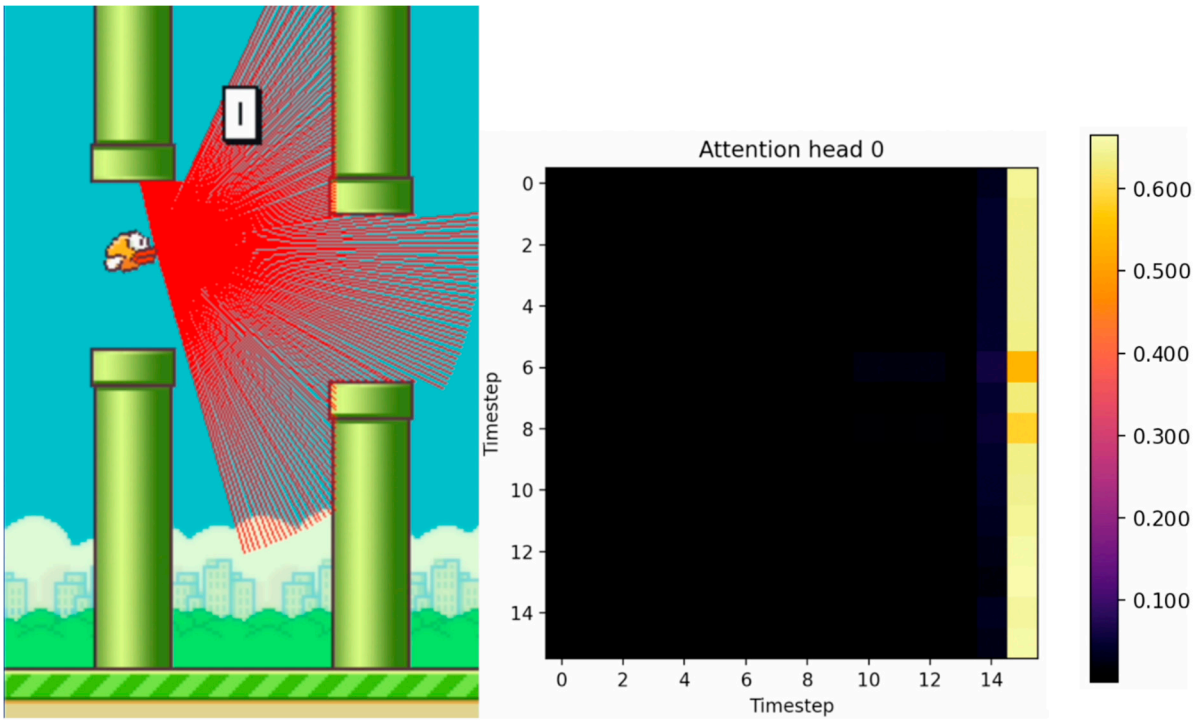


Obrázok 10. Agent so 16 časovými krokmi vstupujúci do medzery medzi hornou a spodnou rúrou. (Čím jasnejšia farba, tým vyššia hodnota pozornosti).

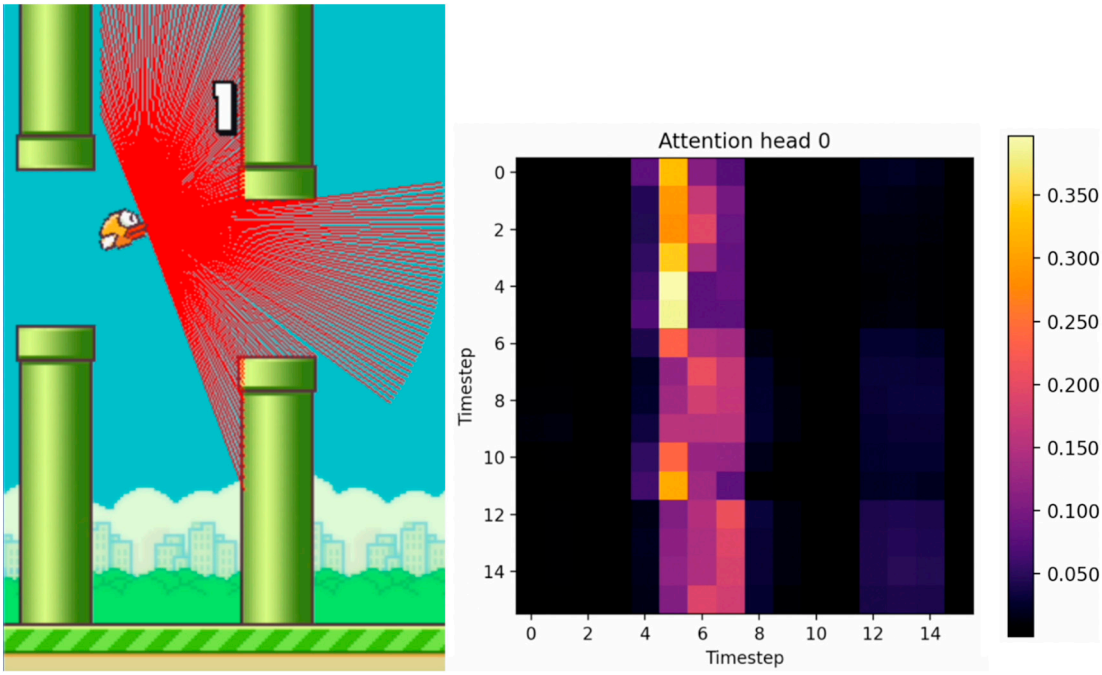
Z analýzy politiky agenta je zrejmé, že agent riskuje a pri prechode cez medzeru medzi rúrami sa približuje k hornému alebo dolnému potrubiu. Na vyriešenie tohto problému je potrebné určiť pre agenta zónu, za ktorou v prípade zistenia prekážok bude agent potrestaný −0,5.

Rovnako tak to isté −0,5 pokuta stanovená v [5] pre agenta, ktorý dosiahne hornú časť obrazovky, sa aplikuje aj na prekážky v súkromnej zóne agenta. V tomto hernom prostredí sú všetky objekty považované za prekážky.

Naopak, ak agent udržiava vzdialenosť od prekážok nad kritickým prahom, je odmenený odmenou „stále naživе“ v hodnote +0,1, podobne ako [44].



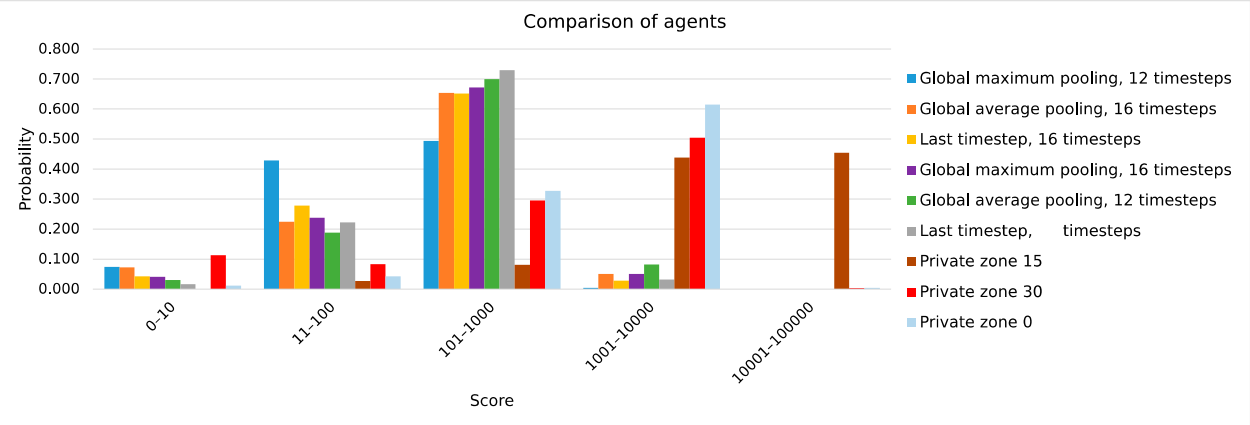
Obrázok 11. Agent so 16 časovými krokmi v medzere medzi horným a spodným potrubím. (Čím jasnejšia farba, tým vyššia hodnota pozornosti).



Obrázok 12. Agent so 16 časovými krokmi prešiel medzerou medzi horným a spodným potrubím. (Čím jasnejšia farba, tým vyššia hodnota pozornosti).

Obrázok13zobrazuje histogram skóre agenta v rôznych technikách redukcie funkcií a veľkostiach súkromných zón. Experimenty zahŕňajúce rôzne metódy redukcie funkcií nezahŕňali penalizáciu vo funkcii odmeny za priblíženie sa k prekážkam príliš blízko. Medzitým experimenty s rôznymi veľkosťami súkromných zón využívali globálne priemerné združovanie so 16 časovými krokmi na redukciu funkcií. Pri porovnaní použitia globálneho maximálneho združovania s globálnym priemerným združovaním je zrejmé, že agent má vyššiu pravdepodobnosť

skóre pod 10 pri použití predchádzajúcej metódy. Hlboká sieť Q vo všeobecnosti nadhodnocuje predpovedané hodnoty Q [45]. V dôsledku toho môže použitie globálneho maximálneho združovania viesť k nadhodnoteniu hodnôt Q a politike, ktorá je pre agenta náchylnejšia na riziko.



Obrázok 13. Histogram skóre.

V tejto štúdii sa pozorovalo, že keď sa použil iba posledný časový krok, podobný tokenu triedy v transformátore videnia [46], globálne priemerné združovanie fungovalo podobne ako globálne maximálne združovanie.

Najstabilnejšie ovládanie Flappy Bird spomedzi testovaných možností rysovej červenej podanie bolo dosiahnuté prostredníctvom metódy globálneho priemerného znižovania. Poskytla h najvyššia maximálne skóre a priemerné skóre v porovnaní s inými metódami zníženia vlastností ión. In na rozdiel od globálneho maximálneho združovania, globálne priemerné združovanie zaťažuje aktivát ión podľa kombinácia maximálnych a nemaximálnych aktivácií [47]. Toto správanie vedie k zníženiu vstúpiť nadhodnotenie Q-hodnôt predpovedaných modelom a menej riziková politika pre agent.

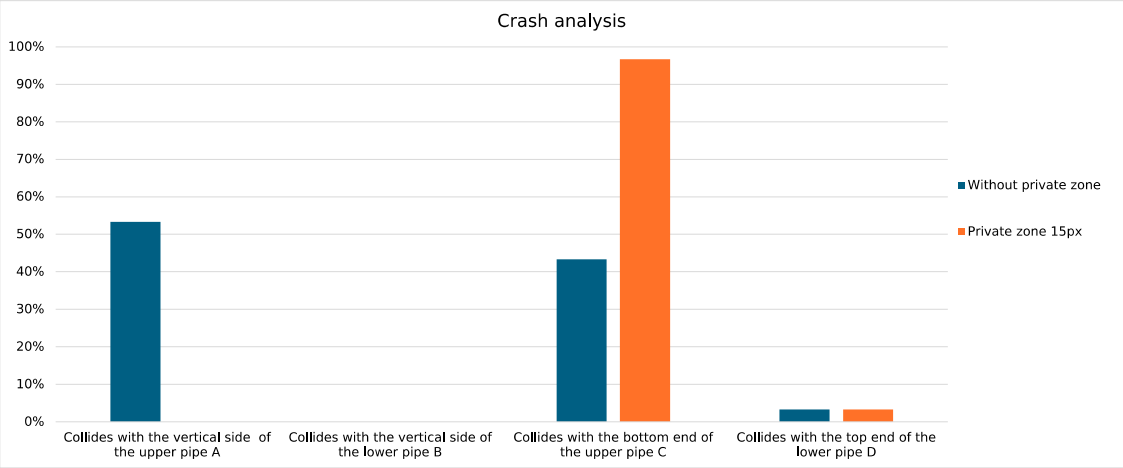
Zistilo sa, že výsledkom je optimálna veľkosť súkromnej zónye d v agentovi ach eving skóre, ktoré bolo mnohonásobne vyššie. Pravdepodobnosť získania skóre menšieho th 100 bola extrémne nízka. Okrem toho bola pozorovaná vysoká pravdepodobnosť získania vyššieho skóre, n 1000 ako bolo pozorované v porovnaní s agentmi bez súkromnej zóny.

Tabuľka3predstavuje porovnanie rôznych veľkostí súkromných zón. Z niekoľkých sekcia možností výsledky naznačujú optimálnu veľkosť súkromnej zóny 15. Nadmerné y veľké hodnoty veľkostí súkromných zón by tiež penalizovali agenta za prelet cez pí peline medzera, kým neprejde svojim stredom, čo sa počíta ako vysoká kladná odmena +1,0 s vylúčením ostatných hodnôt funkcie odmeny [48].

Tabuľka 3. Porovnanie skóre s rôznymi veľkosťami súkromných zón.

Súkromná zóna	Najvyššie skóre	Priemerné skóre
žiadne	2970	380,284
0	10 250	2138,858
15	**74,755**	**13 156,590**
30	11,383	1645,654

Obrázok14predstavuje analýzu zrážky pre kolízie Flappy Bird bez súkromnej zóny as optimálnou veľkosťou súkromnej zóny. Zatiaľ čo skóre v súkromnej zóne je o niekoľko rádov lepšie, analýza rozdrvenia ukazuje, že stále existuje priestor na zlepšenie. Robustné riešenie by malo mať rovnakú pravdepodobnosť zasiahnutia potenciálnych prekážok, pričom výsledky ukazujú, že Flappy Bird má tendenciu naraziť takmer výlučne do spodného konca hornej rúry. Zavedením súkromnej zóny sa minimalizovali potenciálne miesta dopadu, čo umožňuje budúce zameranie sa na potlačenie týchto kolízií. Jedným z prístupov k dosiahnutiu tohto cieľa je navrhnúť robustnejšiu funkciu odmeňovania.



Obrázok 14. Analýza zrútenia so súkromnou zónou a bez nej.

Tabuľka4zobrazuje hyperparametre použité vo všetkých experimentoch perfo rmed. ich hodnoty sú nastavené na základe kombinácie odporúčaných nastavení. Odporúčanie opravená veľkosť vyrovnávacej pamäte pre prehrávanie a diskontný faktor sú prevzaté z [49]. Multi kliešť z Dimenzia bloku MLP, typ rozvrhu rýchlosti učenia a prechodový klip g sú založené dňa [50]. Súkromná zóna s hodnotou Žiadne označuje neprítomnosť pera alty pravidlo pre približujúceho sa k prekážkam vo funkcii odmeňovania. Ostatné číselné hodnoty súkromný veľkosť zóny vyjadruje x rovnice (12).

Tabuľka 4. Hyperparametre modelu agenta a študenta.

Hyperparameter	Popis	Hodnota
prístav	Port databázového servera	8000
max_replay_size	Maximálna databázová pamäť Pomer	1 000 000
sample_per_insert	vzoriek na vloženie pre reverb Počiatočná	32
temp_init	Boltzmannova teplota na prieskum	0,500
temp_min	Minimálna Boltzmannova teplota	0,010
temp_decay	Pokles Boltzmannovej teploty Kroky	0,999999
warmup_steps	zahrievania pre rýchlosť učenia kosínusový plánovač	1000
vlak_kroky	Tréningové kroky	1 000 000
veľkosť_dávky	Veľkosť dávky	256
gama	Diskontný faktor	0,990
tau	Faktor Tau (pre model EMA)	0,005
počet_vrstiev	Num. blokov kódovača	2
embed_dim	Rozmer vloženia	128
ff_mult	Násobiteľ rozmeru bloku MLP	4
num_heads	Num. hláv pozornosti	6
miera_učenia	Miera učenia	3×10^{-4}
global_clipnorm	Globálne normalizované orezanie gradientu	1
úbytok hmotnosti	Úbytok hmotnosti pre optimalizátor AdamW	1×10^{-4}
frame_stack	Veľkosť krátkodobej (epizodickej) pamäte	16 alebo 12
player_private_zone	Veľkosť súkromnej zóny agenta	Žiadne, 0, 15 alebo 30

4. Diskusia

Využitie transformátorovej neurónovej siete na ovládanie simulovaného agenta prostredníctvom vrhania lúčov ako jednoduchý senzor LIDAR má potenciálne rôznorodé aplikácie v niekoľkých doménach. Technológia diaľkového snímania integrovaná s pokročilým ovládaním na základe AI môže byť prospešná v nasledujúcich kontextoch:

Vo virtuálnej realite a hrách môžu avatari alebo postavy ťažiť z prirodzenejšej a citlivejšej interakcie.

Metóda na zlepšenie navigácie pomocou odlievania lúčov v 2D by sa mohla potenciálne rozšíriť tak, aby využívala skutočný LIDAR v 3D priestore. V budúcnosti by to mohlo viesť k pokroku v robotike; autonómne vozidlá, ako sú samoriadiace autá, drony alebo akékoľvek mobilné roboty, ktoré vyžadujú efektívnu navigáciu; a schopnosti vyhýbať sa prekážkam. V oblastiach postihnutých katastrofou môžu takéto roboty pomáhať pri pátracích a záchranných misiách. Pokročilí agenti môžu zefektívniť úlohy, ako je riadenie zásob a manipulácia s materiálom v skladoch. Okrem toho by robotické ramená mohli lepšie manipulovať s objektmi v dynamických prostrediach.

V každom z týchto kontextov by integrácia vrhania lúčov a riadenia neurónovej siete transformátora mala umožniť agentovi robiť informované rozhodnutia na základe časových a priestorových informácií.

Pri zvažovaní výberu algoritmických postupov a hyperparametrov existuje široký priestor na skúmanie a experimentovanie s rôznymi možnosťami.

Pri ukladaní vysokorozmerných stavov do epizodickej pamäte by bolo vhodnejšie extrahovať iba dôležité funkcie na ukladanie. Na tento účel by sa na kompresiu vstupného vektora mohol použiť model AutoEncodertype.

Ďalšou úvahou je inicializácia epizodickej pamäte. V súčasnosti duplikuje počiatočný stav, ale jedna alternatíva zahŕňa vytvorenie vloženia pre prázdny stav obsahujúci epizodickú pamäť na začiatku každej epizódy. Ďalšou možnosťou je dynamicky upravovať počet časových krokov vzhľadom na vstup do pohybového transformátora a zároveň zabezpečiť správne priradenie polohového vloženia pre inkrementujúce stavy z časových krokov.

Sľubným smerom výskumu je skúmanie vplyvu kosínusovej podobnosti na optimálny počet časových krokov. To zahŕňa skúmanie, či podobnosť vložených časových krokov môže znížiť potrebný počet časových krokov. Je potrebná ďalšia štúdia na overenie účinku pozičného zabudovania pri skracovaní časových krokov v rôznych herných prostrediach.

V prípade hry Flappy Bird by budúci výskum mal vyskúšať aj možnosť pridania váženej odmeny za to, že sa bude držať bezpečnejšej vzdialenosti od hornej rúry viac ako od iných prekážok. Tento smer výskumu vyplýva z výsledkov analýzy chýb. Predmetom ďalšieho výskumu je tiež štúdium a náprava porúch po tom, čo agent vykonal veľmi veľký počet krokov v prostredí. Potenciálne zlepšenia možno očakávať v algoritmoch založených na princípe hlbokého Q učenia, ako je napríklad dvojité hlboké učenie Q alebo dvojité hlboké učenie Q. Mala by sa tiež skontrolovať numerická nestabilita, ako aj pokročilejšie modely typu herca a kritika, ako sú A2C alebo PPO.

Ďalšie vylepšenia by sa dali dosiahnuť vytvorením dynamickej súkromnej zóny okolo agenta. Súkromnú zónu je možné vymedziť predikciou hlbokej neurónovej siete, či objekty prekračujúce hranicu súkromnej zóny sú prekážkami alebo pomôckami pri dosahovaní konkrétnej úlohy. Takýto model by mohol priamo upraviť komplexnú funkciu odmeňovania potrebnú na dokončenie úlohy bez toho, aby bol agent vystavený rizikovému správaniu.

5. Závery

Naša štúdia predstavuje nové riadenie navádzania pomocou senzorov LIDAR reprezentovaných metódou vrhania lúčov na detekciu prekážok a navigáciu agentov v prostrediach naplnených prekážkami. Navrhnutý model pohybového transformátora efektívne zachytil časovú dynamiku medzi hodnotami senzorov. Zistenia demonštrujú schopnosť modelu adaptívne reagovať na pohyb agenta medzi potrubiami, ako sa odráža v matici pozornosti. Mechanizmus pozornosti modelu uprednostňuje minulé alebo súčasné dáta senzorov alebo ich kombináciu na základe priestorového rozloženia potrubí v okolí. Okrem toho výsledky ukazujú, že použitie techník priemernej redukcie pomáha zmierniť riziko nadhodnotenia hodnôt Q. Okrem toho začlenenie súkromnej zóny pre agenta prispieva k formulovaniu menej riskantnej navigačnej politiky.

V tomto dokumente je priemerné skóre (počet prechodov cez medzery v potrubí) získané agentom bez súkromnej zóny o 182 percent lepšie v porovnaní s najlepšími výsledkami získanými konkurentmi. Najvyššie skóre dosiahnuté agentom bez súkromnej zóny v porovnaní s výsledkami najlepších konkurentov je o 199 percent lepšie. Agent

so súkromnou zónou 15 pixelov dosiahol priemerné skóre, ktoré bolo o 6286 percent lepšie ako priemerné skóre agentov najlepších konkurentov a maximálne skóre, ktoré bolo o 5014 percent lepšie ako najlepšie výsledky konkurencie z hľadiska maximálneho skóre agentov.

Doplnkové materiály: Nasledujúce podporné informácie si môžete stiahnuť na nasledujúcej adrese: interaktívne tabuľky:https://wandb.ai/markub/rl-toolkit/groups/FlappyBird-v0(prístup 16. októbra 2023); zdrojové kódy:https://github.com/markub3327/rl-toolkit(prístup 16. októbra 2023) a https://github.com/markub3327/flappy-bird-gymnasium(prístup 16. októbra 2023); YouTube video: https://youtu.be/aZQxuDCyHoI(prístup 22. decembra 2023).

Autorské príspevky: Konceptualizácia, IDL a JP; metodika, MK; softvér, MK; validácia, MK; formálna analýza, JP; vyšetrovanie, MK; zdroje IDL; správa údajov, MK; písanie príprava originálu návrhu, MK; písanie – recenzia a úprava, JP; vizualizácia, MK; supervízia, IDL a JP; administrácia projektov, IDL; získanie financovania, IDL Všetci autori si prečítali a súhlasili s publikovanou verziou rukopisu.

Financovanie: Tento výskum bol financovaný Kultúrnou a vzdelávacou grantovou agentúrou MŠVVaŠ SR, číslo grantu KEGA 020UCM-4/2022, a projektom Erasmus+ FAAI: Budúcnosť je v aplikovanej umelej inteligencii—2022-1-PL01-KA220-HED-000088359, prac. balík WP4.

Vyhlásenie inštitucionálnej revíznej rady: Neuplatňuje sa.

Vyhlásenie informovaného súhlasu: Neuplatňuje sa.

Vyhlásenie o dostupnosti údajov:Údaje sú obsiahnuté v článku.

Konflikty záujmov: Autori nedeklarujú žiadny konflikt záujmov.

Referencie

1. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, AN; Kaiser, Ł.; Polosukhin, I. Pozornosť je všetko, čo potrebujete. *Adv. Neural Inf. Proces. Syst.***2017**,*30*, 5998-6008. Dostupné online:https://proceedings.neurips.cc/paper_files/paper/2017 /file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf(prístup 10. decembra 2023).
2. Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Sú transformátory účinné na predpovedanie časových radov?*Proc. AAAI Conf. Artif. Intell.***2023**,*37*, 11121–11128. Dostupné online:https://ojs.aaai.org/index.php/AAAI/article/view/26317/26089(prístup 10. decembra 2023). [CrossRef]
3. Wei, S. Posilňovacie vzdelávanie pre zlepšenie hry Flappy Bird.*Zdôrazňuje Sci. Ing. Technol.***2023**,*34*, 244-249. Dostupné online: https://drpress.org/ojs/index.php/HSET/article/download/5479/5298(prístup 10. decembra 2023). [CrossRef]
4. Pilcer, LS; Hoorelbeke, A.; Andigne, AD Hranie Flappy Bird s hlbokým posilňovaním učenia.*IEEE Trans. Neurónová sieť***2015**,*16*, 285-286. Dostupné online:https://www.researchgate.net/profile/Louis-Samuel-Pilcer/publication/324066514_Playing_Flappy_Bird_with_Deep_Reinforcement_Learning/links/5abbc2230f7e9bfc045592Relapinwith-Bepird-Learning-F (prístup 10. decembra 2023).
5. Yang, K. Použitie DQN a Double DQN na hranie Flappy Bird. In Proceedings of the 2022 International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID 2022), Si-an, Čína, 15. – 17. apríla 2022; Atlantis Press: Amsterdam, Holandsko, 2022; s. 1166–1174. Dostupné online:https://www.atlantis-press.com/article/125977189.pdf(prístup 10. decembra 2023).
6. Chen, K. Deep Reinforcement Learning for Flappy Bird. CS 229 Záverečné projekty strojového učenia. 2015. Dostupné na internete: https://cs229.stanford.edu/proj2015/362_report.pdf(prístup 10. decembra 2023).
7. Vu, T.; Tran, L. FlapAI Bird: Školenie agenta na hranie Flappy Bird pomocou techník posilňovania.*arXiv***2020**arXiv:2003.09579.
8. Li, J.; Yin, Y.; Chu, H.; Zhou, Y.; Wang, T.; Fidler, S.; Li, H. Učíme sa vytvárať rôzne tanečné pohyby s transformátorom.*arXiv* **2020**, arXiv:2008.08171.
9. Shi, S.; Jiang, L.; Dai, D.; Schiele, B. Pohybový transformátor s globálnou lokalizáciou zámeru a lokálnym spresnením pohybu.*Adv. Neural Inf. Proces. Syst.***2022**,*35*, 6531-6543. Dostupné online:https://proceedings.neurips.cc/paper_files/paper/2022/file/2ab47c960bfee4f86dfc362f26ad066a-Paper-Conference.pdf(prístup 10. decembra 2023).
10. Hu, M.; Zhu, X.; Wang, H.; Cao, S.; Liu, C.; Song, Q. STDFormer: Priestorovo-temporálny pohybový transformátor pre sledovanie viacerých objektov.*IEEE Trans. Circuits Syst. Video Technol.***2023**,*33*, 6571-6594. Dostupné online:https://ieeexplore.ieee.org/iel7/76/4358651/10091152.pdf(prístup 10. decembra 2023). [CrossRef]
11. Esslinger, K.; Platt, R.; Amato, C. Deep Transformer Q-Networks pre čiastočne pozorovateľné učenie zosilnenia.*arXiv***2022** arXiv:2206.01078.
12. Meng, L.; Goodwin, M.; Yazidi, A.; Engelstad, P. Hlboké posilňovanie pomocou Swin Transformer.*arXiv***2022**arXiv:2206.15269.

13. Chen, L.; Lu, K.; Rajeswaran, A.; Lee, K.; Grover, A.; Laskin, M.; Abbeel, P.; Srinivas, A.; Mordatch, I. Rozhodovací transformátor: Posilnenie učenia prostredníctvom sekvenčného modelovania.*Adv. Neural Inf. Proces. Syst.***2021**,*34*, 15084–15097. Dostupné online: https://proceedings.neurips.cc/paper_files/paper/2021/file/7f489f642a0ddb10272b5c31057f0663-Paper.pdf(prístup 10. decembra 2023).

14. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; a kol. Obrázok stojí za to 16×16 slov: Transformátory na rozpoznávanie obrazu v mierke.*arXiv***2020**arXiv:2010.11929.
15. Liu, R.; Ji, C.; Niu, J.; Guo, B. Výskum metódy detekcie narušenia založenej na 1D-ICNN-BiGRU. In*Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2022; Zväzok 2347, s. 012001. Dostupné na internete:https://iopscience.iop.org/article/10.1088/1742-6596/2347/1/012001/pdf(prístup 10. decembra 2023).
16. Crocioni, G.; Pau, D.; Delorme, JM; Gruosso, G. Odhad parametrov Li-ion batérií s malými neurónovými sieťami zabudovanými do inteligentných mikrokontrolérov internetu vecí.*Prístup IEEE***2020**,*8*, 122135–122146. Dostupné online:https://ieeexplore.ieee.org/iel7/6287639 /6514899/09133084.pdf(prístup 10. decembra 2023). [CrossRef]
17. Gholamalinezhad, H.; Khosravi, H. Pooling Methods in Deep Neural Networks, a Review.*arXiv***2020**, arXiv:2009.07485.
18. Anders, K.; Winiwarter, L.; Lindenbergh, R.; Williams, JG; Vos, SE; Höfle, B. 4D objekty podľa zmeny: Časopriestorová segmentácia zmeny geomorfného povrchu z časového radu LiDAR.*ISPRS J. Photogramm. Remote Sens.***2020**,*159*, 352-363. Dostupné online: https://www.sciencedirect.com/science/article/pii/S0924271619302850(prístup 10. decembra 2023). [CrossRef]
19. Wang, Z.; Schaul, T.; Hessel, M.; Hasselt, H.; Lanctot, M.; Freitas, N. Súbojové sieťové architektúry pre hlboké posilňovanie učenia. In Proceedings of the International Conference on Machine Learning, PMLR, New York City, NY, USA, 19. – 24 June 2016; s. 1995–2003. Dostupné online:http://proceedings.mlr.press/v48/wangf16.pdf(prístup 10. decembra 2023).
20. Haarnoja, T.; Tang, H.; Abbeel, P.; Levine, S. Posilňovanie učenia sa zásadami založenými na hlbokej energii. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6. – 11. august 2017; s. 1352–1361. Dostupné online: http://proceedings.mlr.press/v70/haarnoja17a/haarnoja17a.pdf(prístup 10. decembra 2023).
21. Peng, B.; Sun, Q.; Li, SE; Kum, D.; Yin, Y.; Wei, J.; Gu, T. Kompletná autonómna jazda prostredníctvom dvojitej hlbokej siete Q. *Automot. Innov.***2021**,*4*, 328-337. Dostupné online:https://link.springer.com/content/pdf/10.1007/s42154-021-00151-3.pdf (prístup 10. decembra 2023). [CrossRef]
22. Liu, F.; Li, S.; Zhang, L.; Zhou, C.; Ye, R.; Wang, Y.; Lu, J. 3DCNN-DQN-RNN: Rámec učenia sa hlbokého posilnenia pre sémantickú analýzu rozsiahlych 3D mračien bodov. In Proceedings of the IEEE International Conference on Computer Vision, Benátky, Taliansko, 22. – 29. október 2017; IEEE: Piscataway, NJ, USA, 2017; s. 5678–5687. Dostupné online:https://openaccess.thecvf.com/ content_ICCV_2017/papers/Liu_3DCNN-DQN-RNN_A_Deep_ICCV_2017_paper.pdf(prístup 10. decembra 2023).
23. Saleh, RA; Saleh, AK Štatistické vlastnosti funkcie Log-Cosh Loss, ktorá sa používa v strojovom učení.*arXiv***2022**arXiv:2208.04564.

24. Tarvainen, A.; Valpola, H. Priemerní učitelia sú lepšími vzormi: Ciele konzistencie s priemerom hmotnosti zlepšujú výsledky hlbokého učenia sa čiastočne pod dohľadom.*Adv. Neural Inf. Proces. Syst.***2017**,*30*, 1195-1204. Dostupné online:https://proceedings.neurips.cc/paper/ 2017/file/68053af2923e00204c3ca7c6a3150cf7-Paper.pdf(prístup 10. decembra 2023).
25. Tummala, S.; Kadry, S.; Bukhari, SAC; Rauf, HT Klasifikácia mozgového nádoru zo zobrazovania magnetickou rezonanciou pomocou zostavy transformátorov videnia.*Curr. Oncol.***2022**,*29*, 7498-7511. Dostupné online:https://www.mdpi.com/1718-7729/29/10/590/ htm (prístup 10. decembra 2023). [CrossRef]
26. Wang, X.; Yang, Z.; Chen, G.; Liu, Y. A Reinforcement Learning Method of Solving Markov Decision Processes: Adaptive Exploration Model Based on Temporal Difference Error.*Elektronika***2023**,*12*, 4176. Dostupné na internete:https://www.mdpi.com/20 79-9292/12/19/4176(prístup 10. decembra 2023). [CrossRef]
27. Feng, H.; Yang, B.; Wang, J.; Liu, M.; Yin, L.; Zheng, W.; Yin, Z.; Liu, C. Identifikácia malígnych ultrazvukových obrazov prsníka pomocou náplasti ViT.*Appl. Sci.***2023**,*13*, 3489. Dostupné na internete:https://www.mdpi.com/2076-3417/13/6/3489(prístup 10. decembra 2023). [CrossRef]
28. Devlin, J.; Chang, MW; Lee, K.; Toutanova, K. Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**arXiv:1810.04805.
29. On, K.; Zhang, X.; Ren, S.; Sun, J. Hlboké zvyškové učenie pre rozpoznávanie obrazu. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27. – 30. júna 2016; IEEE: Piscataway, NJ, USA, 2016; s. 770–778. Dostupné online: https://openaccess.thecvf.com/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper. pdf(prístup 10. decembra 2023).
30. Hasan, F.; Huang, H. MALS-Net: Multi-head pozornosť-založená LSTM sekvenčná sieť pre modelovanie sociálno-časovej interakcie a predpovedanie trajektórie.*Senzory***2023**,*23*, 530. Dostupné na internete:https://www.mdpi.com/1424-8220/23/1 / 530/pdf(prístup 10. decembra 2023). [CrossRef]
31. Mogan, JN; Lee, CP; Lim, KM; Muthu, KS Gait-ViT: Rozpoznávanie chôdze pomocou Vision Transformer.*Senzory***2022**,*22*, 7362. Dostupné na internete:https://www.mdpi.com/1424-8220/22/19/7362/pdf(prístup 10. decembra 2023). [CrossRef]
32. Hendrycks, D.; Gimpel, K. Gaussian Error Linear Units (Gelus).*arXiv***2016**arXiv:1606.08415.
33. Sun, W.; Wang, H.; Xu, J.; Yang, Y.; Yan, R. Efektívny konvolučný transformátor pre vysoko presnú diagnostiku porúch planétovej prevodovky.*IEEE Open J. Instrument. Meas.***2022**,*1*, 1–9. Dostupné online:https://ieeexplore.ieee.org/iel7/9552935/9775186/09828477.pdf(prístup 10. decembra 2023). [CrossRef]

34. Cassirer, A.; Barth-Maron, G.; Brevdo, E.; Ramos, S.; Boyd, T.; Sottiaux, T.; Kroiss, M. Reverb: Rámec pre prehrávanie skúseností. *arXiv***2021** arXiv:2102.04736.
35. Hoffman, MW; Shahriari, B.; Aslanides, J.; Barth-Maron, G.; Momčev, N.; Sinopalnikov, D.; Stańczyk, P.; Ramos, S.; Raichuk, A.; Vincent, D.; a kol. Acme: Výskumný rámec pre distribuované posilňovanie učenia.*arXiv***2020**, arXiv:2006.00979.
36. Lapán, M.*Praktické učenie sa hlbokého posilňovania: aplikujte moderné metódy RL s hlbokými sieťami Q, iteráciou hodnôt, gradientmi politík, TRPO, AlphaGo Zero a ďalšími*; Packt Publishing Ltd.: Birmingham, Spojené kráľovstvo, 2018.
37. Singh, A.; Yang, L.; Hartikainen, K.; Finn, C.; Levine, S. End-to-End Robotic Reinforcement Learning without Reward Engineering. *arXiv* **2019**arXiv:1904.07854.
38. Capellier, E.; Davoine, F.; Cherfaoui, V.; Li, Y. Dôkazné hlboké učenie pre ľubovoľnú klasifikáciu objektov LIDAR v kontexte autonómneho riadenia. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paríž, Francúzsko, 9. – 12. júna 2019; IEEE: Piscataway, NJ, USA, 2019; s. 1304–1311. Dostupné online:https://hal.science/hal-02322434/file/IV19-Edouard.pdf(prístup 10. decembra 2023).

39. Skrinárová,J.; Huraj, Ľ.; Siládi, V. Model neurónového stromu na klasifikáciu zdrojov výpočtovej siete pomocou plánovania úloh PSO. *Neurónová sieť Svet***2013**,*23*, 223. Dostupné na internete:https://www.proquest.com/docview/1418215646/fulltextPDF/AF8F42E64A49412CPQ/1?accountid=49441&sourcetype=Scholarly%20Journals(prístup 10. decembra 2023). [CrossRef]
40. Sualeh, M.; Kim, GW Dynamická multilidarová detekcia a sledovanie viacerých objektov.*Senzory***2019**,*19*, 1474. Dostupné na internete: https://www.mdpi.com/1424-8220/19/6/1474/pdf(prístup 10. decembra 2023). [CrossRef]
41. Kyselica, D.; Šilha, J.; Ďurikovič, R.; Bartková,D.; Tóth, J. K spracovaniu obrazu reentry udalosti.*J. Appl. Matematika Stat. informovať*.**2023**,*19*, 47-60. Dostupné online:https://sciendo.com/article/10.2478/jamsi-2023-0003(prístup 10. decembra 2023). [CrossRef]
42. Orkphol, K.; Yang, W. Zjednoznačnenie slov pomocou kosínusovej podobnosti spolupracuje s Word2vec a WordNet.*Internet budúcnosti* **2019**,*11*, 114. Dostupné na internete:https://www.mdpi.com/1999-5903/11/5/114/pdf(prístup 10. decembra 2023). [CrossRef]

43. Appiah, N.; Vare, S. Playing Flappy Bird with Deep Reinforcement Learning. 2018. Dostupné na internete:http://vision.stanford.edu/learning/cs231n/reports/2016/pdfs/111_Report.pdf(prístup 10. decembra 2023).
44. Li, L.; Jiang, Z.; Yang, Z. Hranie modifikovaného Flappy Bird s hlbokým posilňovaním učenia. 2023. Dostupné na internete: https: //github.com/SeVEnMY/DeepLearningFinal(prístup 10. decembra 2023).
45. Hasselt, H. Dvojité Q-Learning.*Adv. Neural Inf. Proces. Syst.***2010**,*23*, 2613-2621. Dostupné online:https://proceedings.neurips.cc/paper_files/paper/2010/file/091d584fced301b442654dd8c23b3fc9-Paper.pdf(prístup 10. decembra 2023).
46. Al Rahhal, MM; Bazi, Y.; Jomaa, RM; AlShibli, A.; Alajlan, N.; Mekhalfi, ML; Melgani, F. Detekcia COVID-19 v Ct/röntgenových snímkach pomocou transformátorov videnia.*J. Pers. Med.***2022**,*12*, 310. Dostupné na internete:https://www.mdpi.com/2075-4426/12/2/310 (prístup 10. decembra 2023). [CrossRef] [PubMed]
47. Passricha, V.; Aggarwal, RK Porovnávacia analýza stratégií združovania pre konvolučnú neurónovú sieť založenú na hindskom ASR.*J. Ambient. Intell. Humaniz. Výpočet.***2020**,*11*, 675-691. Dostupné online:https://link.springer.com/article/10.1007/s12652-019-013 25-r. (prístup 10. decembra 2023). [CrossRef]
48. Mazumder, S.; Liu, B.; Wang, S.; Zhu, Y.; Yin, X.; Liu, L.; Li, J.; Huang, Y. Riadený prieskum v hlbokom posilňovaní učenia. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6. – 9. máj 2019; Dostupné online:https://openreview.net/forum?id=SJMeTo09YQ(prístup 10. decembra 2023).
49. Hessel, M.; Modayil, J.; Van Hasselt, H.; Schaul, T.; Ostrovski, G.; Dabney, W.; Horgan, D.; Piot, B.; Azar, M.; Silver, D. Rainbow: Kombinácia vylepšení v hlbokom posilňovaní učenia.*AAAI Conf. Artif. Intell.***2018**,*32*, 1. Dostupné na internete: https://ojs.aaai.org/index.php/AAAI/article/download/11796/11655(prístup 10. decembra 2023). [CrossRef]
50. Bao, H.; Dong, L.; Piao, S.; Wei, F. Beit: Bert Pre-Training of Image Transformers.*arXiv***2021**arXiv:2106.08254.