

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1678

UMJETNA INTELIGENCIJA U ZATVORENIKOVOJ DILEMI

Petar Belošević

Zagreb, lipanj, 2024.

Zagreb, 4. ožujka 2024.

ZAVRŠNI ZADATAK br. 1678

Pristupnik: **Petar Belošević (0036538383)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentor: izv. prof. dr. sc. Marko Čupić

Zadatak: **Računalni inteligentni igrač za igranje Zatvorenikove dileme**

Opis zadatka:

Zatvorenikova dilema je misaoni eksperiment koji igraju dva racionalna agenta koji međusobno mogu ili surađivati ili igrati protiv drugog agenta, a sve u svrhu maksimiziranja nagrade. U okviru ovog završnog rada potrebno je dati detaljni opis problema, istražiti i opisati njegova postojeća rješenja te implementirati računalnog igrača koji nekom tehnikom strojnog učenja pokušava naučiti dobru strategiju igranja. Radu priložiti izvorni i izvršni kod, uz potrebna objašnjenja i dokumentaciju. Citirati korištenu literaturu i navesti dobivenu pomoć.

Rok za predaju rada: 14. lipnja 2024.

Sadržaj

1. Uvod	2
2. Zatvorenikova dilema	4
2.1. Dobra strategija	5
3. Umjetne neuronske mreže	8
3.1. Elmanova neuronska mreža	9
4. Evolucijsko računanje	11
4.1. Optimizacija	11
4.2. Genetski algoritam	12
5. Implementacija inteligentnog igrača	14
5.1. Učenje inteligentnog igrača	14
6. Definiranje pokusa	16
7. Rezultati i rasprava	17
8. Zaključak	18
Literatura	19
Sažetak	21
Abstract	22
A: The Code	23

1. Uvod

Neki od radova koje ćemo citirati su [?, ?, ?, ?, ?, ?]. Trebaju nam samo radi testiranja kako izgleda referenciranje rada s konferencije, rada iz časopisa, knjige i Internetske stranice.



Slika 1.1. Moja prva slika

Referenciramo se na sliku 1.1. u sredini rečenice, zatim prije zareza 1.1., te zatim na kraju rečenice 1.1. Upravo smo testirali radi li naredba `\ref` ispravno u slučaju kada nakon nje slijedi točka.

Sada slijedi jedna jednačba:

$$\int_{-\infty}^{+\infty} f(t) dt = F(\omega) \tag{1.1}$$

Jednačba (1.1) je moja prva jednačba koja definira par $f(t) \longleftrightarrow F(\omega)$ ili $F(\omega) \longleftrightarrow f(t)$.

[1] [2] [3]

Zatvorenikova dilema jedan je od najpoznatijih problema iz područja teorije igara. Dilema proučava interakciju između dvaju pojedinaca kroz igru suradnje i izdaje. Zatvorenikova dilema se može pronaći u korijenu mnogih interakcija koje se pojavljuju u društvu, ali i u životinjskom svijetu. Glavno pitanje je kako ta interakcija utječe na dobrobit oba igrača. Paradoks u Zatvorenikove dileme je činjenica da racionalno razmišljanje navodi svakog pojedinca da odabire opciju koja ne odgovara niti jednom od igrača [4].

U stvarnom životu su takve interakcije često ponavljajuće prirode. Tada je potrebno imati na umu da će druga strana vjerojatno imati iskustvo prošlih interakcija što može utjecati na njihovu odluku u novoj interakciji. Stoga je interesantno razmatranje malo složenijeg problema iterirajuće Zatvorenikove dileme. Zanimljivo je da se u iterirajućem problemu mijenja pristup optimalnoj strategiji u odnosu na jednokratnu Zatvorenikovu dilemu.

U korijenu Zatvorenikove dileme je napetost između individualnog racionalizma (u pogledu da je objema stranama smisleno biti sebičan) i grupnog racionalizma (objema stranama je isplativija obostrana suradnja nego obostrana izdaja) [1].

Mnogo je radova na pisano na temu optimalnog načina igranja Zatvorenikove dileme. Tako je Robert Axelrod iz Sveučilišta u Michiganu održao dva turnira u kojima su sudionici predavali svoje strategije za igranje igre u obliku računalnih programa [1] [2]. Analizom turnira došlo se zanimljivih zaključaka. Axelrod je proučavanjem strategija i njihovih rezultata izlučio nekoliko karakteristika koje imale tendenciju davati bolje rezultate. Te karakteristike su ljubaznost, opraštanje i provokabilnost [2].

Glavni cilj ovog rada je napraviti model inteligentnog igrača koji će naučiti igrati iterirajuću Zatvorenikovu dilemu. Cilj ovoga je proučiti kakvo će ponašanje razviti inteligentni igrač i kako će igrati protiv nekih drugih poznatih strategija. Također će biti zanimljivo analizirati ponašanje inteligentnog igrača i vidjeti je li razvio karakteristike koje su se pokazale poželjnima u Axelrodovim turnirima.

2. Zatvorenikova dilema

Zatvorenikova dilema je poznati problem iz područja teorije igara koji proučava odnose između pojedinaca. Radi se o igri sa dva igrača koji imaju na izboru dva poteza: suradnja sa drugim igračem, ili izdaja drugog igrača. S obzirom na odabir oba igrača, svakom od njih se daje određena kazna ili nagrada.

Dilema je dobila naziv po ilustrativnom opisu Alberta Tuckera [5]. Dilema je opisana u kontekstu odvojenog ispitivanja dvojice osumnjičenih kriminalaca. Svaki od osumnjičenih može priznati krivnju (ekvivalentno izdaji) ili šutjeti (ekvivalentno suradnji). Sukladno njihovim potezima dodjeljuju im se zatvorske kazne.

Dilema se naravno može staviti i opisati kroz razne slikovite kontekste. No, u ovom radu ćemo radi jednostavnosti dilemu gledati samo kao igru u kojoj igrači skupljaju bodove.

Još raspisati?

Generalna pravila za "bodovanje" je da obostrana suradnja i obostrana izdaja daju simetričnu podjelu bodova, uz to da suradnja daje veći broj bodova. U slučaju da jedan igrač surađuje dok ga drugi izdaje dolazi do nesimetrične podjele bodova. Igrač koje je izdan mora dobiti manje bodove nego bi dobio obostranom izdajom. Igrač koji ga je izdao tim potezom mora dobiti više nego bi dobio obostranom suradnjom. Tipična raspodjela bodova [1] koja će se koristiti i u ovom radu dana je u tablici 2.1.

Tablica 2.1.: Bodovanje odluka igrača u Zatvorenikovo dilemi

Napomena: Bodovi Igrača 1 su dani prvim brojem u svakom od parova.

		Igrač 2	
		Suradnja	Izdaja
Igrač 1	Suradnja	3, 3	0, 5
	Izdaja	5, 0	1, 1

Važno je primijetiti da je Zatvorenikova dilema igra s promjenjivim ishodom [4]. To znači da dobitak jednog igrača nije nužno gubitak drugog igrača i gubitak jednog igrača ne mora nužno biti dobitak drugog igrača [3]. Drugim riječima, mogući su ishodi u kojima oba igrača profitiraju ili oba igrača gube. Upravo ova karakteristika omogućuje suradnji da bude poželjan pristup igranju.

Zbog ove karakteristike Zatvorenikova dilema dobro opisuje mnoge probleme iz stvarnog života i objašnjava zašto je ponekad suradnja sa neprijateljem poželjna. Poznati je takav primjer problem prekomjernog naoružavanja i međusobnog nepovjerenja SAD-a i SSSR-a tijekom hladnog rata [1] Upravo je i taj problem bio jedan od motiva nastanka i proučavanja Zatvorenikove dileme.

2.1. Dobra strategija

Pitanje koje se prirodno nameće u Zatvorenikovo dilemi je dosta očito: Kako dobro igrati ovu igru? Kojom strategijom pristupiti ovoj igri?

U svrhu istraživanja optimalnih strategija Robert Axelrod je 1980. organizirao turnir [1] u koji je pozvao znanstvenike iz raznih područja koji su se bavili Zatvorenikovom dilemom. Učesnici su za turnir izradili strategije igranja Zatvorenikove dileme u obliku računalnog programa. Svaka strategija je igrala Zatvorenikovu dilemu u 200 iteracija sa svakom drugom strategijom, sa samom sobom i sa dodatnom strategijom koja je donosila nasumične odluke. Kasnije te godine Axelrod je organizirao još jedan sličan turnir u svrhu provođenja dodatne analize [2].

Pobjednik u oba turnira je bila strategija zvana "Tit for Tat", također poznata i kao

"Copycat". Strategija je vrlo jednostavna. Na prvom potezu uvijek surađuje, a dalje uvijek kopira suparnikov prethodni potez [1].

Axelrod je analizom rezultata oba turnira pronašao nekoliko karakteristika koje su bile ključne za uspješnost strategija u oba turnira. Pronađene karakteristike su:

- **Ljubaznost** (eng. niceness) [1] - strategija je ljubazna ako nikada neće prva izdati drugog igrača
- **Sklonost opraštanju** (eng. forgiveness) [1] - sklonost strategije da surađuje nakon što ju je drugi igrač izdao
- **Provokabilnost** (eng. provocability) [2] - sklonost da strategija izda drugog igrača nakon što ju je drugi igrač izdao

Strategija "Tit for Tat" je bila uspješna jer je imala sve navedene karakteristike. "Tit for Tat" je ljubazna strategija jer počinje sa suradnjom i izdaje samo ako ju je drugi igrač izdao prvi. Strategija oprašta jer će nakon što bude jednom izdana uzvratiti izdajom, ali će nakon toga nastaviti surađivati dok ponovno ne bude izdana. Također, strategija je provokabilna jer će nakon što bude izdana uzvratiti izdajom.

Axelrod je u svojem radu napomenuo važnu stvar, a to je da ne postoji najbolja strategija za Zatvorenikovu dilemu. To proizlazi iz jednostavne činjenice da performansa strategije uvelike ovisi o strategijama s kojima ta strategija igra. Dakle, okruženje u kojem se nalazi je od presudne važnosti [1].

No kroz detaljnu analizu drugog turnira Axelrod je ustanovio da su navedene karakteristike generalno poželjne i u pravilu donose dobre rezultate. Također, iako "Tit for Tat" nije univerzalno najbolja strategija, pokazuje se da generalno daje odlične rezultate u raznim okruženjima zbog dobro kombiniranih poželjnih karakteristika [2].

Axelrod je također napomenuo da je kod kreiranja strategije potrebna analiza na barem 3 dubine [1]. Prva razina je direktna posljedica trenutne odluke, to jest, koliko bodova igrač dobiva na temelju svoje odluke. Druga razina je uzimanje u obzira da će drugi igrač možda odlučiti kazniti izdaju. Treća razina razmatra da reagiranje na izdaju može prouzrokovati eho efekt međusobnih izdaja koji će dugoročno štetiti i jednom i drugom

igraču.

3. Umjetne neuronske mreže

Umjetne neuronske mreže su jedan od mnogih koncepata u strojnom učenju, koje je jedno od najvećih grana umjetne inteligencije. Prvi puta su ih predložili Warren McCulloch i Walter Pitts sa Sveučilišta u Chicagu davne 1944. Prva neuronska mreža koja se mogla trenirati je bila Perceptron, predložen od strane psihologa Franka Rosenblatta sa Sveučilišta Cornell 1957. [6]

Umjetne neuronske mreže su inspirirane ljudskim mozgom - sastoje se od velikog broja jednostavnih umjetnih neurona koji su gusto povezani. [6]

Područje koje se bavi umjetnim neuronskim mrežama zove se neuro-računarstvo, koje čini granu računarstva iz skupine mekog-računarstva. Meko računarstvo objedinjuje pristupe čiji je cilj izgraditi tehničke sustave za rješavanje teških problema sa čestim šumom i nepouzdanošću u ulaznim podacima. [7]

Umjetni neuron je najmanja funkcionalna jedinka neuronske mreže i modelira biološki neuron u ljudskom mozgu. Umjetni neuroni imaju ulogu jednostavne procesne jedinice. Svaki umjetni neuron na ulazu prima nekakve vrijednosti (izlazi drugih neurona ili ulazni podaci). Ulazni podaci u neuron se prvo množe sa težinama veza preko kojih su došli do trenutnog neurona. Dobiveni umnošci se zbrajaju, te im se dodatno pribraja takozvani pomak. Tako nastala suma se provlači kroz prijenosnu funkciju. Rezultat prijenosne funkcije se postavlja na izlaz umjetnog neurona te se propagira dalje na ulaze drugih neurona ili na izlaz neuronske mreže. Postoje razne prijenosne funkcije različitih složenosti koje se mogu koristiti kod umjetnih neurona, a neke od njih su: funkcija identiteta, funkcija skoka, sigmoidna funkcija, tangens hiperbolni, funkcija zglobnica, funkcija propusna zglobnica [7].

Neka slika neurona (i prijenosnih funkcija)

Umjetni neuroni se povezuju u paralelne strukture različitih arhitektura koje omogućuju paralelnu obradu podataka [7]. Arhitektura umjetne neuronske mreže opisuje kako su povezani umjetni neuroni u mreži i koliko ih ima. Postoje razne arhitekture neuronskih mreža, što jednostavnih, što kompleksnih, a svaka ima svoje prednosti i područja u kojima ima primjenu.

Umjetni neuroni se u umjetnim neuronskim mrežama obično organiziraju u slojeve. Tako razlikujemo ulazni sloj, skriveni slojeve i izlazni sloj. Skriveni slojevi i izlazni sloj se sastoje od umjetnih neurona dok ulazni sloj zapravo ne sadrži umjetne neurone kakvi su opisani u odjeljku iznad. Neuroni ulaznog sloja na svoje ulaze samo primaju ulazne podatke neuronske mreže te ih dalje prosljeđuju neuronima prvog skrivenog sloja.

Umjetne neuronske mreže danas imaju vrlo široku primjenu, od obrade slika i signala, prepoznavanja uzoraka, kompresije slika, upravljanja, financija... [7]. Glavne prednosti neuronskih mreža su vrlo dobra sposobnost generalizacije i učenja, čak i uz neprecizne ulazne podatke. Njihova glavna mana je nemogućnost interpretacije njihovog ponašanja, iako su računalni znanstvenici počeli razvijati metode za dedukciju strategija koje neuronske mreže nauče [6]. Znanje koje neuronske mreže nauče iz podataka je implicitno pohranjeno u način povezivanja neurone i težine veza između njih [7]. Stoga je teško razumjeti zašto je neuronska mreža za neki ulaz dala baš taj izraz. Također, teško je prepoznati kakvu konkretnu ulogu u sveukupnoj obradi ulaza ima pojedini neuron u mreži.

3.1. Elmanova neuronska mreža

Elmanova neuronska mreža je jednostavna povratna neuronska mreža (*eng. Recurrent Neural Network*). Glavna karakteristika takvih mreža je da imaju neku vrstu povratne veze ili vremenski odgođene veze [8].

Elmanova neuronska mreža se sastoji od ulaznog sloja, jednog skrivenog sloja i izlaznog sloja. Skriveni sloj ove mreže ima dodatan skriveni kontekst. Kontekst čine neuroni koji služe za pohranu kontekstnih podataka. Dodatno postoje veze iz izlaza skrivenog sloja prema kontekstu i veze iz konteksta prema ulazu u skriveni sloj. Izlaz iz skrivenog sloja se preko tih veza sprema u kontekst. Kod obrade sljedećeg ulaza u neuronsku mrežu

skrivenom se sloju na ulaz dodatno stavljaju vrijednosti iz konteksta preko pripadajućih veza te se ukupan ulaz mreže i konteksta dalje obrađuje. [9]

Neka slika Elmanove mreže

Ovime se omogućava da Elmanova mreža može pamtit i prošla stanja. Na ovaj način prijašnji ulazi u neuronsku mrežu utječu na rezultat obrade nekog budućeg ulaza.

Ovo je korisno svojstvo za model inteligentnog igrača Zatvorenikove dileme, s obzirom da je prirodno da prijašnje interakcije sa drugim igračem utječu na nove odluke.

4. Evolucijsko računanje

Evolucijsko računanje je grana umjetne inteligencije koja se najčešće bavi rješavanjem optimizacijskih problema. Začeci ove grane sežu još u kasne 50-te godine prošlog stoljeća. Evolucijsko računanje se može podijeliti u tri grane [10]:

- *evolucijske algoritme*
- *algoritme rojeva*
- *ostale algoritme*

Neki od algoritama iz grane evolucijskih algoritama su evolucijske strategije, evolucijsko programiranje, genetski algoritam i genetsko programiranje [10].

U grani algoritma rojeva nalaze se mravlji algoritmi, algoritam roja čestica, algoritmi pčela i drugi [10].

Neki do značajnih algoritama iz grane ostali algoritmi pripadaju umjetni imunološki algoritmi, algoritam diferencijske evolucije i algoritam harmonijske pretrage [10].

Područje evolucijskog računanja također pripada području metaheuristika. Metaheuristika je skup algoritamskih koncepata koji se koriste za definiranje heurističkih metoda primjenjivih na široki skup problema [10]. Heurističke metode su algoritmi koji pronalaze rješenja problema koja su zadovoljavajuće dobra, ali ne nužno optimalna. Heurističke metode obično imaju relativnu nisku računsku složenost [10].

4.1. Optimizacija

Kao što je ranije navedeno, algoritmi iz područja evolucijskog računanja se često koriste u problemima optimizacije. Optimizaciju možemo definirati kao postupak pronalaženja

najboljeg rješenja problema. Uz svako rješenje se pridružuje funkcija dobrote ili funkcija kazne. Ako se rješenjima pridružuje funkcija dobrote, optimizacijski postupak će tražiti rješenje tako da pokuša maksimizirati funkciju dobrote. Ako se pak rješenjima pridružuje funkcija kazne, optimizacijski postupak će pokušati minimizirati funkciju kazne.[10]

Bitno je reći da kada gledamo optimizacijske algoritme iz područja evolucijskog računanja, niti jedan od njih se ne može istaknuti kao generalno najbolji algoritam. Svaki od tih algoritama je dobar u određenim primjenama, dok u drugim primjenama drugi optimizacijski algoritmi daju bolje rezultate. U prosjeku nema značajnih razlika u globalnim performansama kada se gleda prosjek po svim mogućim funkcijama cijene.[10]

4.2. Genetski algoritam

Genetski algoritam je algoritam iz grane evolucijskih algoritama. Općenito, svi pristupi iz grane evolucijskih algoritama crpe inspiraciju iz Darwinove teorije o postanku vrsta, koja se temelji na pet pretpostavki [10]:

1. potomaka uvijek ima više nego je potrebno
2. veličina populacije je približno stalna u vremenu
3. hrana je ograničen resurs
4. ne postoje dvije potpuno identične jedinke kod vrsta koje se seksualno razmnožavaju - postoje varijacije
5. većina varijacija se prenosi nasljeđem

Genetski algoritam je jedan od algoritama koji rješavaju optimizacijske probleme, a koji izravno utjelovljuje navedene postavke. Genetski algoritam zapravo nije jedinstven, već se može implementirati na različite načine. [10]

Ideja genetskog algoritma je modeliranje evolucijskog procesa. U algoritmu postoji populacija jedinki, koje zapravo predstavljaju neka od mogućih rješenja problema. Cilj algoritma je kroz selekciju roditelja, stvaranje novih jedinki korištenjem operatora križanja nad roditeljima i korištenje operatora mutiranja nad novim jedinkama. Selekcija

roditelja bi trebala ovisiti o nekakvoj metrici uspješnosti - uspješniji roditelji bi trebali imati veće izgleda da budu odabrani za križanje. Takvim postupkom se očekuje da će se kontinuiranim križanjem uspješnih roditelja i mutiranjem njihove djece stvarati bolja rješenja. Ovo je zapravo svojevrstan ekvivalent prirodnoj selekciji, to jest preživljavanju najboljih.

Varijacije u implementaciji se mogu pojavljivati u načinu mutiranja, križanja, selekcije, ali i magnitudi izmjena u populaciji u jednoj iteraciji algoritma.

Tako u jednoj krajnosti imamo takozvani *eliminacijski genetski algoritam* u kojem se u jednoj iteraciji samo jedna jedinka zamjenjuje jednim novonastalim djetetom. Novo dijete već u sljedećem koraku može biti roditelj.

U drugoj krajnosti imamo *generacijski genetski algoritam*. U njemu se u jednoj iteraciji iz jedne populacije biraju roditelji čija se djeca spremaju u novu populaciju sve dok nova populacije ne dosegne veličinu populacije roditelja. Tada populacija djece zamjenjuje populaciju roditelja. U ovoj varijanti dijete može biti roditelj tek kada populacija djece postane populacija roditelja, to jest u sljedećoj iteraciji. [10]

Operator križanja se primjenjuje nad dvije jedinke (roditelji) i iz njih stvara dijete. Može se implementirati na razne načine, ali generalno ovaj operator često vrši kontrakciju nad populacijom. To znači da se primjenom ovog operatora obično gubi raznolikost između jedinki te one u prosjeku postaju sve sličnije. [10]

Operator mutacije ima upravo suprotno svojstvo. On vrši nasumične izmjene nad jedinkama i time im vraća raznolikost. Njegova važna uloga je izbaciti populaciju iz lokalnih maksimuma u prostoru rješenja [10].

5. Implementacija inteligentnog igrača

Inteligentni igrač Zatvorenikove dileme modeliran je jednostavnom Elmanovom neuronskom mrežom. Mreža se sastoji od ulaznog sloja, jednog skrivenog sloja sa kontekstom i izlaznog sloja. Svi slojevi su potpuno povezani. Veze između izlaza skrivenog sloja i konteksta su jedan-na-jedan, to jest izlaz jednog neurona se preslikava u "kopiju" tog neurona u kontekstu. To implicitno znači da skriveni kontekst ima jednak broj neurona kao i sam skriveni sloj. Veze između konteksta i ulaza u neurone skrivenog sloja su svaki-sa-svakim, isto kao i između ostalih slojeva u mreži.

Korištena mreža je arhitekture $1 \times 8 \times 1$. Ulaz u mrežu je brojčana vrijednost odluke suparničkog igrača iz prijašnje runde. Ako prijašnje runde još nije bilo, tada se na ulaz dovodi 0. Odluka o suradnji je brojčano kodirana vrijednošću 1, a odluka o izdaju vrijednošću -1. Izlaz mreže je brojčana vrijednost odluke koju će inteligentni igrač odigrati u ovoj rundi Zatvorenikove dileme.

5.1. Učenje inteligentnog igrača

S obzirom na karakteristike Zatvorenikove dileme i dinamiku okruženja u kojem inteligentni igrač mora donositi odluke, često korišteno nadzirano učenje neuronske mreže algoritmom propagacije unazad ovdje nije praktično.

Ako bismo koristili algoritam propagacije unazad, morali bismo za svaki ulaz definirati očekivani (najbolji) izlaz, to jest optimalnu odluku u toj situaciji. S obzirom da u ovom problemu postoje samo 3 moguća ulaza u mrežu (0 ako se radi o prvoj rundi, 1 ako je u prijašnjoj rundi suparnički igrač surađivao i -1 ako je izdao ovog igrača), skup za učenje bi bio premalen. Također, problem u tom pristupu je to što bismo time inteligentnom

igraču zapravo predstavljali problem Zatvorenikove dileme sa samo jednom iteracijom, a taj problem ima potpuno različit pristup u odnosu na iterirajuću Zatvorenikovu dilemu, kao što je prije raspravljeno. Još jedan problem je u tome što je zapravo teško odrediti što je najbolji potez koji igrač može odigrati u nekom okruženju.

Bolji pristup je zato neka vrsta podržanog učenja. Općenito se kod podržanog učenja razmatra interakcija agenta i njegove okoline. Agent dobiva informacije iz okolina i na temelju njih vrši akcije. Za izvršene akcije agent dobiva nagrade ili kazne. Cilj podržanog učenja je razviti strategiju ponašanja agenta koja će maksimizirati njegovu nagradu. ???

Zbog toga je za ovaj problem odlučeno koristiti jednu drugu tehniku, a to je učenje neuronske mreže genetskim algoritmom. Ideja ove tehnike je koristiti genetski algoritam kao algoritam optimizacije nad neuronskim mrežama. Cilj optimizacije je pronaći težine veza između neurona (i pomake svakog neurona) koje će davati najbolje (ili barem dovoljno dobre) performanse neuronske mreže u igri Zatvorenikove dileme.

6. Definiranje pokusa

7. Rezultati i rasprava

8. Zaključak

Na kraju rada piše se kratak zaključak, duljine do najviše jedne stranice.

Literatura

- [1] R. Axelrod, “Effective choice in the prisoner’s dilemma”, *The Journal of Conflict Resolution*, sv. 24, br. 1, str. 3–25, ožujak 1980. [Mrežno]. Adresa: <https://doi.org/10.1177/002200278002400101>
- [2] —, “More effective choice in the prisoner’s dilemma”, *The Journal of Conflict Resolution*, sv. 24, br. 3, str. 379–403, rujan 1980. [Mrežno]. Adresa: <https://doi.org/10.1177/002200278002400301>
- [3] Igra. Hrvatska enciklopedija, mrežno izdanje. Leksikografski zavod Miroslav Krleža, 2013–2024. <https://enciklopedija.hr/clanak/70136>; pristupljeno 8.3.2024.
- [4] Zatvorenikova dilema. Hrvatska enciklopedija, mrežno izdanje. Leksikografski zavod Miroslav Krleža, 2013–2024. <https://enciklopedija.hr/clanak/zatvorenikova-dilema>; pristupljeno 9.3.2024.
- [5] M. E. I. Barković Bojanić, “Teorija igara i pravo”, *Pravni vjesnik*, sv. 29, br. 1, str. 59–76, travanj 2013. [Mrežno]. Adresa: <https://hrcak.srce.hr/111004>
- [6] Explained: Neural networks. MIT News, 2017. <https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414>; pristupljeno 23.5.2024.
- [7] M. Čupic, Umjetne neuronske mreže, skripta iz kolegija Uvod u umjetnu inteligenciju, 2018. <http://java.zemris.fer.hr/nastava/ui/ann/ann-20180604.pdf>; pristupljeno 24.5.2024.
- [8] Implementation of Elman Recurrent Neural Network in WEKA. ICT Research Blog by John Salatas, 2018. <https://jsalatas.ictpro.gr/implementation-of-elman->

recurrent-neural-network-in-weka/; pristupljeno 22.5.2024.

- [9] J. L. Elman, "Finding structure in time", *Cognitive Science*, sv. 14, br. 2, str. 179–211, ožujak 1990. https://doi.org/https://doi.org/10.1207/s15516709cog1402_1
- [10] M. Čupic, Evolucijski računarstvo, skripta iz kolegija Uvod u umjetnu inteligenciju, 2019. <http://java.zemris.fer.hr/nastava/ui/evo/evo-20190604.pdf>; pristupljeno 24.5.2024.

Sažetak

Umjetna inteligencija u Zatvorenikovo dilemi

Petar Belošević

Unesite sažetak na hrvatskom.

Naslov, sažetak, ključne riječi (na hrvatskom jeziku) Sažetak opisuje sadržaj rada, prepričan u stotinjak riječi.

Ključne riječi: Zatvorenikova dilema, umjetna inteligencija, umjetne neuronske mreže, inteligentni igrač

Abstract

AI in Prisoner's Dilemma

Petar Belošević

Enter the abstract in English.

Title, summary, keywords (na engleskom jeziku)

Keywords: Prisoner's dilemma, Artificial intelligence, Artificial neural networks, Intelligent player

Privitak A: The Code

Privitak je također opcionalno poglavlje (u dogovoru s mentorom). Sadržaj koji se stavlja u privitak je, općenito, nešto što je, kao cjelinu, prikladno izdvojiti iz sadržaja samog rada. Mogući primjer je tehnička dokumentacija vezana uz završni rad - npr. električka i položajna shema sklopa, sastavnica, predložak tiskane veze, plan bušenja, ispis programa s detaljnim opisom. Drugi primjer uključuju upute za korištenje rezultata rada (softvera ili hardvera), detaljni ispisi mjerenja čiji su rezultati sažeto ili grafički prikazani u radu. Ako se radi o softveru, uobičajeno je navesti podatke o platformi na kojoj se izvodi (npr., karakteristike uređaja i operacijskog sustava te pomoćnog softvera), kao i upute za instalaciju.

U privitku nemojte koristiti stilove razine Heading, već samo (nenumерirani) stil Podnaslov. Na primjer:

Instalacija programske podrške

Upute za korištenje programske podrške