VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ STUDIJŲ PROGRAMA

Skatinamojo mokymosi modelio tyrimas alternatyviai ir augmentinei komunikacijai

Research on a reinforcement learning model for alternative and augmentative communication

Kursinis darbas

Atliko: 4 kurso 4 grupės studentė
Paulina Ivanauskaitė

Darbo vadovas: doc. dr. Asta Slotkienė

Vilnius – 2026

Turinys

Santrumpos

AAC – (angl. *Augmentative and Alternative Communication*) Alternatyvi ir augmentatyvi komunikacija

ABA – (angl. *Applied Behavior Analysis*) Taikomoji elgesio analizė

ASS – Autizmo spektro sutrikimas

DI – Dirbtinis intelektas

LLM – (angl. *Large Language Model*) Didysis kalbos modelis

MDP – (angl. *Markov Decision Process*) Markovo sprendimų procesas

PECS – (angl. *Picture Exchange Communication System*) Paveikslėlių mainų komunikacijos sistema

RL – (angl. *Reinforcement Learning*) Skatinimasis mokymasis

SARSA – (angl. *State-Action-Reward-State-Action algorithm*) Būsena-Veiksmas-Atlygis-Būsena-Veiksmas algoritmas

TD – (angl. *Temporal Difference*) Skirtumų laike algoritmas

Įvadas

Autizmo spektro sutrikimas (ASS) yra neurologinė raidos būklė, pasireiškianti socialinės komunikacijos ir sąveikos sunkumais. Pasaulio sveikatos organizacijos duomenimis, maždaug 1 iš 100 vaikų turi autizmo spektro sutrikimą, iš kurių reikšminga dalis yra neverbaliniai arba minimaliai verbaliniai [AAA24]. Šiems asmenims tradicinė verbalinė komunikacija tampa itin sudėtinga arba neįmanoma, todėl alternatyvios ir augmentatyvios komunikacijos (AAC) sistemos tampa gyvybiškai svarbios kasdieniam gyvenimui ir socialinei integracijai [IG24].

Paveikslėlių apsikeitimo komunikacijos sistema (angl. *Picture Exchange Communication System*, PECS) yra vienas iš plačiausiai taikomų ir empiriškai pagrįstų AAC metodų, leidžiantis asmenims su kompleksiniais komunikacijos poreikiais reikšti savo norus, poreikius ir mintis naudojant vizualines komunikacijos korteles [TOS+23]. PECS sistema buvo sukurta 1985 metais ir nuo to laiko plačiai taikoma vaikams su autizmu bei kitomis raidos negaliomis [Ter22]. Nepaisant PECS efektyvumo, tradicinės sistemos remiasi fiksuotomis taisyklėmis ir neadaptuojasi prie individualių vartotojo poreikių, mokymosi progreso ar konteksto pokyčių.

Pastaraisiais metais dirbtinio intelekto (DI) ir mašininio mokymosi technologijos atvėrė naujas galimybes AAC sistemų tobulinimui [KWL+23]. Pereira ir kt. [PPZ+24] parodė, kad transformeriais pagrįsti kalbos modeliai gali reikšmingai pagerinti komunikacijos kortelių prognozavimą AAC sistemose, adaptuojantis prie individualių vartotojų žodynų. Holyfield ir kt. [HMC+24] tyrė dirbtinio intelekto taikymą automatizuotam augmentatyviam įvesčiai vaikams su autizmu, parodydami, kad DI gali padėti įveikti komunikacijos partnerių laiko ir mokymosi barjerus. Tačiau skatinimojo mokymosi (angl. *Reinforcement Learning*, RL) algoritmų taikymas PECS kortelių parinkimo kontekste lieka mažai ištirtas, nors RL metodai natūraliai tinka sprendimų priėmimui remiantis grįžtamuoju ryšiu [SB18].

Darbo aktualumas. Tradicinės PECS sistemos pateikia korteles fiksuota tvarka arba pagal paprastas taisykles, neatsižvelgiant į individualius vaiko poreikius ir ankstesnę sąveikos istoriją. Tai lemia, kad vaikas dažnai turi peržiūrėti daug neaktualių kortelių, kol randa norimą, kas mažina komunikacijos efektyvumą ir gali sukelti nepasitenkinimą. Prisitaikančios sistemos, gebančios mokytis iš vaiko reakcijų ir optimizuoti kortelių pateikimo tvarką, galėtų reikšmingai pagerinti komunikacijos greitį ir tikslumą.

Darbo tikslas – ištirti skirtingų skatinamojo mokymosi algoritmų panaudojimo efektyvumą PECS komunikacijos kortelių parinkimo sistemoje.

Darbo uždaviniai:

1. Išanalizuoti skatinimojo mokymosi algoritmų teorinius pagrindus ir įvertinti jų tinkamumą AAC sistemai.
2. Sukurti eksperimentinę aplinką su skirtingais skatinamojo mokymosi algoritmais, kortelių parinkimo strategijomis, tyrinėjimo bei panašumo metodais ir parametrų kombinacijomis.
3. Atlikti išsamų eksperimentą, lyginant algoritmų efektyvumą pagal atitikties rodiklį.
4. Išanalizuoti gautus rezultatus ir pateikti rekomendacijas dėl optimalių parametrų ir strategijų praktiniam PECS sistemų tobulinimui.

1. Teorinė dalis

1.1. Alternatyvi ir augmentatyvi komunikacija

Alternatyvi ir augmentatyvi komunikacija (AAC) apima įvairius metodus, priemones ir strategijas, skirtus papildyti arba pakeisti natūralią kalbą asmenims su kompleksiniais komunikacijos poreikiais [PPZ+24]. AAC sistemos gali būti klasifikuojamos į žemos technologijos (pvz., paveikslėlių lentos, gestai, rašymas) ir aukštos technologijos (pvz., planšetiniai kompiuteriai su kalbos generavimo programomis, specializuoti komunikatoriai) sprendimus [IG24].

Konadl ir kt. [KWL+23] atliko sisteminę literatūros apžvalgą, analizuodami dirbtinio intelekto taikymą AAC sistemose. Tyrėjai nustatė, kad DI pagerintos AAC sistemos gali reikšmingai pagerinti komunikacijos efektyvumą ir vartotojų autonomiją, tačiau egzistuoja spragos tyrimuose, ypač kalbant apie pokalbio pradžios ir pabaigos fazes bei informatyvumo kontekstą. Valencia ir kt. [VCK+23] tyrė didelių kalbos modelių (LLM) taikymą AAC sistemose ir nustatė, kad AAC vartotojai yra entuziastingi, bet atsargūs dėl tokių technologijų naudojimo savo komunikacijai.

Martin ir kt. [MN24] pabrėžė, kad AAC tyrimai dažniausiai orientuoti į vaikus, tuo tarpu suaugusieji su autizmu, ypač tie, kurie diagnozuojami vėliau gyvenime, susiduria su specifiniais iššūkiais. Tyrėjai akcentavo privatumo problemas ir poreikį kurti AAC sistemas, kurios gerbtų vartotojų duomenis ir tapatybę.

1.2. PECS sistema ir jos panaudojamumas

Paveikslėlių apsikeitimo komunikacijos sistema (PECS) buvo sukurta 1985 metais Lori Frost ir Andrew Bondy kaip struktūruotas alternatyvios komunikacijos metodas, pagrįstas taikomosios elgesio analizės (angl. *Applied Behavior Analysis*, ABA) principais [Ter22]. Sistema susideda iš šešių progresyvių fazių:

1. I fazė – Fiziniai mainai: vaikas mokosi fiziškai paimti paveikslėlį ir paduoti jį komunikacijos partneriui mainais į norimą objektą.
2. II fazė – Atstumo didinimas: vaikas mokosi naudoti įgūdį įvairiose aplinkose ir su skirtingais partneriais, įveikdamas atstumo barjerus.
3. III fazė – Paveikslėlių atskyrimas: vaikas mokosi atskirti ir pasirinkti tinkamą paveikslėlį iš kelių galimų variantų.
4. IV fazė – Sakinių struktūra: vaikas mokosi konstruoti paprastus sakinius naudodamas „Aš noriu" kortelę kartu su norimo objekto kortele.
5. V fazė – Atsakymas į klausimus: vaikas mokosi atsakyti į klausimą „Ko tu nori?" naudodamas PECS sistemą.
6. VI fazė – Komentavimas: vaikas mokosi spontaniškai komentuoti aplinką, atsakydamas į klausimus „Ką tu matai?", „Ką tu girdi?" ir pan.

Alfuraih ir kt. [AAA24] tyrė PECS efektyvumą ugdant prašymo įgūdžius vaikams su daugybine negalia. Tyrimas, atliktas Saudo Arabijos Autizmo meistriškumo centre, parodė, kad PECS yra efektyvi priemonė vystant ir generalizuojant komunikacijos įgūdžius. Tamanaha ir

kt. [TOS+23] įvertino PECS įgyvendinimo programą vaikams su autizmo spektro sutrikimu Brazilijoje ir nustatė, kad visos 22 tyrimo dalyvės sėkmingai pasiekė pirmąsias tris PECS fazes per 24 terapijos sesijas.

Siriraj ligoninės tyrėjų komanda [WVW25] analizavo PECS mokymo atitikties prognostinius veiksnius ribotų išteklių aplinkoje. Tyrimas, apėmęs 61 vaiką su komunikacijos sutrikimais, nustatė, kad receptyviosios ir ekspresyviosios kalbos raidos koeficientai (DQ) bei pradinis klinikinės būklės sunkumas buvo reikšmingi sėkmingo PECS mokymo prognoziniai veiksniai.

1.3. Skatinimasis mokymasis

Skatinamasis mokymasis (RL) yra mašininio mokymosi paradigma, kurioje agentas mokosi priimti sprendimus sąveikaudamas su aplinka ir gaudamas atlygius (arba baudas) už savo veiksmus [SB18]. Skirtingai nuo prižiūrimo mokymosi, kur modelis mokosi iš pažymėtų pavyzdžių, RL agentas mokosi per bandymus ir klaidas, optimizuodamas ilgalaikį kaupiamąjį atlygį.

Formaliai, RL problema apibrėžiama kaip Markovo sprendimų procesas (MDP), kurį sudaro [SB18]:

- $\mathcal{S}$ – baigtinė būsenų aibė
- $\mathcal{A}$ – baigtinė veiksmų aibė
- $P(s'|s,a)$ – perėjimo tikimybės funkcija
- $R(s,a,s')$ – atlygio funkcija
- $\gamma \in [0,1]$ – nuolaidos faktorius

Agento tikslas yra rasti strategiją $\pi : \mathcal{S} \to \mathcal{A}$, kuri maksimizuoja tikėtiną kaupiamąjį atlygį.

1.3.1. Veiksmo-vertės funkcija (Q-funkcija)

Veiksmo-vertės funkcija, dar vadinama Q-funkcija, apibrėžia tikėtiną kaupiamąjį atlygį, atlikus veiksmą a būsenoje s ir toliau sekant strategiją π [SB18]:

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R_{t+1} \mid S_0 = s, A_0 = a\right] \tag{1}$$

kur:

- $Q^{\pi}(s,a)$ – veiksmo a vertė būsenoje s pagal strategiją π
- $\mathbb{E}_{\pi}$ – tikėtina reikšmė sekant strategiją π
- γ – nuolaidos faktorius, apibrėžiantis ateities atlygių svarbą
- R_{t+1} – atlygis laiko žingsnyje $t+1$
- $\sum_{t=0}^{\infty} \gamma^t R_{t+1}$ – diskontuotas kaupiamasis atlygis

1.3.2. Temporal Difference mokymasis

Temporal Difference (TD) mokymasis yra RL metodų klasė, kuri atnaujina vertės įverčius remiantis dalinėmis patirties sekvencijomis, nelaukiant viso epizodo pabaigos [SB18]. Bendra TD atnaujinimo taisyklė išreiškiama formule:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha \cdot \left[\underbrace{r_{t+1} + \gamma \cdot V(s_{t+1})}_{\text{TD tikslas}} - Q(s_t,a_t) \right] \tag{2}$$

kur:

- $\alpha \in (0,1]$ – mokymosi greitis (angl. *learning rate*), kontroliuojantis, kiek naujos informacijos įtraukiama į Q įvertį
- r_{t+1} – gautas atlygis po veiksmo a_t atlikimo
- γ – nuolaidos faktorius (angl. *discount factor*)
- $V(s_{t+1})$ – kitos būsenos vertė (skiriasi priklausomai nuo konkretaus algoritmo)
- $[\text{TD tikslas} - Q(s_t,a_t)]$ – TD paklaida, matuojanti skirtumą tarp tikėtino ir faktinio įverčio

1.4. TD valdymo algoritmai

Šiame skyriuje pateikiami trys pagrindiniai TD valdymo algoritmai, naudojami šiame darbe: Q-Learning, SARSA ir Expected-SARSA. Algoritmų pseudokodas pateikiamas remiantis Sutton ir Barto [SB18] vadovėliu, o Expected-SARSA – van Seijen ir kt. [SHW+09] tyrimu.

1.4.1. Q-Learning algoritmas

Q-Learning yra off-policy TD valdymo algoritmas, pasiūlytas Watkins [Wat89]. Jo pagrindinė savybė – naudoti maksimalią kitos būsenos Q reikšmę, nepriklausomai nuo faktiškai pasirinkto veiksmo:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t) \right] \tag{3}$$

kur:

- $Q(S_t, A_t)$ – veiksmo-vertės funkcija: tikėtinas kaupiamasis atlygis atlikus veiksmą A_t būsenoje S_t
- $\alpha \in (0,1]$ – mokymosi greitis, kontroliuojantis naujų įverčių įtaką
- R_{t+1} – atlygis, gautas po veiksmo A_t atlikimo
- $\gamma \in [0,1]$ – nuolaidos faktorius, apibrėžiantis ateities atlygių svarbą
- S_{t+1} – nauja būsena, į kurią pereita po veiksmo
- $\max_a Q(S_{t+1}, a)$ – didžiausia Q reikšmė kitoje būsenoje (nepriklausomai nuo faktiškai pasirinkto veiksmo)

1 algoritmas. Q-Learning algoritmas [SB18]

Inicializuoti $Q(s,a)$ visiems $s \in \mathcal{S}$, $a \in \mathcal{A}$ ▷ pvz., ε-greedy strategiją
for kiekvienas epizodas **do**
    Inicializuoti būseną S
    for kiekvienas epizodo žingsnis **do**
        Pasirinkti veiksmą A iš S naudojant Q išvestą strategiją (pvz., ε-greedy)
        Atlikti veiksmą A, stebėti apdovanojimą R ir naują būseną S'
        $Q(S,A) \leftarrow Q(S,A) + \alpha\left[R + \gamma \max_a Q(S',a) - Q(S,A)\right]$
        $S \leftarrow S'$
    end for
end for

1.4.2. SARSA algoritmas

SARSA yra on-policy TD valdymo algoritmas, pasiūlytas Rummery ir Niranjan [RN94]. Pavadinimas kyla iš sekos $(S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1})$. Skirtingai nuo Q-Learning, SARSA naudoja faktiškai pasirinktą kitą veiksmą:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\left[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)\right] \tag{4}$$

kur:

- A_{t+1} – faktiškai pasirinktas kitas veiksmas pagal esamą strategiją (skirtumas nuo Q-Learning)
- $Q(S_{t+1}, A_{t+1})$ – kito veiksmo Q reikšmė (ne maksimali, o faktinio pasirinkto veiksmo)
- Kiti kintamieji (α, γ, R_{t+1}, S_{t+1}) apibrėžti kaip Q-Learning algoritme

2 algoritmas. SARSA algoritmas [SB18]

Inicializuoti $Q(s,a)$ visiems $s \in \mathcal{S}$, $a \in \mathcal{A}$ (pvz., $Q(s,a) = 0$)
for kiekvienas epizodas **do**
    Inicializuoti būseną S
    Pasirinkti veiksmą A iš S naudojant Q išvestą strategiją ▷ pvz., ε-greedy strategiją
    for kiekvienas epizodo žingsnis **do**
        Atlikti veiksmą A, stebėti apdovanojimą R ir naują būseną S'
        Pasirinkti veiksmą A' iš S' naudojant Q išvestą strategiją
        $Q(S,A) \leftarrow Q(S,A) + \alpha\left[R + \gamma Q(S',A') - Q(S,A)\right]$
        $S \leftarrow S'$; $A \leftarrow A'$
    end for
end for

1.4.3. Expected-SARSA algoritmas

Expected-SARSA yra SARSA variacija, kuri vietoj vieno pasirinkto veiksmo Q reikšmės naudoja tikėtiną Q reikšmę pagal dabartinę strategiją [SHW+09]. Tai sumažina dispersiją, nes nebepriklausoma nuo atsitiktinio kito veiksmo parinkimo:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \sum_a \pi(a|S_{t+1}) Q(S_{t+1}, a) - Q(S_t, A_t) \right] \tag{5}$$

kur:

- $\pi(a|S_{t+1})$ – strategija: tikimybė pasirinkti veiksmą a būsenoje S_{t+1}
- $\sum_a \pi(a|S_{t+1}) Q(S_{t+1}, a)$ – tikėtina Q reikšmė pagal strategiją π (visų veiksmų svertinis vidurkis)
- Kiti kintamieji apibrėžti kaip Q-Learning algoritme

ε-greedy strategijai tikėtina Q reikšmė apskaičiuojama:

$$\mathbb{E}_\pi[Q(S_{t+1}, \cdot)] = (1 - \varepsilon) \max_a Q(S_{t+1}, a) + \varepsilon \cdot \frac{1}{|\mathcal{A}|} \sum_a Q(S_{t+1}, a) \tag{6}$$

kur:

- $\varepsilon \in [0,1]$ – tyrinėjimo tikimybė
- $(1 - \varepsilon)$ – tikimybė pasirinkti geriausią veiksmą (išnaudojimas)
- $|\mathcal{A}|$ – veiksmų aibės dydis (galimų veiksmų skaičius)
- $\frac{\varepsilon}{|\mathcal{A}|}$ – tikimybė pasirinkti bet kurį konkretų veiksmą tyrinėjimo metu

3 algoritmas. Expected-SARSA algoritmas [SHW+09]

Inicializuoti $Q(s,a)$ visiems $s \in \mathcal{S}$, $a \in \mathcal{A}$ (pvz., $Q(s,a) = 0$)
for kiekvienas epizodas **do**
    Inicializuoti būseną S
    for kiekvienas epizodo žingsnis **do**
        Pasirinkti veiksmą A iš S naudojant Q išvestą strategiją (pvz., ε-greedy)
        Atlikti veiksmą A, stebėti apdovanojimą R ir naują būseną S'
        Apskaičiuoti tikėtiną Q: $\bar{Q} \leftarrow \sum_a \pi(a|S') Q(S', a)$
        $Q(S, A) \leftarrow Q(S, A) + \alpha \left[R + \gamma \bar{Q} - Q(S, A) \right]$
        $S \leftarrow S'$
    end for
end for

1.4.4. Double Q-Learning algoritmas

Double Q-Learning yra Q-Learning modifikacija, pasiūlyta van Hasselt [Has10], skirta spręsti Q reikšmių pervertinimo (angl. *overestimation*) problemą. Standartinis Q-Learning naudoja tą

pačią Q funkciją ir veiksmui pasirinkti, ir jo vertei įvertinti, kas sistemingai veda prie per didelių Q įverčių. Double Q-Learning sprendžia šią problemą naudodamas dvi nepriklausomas Q funkcijas:

$$Q_1(S_t, A_t) \leftarrow Q_1(S_t, A_t) + \alpha \left[R_{t+1} + \gamma Q_2(S_{t+1}, \arg\max_a Q_1(S_{t+1}, a)) - Q_1(S_t, A_t) \right] \quad (7)$$

kur:

- Q_1, Q_2 – dvi nepriklausomos veiksmo-vertės funkcijos
- $\arg\max_a Q_1(S_{t+1}, a)$ – geriausias veiksmas pagal Q_1 (veiksmo pasirinkimas)
- $Q_2(S_{t+1}, \cdot)$ – veiksmo vertės įvertinimas naudojant kitą Q funkciją
- Kiti kintamieji apibrėžti kaip Q-Learning algoritme

Kiekviename žingsnyje su 50% tikimybe atnaujinama Q_1 arba Q_2, sukeičiant jų vaidmenis.

4 algoritmas. Double Q-Learning algoritmas [Has10]

Inicializuoti $Q_1(s,a)$ ir $Q_2(s,a)$ visiems $s \in \mathcal{S}$, $a \in \mathcal{A}$
for kiekvienas epizodas **do**
    Inicializuoti būseną S
    for kiekvienas epizodo žingsnis **do**
        Pasirinkti veiksmą A iš S naudojant $(Q_1 + Q_2)$ išvestą strategiją
        Atlikti veiksmą A, stebėti atlygį R ir naują būseną S'
        if su tikimybe 0.5 **then**
            $Q_1(S, A) \leftarrow Q_1(S, A) + \alpha \left[R + \gamma Q_2(S', \arg\max_a Q_1(S', a)) - Q_1(S, A) \right]$
        else
            $Q_2(S, A) \leftarrow Q_2(S, A) + \alpha \left[R + \gamma Q_1(S', \arg\max_a Q_2(S', a)) - Q_2(S, A) \right]$
        end if
        $S \leftarrow S'$
    end for
end for

1.4.5. SARSA(λ) algoritmas

SARSA(λ) yra SARSA išplėtimas su tinkamumo pėdsakais (angl. *eligibility traces*), leidžiantis efektyviau paskirstyti atlygį per kelis ankstesnius veiksmus [SB18; SS96]. Parametras $\lambda \in [0,1]$ kontroliuoja, kiek toli į praeitį paskirstomas atlygis:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \delta_t e_t(S_t, A_t) \quad (8)$$

kur:

- $\delta_t = R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)$ – TD paklaida
- $e_t(s, a)$ – tinkamumo pėdsakas būsenai-veiksmui porai
- $\lambda \in [0,1]$ – pėdsakų skilimo parametras

Tinkamumo pėdsakai atnaujinami kiekviename žingsnyje:

$$e_t(s,a) = \begin{cases} \gamma\lambda e_{t-1}(s,a) + 1 & \text{jei } s = S_t \text{ ir } a = A_t \\ \gamma\lambda e_{t-1}(s,a) & \text{priešingu atveju} \end{cases} \tag{9}$$

Kai $\lambda = 0$, SARSA(λ) tampa standartiniu SARSA. Kai $\lambda = 1$, algoritmas artėja prie Monte Carlo metodo.

5 algoritmas. SARSA(λ) algoritmas [SB18]

Inicializuoti $Q(s,a)$ visiems $s \in \mathcal{S}$, $a \in \mathcal{A}$
for kiekvienas epizodas **do**
    Inicializuoti $e(s,a) = 0$ visiems s, a
    Inicializuoti būseną S, pasirinkti veiksmą A
    for kiekvienas epizodo žingsnis **do**
        Atlikti veiksmą A, stebėti atlygį R ir naują būseną S'
        Pasirinkti veiksmą A' iš S' naudojant Q išvestą strategiją
        $\delta \leftarrow R + \gamma Q(S', A') - Q(S, A)$
        $e(S, A) \leftarrow e(S, A) + 1$
        for visiems $s \in \mathcal{S}$, $a \in \mathcal{A}$ **do**
            $Q(s, a) \leftarrow Q(s, a) + \alpha\delta \cdot e(s, a)$
            $e(s, a) \leftarrow \gamma\lambda \cdot e(s, a)$
        end for
        $S \leftarrow S'$; $A \leftarrow A'$
    end for
end for

1.4.6. Algoritmų palyginimas

Remiantis Sutton ir Barto [SB18], Van Seijen ir kt. [SHW+09], Van Hasselt [Has10] bei Singh ir Sutton [SS96] darbais, pagrindiniai skirtumai tarp penkių TD valdymo algoritmų pateikti 1 lentelėje.

1 lentelė. TD valdymo algoritmų palyginimas (sudaryta autoriaus remiantis šaltiniais)

Savybė	Q-Learning	SARSA	Exp.-SARSA	Double Q	SARSA(λ)
Tipas	Off-policy	On-policy	On/Off	Off-policy	On-policy
Kitos būs. Q	$\max_a Q$	$Q(S', A')$	$\mathbb{E}_\pi[Q]$	$Q_2(\arg\max Q_1)$	$Q(S', A')$
Dispersija	Maža	Aukšta	Maža	Maža	Vidutinė
Pervertinimas	Taip	Ne	Ne	Ne	Ne
Q lentelių sk.	1	1	1	2	1
Papild. param.	-	-	-	-	λ
Atmintis	Maža	Maža	Maža	Dviguba	Didelė

Van Seijen ir kt. [SHW+09] įrodė, kad Expected-SARSA konverguoja tomis pačiomis sąlygomis kaip SARSA, bet pasiekia geresnius rezultatus deterministinėse aplinkose dėl nulinės

dispersijos. Van Hasselt [Has10] parodė, kad Double Q-Learning efektyviai sprendžia Q reikšmių pervertinimo problemą stochastinėse aplinkose. Singh ir Sutton [SS96] nustatė, kad SARSA(λ) su tinkamumo pėdsakais gali reikšmingai pagreitinti mokymąsi užduotyse su uždelstais atlygiais.

1.5. Skatinamojo mokymosi tinkamumas AAC sistemoms

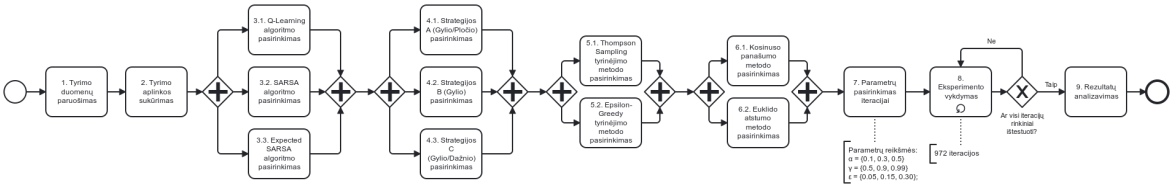
Skatinamasis mokymasis yra ypač tinkamas AAC ir PECS sistemoms dėl kelių priežasčių:

1. Natūralus grįžtamasis ryšys: vaiko reakcija (pasirinkimas arba atmetimas) tiesiogiai atitinka RL atlygio/baudos mechanizmą
2. Prisitaikymas: sistema mokosi iš kiekvienos sąveikos ir prisitaiko prie individualių vaiko poreikių
3. Tyrinėjimo ir išnaudojimo balansas: sistema gali balansuoti tarp žinomų efektyvių kortelių naudojimo ir naujų galimybių tyrinėjimo
4. Konteksto įskaitymas: būsena gali apimti ankstesnių atmetimų informaciją, leidžiant sistemai keisti strategiją
5. Ilgalaikis mokymasis: Q reikšmės kaupia informaciją apie kortelių efektyvumą per ilgą laiką

2. Tyrimo metodika

2.1. Tyrimo schema

Tyrimo eiga sudaryta kaip nuoseklus iteracinis procesas, apimantis duomenų paruošimą, tyrimo aplinkos sukūrimą, skirtingų skatinamojo mokymosi algoritmų ir strategijų taikymą, eksperimento vykdymą bei rezultatų analizę. Visa tyrimo eiga sudaryta pagal devynis pagrindinius etapus, pateiktus 1 paveiksle.



1 pav. Tyrimo eigos schema

2.2. Tyrimo duomenų paruošimas

Duomenų bazę sudaro 17 vaisių/produktų, kurių kiekvienas aprašomas 6 savybėmis skalėje $[0, 1]$. Savybių aprašymas pateiktas 2 lentelėje, o pilna produktų matrica – 3 lentelėje.

2 lentelė. Produktų savybių aprašymas

Savybė	Aprašymas	Skalė
Kietumas	Vaisiaus fizinis kietumas	0 (minkštas) - 1 (kietas)
Saldumas	Saldumas	0 (nesaldus) - 1 (labai saldus)
Rūgštingumas	Rūgštumas	0 (nerūgštus) - 1 (labai rūgštus)
Forma	Formos neįprastumas	0 (apvalus) - 1 (pailgas)
Tekstūra	Paviršiaus tekstūra	0 (lygus) - 1 (šiurkštus)
Spalva	Spalvos tonas	0 (raudona) - 1 (mėlyna)

Savybių reikšmės buvo priskirtos subjektyviai, remiantis realiais vaisių fiziniais ir organoleptiniais požymiais. Kiekviena savybė buvo normalizuota į intervalą [0, 1], kur kraštinės reikšmės atitinka ekstremalius atvejus (pvz., 0 = visiškai minkštas, 1 = labai kietas).

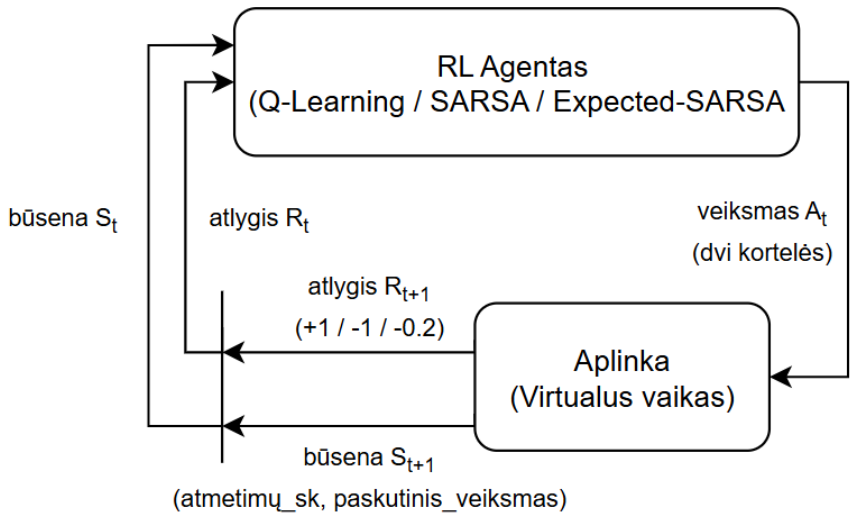
3 lentelė. Produktų savybių matrica

Produktas	Kietumas	Saldumas	Rūgšt.	Forma	Tekstūra	Spalva
Obuolys	0.8	0.7	0.4	0.0	0.2	0.00
Žaliasis obuolys	0.8	0.5	0.7	0.0	0.2	0.66
Kriaušė	0.6	0.8	0.2	0.3	0.3	0.50
Bananas	0.3	0.9	0.1	1.0	0.0	0.33
Persikai	0.4	0.8	0.2	0.0	0.8	0.15
Mangas	0.5	0.9	0.3	0.2	0.1	0.25
Apelsinas	0.6	0.7	0.5	0.0	0.4	0.20
Mandarinas	0.5	0.8	0.4	0.0	0.3	0.20
Arbūzas	0.3	0.8	0.1	0.0	0.0	0.00
Melionas	0.4	0.7	0.1	0.0	0.1	0.40
Ananasas	0.6	0.8	0.6	0.5	0.9	0.33
Kiviai	0.5	0.6	0.5	0.0	0.6	0.66
Vyšnios	0.7	0.7	0.3	0.0	0.1	0.00
Braškės	0.4	0.8	0.2	0.4	0.5	0.00
Mėlynės	0.5	0.6	0.4	0.0	0.1	1.00
Avietės	0.3	0.7	0.3	0.1	0.7	0.05
Vynuogės	0.6	0.8	0.2	0.0	0.0	0.80

Šios savybės leidžia skaičiuoti panašumus tarp produktų ir taikyti strategijas, pagrįstas semantiniu artumu.

2.3. Tyrimo aplinkos sukūrimas

Sukuriama skatinamojo mokymosi aplinka, modeliuojanti PECS kortelių parinkimo procesą. Tyrimo programinis kodas viešai prieinamas GitHub repozitorijoje. Sistema realizuoja klasikinę RL agento ir aplinkos sąveiką, formalizuotą kaip Markovo sprendimų procesą (MDP). PECS sistemos agento ir aplinkos sąveikos schema pateikta 2 paveiksle.



2 pav. PECS sistemos agento ir aplinkos sąveikos schema

2.3.1. MDP specifikacija

Būsenų aibė $\mathcal{S}$. Būsena apibrėžiama kaip pora:

$$S = (\text{atmetimų\_sk}, \text{paskutinis\_veiksmas}) \tag{10}$$

kur:

- atmetimų_sk $\in \{0, 1, 2, 3\}$ – kiek kartų vaikas atmetė korteles šioje sesijoje
- paskutinis_veiksmas $\in \{\texttt{start}, \texttt{success}, \texttt{rejection}\}$ – paskutinės sąveikos rezultatas

Iš viso galimos $4 \times 3 = 12$ unikalių būsenų.

Veiksmų aibė $\mathcal{A}$. Veiksmas – tai vienos kortelės pasirinkimas iš 17 galimų produktų kortelių. Q-lentelė saugo Q reikšmes kiekvienai kortelei atskirai: $Q(s, a)$, kur $a \in \{1, 2, \ldots, 17\}$. Kiekviename žingsnyje agentas pasirenka dvi korteles su aukščiausiomis Q reikšmėmis (arba pagal tyrinėjimo strategiją).

Atlygių funkcija R. Atlygių sistema apibrėžta taip:

- Atitiktis (vaikas pasirinko kortelę): $r = +1.0$ pasirinktai kortelei
- Neatitiktis (vaikas atmetė abi korteles):
  - Pirmoji kortelė (aukštesnė Q reikšmė): $r_1 = -1.0$
  - Antroji kortelė (žemesnė Q reikšmė): $r_2 = -0.2$

Asimetrinis baudų paskirstymas (-1.0 prieš -0.2) motyvuotas tuo, kad pirmoji kortelė buvo „pagrindinė" sistemos prognozė, o antroji – papildoma alternatyva. Tai leidžia sistemai greičiau išmokti vengti neteisingų „pagrindinių" pasirinkimų.

Dvigubas Q reikšmių atnaujinimas. Po neatitikties atnaujinamos abiejų rodomų kortelių Q reikšmės. Tai skiriasi nuo standartinio RL, kur atnaujinama tik vieno veiksmo Q reikšmė. Adaptacija būtina, nes PECS sistemoje vaikas mato abi korteles ir jas abi atmeta – todėl abi kortelės gauna neigiamą grįžtamąjį ryšį.

2.3.2. Virtualus vaikas ir norų sąrašas

Eksperimentui sukurtas virtualus vaikas su fiksuotu 100 sesijų norų sąrašu. Norų sąrašas buvo aptartas ir sudarytas bendradarbiaujant su specialiuoju pedagogu, siekiant užtikrinti, kad pasiskirstymas atspindėtų realius vaiko su autizmo spektro sutrikimu pasirinkimus. Sąraše dominuoja tam tikri produktai (pvz., Bananas – 43% sesijų), o kiti pasirodo rečiau, kas atitinka tipinį autizmo spektro vaikų elgesį, kur teikiama pirmenybė tam tikriems maisto produktams.

Virtualus vaikas elgiasi deterministiškai – jei norimas produktas yra tarp rodomų kortelių, jis jį pasirenka. Priešingu atveju, atmetamos abi kortelės:

6 algoritmas. Virtualaus vaiko reakcijos algoritmas

```
1: function ReactToCards(kortelė1, kortelė2, norimas)
2:     if norimas = kortelė1 then
3:         return kortelė1                          ▷ Atitiktis – pasirinko pirmą
4:     else if norimas = kortelė2 then
5:         return kortelė2                          ▷ Atitiktis – pasirinko antrą
6:     else
7:         return NULL                              ▷ Neatitiktis – nėra norimo
8:     end if
9: end function
```

Deterministinis vaiko elgesys užtikrina eksperimento reprodukuojamumą ir leidžia tiksliai palyginti skirtingų algoritmų ir parametrų efektyvumą.

2.4. Mokymosi algoritmo pasirinkimas

Šiam eksperimentui pasirinkti trys algoritmai: Q-Learning, SARSA ir Expected-SARSA. Pasirinkimą lėmė PECS kortelių parinkimo aplinkos specifika – sukurta sistema yra deterministinė, t.y. virtualus vaikas visada vienodai reaguoja į tas pačias korteles. Tokioje aplinkoje Q-Learning pervertinimo problema yra minimali, todėl Double Q-Learning papildomas sudėtingumas (dviejų Q lentelių palaikymas) neduoda reikšmingo pranašumo. Be to, atlygiai PECS sistemoje gaunami iš karto po kiekvieno veiksmo, o ne uždelstai epizodo pabaigoje, todėl SARSA(λ) tinkamumo pėdsakų mechanizmas nėra būtinas ir tik padidintų skaičiavimų apimtį bei atminties poreikį.

Šių algoritmų teoriniai pagrindai ir pseudokodas pateikti 1.4 skyriuje: Q-Learning (Algoritmas 1), SARSA (Algoritmas 2) ir Expected-SARSA (Algoritmas 3). Šiame skyriuje aprašoma, kaip šie algoritmai buvo adaptuoti PECS kortelių parinkimo sistemai.

2.4.1. Q-Learning taikymas

Q-Learning algoritmas PECS sistemoje realizuotas identiškai teoriniam pseudokodui (žr. Algoritmą 1 ir formulę 3). Kitos būsenos Q reikšmė skaičiuojama kaip maksimali reikšmė iš visų 17 galimų kortelių.

2.4.2. SARSA taikymas

SARSA algoritmas PECS sistemoje adaptuotas dėl dviejų kortelių veiksmų erdvės. Standartinis SARSA (žr. Algoritmą 2 ir formulę 4) naudoja vieną kitą veiksmą A', tačiau PECS sistemoje kiekviename žingsnyje pateikiamos dvi kortelės. Todėl kitos būsenos Q reikšmė skaičiuojama kaip dviejų kitų veiksmų vidurkis:

$$Q_{\text{next}} = \frac{Q(S', A'_1) + Q(S', A'_2)}{2} \tag{11}$$

kur A'_1 ir A'_2 – dvi kortelės, kurias agentas pasirinktų kitame žingsnyje pagal esamą strategiją.

2.4.3. Expected-SARSA taikymas

Expected-SARSA algoritmas PECS sistemoje realizuotas identiškai teoriniam pseudokodui (žr. Algoritmą 3 ir formules 5, 6). Su ε-greedy strategija, tikėtina Q reikšmė skaičiuojama kaip svertinis vidurkis: $(1 - \varepsilon)$ dalis tenka maksimaliai Q reikšmei (išnaudojimas), o ε dalis padalinama tolygiai visiems veiksmams (tyrinėjimas).

2.5. Strategijos pasirinkimas

Sistemoje realizuotos trys kortelių parinkimo strategijos, kurios skiriasi elgsena po neatitikties. Strategijos apibrėžia, kaip sistema reaguoja į vaiko atmetimą ir kokias korteles siūlo kitame bandyme.

2.5.1. Strategija A (Gylio / Pločio)

- Rinkinys 1 (atmetimų = 0): Dažniausiai pasirenkamos 2 kortelės pagal Q reikšmes arba Thompson Sampling
- Rinkinys 2+ (atmetimų $\geq$ 1): 1 panašiausia į atmestąsias kortelė + 1 priešingiausia (mažiausiai panaši) kortelė

Ši strategija derina *gylio* (panašumo) ir *pločio* (priešingumo) paiešką. Po neatitikties sistema siūlo vieną panašią kortelę (gilinimasis į tą pačią kategoriją) ir vieną priešingą (išplėtimas į kitą kategoriją), taip balansuojant tarp dviejų hipotezių apie vaiko pageidavimus.

2.5.2. Strategija B (Gylio)

- Rinkinys 1: Dažniausiai pasirenkamos 2 kortelės
- Rinkinys 2+: 2 panašiausios į atmestąsias kortelės

Strategija B remiasi prielaida, kad jei vaikas atmetė tam tikrą kortelę, jis gali norėti kažko panašaus. Sistema „gilina" paiešką toje pačioje semantinėje kategorijoje, t. y. jei atmetė obuolį, siūlo kriaušę ar žalią obuolį.

2.5.3. Strategija C (Gylio / Dažnio)

- Rinkinys 1: Dažniausiai pasirenkamos 2 kortelės
- Rinkinys 2 (atmetimų = 1): 1 panašiausia + 1 iš dažniausiai pasirinktų kortelių (dar nerodyta)
- Rinkinys 3 (atmetimų = 2): 2 panašiausios

Strategija C bando subalansuoti panašumo naudojimą (gylis) su dažnio informacija iš Q reikšmių (dažniausiai pasirinktos kortelės). Pirmame pakartotiniame bandyme siūlo vieną panašią ir vieną dažnai pasirenkamą kortelę, o jei ir tai nepavyksta – pereina prie grynos gylio strategijos.

2.6. Tyrinėjimo metodo pasirinkimas

Viena iš fundamentalių RL problemų yra tyrinėjimo ir išnaudojimo balansas (angl. *exploration-exploitation trade-off*). Agentas turi balansuoti tarp [SB18]:

- Išnaudojimo: naudoti jau turimą žinojimą siekiant maksimalaus atlygio
- Tyrinėjimo: bandyti naujus veiksmus siekiant atrasti potencialiai geresnius sprendimus

Šiam eksperimentui pasirinkti du tyrinėjimo metodai: Epsilon-Greedy ir Thompson Sampling. Epsilon-Greedy yra klasikinis ir paprasčiausias metodas, plačiai naudojamas RL tyrimuose, todėl jo įtraukimas leidžia palyginti rezultatus su kitais darbais. Thompson Sampling pasirinktas kaip sudėtingesnis, Bajeso metodu pagrįstas metodas, kuris automatiškai pritaiko tyrinėjimo intensyvumą pagal neapibrėžtumo lygį. Tai ypač aktualu PECS sistemoje, kur skirtingos kortelės turi labai nevienodą pasirinkimo dažnį. Abu metodai kartu leidžia įvertinti, ar sudėtingesnis prisitaikantis tyrinėjimas duoda reikšmingą pranašumą prieš paprastą fiksuoto ε strategiją.

2.6.1. Thompson Sampling metodo pasirinkimas

Thompson Sampling yra Bajeso metodu pagrįsta tyrinėjimo strategija, kuri remiasi tikimybiniu modeliu apie kiekvieno veiksmo tikėtiną atlygį [RRK+18]. Skirtingai nuo ϵ-greedy, Thompson Sampling automatiškai balansuoja tyrinėjimą ir išnaudojimą pagal neapibrėžtumo lygį, todėl čia ϵ reikšmė nėra naudojama.

Algoritmas veikia taip [RRK+18]:

1. Kiekvienam veiksmui a palaikomas Beta pasiskirstymas $\text{Beta}(\alpha_a, \beta_a)$, kur α_a – atitikčių skaičius, β_a – neatitikčių skaičius
2. Kiekviename žingsnyje imama atsitiktinė reikšmė $\theta_a \sim \text{Beta}(\alpha_a, \beta_a)$ kiekvienam veiksmui
3. Pasirenkamas veiksmas su didžiausia atsitiktine reikšme:

$$a_t = \arg\max_a \theta_a, \quad \text{kur } \theta_a \sim \text{Beta}(\alpha_a + 1, \beta_a + 1) \tag{12}$$

Po atgalinio ryšio atnaujinami parametrai [RRK+18]:

$$\alpha_a \leftarrow \alpha_a + \mathbf{1}[\text{atitiktis}], \quad \beta_a \leftarrow \beta_a + \mathbf{1}[\text{neatitiktis}] \tag{13}$$

2.6.2. Epsilon-Greedy metodo pasirinkimas

Epsilon-greedy (ϵ-greedy) yra paprasčiausia ir plačiausiai naudojama tyrinėjimo strategija. Ji su tikimybe ϵ pasirenka atsitiktinį veiksmą, o su tikimybe $(1 - \epsilon)$ – geriausią žinomą veiksmą [SB18]:

$$a_t = \begin{cases} \text{atsitiktinis veiksmas iš } \mathcal{A} & \text{su tikimybe } \epsilon \\ \arg\max_a Q(s_t, a) & \text{su tikimybe } 1 - \epsilon \end{cases} \tag{14}$$

kur:

- $\epsilon \in [0,1]$ – tyrinėjimo tikimybė (tyrinėjimo intensyvumas)
- $\arg\max_a Q(s_t, a)$ – veiksmas su didžiausia Q reikšme dabartinėje būsenoje

2.7. Panašumo ir atstumo metodų pasirinkimas

Kortelių parinkimo strategijose naudojamas panašumo skaičiavimas tarp produktų savybių vektorių. Panašumo metrikos leidžia sistemai rasti korteles, kurios yra semantiškai artimos atmestoms kortelėms (potencialiai vaikas nori kažko panašaus) arba priešingos (vaikas nori kažko visiškai kitokio).

Šiam eksperimentui pasirinktos dvi plačiausiai naudojamos panašumo metrikos: Kosinuso panašumas ir Euklido atstumas. Kosinuso panašumas matuoja kampą tarp vektorių ir yra nepriklausomas nuo jų ilgio, todėl tinka palyginti produktus, kurių savybių intensyvumai skiriasi. Euklido atstumas matuoja tiesioginį geometrinį atstumą tarp taškų ir yra aiškiau suprantamas. Šių dviejų metrikų palyginimas leidžia įvertinti, ar panašumo skaičiavimo būdas turi reikšmingą įtaką PECS sistemos efektyvumui, ar rezultatai yra atsparūs metrikos pasirinkimui.

2.7.1. Kosinuso panašumo metodo pasirinkimas

Kosinuso panašumas yra plačiai naudojama metrika informacijos paieškos ir duomenų gavybos srityse, matuojanti kampą tarp dviejų vektorių n-dimensinėje erdvėje [MRS08]. Ši metrika yra nepriklausoma nuo vektorių ilgio, todėl tinka palyginti objektus su skirtingais savybių intensyvumais [HKP12]:

$$\text{cosine}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \times \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{15}$$

kur:

- $\mathbf{A}, \mathbf{B}$ – produktų savybių vektoriai
- $\mathbf{A} \cdot \mathbf{B}$ – skaliarinė sandauga (dot product)
- $\|\mathbf{A}\|$ – vektoriaus Euklido norma
- Rezultatas: $[0, 1]$, kur 1 = visiškai identiški vektoriai (0° kampas), 0 = nepriklausomi (90° kampas)

2.7.2. Euklido atstumo metodo pasirinkimas

Euklido atstumas yra klasikinė metrika, matuojanti tiesioginį geometrinį atstumą tarp dviejų taškų n-dimensinėje erdvėje [HKP12]. Tai yra viena iš dažniausiai naudojamų atstumo metrikų duomenų gavime ir mašininiame mokyme:

$$\text{distance}(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2} \tag{16}$$

Kadangi atstumas yra atvirkščiai proporcingas panašumui, jis konvertuojamas į panašumo metriką [HKP12]:

$$\text{similarity} = \frac{1}{1 + \text{distance}(\mathbf{A}, \mathbf{B})} \tag{17}$$

kur rezultatas yra intervale $(0, 1]$, ir 1 reiškia identiškus vektorius (atstumas = 0).

2.8. Parametrų parinkimas iteracijai

Trys pagrindiniai skatinamojo mokymosi parametrai turi skirtingą įtaką algoritmo elgsenai:

Mokymosi greitis $\alpha \in (0, 1]$ kontroliuoja, kiek naujai gauta informacija pakeičia seną Q reikšmę. Literatūroje rekomenduojamos reikšmės paprastai yra intervale 0.1-0.5 [SB18]. Per didelis mokymosi greitis (pvz., $\alpha = 0.8$ ar $\alpha = 1.0$) sukelia Q reikšmių nestabilumą – agentas „pamiršta" ankstesnę patirtį ir per daug reaguoja į kiekvieną naują atlygį. Per mažas greitis (pvz., $\alpha = 0.01$) lėtina konvergavimą. Eksperimentui pasirinktos reikšmės 0.1, 0.3 ir 0.5 apima rekomenduojamą diapazoną nuo konservatyvaus iki agresyvaus mokymosi.

Nuolaidos faktorius $\gamma \in [0, 1]$ apibrėžia, kaip stipriai agentas vertina būsimus atlygius lyginant su dabartiniais. PECS sistemoje sesijos yra trumpos (maksimaliai 3 bandymai), todėl labai maža γ (pvz., 0.1 ar 0.3) būtų per trumparegiška – agentas nevertintų, kaip dabartinis pasirinkimas paveiks sekančius bandymus. Standartinės reikšmės literatūroje yra 0.9-0.99 [SB18]. Eksperimentui pasirinktos 0.5 (kaip žemesnė riba palyginimui), 0.9 (subalansuotas požiūris) ir 0.99 (beveik pilnas ateities vertinimas).

Tyrinėjimo tikimybė $\varepsilon \in [0, 1]$ (taikoma tik Epsilon-Greedy metodui) nustato, kaip dažnai agentas renkasi atsitiktinį veiksmą vietoj geriausio žinomo. Per didelis ε (pvz., 0.5 ar 0.7) reikštų, kad agentas daugiau laiko elgiasi atsitiktinai nei naudoja išmoktą žinojimą – tai neefektyvu ir paneigia mokymosi prasmę. Standartinės reikšmės literatūroje yra 0.01-0.3 [SB18]. Eksperimentui pasirinktos 0.05 (minimalus tyrinėjimas, daugiausia išnaudojimas), 0.15 (subalansuotas požiūris) ir 0.30 (aktyvesnis tyrinėjimas, kaip viršutinė praktiškai naudinga riba).

Eksperimentui sukurta išsami parametrų matrica, apimanti visas galimas kombinacijas. Testuojamumo požiūriu pasirinktas „plataus" testavimo metodas – tiriama daug skirtingų parametrų kombinacijų, siekiant nustatyti optimalias reikšmes.

4 lentelė. Eksperimento parametrų erdvė

Parametras	Reikšmės
Modelis	Q-Learning, SARSA, Expected-SARSA
Alpha (α) – mokymosi greitis	0.1, 0.3, 0.5
Gamma (γ) – nuolaidos faktorius	0.5, 0.9, 0.99
Epsilon (ϵ) – tyrinėjimo tikimybė	0.05, 0.15, 0.30
Exploration metodas	Epsilon-Greedy, Thompson Sampling
Strategija	A (Gylio/Pločio), B (Gylio), C (Gylio/Dažnio)
Panašumo metodas	Kosinuso panašumas, Euklido atstumas

2.9. Eksperimento vykdymas

2.9.1. Eksperimento iteracijų matrica

Kiekviena eksperimento iteracija vykdoma su viena konkrečia parametrų kombinacija. Kombinacija sudaroma pasirenkant po vieną reikšmę iš kiekvieno parametro: pirmiausia fiksuojamas vienas iš 3 algoritmų, tada viena iš 3 α reikšmių, viena iš 3 γ reikšmių, viena iš 3 ε reikšmių, vienas iš 2 tyrinėjimo metodų, viena iš 3 strategijų ir vienas iš 2 panašumo metodų. Kadangi kiekvienas parametras parenkamas nepriklausomai nuo kitų, bendras galimų kombinacijų skaičius apskaičiuojamas sudauginant visų parametrų reikšmių skaičius: $3 \times 3 \times 3 \times 3 \times 2 \times 3 \times 2 = 972$ unikalios kombinacijos. Eksperimento rezultatai saugomi matricoje $M_{972 \times m}$, kur kiekviena eilutė i (kur $i \in \{1, 2, \ldots, 972\}$) atitinka unikalią parametrų kombinaciją:

$$M = \begin{bmatrix} 1 & \text{modelis}_1 & \alpha_1 & \gamma_1 & \epsilon_1 & \ldots & \text{atitiktis}_1 \\ 2 & \text{modelis}_2 & \alpha_2 & \gamma_2 & \epsilon_2 & \ldots & \text{atitiktis}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 972 & \text{modelis}_{972} & \alpha_{972} & \gamma_{972} & \epsilon_{972} & \ldots & \text{atitiktis}_{972} \end{bmatrix} \tag{18}$$

Matrica prasideda nuo indekso 1 (pirmoji kombinacija) ir baigiasi indeksu 972 (paskutinė kombinacija).

2.9.2. Sesijos eiga

Sesija – tai vienas sąveikos ciklas tarp PECS sistemos ir virtualaus vaiko, kurio metu sistema bando parinkti kortelę, atitinkančią vaiko norą. Kiekvienos sesijos pradžioje iš norų sąrašo nustatomas vienas konkretus produktas, kurio vaikas „nori" šios sesijos metu. Eksperimente vykdomos 100 sesijų, kiekvienai sesijai priskiriant produktą pagal iš anksto sudarytą norų sąrašą.

Kiekviena sesija susideda iš maksimaliai 3 rinkinių (bandymų). Kiekviename rinkinyje sistema pateikia dvi korteles, o virtualus vaikas arba pasirenka vieną iš jų (jei norimas produktas yra tarp pateiktų), arba atmeta abi. Sesija laikoma sėkminga (atitikties atvejis), jei bent viename rinkinyje vaikas pasirenka kortelę:

7 algoritmas. Vienos sesijos eiga

norimas ← VaikoNoruSarasas[sesijos_nr]
atmetimai ← 0
for rinkinys := 1 to 3 **do**
    (k_1, k_2) ← ModelioPasirinkimas(būsena, strategija)
    rezultatas ← VaikasReaguoja$(k_1, k_2$, norimas)
    if rezultatas ≠ `NULL` **then**
        Atlygis ← $+1.0$
        break ▷ Sesija – atitiktis
    else
        Atlygis ← -1.0
        atmetimai ← atmetimai + 1
    end if
    AtnaujintiQ(būsena, (k_1, k_2), Atlygis)
end for

2.10. Rezultatų analizavimas

2.10.1. Vertinimo metrikos

Eksperimento rezultatai vertinami pagal šias pagrindines metrikas:

Atitikties rodiklis yra pagrindinė eksperimento metrika, rodanti sistemos efektyvumą. Ji apskaičiuojama kaip sėkmingų sesijų (kuriose bent viename iš 3 bandymų virtualusvaikas pasirinko pateiktą kortelę) dalis iš visų sesijų:

$$\text{Atitikties rodiklis} = \frac{\text{sėkmingų sesijų skaičius}}{\text{visų sesijų skaičius}} \times 100\% = \frac{n_{\text{atitiktis}}}{100} \times 100\% \tag{19}$$

kur $n_{\text{atitiktis}}$ – sesijų, kuriose norimas produktas (pagal vaiko norų sąrašą) buvo tarp pateiktų kortelių ir vaikas jį pasirinko, skaičius. Pavyzdžiui, jei iš 100 sesijų 65 baigėsi sėkmingai, atitikties rodiklis yra 65%.

Mokymosi kreivė parodo, kaip sistemos efektyvumas keičiasi laikui bėgant. Ji apskaičiuojama naudojant slenkančio vidurkio metodą – kiekvienam taškui skaičiuojamas paskutinių w sesijų atitikties vidurkis:

$$\text{Mokymosi kreivė}(t) = \frac{1}{w} \sum_{i=t-w+1}^{t} \mathbf{1}[\text{sesija } i \text{ sėkminga}] \tag{20}$$

kur w – slenkančio lango dydis (šiame eksperimente $w = 10$), o $\mathbf{1}[\cdot]$ – indikatoriaus funkcija, grąžinanti 1, jei sąlyga tenkinama, ir 0 priešingu atveju. Kylanti mokymosi kreivė rodo, kad sistema mokosi ir gerėja.

3. Tyrimo rezultatai

Šiame skyriuje pateikiami eksperimento rezultatai, gauti įvykdžius visas 972 parametrų kombinacijas. Kiekviena kombinacija buvo testuojama 100 sesijų simuliacija su virtualiu vaiku, iš viso sugeneruojant 222,608 sąveikos įrašus.

Visame šiame skyriuje sistemos efektyvumas išreiškiamas atitikties rodikliu (procentais), kuris parodo sėkmingų sesijų dalį iš 100 (žr. formulę 19).

3.1. Bendrieji rezultatai

Eksperimento metu buvo išbandytos visos 972 unikalios parametrų kombinacijos. Atitikties rodiklio statistika pateikta 5 lentelėje.

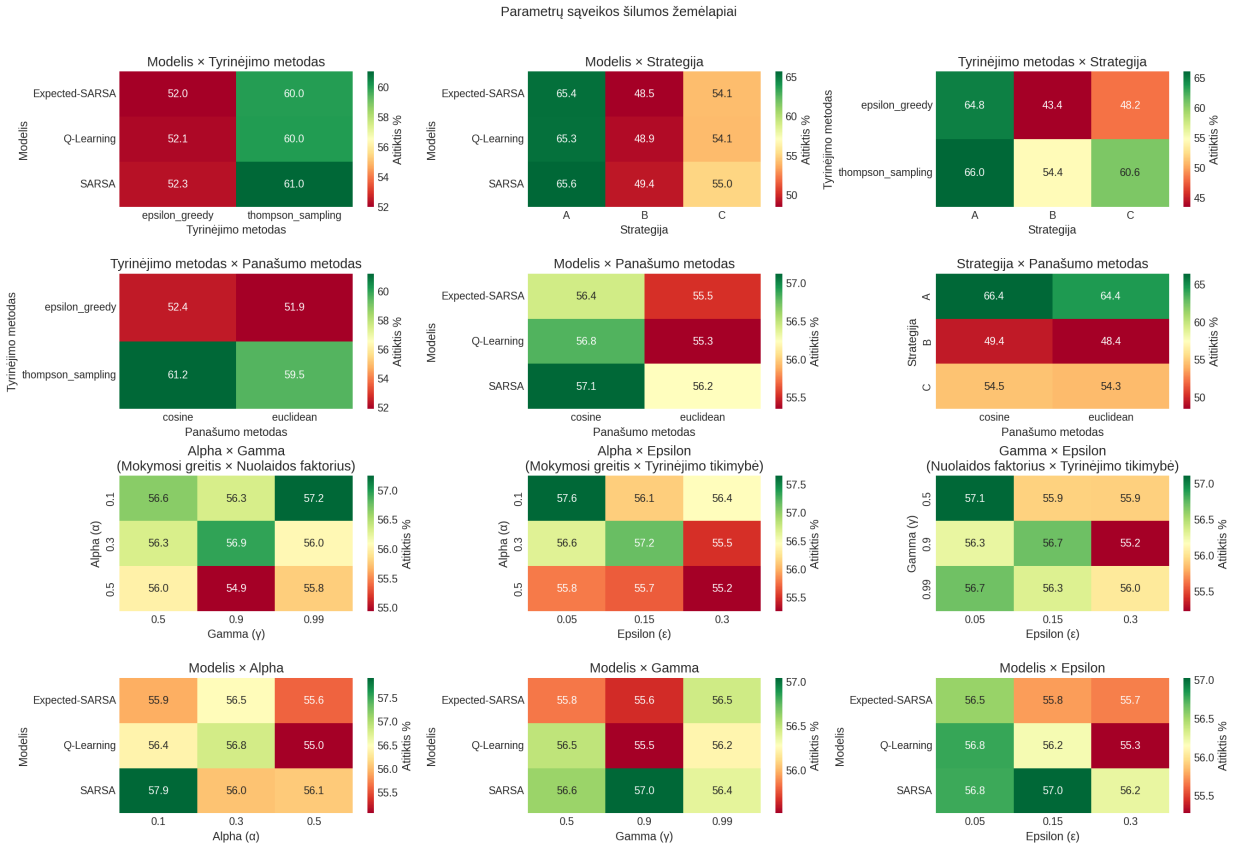
5 lentelė. Atitikties rodiklio suvestinė statistika

Statistika	Reikšmė
Minimumas	26.0%
Maksimumas	75.0%
Vidurkis	56.2%
Mediana	58.0%
Standartinis nuokrypis	10.0%

Analizuojant bendrą rezultatų pasiskirstymą, 75.4% visų sąveikų baigėsi neatitiktimi (virtualus vaikas atmetė siūlomas korteles), o 24.6% – atitiktimi (virtualus vaikas pasirinko vieną iš siūlomų kortelių). Šis santykis atspindi eksperimento sudėtingumą – sistemai reikėjo atspėti vieną iš 17 produktų, turint tik 2 korteles viename rinkinyje.

3.2. Parametrų įtakos analizė

Prieš detalią kiekvieno parametro analizę, svarbu įvertinti bendrą parametrų įtaką rezultatams. Parametrų sąveikos šilumos žemėlapiai (3 pav.) atskleidžia, kurie parametrai daro didžiausią įtaką atitikties rodikliui.



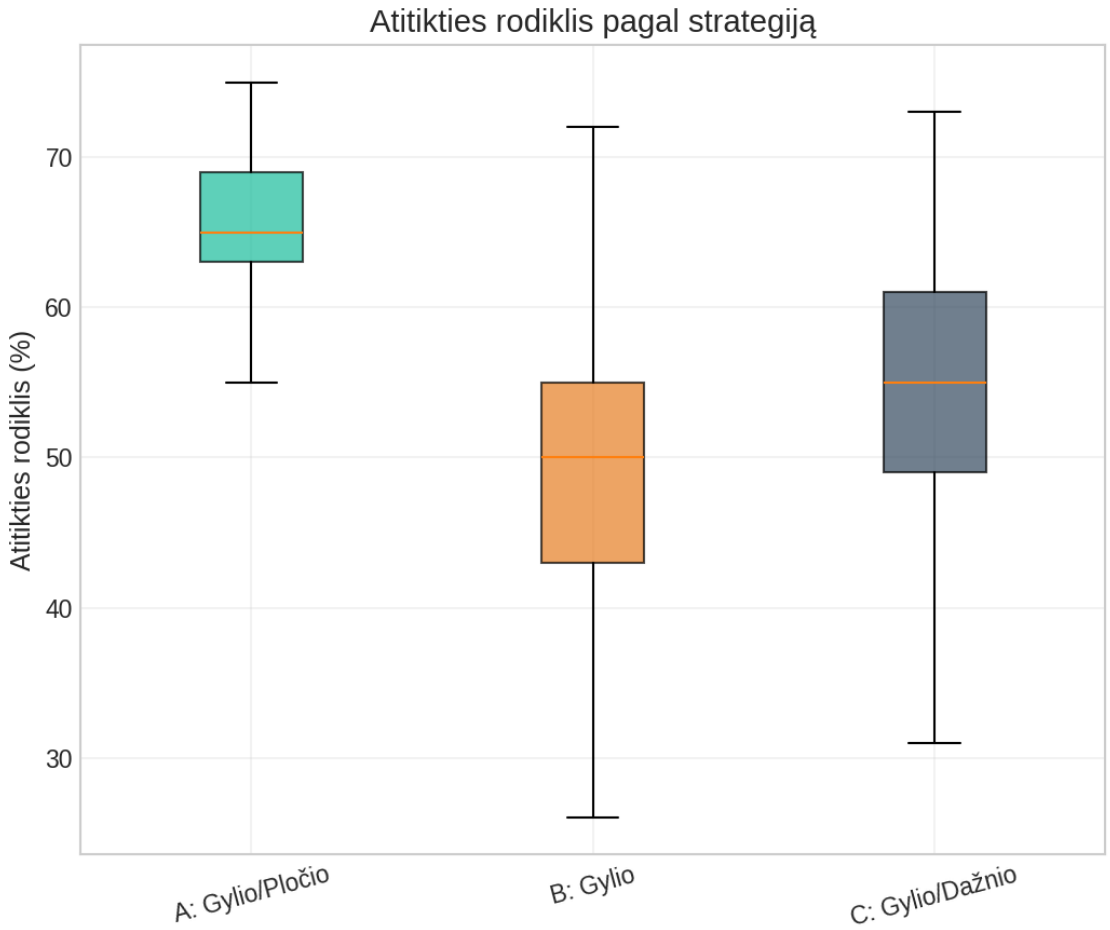
3 pav. Parametrų sąveikos šilumos žemėlapiai

Šilumos žemėlapiai aiškiai atskleidžia parametrų svarbos hierarchiją:

- Strategija – didžiausia įtaka: strategija A pasiekia 64-66%, tuo tarpu B ir C tik 43-55% (skirtumas iki 23 procentinių punktų)
- Tyrinėjimo metodas – didelė įtaka: Thompson Sampling pasiekia 59-66%, Epsilon-Greedy tik 48-65% (skirtumas iki 11 procentinių punktų)
- Algoritmas, α, γ, ϵ – minimali įtaka: visų reikšmės svyruoja 55-57% ribose (skirtumas <2 procentiniai punktai)
- Panašumo metodas – minimali įtaka: Kosinuso panašumas 56-66%, Euklido atstumas 55-64% (skirtumas <2 procentiniai punktai)

3.3. Strategijos

Kortelių parinkimo strategija pasirodė esanti svarbiausias parametras, lemiantis sistemos efektyvumą. Šilumos žemėlapis „Modelis × Strategija" rodo, kad nepriklausomai nuo pasirinkto algoritmo, strategija A nuosekliai pasiekia 65-66% atitikties rodiklį.



4 pav. Kortelių parinkimo strategijų efektyvumo palyginimas

6 lentelė. Strategijų efektyvumo statistika

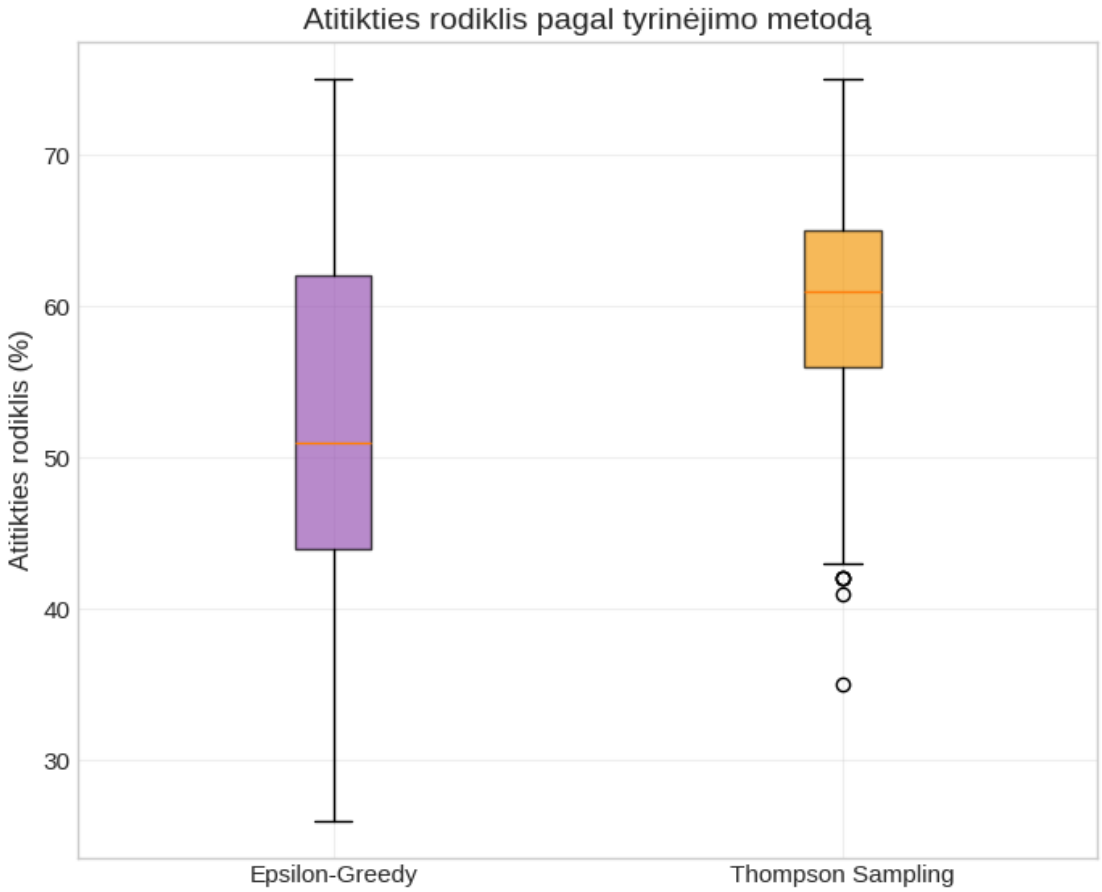
Strategija	Vidurkis	Std	Min	25%	50%	75%	Max
A (Gylio/Pločio)	65.40	4.12	53.0	63.0	66.0	68.0	75.0
B (Gylio)	48.90	7.89	26.0	43.0	49.0	55.0	65.0
C (Gylio/Dažnio)	54.40	8.45	32.0	48.0	55.0	61.0	71.0

Strategija A (Gylio/Pločio) pasiekė aukščiausią vidutinį atitikties rodiklį (65.40%) su mažiausia dispersija (std = 4.12). Strategijos B ir C pasiekė žymiai prastesnius rezultatus (48.90% ir 54.40%) su didesne dispersija, kas rodo jų nestabilumą.

Šie rezultatai patvirtina, kad diversifikuota paieška (kombinuojant panašumo ir priešingumo principus) yra ne tik efektyvesnė, bet ir stabilesnė – ji mažiau priklauso nuo kitų parametrų pasirinkimo.

3.4. Tyrinėjimo metodai

Tyrinėjimo metodas taip pat turi reikšmingą įtaką rezultatams. Šilumos žemėlapis „Tyrinėjimo metodas × Panašumo metodas" rodo, kad Thompson Sampling nuosekliai pranoksta Epsilon-Greedy (60% prieš 52%).



5 pav. Tyrinėjimo metodų efektyvumo palyginimas

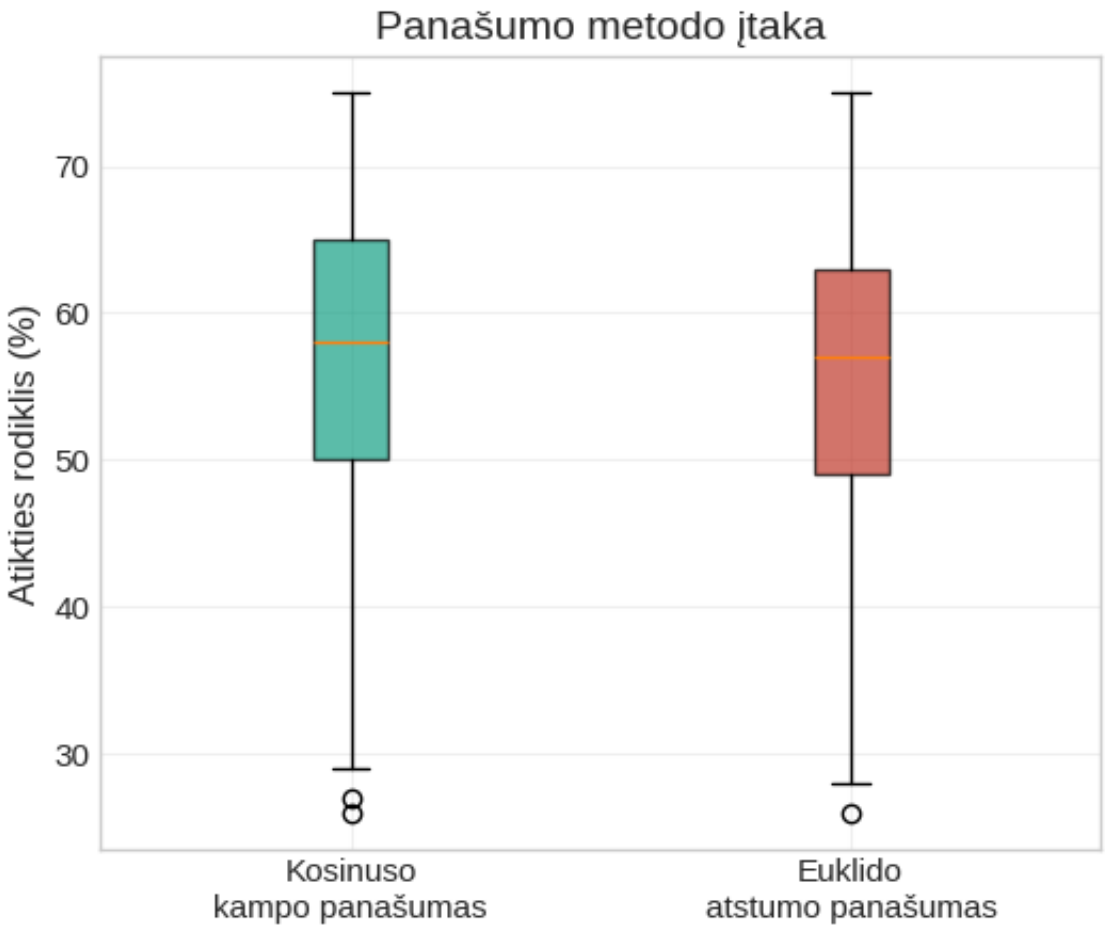
7 lentelė. Tyrinėjimo metodų efektyvumo statistika

Metodas	Vidurkis	Std	Min	25%	50%	75%	Max
Thompson Sampling	60.35	7.23	38.0	55.0	61.0	66.0	75.0
Epsilon-Greedy	52.15	10.12	26.0	45.0	53.0	60.0	72.0

Thompson Sampling metodas pasiekė vidutiniškai 8.2 procentų geresnį rezultatą nei Epsilon-Greedy (60.35% vs 52.15%) su mažesne dispersija. Šie rezultatai patvirtina, kad automatinis tyrinėjimo ir išnaudojimo balansavimas pagal neapibrėžtumo lygį yra efektyvesnis nei fiksuotas ε parametras.

3.5. Panašumo ir atstumo metodai

Šilumos žemėlapiai „Strategija × Panašumo metodas” ir „Modelis × Panašumo metodas” rodo, kad panašumo metodas (Kosinuso prieš Euklido) turi minimalią įtaką rezultatams.



6 pav. Panašumo metodų įtaka atitikties rodikliui

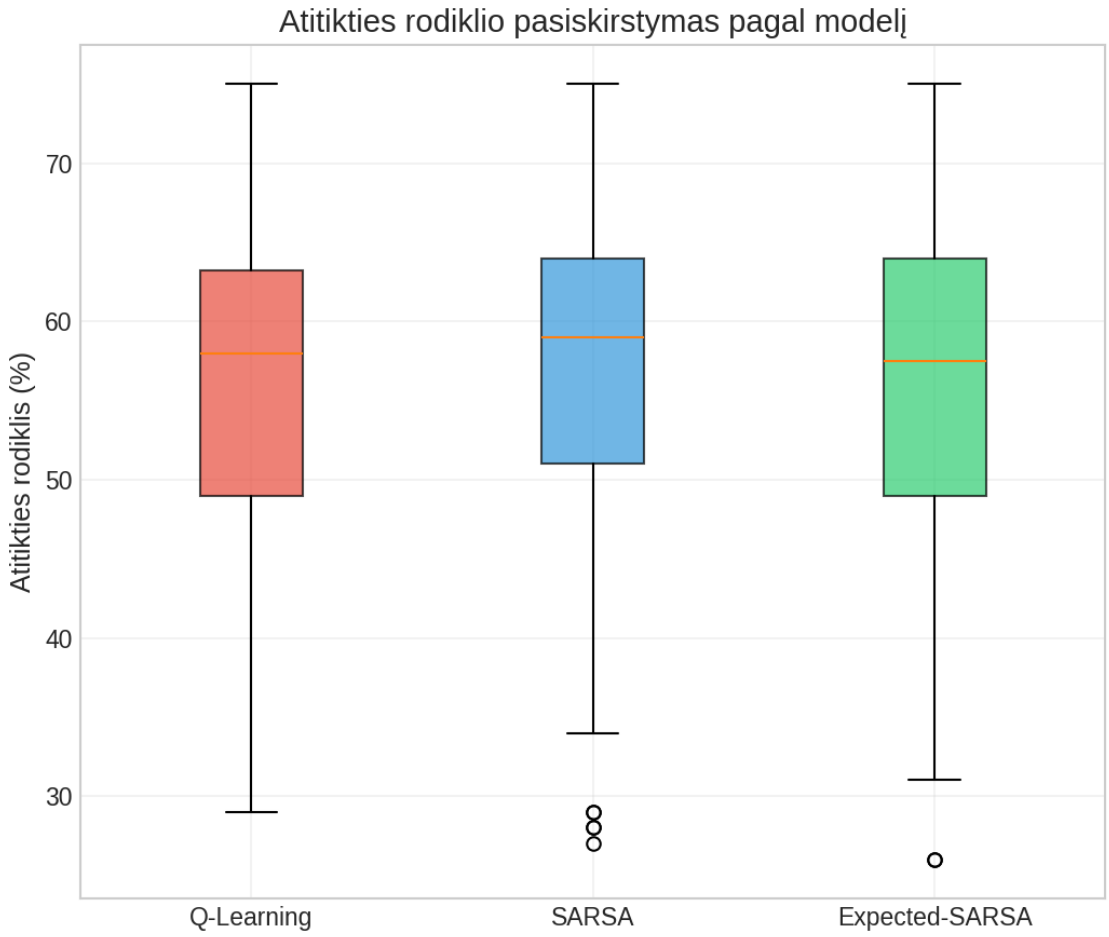
8 lentelė. Panašumo metodų efektyvumo statistika

Metodas	Vidurkis	Std	Min	25%	50%	75%	Maxs
Kosinuso panašumas	56.77	9.85	27.0	50.0	58.0	64.0	75.0
Euklido atstumas	55.73	10.15	26.0	49.0	57.0	63.0	75.0

Kosinuso panašumas minimaliai pranoksta Euklido atstumą (56.77% vs 55.73%, skirtumas 1 procentas). Abu metodai pasiekia tą patį maksimalų rezultatą (75%), todėl panašumo metodo pasirinkimas nėra kritinis veiksnys.

3.6. Mokymosi algoritmai

Trys skatinamojo mokymosi algoritmai parodė labai panašius rezultatus. Šilumos žemėlapis „Modelis × Tyrinėjimo metodas" rodo, kad visi algoritmai pasiekia 52–61% priklausomai nuo tyrinėjimo metodo, bet tarpusavyje praktiškai nesiskiria.



7 pav. Skatinamojo mokymosi algoritmų palyginimas

9 lentelė. Algoritmų efektyvumo statistika

Algoritmas	Vidurkis	Std	Min	25%	50%	75%	Max
Expected-SARSA	55.98	10.16	26.0	49.0	57.5	64.00	75.0
Q-Learning	56.08	9.88	29.0	49.0	58.0	63.25	75.0
SARSA	56.67	10.02	27.0	51.0	59.0	64.00	75.0

Visi trys algoritmai pasiekė praktiškai vienodus rezultatus – vidurkiai svyruoja nuo 55.98% iki 56.67% (skirtumas <1 procentas). Tai rodo, kad šioje konkrečioje PECS kortelių parinkimo užduotyje algoritmo pasirinkimas nėra kritinis veiksnys – kur kas svarbiau yra strategijos ir tyrinėjimo metodo pasirinkimas.

3.7. Alpha, Gamma ir Epsilon parametrai

Šilumos žemėlapiai „Alpha × Gamma", „Alpha × Epsilon" ir „Gamma × Epsilon" rodo, kad parametrai turi minimalią įtaką rezultatams – visos reikšmės svyruoja siaurame 55-57% intervale.

10 lentelė. Mokymosi greičio (α) įtakos statistika

α	Vidurkis	Std	Min	25%	50%	75%	Max
0.1	56.52	9.94	27.0	50.0	58.0	64.0	75.0
0.3	56.33	10.08	26.0	49.0	58.0	64.0	75.0
0.5	55.87	10.06	27.0	49.0	57.0	63.0	75.0

11 lentelė. Nuolaidos faktoriaus (γ) įtakos statistika

γ	Vidurkis	Std	Min	25%	50%	75%	Max
0.50	56.12	10.15	26.0	49.0	58.0	64.0	75.0
0.90	56.19	9.98	27.0	49.0	58.0	64.0	75.0
0.99	56.41	9.96	27.0	50.0	58.0	64.0	75.0

12 lentelė. Tyrinėjimo tikimybės (ε) įtakos statistika

ε	Vidurkis	Std	Min	25%	50%	75%	Max
0.05	56.48	9.89	28.0	50.0	58.0	64.0	75.0
0.15	56.27	10.04	26.0	49.0	58.0	64.0	75.0
0.30	55.97	10.15	27.0	49.0	57.0	63.0	75.0

Šie rezultatai rodo, kad PECS kortelių parinkimo kontekste tradiciniai skatinamojo mokymosi hiperparametrai nėra kritiniai – vidurkiai skiriasi mažiau nei 1 procentiniu punktu. Sistema adaptuojasi nepriklausomai nuo konkrečių hiperparametrų reikšmių.

3.8. Konfigūracijos su didžiausiu atitikties rodikliu

Eksperimento metu penkios skirtingos konfigūracijos pasiekė maksimalų 75% atitikties rodiklį. Visos jos naudoja strategiją A, 4 iš jų naudoja Thompson Sampling. Modeliai, parametrai ir panašumo metodai skiriasi:

13 lentelė. Konfigūracijos su didžiausiu atitikties rodikliu (visos su 75% atitikties rodikliu)

#	Algoritmas	Tyrinėjimas	α	γ	ϵ	Panašumas	Atit.
1	SARSA	Thompson	0.1	0.9	0.15	Euklido	75%
2	SARSA	Epsilon	0.1	0.9	0.05	Kosinuso	75%
3	Q-Learning	Thompson	0.3	0.5	0.15	Kosinuso	75%
4	Expected-SARSA	Thompson	0.1	0.9	0.30	Euklido	75%
5	Expected-SARSA	Thompson	0.3	0.99	0.05	Kosinuso	75%

Rezultatai ir išvados

Atlikus eksperimentą su 972 unikaliomis parametrų kombinacijomis ir 222,608 sąveikos įrašais, gauti šie rezultatai:

1. Strategijos pasirinkimas yra svarbiausias veiksnys: Kortelių parinkimo strategija A (Gylio/Pločio), kuri po neatitikties siūlo vieną panašią ir vieną priešingą kortelę, pasiekė vidutiniškai 65.4% atitikties rodiklį – 14 procentinių punktų daugiau nei strategijos B ir C. Tai patvirtina, kad diversifikuota paieška yra efektyvesnė nei gryna gylio paieška.
2. Thompson Sampling pranoksta Epsilon-Greedy: Bajeso metodu pagrįstas Thompson Sampling tyrinėjimo metodas pasiekė 60.4% vidutinį atitikties rodiklį, pranokdamas Epsilon-Greedy metodą 8.4 procentiniais punktais. Be to, Thompson Sampling pasižymi mažesniu nepastovumu, kas patvirtina jo automatinio prisitaikymo privalumus.
3. Panašumo metodai daro nedidelę įtaką: Kosinuso panašumas turi mažą (1-2% procentiniai punktų) persvarą prieš Euklido atstumo metodą.
4. Skatinamojo mokymosi algoritmų skirtumai minimalūs: Visi trys tirti algoritmai (Q-Learning, SARSA, Expected-SARSA) pasiekė praktiškai vienodus rezultatus (56-57% vidurkis, skirtumas <1 procentinis punktas). Tai rodo, kad šioje konkrečioje užduotyje algoritmo pasirinkimas nėra kritinis veiksnys.
5. Parametrai turi minimalią įtaką: Mokymosi greitis (α), nuolaidos faktorius (γ) ir tyrinėjimo tikimybė (ϵ) turėjo mažiau nei 1 procentinio punkto įtaką rezultatams. Sistema adaptuojasi nepriklausomai nuo konkrečių parametrų reikšmių.
6. Optimali konfigūracija pasiekia 75% atitikties rodiklį: Geriausios parametrų kombinacijos pasiekė 75% atitikties rodiklį, kas yra beveik tris kartus geriau nei blogiausios konfigūracijos (26%).

Remiantis šiais rezultatais galima daryti dvi pagrindines išvadas. Pirma, geriausias atitikties rodiklis buvo pasiektas keičiant strategijas, o ne kitus parametrus, kaip algoritmą, panašumo metodą ar parametrus (α, γ, ε). Antra, iš trijų tirtų strategijų labiausiai pasiteisino ejimas į gylį ir plotį vienu metu (strategija A), kuri diversifikuoja paiešką siūlydama ir panašias, ir priešingas korteles, o ne tik gilinasi į vieną kryptį kaip strategijos B ir C.

Rekomendacijos

Remiantis tyrimo rezultatais, pateikiamos šios rekomendacijos tolimesniam PECS sistemų tobulinimui ir skatinamojo mokymosi algoritmų taikymui AAC kontekste:

Praktiniam taikymui:

1. Strategijos pasirinkimas yra prioritetas: Rekomenduojama naudoti strategiją A (Gylio/Pločio), kuri po neatitikties siūlo vieną panašią ir vieną priešingą kortelę. Ši strategija pasiekė 65.4% vidutinį atitikties rodiklį ir yra stabili skirtingose konfigūracijose.
2. Thompson Sampling vietoj Epsilon-Greedy: Rekomenduojama naudoti Thompson Sampling tyrinėjimo metodą, kuris automatiškai balansuoja tyrinėjimą ir išnaudojimą. Šis metodas pranoko Epsilon-Greedy 8.4 procentiniais punktais.
3. Algoritmo pasirinkimas nėra kritinis: Galima naudoti bet kurį iš trijų algoritmų (Q-Learning, SARSA, Expected-SARSA), nes jų rezultatai praktiškai nesiskiria. SARSA rekomenduojamas dėl minimaliai geresnių rezultatų ir teorinio konservatyvumo.
4. Parametrai gali būti standartiniai: Rekomenduojama pradėti su $\alpha = 0.1$, $\gamma = 0.9$, $\epsilon = 0.15$, tačiau sistema veiks efektyviai ir su kitomis reikšmėmis, nes hiperparametrų įtaka minimali.

Tolimesniems tyrimams:

1. Didesnė produktų bazė: Rekomenduojama išplėsti eksperimentą su didesne produktų duomenų baze (50-100 kortelių), kas labiau atitiktų realias PECS sistemas.
2. Konteksto įtraukimas: Verta ištirti papildomų konteksto požymių (dienos laikas, aplinka, ankstesnės sesijos) įtraukimą į būsenos reprezentaciją.
3. Prisitaikančios strategijos: Siūloma ištirti dinamiškas strategijas, kurios automatiškai persijungia tarp gylio ir pločio paieškos pagal mokymosi eigą.

Sistemų kūrėjams:

1. Produktų savybių svarba: Svarbu investuoti į kokybišką produktų savybių aprašymą, nes strategijos A efektyvumas tiesiogiai priklauso nuo panašumo skaičiavimo tikslumo.
2. Privatumo užtikrinimas: Kadangi sistema kaupia informaciją apie vaiko pasirinkimus, būtina užtikrinti duomenų saugumą ir laikytis privatumo reikalavimų, ypač dirbant su pažeidžiamomis grupėmis.

Šaltiniai

[AAA24] R. K. Alfuraih, N. S. Almalki, F. M. AlNemary. Effectiveness of picture exchange communication system in developing requesting skills for children with multiple disabilities. *Frontiers in Psychology*. 2024, tomas 15, p. 1434478.

[Has10] H. van Hasselt. Double Q-learning. Iš: *Advances in Neural Information Processing Systems 23 (NIPS 2010)*. Curran Associates, Inc., 2010, p. 2613–2621.

[HKP12] J. Han, M. Kamber, J. Pei. *Data Mining: Concepts and Techniques*. 3-as leidimas. Waltham, MA: Morgan Kaufmann Publishers, 2012. 744 psl.

[HMC+24] C. Holyfield, S. MacNeil, N. Caldwell, T. O. Zimmerman, E. Lorah, E. Dragut, S. Vucetic. Leveraging Communication Partner Speech to Automate Augmented Input for Children on the Autism Spectrum Who Are Minimally Verbal: Prototype Development and Preliminary Efficacy Investigation. *American Journal of Speech-Language Pathology*. 2024, tomas 33, numeris 3, p. 1174–1192.

[IG24] A. Iannone, D. Giansanti. Breaking Barriers—The Intersection of AI and Assistive Technology in Autism Care: A Narrative Review. *Journal of Personalized Medicine*. 2024, tomas 14, numeris 1, p. 41. Prieiga per internetą: `https://doi.org/10.3390/jpm14010041`.

[KWL+23] D. Konadl, J. Wörner, L. Luttner, S. Leist. Artificial Intelligence in Augmentative and Alternative Communication Systems – A Literature-Based Assessment and Implications of Different Conversation Phases and Contexts. Iš: *Proceedings of the 31st European Conference on Information Systems (ECIS 2023)*. Kristiansand, Norway: AIS Electronic Library, 2023.

[MN24] L. J. Martin, M. Nagalakshmi. *Aging Up AAC: An Introspection on Augmentative and Alternative Communication Applications for Autistic Adults*. 2024. [žiūrėta 2024-12-20]. Prieiga per internetą: `https://arxiv.org/abs/2404.17730`.

[MRS08] C. D. Manning, P. Raghavan, H. Schütze. *Introduction to Information Retrieval*. Cambridge, UK: Cambridge University Press, 2008. 482 psl.

[PPZ+24] J. A. Pereira, J. A. Pereira, C. Zanchettin, R. do Nascimento Fidalgo. PrAACT: Predictive Augmentative and Alternative Communication with Transformers. *Expert Systems with Applications*. 2024, tomas 240, p. 122417.

[RN94] G. A. Rummery, M. Niranjan. *On-line Q-Learning Using Connectionist Systems*. Cambridge, UK, 1994. Technical Report, CUED/F-INFENG/TR 166. Cambridge University Engineering Department.

[RRK+18] D. J. Russo, B. V. Roy, A. Kazerouni, I. Osband, Z. Wen. A Tutorial on Thompson Sampling. *Foundations and Trends in Machine Learning*. 2018, tomas 11, numeris 1, p. 1–96.

[SB18] R. S. Sutton, A. G. Barto. *Reinforcement Learning: An Introduction*. 2-as leidimas. Cambridge, MA: MIT Press, 2018. 552 psl.

[SHW+09] H. van Seijen, H. van Hasselt, S. Whiteson, M. Wiering. A Theoretical and Empirical Analysis of Expected Sarsa. Iš: *Proceedings of the IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL 2009)*. Nashville, TN: IEEE, 2009, p. 177–184.

[SS96] S. P. Singh, R. S. Sutton. Reinforcement Learning with Replacing Eligibility Traces. *Machine Learning*. 1996, tomas 22, numeris 1–3, p. 123–158.

[Ter22] L. Tereshko. Picture Exchange Communication Systems (PECS): A treatment summary. *Science in Autism Treatment*. 2022, tomas 19, numeris 10.

[TOS+23] A. C. Tamanaha, D. O. F. Olivatti, S. C. da Silva, S. C. P. Vieira, J. Perissinoto. Picture Exchange Communication System (PECS) Implementation Program for children with autism spectrum disorder. *CoDAS*. 2023, tomas 35, numeris 4, e20210305.

[VCK+23] S. Valencia, R. Cave, K. Kallarackal, K. Seaver, M. Terry, S. K. Kane. "The Less I Type, the Better": How AI Language Models Can Enhance or Impede Communication for AAC Users. Iš: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2023, p. 1–14. Prieiga per internetą: `https://doi.org/10.1145/3544548.3581560`.

[Wat89] C. J. C. H. Watkins. *Learning from Delayed Rewards*. Cambridge, UK, 1989. Disertacija. King's College, University of Cambridge.

[WVW25] P. Wannapaschaiyong, T. Vivattanasinchai, A. Wongkwanmuang. Predictors of successful Picture Exchange Communication System training in children with communication impairments: insights from a real-world intervention in a resource-limited setting. *BMJ Paediatrics Open*. 2025, tomas 9, numeris 1, e003282. Prieiga per internetą: `https://doi.org/10.1136/bmjpo-2024-003282`.