Matematinių uždavinių klasifikavimas taikant natūralios kalbos apdorojimo ir mašininio mokymosi metodus

Emilija Bareikaitė[1], Gražina Korvel[1], Ieva Kilienė[2]

[1] Vilniaus universitetas, Matematikos ir informatikos fakultetas, Duomenų mokslo ir skaitmeninių technologijų institutas, Akademijos g. 4, Vilnius, Lietuva
[2] Vilniaus universitetas, Matematikos ir informatikos fakultetas, Taikomosios matematikos institutas, Naugarduko g. 24, Vilnius, Lietuva
emilija.bareikaite@mif.stud.vu.lt

Santrauka. Matematiniai tekstiniai uždaviniai yra svarbi matematikos mokymo dalis, tačiau jų analizė dažnai atliekama rankiniu būdu, todėl yra sudėtinga ir imli laikui. Šiame darbe tiriamos galimybės automatizuoti lietuviškų pradinių klasių matematinių uždavinių klasifikavimą taikant natūralios kalbos apdorojimo ir mašininio mokymosi metodus. Eksperimentuose nagrinėti skirtingi teksto paruošimo metodai, teksto vektorizavimo (žodžių maišo ir TF-IDF vektoriai) bei klasifikavimo modeliai. Rezultatai parodė, kad kategorijų klasifikavimui geriausiai tinka žodžių maišo vektorizavimo metodas kartu su atraminių vektorių klasifikatoriumi, pasiekiant 93,33 % tikslumą. Tipų klasifikavime geriausi rezultatai gauti taikant trijų žingsnių hierarchinę architektūrą su žodžių maišo vektoriais ir kalbos modelio įterpiniais (pasiektas 0,68 makro F1 įvertis). Šis tyrimas prisideda prie efektyvesnės lietuviškų matematinių tekstinių uždavinių analizės.

Raktiniai žodžiai: matematinių tekstinių uždavinių klasifikavimas, natūralios kalbos apdorojimas, mašininis mokymasis, vektorizavimas, hierarchinė klasifikacija.

1 Įvadas

Matematika yra viena svarbiausių mokslo šakų, ugdanti loginį mąstymą, bei taikoma įvairiose technologijų, kasdienio gyvenimo srityse. Vis dėlto, Lietuvoje matematikos mokymosi rezultatai vis dar išlieka opi problema: 2025 m. išplėstinio kurso matematikos brandos egzamino neišlaikė 14,8 %, o bendrojo kurso – 41,5 % mokinių [8]. Dėl šios priežasties vis daugiau dėmesio skiriama matematikos mokymo turinio, vadovėlių ir uždavinių tyrimams. Tokie tyrimai dažnai remiasi matematinių tekstinių uždavinių klasifikavimu pagal

DOI: https://doi.org/10.15388/LMITT.2026.2

jų struktūrą ar sprendimo tipą. Tačiau nemaža dalis šios analizės atliekama rankiniu būdu, todėl procesas reikalauja daug laiko ir riboja nagrinėjamų duomenų apimtį. Natūralios kalbos apdorojimo metodai ir mašininio mokymosi modeliai leidžia automatizuoti šį procesą ir sudaro galimybes analizuoti didesnius uždavinių rinkinius.

Anglų kalbos matematinių uždavinių automatinis klasifikavimas jau yra plačiai taikomas naudojant klasikinius natūralios kalbos apdorojimo (žodžių maišo (angl. *bag-of-words*) vektoriai, lingvistinių požymių ir taisyklių konstravimas, teksto paruošimo strategijos) ir mašininio mokymosi metodus (atraminių vektorių klasifikatorius (angl. *support vector machine*), logistinė regresija, naivusis Bajeso modelis), pasiekusius 0,79–0,94 tikslumo rodiklius [2],[5],[6]. Tačiau šie metodai dažnai reikalauja gilios teksto analizės, kruopštaus lingvistinių požymių kūrimo ir gali būti jautrūs skirtingų kalbų ypatumams.

Pastaraisiais metais daug dėmesio skiriama pažangesniems metodams, pagrįstiems giliojo mokymosi modeliais [7] ir transformerių architektūromis, tokioms kaip BERT ar GPT-3 [1],[13]. Šie modeliai dažnai taikomi platesnėje užduotyje – automatiniame matematinių uždavinių sprendime, kur klasifikavimas yra tarpinis žingsnis. Nors tokie metodai demonstruoja aukštus rezultatus (metrikos, skirtos tikslumui vertinti, virš 95 %), jų efektyviam veikimui reikia didelių duomenų kiekių, skaičiavimo išteklių ir kruopštaus hiperparametrų parinkimo.

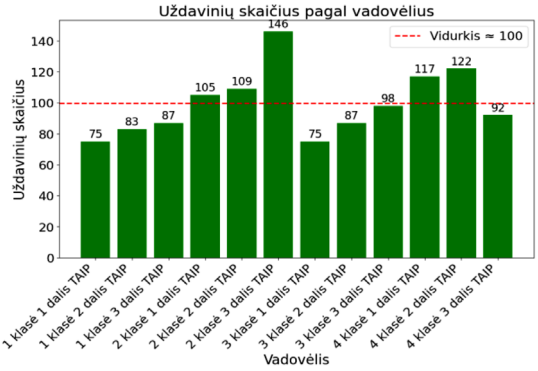
Vis dėlto dauguma ankstesnių tyrimų nagrinėja anglų kalbos matematinius uždavinius, todėl kitų kalbų, ypač morfologiškai sudėtingesnių, analizė išlieka mažiau ištirta. Šiame darbe nagrinėjami lietuviški matematiniai tekstiniai uždaviniai iš pradinių klasių vadovėlių ir tiriamos galimybės juos klasifikuoti naudojant natūralios kalbos apdorojimo metodus bei mašininio mokymosi modelius. Darbe analizuojamas skirtingų teksto reprezentacijų, klasifikavimo modelių ir įvairių modeliavimo technikų efektyvumas, siekiant nustatyti tinkamiausius metodus lietuviškų matematikos uždavinių klasifikavimui.

2 Duomenys

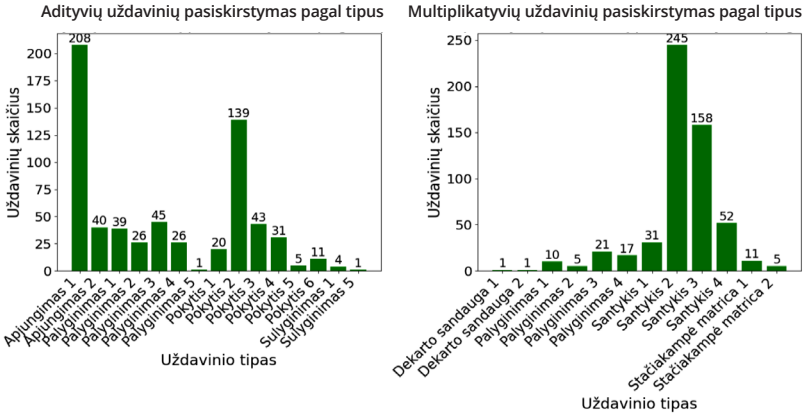
Šiame darbe nagrinėjami Lietuvos 1–4 klasių vadovėlių [12], [9], [10], [11] vieno žingsnio aritmetiniai tekstiniai uždaviniai, anotuoti remiantis darbe [4] pasiūlyta metodika. Uždaviniai yra pirmiausia skaidomi į adityvius ir multiplikatyvius, o vėliau klasifikuojami, remiantis uždavinio situacijoje aprašomu (ne matematiniu) veiksmu ar sąryšiu [4]. Adityviuose uždaviniuose naudojami sudėties ir atimties veiksmai, jie yra skirstomi į keturias grupes (pokyčio, apjungimo, pa-

lyginimo ir sulyginimo) ir į 20 skirtingų tipų. Multiplikatyvūs uždaviniai naudoja daugybos ir dalybos veiksmus ir taip pat skaidomi į keturias grupes (santykio, palyginimo, Dekarto sandaugos ir stačiakampės matricos) bei 14 tipų.

Iš dvylikos vadovėlių buvo surinkti 1196 uždaviniai: 639 adityvūs ir 557 multiplikatyvūs. 1 paveikslas vaizduoja uždavinių pasiskirstymą pagal vadovėlius. Pastebima, jog daugiausiai uždavinių surinkta iš 2 klasės vadovėlių, ypač 3 dalies, o mažiausiai – iš 1 klasės vadovėlių. Be to, fiksuota, kad tipų pasiskirstymas yra stipriai nesubalansuotas – kai kurių tipų stebimi tik keli pavyzdžiai, o dalies tipų visai nėra (žr. 2 pav.). Toliau tyrime duomenys buvo padalinti į mokymo ir testavimo aibes, santykiu 80:20 (testavimo aibė buvo naudojama modelio vertinimui).



1 pav. Uždavinių skaičius pagal vadovėlius.



2 pav. Atitinkamų kategorijų uždavinių pasiskirstymas pagal tipus.

3 Metodinė dalis

Šį tyrimą sudaro keli pagrindiniai etapai: tekstinių duomenų paruošimas, teksto reprezentavimas skaitiniais vektoriais, klasifikavimo modelių taikymas ir eksperimentų rezultatų vertinimas.

3.1 Tekstinių duomenų paruošimas

Šiame darbe taikomos trys skirtingos teksto paruošimo strategijos. Naudojant pirmąją strategiją, panaikinami skyrybos ženklai, papildomi tarpai, didžiosios raidės paverčiamos mažosiomis. Taikant likusias strategijas, atliekami papildomi žingsniai tekstui, paruoštam naudojant pirmąją strategiją. Trečioji strategija paremta literatūroje aprašytu metodu, kai iš teksto pašalinamos beveik visos pagrindinės kalbos dalys (daiktavardžiai, tikriniai daiktavardžiai, būdvardžiai ir kt.) [2]. Kadangi šis metodas buvo taikytas anglų kalbos uždaviniams, buvo sukurta ir papildoma strategija (toliau vadinama antrąja), labiau pritaikyta lietuvių kalbai: joje iš teksto šalinami tik tikriniai daiktavardžiai ir jaustukai.

3.2 Teksto vektorizavimas

Norint taikyti mašininio mokymosi metodus, tekstiniai duomenys buvo paversti skaitiniais požymiais. Šiame darbe tam buvo taikomi du metodai: žodžių maišo ir TF-IDF vektorizavimas. Abiem metodams taikytos unigramų ir bigramų kombinacijos, naudojant minimalų dviejų dokumentų dažnio slenkstį.

Žodžių maišo metodas dokumentą vaizduoja vektoriumi, kurio elementai atitinka skirtingų žodžių dažnį tekste, neatsižvelgiant į gramatiką ir žodžių tvarką [2]. TF-IDF metodas įvertina ne tik žodžio pasikartojimą dokumente, bet ir jo paplitimą visame dokumentų rinkinyje. Šis metodas sumažina labai dažnų, tačiau mažai informatyvių žodžių svorį ir padidina retesnių, labiau išskiriančių terminų svarbą. TF-IDF svoris apskaičiuojamas pagal formulę:

$$TF - IDF(t, d) = TF(t, d) \cdot IDF(t),$$

kur TF – termino dažnis, IDF – atvirkštinis dokumentų dažnis, t – nagrinėjamas terminas, o d – dokumentas. Termino dažnis skaičiuojamas pagal formulę:

$$TF(t, d) = \frac{termino\ t\ pasikartojimų\ skaičius\ dokumente\ d}{bendras\ terminų\ skaičius\ dokumente}.$$

O atvirkštinis dokumentų dažnis, skaičiuojamas naudojantis formule:

$$IDF(t) = \log\left(\frac{bendras\ dokumentų\ skaičius}{dokumentų, kuriuose\ yra\ nagrinėjamas\ terminas\ t, skaičius}\right).$$

3.3 Klasifikavimo modeliai

Matematinių uždavinių klasifikavimo užduočiai spręsti šiame darbe taikyti skirtingi mašininio mokymosi modeliai: k-artimiausių kaimynų modelis, logistinė regresija, atraminių vektorių klasifikatorius, naivusis Bajeso klasifikatorius ir atsitiktinių miškų modelis [2], [6].

K-artimiausių kaimynų (angl. *k-nearest neighbour*s) modelis klasifikuoja naują objektą pagal artimiausių mokymo duomenų kaimynų klases požymių erdvėje. Klasė nustatoma pagal dažniausiai pasitaikančią klasę tarp *k* artimiausių kaimynų. Logistinė regresija modeliuoja klasės tikimybę naudojant logistinę funkciją:

$$P(y = 1|x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n)}},$$

kur $x_1,\ldots,x_n$ yra požymiai, o $\beta_0,\ldots,\beta_n$ – modelio parametrai. Atraminių vektorių klasifikatorius randa hiperplokštumą, kuri geriausiai atskiria skirtingų klasių duomenis požymių erdvėje. Naivusis Bajeso klasifikatorius remiasi Bajeso teorema ir daro prielaidą, kad požymiai yra tarpusavyje nepriklausomi. Klasės tikimybė apskaičiuojama pagal formulę:

$$P(y \mid x) = \frac{P(x \mid y)P(y)}{P(x)},$$

kur *x* – požymių vektorius, o *y* – prognozuojama klasė. Atsitiktinių miškų modelis yra ansamblinis metodas, sudarytas iš kelių sprendimų medžių. Galutinė klasė nustatoma pagal daugumos medžių sprendimą.

Modelių hiperparametrams parinkti taikyta gardelės paieška (angl. *grid search*). Buvo tiriamos šios hiperparametrų reikšmės: C = 0,5,1,2 (logistinei regresijai ir atraminių vektorių klasifikatoriui), α = 0,5,1 (naiviajam Bajeso klasifikatoriui), medžių skaičius 100 ir 200 (atsitiktinių miškų modeliui) bei kaimynų skaičius k = 3,5,7 (k-artimiausių kaimynų metodui).

3.4 Eksperimentų schema

Siekiant įvertinti skirtingų metodų tinkamumą lietuviškų matematinių tekstinių uždavinių klasifikavimui, atlikti eksperimentai, apimantys skirtingas teksto apdorojimo, reprezentavimo ir modeliavimo strategijas. Eksperimentai skirstomi pagal uždavinio pobūdį:

Kategorijų klasifikavimas:

- Teksto paruošimo strategijų palyginimas.
- Teksto reprezentavimo skaitiniais vektoriais palyginimas.
- Klasifikavimo modelių palyginimas.

Klasifikavimo tikslumas (teisingai priskirtų klasių santykis su visais testavimo aibės duomenimis) buvo naudojamas kaip vertinimo rodiklis šiai daliai.

Tipų klasifikavimas:

- Plokščio ir hierarchinio (dviejų ir trijų žingsnių) klasifikavimo architektūrų analizė.
- Papildomų lingvistinių ir struktūrinių požymių konstravimas (POS žymų dažniai, raktinių žodžių pasikartojimai, skaitmenų dažnis, trupmenų buvimas, vidutinis sakinių ilgis ir t. t.). Eksperimente palikti tik statistiškai reikšmingi skaitiniai požymiai (taikant Kruskal–Wallis testą), o reti kategoriniai kintamieji buvo panaudoti kaip taisyklių sistema, koreguojanti modelio prognozuotas tikimybes.
- Duomenų balansavimo metodai: mažesnių klasių uždavinių dubliavimas ir naujų pavyzdžių generavimas retoms klasėms, naudojant sinonimų žodyną [3].
- Kalbos modelių įterpinių (angl. *embeddings*) taikymas kartu su skaitiniais vektoriais.

Eksperimentai toliau žymimi taip: „E1“ – plokščioji klasifikacija, „E2“ – dviejų žingsnių hierarchinė klasifikacija, „E3“ – trijų žingsnių hierarchinė klasifikacija, „E4“ – trijų žingsnių klasifikacija su visais lingvistiniais ir struktūriniais požymiais bei taisyklėmis, „E5“ – trijų žingsnių klasifikacija su vienu atrinktu požymiu, derintu su skaitiniais vektoriais, „E6“ – „E5“ su mažesnių klasių uždavinių dubliavimas, „E7“ – „E5“ su papildomais uždaviniais, generuotais pagal sinonimus, „E8“ – trijų žingsnių klasifikacija su skaitiniais vektoriais ir kalbos modelio įterpiniais.

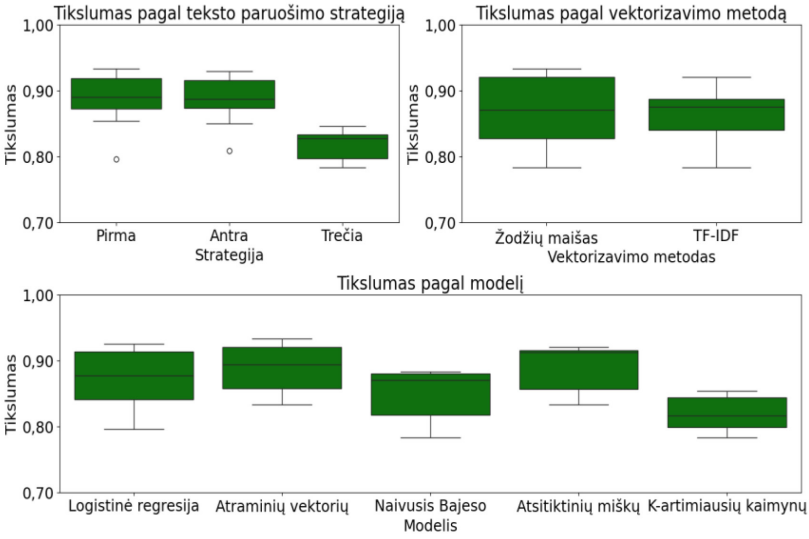
Makro F1 metrika (kiekvienos klasės F1 įverčių vidurkis) buvo naudojama kaip vertinimo rodiklis šiai daliai. F1 metrika skaičiuojama pagal formulę:

$$F1 = \frac{2\,(Preciziškumas \cdot Jautrumas)}{Preciziškumas + Jautrumas},$$

kur jautrumas nusako, kokią dalį visų tikrų tam tikros klasės stebėjimų modelis teisingai identifikuoja, o preciziškumas – kokia dalis modelio tai klasei priskirtų stebėjimų iš tikrųjų jai priklauso.

4 Eksperimentų rezultatai

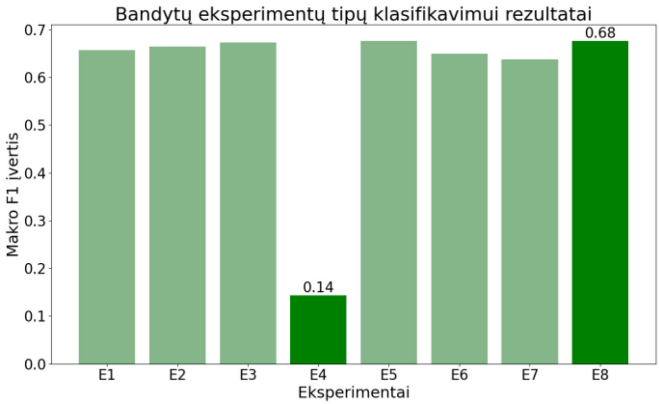
Uždavinių kategorijų klasifikavimo eksperimentų rezultatus galima stebėti 3 paveiksle – čia vaizduojami fiksuoti klasifikavimo tikslumai testavimo duomenims, atsižvelgiant į skirtingus išbandytus metodus. Bendrai, matoma, kad dauguma bandytų kombinacijų siekia virš 80 % klasifikavimo tikslumą. Vertinant teksto paruošimo strategijas, prasčiausi rezultatai fiksuoti taikant trečiąją strategiją, o naudojant pirmąją ir antrąją stebėti panašūs rezultatai. Atsižvelgiant į vektorizavimo metodus, didelių skirtumų tarp žodžių maišo ir TF-IDF taikymo nepastebėta. Pagal naudotus modelius prasčiausiai veikė



3 pav. Skirtingų strategijų klasifikavimo tikslumų palyginimas.

k-artimiausių kaimynų ir naiviojo Bajeso klasifikatoriai, o likę modeliai pasiekė panašius rezultatus. Vis dėlto, geriausią rezultatą fiksavo atraminių vektorių klasifikatorius, naudojęs žodžių maišo vektorizavimą ir pirmąją teksto paruošimo strategiją: pasiektas 93,33 % tikslumas. Remiantis šiais stebėtais rezultatais, toliau darbe nuspręsta taikyti pirmą strategiją teksto paruošimui, žodžių maišo metodą vektorizavimui ir atraminių vektorių klasifikatorių pirmo lygmens modeliui hierarchinėse architektūrose.

Sėkmingai klasifikavus uždavinius į atitinkamas kategorijas, buvo pereita prie uždavinių tipų klasifikavimo. Duomenų analizė atskleidė, kad tipai yra stipriai nesubalansuoti, todėl tolimesniems tyrimams buvo atrinkti tik tie tipai, kuriuose yra bent 17 pavyzdžių. Taip pat, kai buvo įmanoma, modeliuose taikyti subalansuoti klasių svoriai. Atliktų eksperimentų rezultatus testavimo duomenų aibei galima stebėti 4 paveiksle. Iš grafiko galime matyti, kad geriausią rezultatą – 0,68 makro F1 įvertį – pavyko pasiekti, taikant trijų žingsnių hierarchinę architektūrą kartu su žodžių maišo vektoriais ir kalbos modelio „intfloat/multilingual-e5-small“ įterpiniais. Prasčiausias rezultatas fiksuotas, taikant tik trijų žingsnių hierarchinę architektūrą kartu su sukurtais požymiais.



4 pav. Bandytų eksperimentų tipų klasifikavimui rezultatai.

5 Išvados

Kategorijų klasifikavimo rezultatai parodė, kad geriausiai veikia žodžių maišo vektorizavimas kartu su atraminių vektorių klasifikatoriumi ir minimaliu teksto normalizavimu. Taip pat pastebėta, kad tradiciniai mašininio moky-

mosi ir natūralios kalbos apdorojimo metodai gana sėkmingai sprendžia kategorijų klasifikavimo užduotį. Vis dėlto teksto paruošimas, pašalinant daugumą kalbos dalių, lietuvių kalbos duomenims pasirodė neefektyvus.

Tipų klasifikavimo eksperimentai parodė, kad hierarchinės architektūros veikia geriau už plokščią klasifikavimą: dviejų ir trijų žingsnių hierarchinės struktūros pagerino makro F1 nuo 0,656 iki 0,673. Įtraukus papildomus lingvistinius ir struktūrinius požymius, rezultatai smarkiai sumažėjo, tačiau jų derinimas su žodžių maišo vektoriais leido atkurti efektyvumą (0,676). Duomenų balansavimo metodai – dubliavimas ir naujų pavyzdžių generavimas – nepagerino rezultatų, o kalbos modelių įterpiniai kartu su žodžių maišo vektoriais pasiekė geriausią rezultatą. Vis dėlto kalbos modelių įterpinių naudojimas pagerino rezultatą tik nežymiai, todėl galima daryti išvadą, kad tipų klasifikavimui svarbiausias veiksnys yra tinkamų skaitinių teksto reprezentacijų pasirinkimas.

Visgi, vienas pagrindinių šio darbo apribojimų yra ribotas ir nesubalansuotas duomenų rinkinys: kai kurie uždavinių tipai turi labai mažai pavyzdžių arba jų visai nepasitaiko. Ateities darbuose būtų tikslinga išplėsti duomenų rinkinį, kas leistų taikyti pažangesnius metodus, pavyzdžiui, neuroninius tinklus ar didesnius kalbos modelius.

Literatūra

[1] T. T. Aurpa, K. N. Fariha, K. Hossain, S. M. Jeba, M. S. Ahmed, M. R. S. Adib, F. Islam, F. Akter. „Deep transformer-based architecture for the recognition of mathematical equations from real world math problems“. Iš: *Heliyon* 10.20 (2024).

[2] S. Cetintas, L. Si, Y. P. Xin, D. Zhang, J. Y. Park. „Automatic Text Categorization of Mathematical Word Problems.“ Iš: *FLAIRS*. 2009.

[3] G. Yigit, M. F. Amasyali. „Data Augmentation with In-Context Learning and Comparative Evaluation in Math Word Problem Solving“. Iš: *SN Computer Science* 5.5 (2024), puslapis 506.

[4] I. Kilienė. „Tekstinių uždavinių vaidmuo gilinant mokyklinės matematikos žinias“. Disertacija. Vilniaus universitetas, 2024.

[5] S. Mandal, S. Acharya, R. Basak. „Solving arithmetic word problems using natural language processing and rule-based classification“. Iš: *International Journal of Intelligent Systems and Applications in Engineering* 10.1 (2022), puslapiai 87–97.

[6] S. Mandal, S. K. Naskar. „Classifying and solving arithmetic math word problems—an intelligent math solver“. Iš: *IEEE Transactions on Learning Technologies* 14.1 (2021), puslapiai 28–41.

[7] S. Mandal, A. A. Sekh, S. K. Naskar. „Solving arithmetic word problems: A deep learning based approach“. Iš: *Journal of Intelligent & Fuzzy Systems* 39.2 (2020), puslapiai 2521–2531.

[8] Nacionalinė švietimo agentūra. *Egzaminų ir pasiekimų patikrinimų rezultatų analizė.* 2026. URL: https://www.nsa.smsm.lt/pasiekimu-departamentas/egzaminai-ir-pasiekimu-patikrinimai/valstybiniai - brandos - egzaminai / rezultatu – analizes/ (žiūrėta 2026-02-21).
[9] R. Rimšelienė, A. Kavaliauskienė, A. Padarauskienė. *Matematika. Vadovėlis 2 klasei. Serija TAIP!* Šviesa, 2018.
[10] R. Rimšelienė, A. Kavaliauskienė, A. Padarauskienė. *Matematika. Vadovėlis 3 klasei. Serija TAIP!* Šviesa, 2018.
[11] R. Rimšelienė, A. Kavaliauskienė, A. Padarauskienė. *Matematika. Vadovėlis 4 klasei. Serija TAIP!* Šviesa, 2018.
[12] R. Rimšelienė, A. Kavaliauskienė, L. Vilčinskas. *Matematika. Vadovėlis 1 klasei. Serija TAIP!* Šviesa, 2018.
[13] M. Zong, B. Krishnamachari. „Solving math word problems concerning systems of equations with gpt3“. Iš: *Proceedings of the AAAI conference on artificial intelligence*. Tomas 37. 13. 2023, puslapiai 15972–15979.