VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ BAKALAURO STUDIJŲ PROGRAMA

Teksto lietuvių kalba supaprastinimas į lengvai suprantamą kalbą pasitelkiant didelius kalbos modelius

(Simplifying Lithuanian text into Easy-to-Read language using large language models)

Bakalauro baigiamasis darbas

Atliko: Simona Kuoraitė

Darbo vadovas: dr. Valentas Gružauskas

Darbo recenzentas: Tomas Plankis

Vilnius – 2025

Santrauka

Šiame darbe nagrinėjama teksto lietuvių kalba supaprastinimo į lengvai suprantamą kalbą užduotis, naudojant didelius kalbos modelius. Tekstas lengvai suprantama kalba – tai trumpais, aiškiais sakiniais ir paprastais žodžiais parašytas tekstas, pritaikytas žmonėms, turintiems proto negalią arba ribotas kalbos žinias. Šio darbo tikslas – ištirti, kaip galima pritaikyti didelį kalbos modelį lietuvių kalba Lt-Llama- 2 lietuviškų tekstų supaprastinimui į lengvai suprantamą kalbą, siekiant geriausių kiekybinių ir kokybinių rezultatų. Siekiant šio tikslo, buvo atlikta literatūros analizė, parengti specialūs duomenų rinkiniai, pritaikyti modelio adaptavimui, ir atlikti eksperimentai. Modelis buvo adaptuojamas naudojantis kelių pakopų adaptavimo metodą bei testuojamas modeliui pateikiant tekstus originalia kalba bei tekstus su užduočiai pritaikyta užklausa. Šiame darbe palyginti rezultatai, gauti modelio įvesčiai pateikus skirtingai sukonstruotas užklausas. Modelio rezultatai įvertinti naudojant SARI ir BERTScore metrikas, skirtas tekstų vertinimui bei kokybinį didelio kalbos modelio vertinimą. Gauti rezultatai parodė, kad teksto supaprastinimo į lengvai suprantamą kalbą užduočiai dvejomis pakopomis adaptuotas modelis supaprastina tekstč truputį geriau nei neadaptuotas, bet norint pasiekti reikšmingų rezultatų, reikėtų paruošti didesnį ir platesnį duomenų rinkinį bei atlikti daugiau tyrimų adaptuojant modelį šiai užduočiai atlikti.

Raktiniai žodžiai: lengvai suprantama kalba, kalbos supaprastinimas, dideli kalbos modeliai, kelių pakopų modelio adaptavimas, dideli kalbos modeliai kaip vertintojai.

Summary

This paper explores the task of simplifying Lithuanian text into Easy-to-Read language. Easy-to-Read language is a text written in short, clear sentences and simple words, adapted for people with intellectual disabilities or limited language skills. The aim of this work is to investigate how the large language model Lt-Llama-2-7b-hf, trained on Lithuanian language data, can be adapted to the task of simplifying Lithuanian texts into Easy-to-Read language. To achieve this goal, a literature analysis was carried out, specialized datasets were developed to fine-tune the model, applying multi-stage fine-tuning method, and experiments were carried out. The model was tested by presenting the texts in their original language and the texts with different prompts, constructed based on prompt engineering practices. The results were evaluated using the SARI and BERTScore metrics for assessing the quality of simplified texts and a qualitative evaluation of the large language model. The results show that the multi-stage fine-tuned model simplifies texts a little bit better than a not fine-tuned model, but that a larger and more extensive dataset would be needed to achieve significant results, and that more research should be carried out on fine-tuning the model for this task.

Keywords: Easy-to-Read language, language simplification, large language models, multi-stage fine-tuning, large language model as a judge.

Turinys

Santrauka 1
Summary 2
Įvadas 4
1. Teoriniai pagrindai 5
1.1. Socialinės gerovės sektorius 5
1.1.1. Informacijos prieinamumas žmonėms, turintiems negalią 5
1.2. Lengvai suprantama kalba 5
1.3. Natūralios kalbos apdorojimo technologijos 6
1.4. Transformerių architektūra 7
1.5. Modelio adaptavimas 8
1.5.1. Kelių pakopų adaptavimas 8
1.5.2. Išteklių optimizavimas naudojant QLoRA 9
1.5.3. Užklausų kūrimas 10
1.6. Vertinimo metrikos 11
1.6.1. Dideli kalbos modeliai kaip vertintojai 12
2. Metodologija 13
2.1. Duomenų rikiniai 13
2.2. Kalbos supaprastinimas į lengvai suprantamą kalbą 14
2.2.1. Išteklių optimizavimas naudojant QLoRA 14
2.2.2. Mokymo parametrai 15
2.2.3. Užklausų kūrimas 16
2.3. Supaprastintų tekstų vertinimas 17
2.3.1. Kokybės vertinimas naudojant SARI metriką 18
2.3.2. Kokybės vertinimas naudojant BERTScore metriką 19
2.3.3. Modelio rezultatų vertinimas naudojant PPL metriką 19
2.3.4. Kokybės vertinimas naudojant didelį kalbos modelį kaip vertintoją 20
3. Eksperimentai ir rezultatai 21
3.1. Modelio adaptavimas 21
3.2. Užklausų kūrimas 24
Rezultatai 27
Išvados 28
Literatūra 30

Įvadas

Šiuolaikinėje visuomenėje daug dėmesio skiriama informacijos prieinamumui ir negalią turinčių žmonių įtraukimui į visuomenę. Remiantis pasaulio sveikatos organizacija, daugiau nei milijardas žmonių pasaulyje – apie 16 proc. visos pasaulio populiacijos – turi negalią [22]. Daliai iš jų prieiga prie informacijos vis dar išlieka sudėtinga. Viena iš šių žmonių grupių, tai asmenys, turintys intelekto sutrikimų, mokymosi sunkumų ar žmonės, turintys ribotus kalbinius gebėjimus, kaip tik pradedantys mokytis kalbos asmenys. Šie žmonės dažnai susiduria su problema suprantant sudėtingus, neįprastais žodžiais parašytus tekstus.

Lengvai suprantama kalba (angl. Easy to Read, toliau trumpinama kaip ETR), yra informacijos perteikimo būdas, kurio tikslas – pateikti svarbiausią informaciją trumpais, aiškiais sakiniais ir paprastais žodžiais. ETR tekstai rengiami žmonių, remiantis teksto rengimo rekomendacijomis. Šis procesas reikalauja daug laiko ir išteklių. Tačiau šį procesą galima supaprastinti ir pagreitinti pasitelkiant natūralios kalbos apdorojimo (angl. Natural Language Processing, NLP) modelius. Atsiradus transformeriams, kurie NLP srityje suteikė naujų galimybių, tekstų analizė ir generavimas tapo aiškesni bei efektyvesni. Transformerių architektūra pakeitė didelių kalbos modelių, tokių kaip BERT, GPT galimybes analizuoti bei generuoti tekstą greičiau ir kokybiškiau, o modelių apmokymas dideliais duomenų rinkiniais leido pritaikyti modelius konkrečioms užduotims, reikalaujant mažiau išteklių. Dabar šios technologijos pritaikomos įvairiose srityse: nuo automatinių vertimų, iki medicininių įrašų supaprastinimo. Lietuvių kalba ilgą laiką nebuvo sulaukusi pakankamai dėmesio natūralios kalbos apdorojimo srityje, tačiau neseniai lietuvių kalbai pritaikytas didelis kalbos modelis Lt-Llama-2 suteikia naujų galimybių teksto lietuvių kalba apdorojimui ir generavimui, įskaitant tekstų supaprastinimą.

Darbo tikslas – ištirti kaip galima pritaikyti didelį kalbos modelį Lietuvių kalba Lt-Llama-2 lietuviškų tekstų supaprastinimui į lengvai suprantamą kalbą, siekiant geriausių kiekybinių ir kokybinių rezultatų.

Darbo uždaviniai:

- Atlikti literatūros analizę apie teksto supaprastinimo principus ir didelių kalbos modelių taikymą natūralios kalbos apdorojimo užduotims atlikti.
- Paruošti duomenų rinkinius, skirtus modelio Lt-Llama-2 adaptavimui, bei nustatyti tinkamus hiperparametrus pritaikant modelį teksto supaprastinimo užduočiai atlikti.
- Adaptuoti Lt-Llama-2 modelį ir testuoti jo mokymosi rezultatus keičiant mokymo hiperparametrus bei įvestyje pateikiamas užklausas.
- Atlikti supaprastintų tekstų kiekybinę ir kokybinę analizę.

1. Teoriniai pagrindai

1.1. Socialinės gerovės sektorius

Žmonių, turinčių negalią, įtraukimas į visuomenę ir darbo rinką tampa kritiniu aspektu valstybės socialinei gerovei. Socialinis įtraukumas pabrėžia socialinės lygybės, galimybių sudarymo ir paramos sistemos svarbą visiems valstybės gyventojams. Jungtinių Tautų Neigaliųjų teisių konvencija – tarptautinis žmogaus teisių dokumentas, įtvirtinantis asmenų, turinčių negalią, teises ir laisves – pabrėžia įtraukumą kaip pagrindinę žmogaus teisę visiems asmenims, įtraukiant asmenis, turinčius ilagalaikių fizinių, psichikos, intelekto ar jutimo sutrikimų, kurie šiems asmenims gali trukdyti visapusiškai dalyvauti visuomenėje ir būti lygiais su kitais asmenimis.

Remiantis [24] pasaulyje šiuo metu dirba 27 proc. darbingo amžiaus žmonių, turinčių negalią ir 56 proc. neturinčių negalios. Nors šie statistiniai duomenys nėra labai tikslūs, nes dažnai žmonės, turintys negalią nėra įtraukti į nedarbo rodiklius, kadangi dėl nustatytų socialinių normų ir barjerų jie nesiryžta ieškoti darbo, šie skaičiai, kurie realybėje gali būti dar mažesni, parodo, kad negalią turinčių žmonių atskirtis darbo rinkoje išlieka didelė. Lietuvoje negalią turinčių žmonių įtraukumas taip pat išlieka opi problema. Remiantis [24], procentiškas progresas įtraukiant žmones, turinčius negalią, į darbo rinką Lietuvoje, tarp 2015 ir 2019 metų, nesiekia net vidurkio (18 proc.). Šioje skalėje Lietuva vertinama tarp kitų 42 valstybių ir regionų ir yra įvertinta kaip viena žemiausiai esančių valstybių.

1.1.1. Informacijos prieinamumas žmonėms, turintiems negalią

Remiantis [24], tik 6 proc. Jungtinių Tautų šalių teikia viešą informaciją pasiekiamais formatais be jokių apribodjimų žmonėms, turintiems negalią, ir tik 1 proc. valstybių nurodo turinio prieinamumą žmonėms turintiems negalią kai informacija pateikiama internete. Taip pat, pagal [24] viešoji informacija ne visada pateikiama negalią turintiems žmonėms prieinamais formatais, pavyzdžiui, lengvai suprantama ar gestų kalba.

Prieinamumo trūkumas pasižymi ir viešosiose įstaigose, kurios paprastai teikia viešąją informaciją. Pagal [24] tik 49 proc. viešųjų bibliotekų visame pasaulyje turi prieinamumo politiką žmonėms, turintiems negalią, ir tik 15 proc. įstaigų turi prieinamumui skirtą biudžetą.

Vaikai, turintys negalią, taip pat susiduria su sunkumais ir nuo pat vaiksytės jaučia įtraukumo trūkumą. Remiantis [24], nors 11 proc. vaikų pasaulyje, turinčių negalią nelanko pradinio ugdymo mokyklų, šis skaičius padidėja net iki 32 proc. kalbant apie vaikus, nelankančius vyresniųjų klasių.

1.2. Lengvai suprantama kalba

Lengvai suprantama kalba tai tekstas, pasižymintis paprastais žodžiais, trumpais sakiniais. Šie tekstai pritaikomi prie žmonių specialiųjų poreikių, kad pateikta informacija būtų suprantama žmonėms, turintiems atminties, dėmesio sutrikimų, disleksiją, kitų psichikos sutrikimų ar kalbos barjerų, pavyzdžiui, žmonėms, kurie tik pradeda mokytis kalbos [MP14]. ETR leidžia šią plačią žmonių

grupę įtraukti į visuomenę, gerinti jų gerbūvį bei ginti jų teises į lengvai pasiekiamą informaciją. Kai informacija pateikiama prieinamu būdu, ji padeda asmenims geriau suprasti, sudaro sąlygas jiems priimti informuotus sprendimus ir pagerina jų įsitraukimo į visuomenę galimybes.

Rengiant ETR svarbu atsižvelgti į ETR rengimo gaires ir principus. Remiantis lietuviško teksto lengvai suprantama kalba rengimo gairėmis [BDV], ETR yra trijų lygių:

- Pirmojo lygio tekstai skirti žmonėms su sunkesne intelekto negalia, kurie patiria daugiau kognityvinių iššūkių. Šie tekstai yra labai trumpi, aiškūs ir supaprastinti – vidutiniškai 5 žodžių sakiniai perteikia esminę informaciją, dažnai iliustruojamą paveikslėliais.
- Antrojo lygio tekstai skirti žmonėms su mažiau kognityvinių iššūkių, lengvesne intelekto negalia ar besimokantiems lietuvių kalbos. Sakiniai vidutiniškai sudaryti iš 7 žodžių, perteikiant svarbiausią informaciją, o naują informaciją perteikiant palaipsniui, suteikiant konteksto, iliustruojant kasdienio gyvenimo pavyzdžiais.
- Trečiojo lygio tekstai yra skirti žemesnio išsilavinimo, neryškių intelekto sutrikimų turintiems ar pažengusiems lietuvių kalbos besimokantiems asmenims. Šiuose tekstuose pateikiama ir pagrindinė, ir detali informacija, sakinius vidutiniškai sudaro 10 žodžių.

Remiantis [MP14], ETR gali būti pateikta įvairias formatais – spausdintine, elektronine, garsine versijomis.

1.3. Natūralios kalbos apdorojimo technologijos

Natūralios kalbos apdorojimas (toliau trumpinama kaip NLP) yra dirbtinio intelekto sritis, kuri sprendžia sudėtingas užduotis, susijusias su žmogaus kalba, tokias kaip automatizuotas teksto vertimas, atsakymai į klausimus, teksto apibendrinimas ir kt. [LLA22]. NLP sujungia lingvistiką ir mašininį mokymąsi, leidžiant kompiuteriams analizuoti, suprasti ir kurti tekstus natūralia kalba, taip sudarant galimybes žmonėms efektyviai sąveikauti natūralia kalba su kompiuteriu. NLP dažnai yra skiriama į dvi pagrindines šakas: natūralios kalbos supratimą (angl. Natural Language Understanding, toliau trumpinama kaip NLP) ir natūralios kalbos generavimą (angl. Natural Language Generation, toliau trumpinama kaip NLG).

NLU leidžia technologijoms suprasti ir analizuoti kalbą, išskirti jos raktinius žodžius, emocijas, esybes. NLU paremtas lingvistika – kalbotyros mokslu – todėl kalbos analizavimui jam pritaikomos lingvistikos šakos tokios kaip fonologija, morfologija, sintaksinė, semantinė ir kt. [KKK+23].

NLG metu, remiantis išanalizuota kalba, žodis po žodžio generuojamos prasmingos frazės, sakiniai ir pastraipos. Šiuolaikinės NLG sistemos dažniausiai paremtos giliojo mokymosi modeliais, ypač transformerių architektūra, kuri naudojama dideliuose ir plačiai taikomuose kalbos modeliuose, tokiuose kaip OpenAI GPT, Meta Llama-2 ir kiti. Šie modeliai pritaikomi įvairioms teksto generavimo užduotims, įskaitant teksto supaprastinimą, santraukų kūrimą bei pokalbių sistemas, pavyzdžiui, pokalbių robotus (angl. chatbots).

1.4. Transformerių architektūra

Tikrasis proveržis NLP srityje prasidėjo atsiradus modernių natūralios kalbos apdorojimo technologijų pagrindui – transformeriams, kurie 2017 m. buvo pristatyti straipsnyje [VSP+]. Transformeriai išsiskiria nuo tradicinių sekų modelių, tokių kaip pasikartojantys neuroniniai tinklai (angl. Recurrent Neural Networks), nes naudoja savaimio dėmesio (angl. self-attention) mechanizmą. Šis mechanizmas leidžia fiksuoti ryšius tarp bet kurių sekos dalių, nepriklausomai nuo jų pozicijos, kiekvienam žodžiui sekoje priskiriant svarbos koeficientus (angl. attention score), gautus iš žodžių, transformuotų į užklausos (angl. query) ir rakto (angl. key) vektorius, atliekant su jais veiksmus, taip suprantant kiekvieno žodžio svarbą kontekste [VSP+]. Todėl modeliai, pritaikantys šį mechanizmą, apdoroja duomenis paraleliai, o ne nuosekliai, taip užtikrinant didesnį efektyvumą. Remiantis [VSP+], transformerių architektūra grįsti modeliai vertimo užduotims gali būti daug greičiau apmokomi nei modeliai, paremti pasikartojančiomis ar konvoliucinėmis architektūromis. Be to, tiek anglų–vokiečių, tiek anglų–prancūzų kalbų vertimo užduotyse transformeriai pranoksta anksčiau geriausiais laikytus modelius pagal BLEU rodiklį. Pavyzdžiui, anglų–vokiečių kalbų vertimo užduotyje konvoliucinių tinklų modelis ConvS2S pasiekia 26,36 BLEU rezultatą, o transformerių pagrindu sukurtas modelis – 28,4 BLEU rezultatą, parodydamas ženklią kokybės persvarą.

Transformerių architektūrą sudaro dvi pagrindinės dalys: koduotojas (angl. encoder) ir dekoduotojas (angl. decoder). Koduotojas apdoroja įvesties duomenis ir generuoja jų tarpines reprezentacijas, o dekoduotojas, remdamasis šiomis reprezentacijomis, generuoja išvesties sekas. Tokios užduotys kaip teksto klasifikavimas, esybių atpažinimas (angl. named entity recognition) ir sentimentų analizė, dažniausiai atliekamos modelių, paremtų tik koduotoju. Pavyzdžiui, BERT (angl. Bidirectional Encoder Representations from Transformers), kuris buvo pristatytas straipsnyje [DCL+19], leidžia suprasti kontekstą ir teksto prasmę dvikrypčiu būdu, naudojantis savaimio dėmesio mechanizmu. Šis modelis pasiekė reikšmingų rezultatų įvairiose NLP užduotyse, tačiau negali generuoti teksto, nes yra sukurtas naudojant tik koduotojo architektūrą. Tuo tarpu teksto generavimo užduotims, kaip automatiniam vertimui, santraukų kūrimui ar dialogų generavimui, dažniausiai naudojami dekoduotojo arhitektūra grįsti modeliai, tokie kaip Llama-2 [TMS+23] ar GPT-3 [BMR+20]. Šie modeliai remiasi vien dekoduotojo architektūra, kuri leidžia efektyviai apdoroti įvestį ir generuoti nuoseklų atsakymą. Tokie modeliai geba interpretuoti pateiktą užklausą bei generuoti prasmingą atsakymą, pasitelkiant savaimio dėmesio mechanizmu išsaugotą kontekstą visame atsakyme. Kuriant ilgesnius, sudėtingesnius tekstus arba atliekant automatines vertimo užduotis, taip pat naudojami pilnos koduotojo-dekoduotojo architektūros modeliai. Tokiu atveju, modeliai kaip T5 [RSR+23] pirmiausia koduoja įvestį, išanalizuodami jos kontekstą ir prasmę, o vėliau dekodavimo metu panaudoja šią informaciją tikslesniam atsakymo generavimui. Tokie modeliai geba rašyti geresnes santraukas ilgiems tekstams, kada svarbu tekste išlaikyti kontekstą, taip pat gali patys generuoti atsakymus į klausimus.

Vienas iš svarbiausių transformatorių veikimo principų, vykdomų tiek koduotojo, tiek deko-

duotojo, yra teksto skaidymas į teksto vienetus (angl. tokens). Tokie teksto skaidymo į teksto vienetus modeliai kaip „Byte-Pair Encoding“ (toliau trumpinama kaip BPE) ar „WordPiece“ skaido įvesties tekstą į teksto vienetus – žodžius, žodžių dalis ar simbolius, taip leidžiant modeliams efektyviai dirbti su retais ar sudėtingais kalbos dariniais. Pavyzdžiui, modeliuose kaip Llama-2, BPE teksto skaidymas į teksto vienetus leidžia dirbti su įvairiomis kalbos struktūromis apmokant modelį, užtikrinant efektyvų išteklių panaudojimą ir geresnį apmokymo rezultatą [TMS+23].

1.5. Modelio adaptavimas

Transformeriais paremti modeliai gali būti iš anksto apmokyti (angl. pre-trained) su dideliais kiekiais duomenų. Šis išankstinis mokymas paskatina modelį įgyti bendrųjų žinių ir išmokti šablonus. Vėliau šie modeliai gali būti pritaikomi specifinėms užduotims, tokioms kaip teksto supaprastinimas. Tai pasiekiama adaptavimu (angl. fine-tuning) su nedideliu, bet specifinei užduočiai pritaikytu duomenų rinkiniu. Šis procesas leidžia modeliui geriau suprasti konkrečius užduoties reikalavimus ir pateikti išvestis, labiau tinkančias konkrečiai užduočiai. Šis procesas paprastai vyksta pagal perdavimo mokymosi (angl. transfer learning) principą, kai jau esamos modelio žinios yra pritaikomos naujiems tikslams.

Norint tinkamai pritaikyti modelį, pavyzdžiui, Llama-2 ar GPT-3, labai svarbu atlikti tinkamą adaptavimo procesą. Kad būtų pasiektas geriausias rezultatas, svarbūs veiksniai yra pasirinktos mokymo reikšmės, tokios kaip mokymosi greitis (angl. learning rate) ir optimizavimo metodai. Remiantis [TMS+23], Llama-2 modelio adaptavimas buvo vykdomas naudojant kosinuso mokymosi greičio schemą su pradine mokymosi sparta 2×10^{-5}, svorio mažėjimo koeficientu 0,1, duomenų rinkinių dydžiu (angl. batch) 64 ir 4096 teksto vienetų sekos ilgiu. Nuostoliai (angl. loss) skaičiuojami tik rezultatų teksto vienetams, o užklausos teksto vienetai į nuostolius buvo neįtraukiami, taip užtikrinant, kad gradientai būtų skaičiuojami tik atsakymo teksto vienetų atžvilgiu. Modelis buvo adaptuojamas 2 epochas. Ši strategija leido modeliui efektyviai optimizuoti tekstų generavimą, išlaikant nuoseklų atsakymą net ir esant ilgoms tekstų sekoms.

Papildomi tyrimai, kaip [SKY+18], pabrėžia svarbą ne tik pasirinkti teisingą mokymosi greitį, bet ir koreguoti modeliui paduodamo duomenų rinkinio (angl. batch) dydį, siekiant užtikrinti greitesnį ir efektyvesnį modelio apmokymą. Šie tyrimai įrodė, kad didesnių partijų dydžiai kartu su adaptaciniais mokymosi greičiais gali ženkliai pagerinti modelio našumą ir sumažinti apmokymo laiką.

Tinkamai pritaikant šiuos metodus, galima užtikrinti, kad modeliai, paremti dekoduotojo architektūra, kaip Llama-2, būtų sėkmingai pritaikyti įvairioms teksto generavimo užduotims, įskaitant automatines teksto supaprastinimo, santraukų kūrimo, dialogų generavimo ar vertimo užduotis.

1.5.1. Kelių pakopų adaptavimas

Pastaruoju metu daugėja tyrimų, pabrėžiančių kelių pakopų modelio adaptavimo (angl. multistage fine-tuning) strategijos efektyvumą, ypač tais atvejais, kai specifinei užduočiai pritaikytų duo-

menų kiekis yra ribotas. Kelių pakopų modelio adaptavimo metodas pagrįstas su daug duomenų apmokyto modelio adaptavimu konkrečiai užduočiai kelis kartus – pirmiausia naudojant bendresnio pobūdžio duomenis, o galiausiai – konkrečiai užduočiai pritaikytus duomenis. Tokia strategija leidžia modeliui palaipsniui įgyti reikiamų žinių ir stilistinių ypatybių, reikalingų konkrečiam uždaviniui atlikti [GMS+20].

Darbo [GMS+20] autorius pabrėžė, kad prie srities prisitaikantis išankstinis apmokymas (angl. domain-adaptive pretraining, toliau trumpinama kaip DAPT) ir konkrečiai užduočiai pritaikytas išankstinis apmokymas (angl. task-adaptive pretraining, toliau trumpinama kaip TAPT), atitinkantis kelių pakopų adaptavimą, žymiai pagerino rezultatų kokybę teksto klasifikavimo ir analizės užduotyse. Pavyzdžiui, „CHEMPROT“ užduotyje „RoBERTa“ modelis, nenaudojant DAPT ir TAPT metodų, pasiekė 81.9 „micro-F1“ vertinimo rodiklio rezultatą, o taikant DAPT ir TAPT – 84.4. Panašus pagerėjimas pastebėtas ir „ACL-ARC“ užduotyje, pagerinus „macro-F1“ vertinimo rodiklio rezultatą nuo 63.0 iki 75.6. Tyrimo metu, pirmąjai pakopai modeliai buvo papildomai apmokyti naudojant tikslinės srities (angl. domain-specific) duomenis, o antrąjai – konkrečios užduoties (angl. task-specific) duomenis. Nors tikslinės srities duomenų kiekis buvo nedidelis, rezultatai parodė, kad dviejų pakopų adaptavimas leidžia pasiekti geresnių rezultatų nei tiesioginis modelio adaptavimas su mažu, konkrečiai užduočiai pritaikytu, duomenų rinkiniu.

Visgi, taikant kelių pakopų adaptavimą būtina atsižvelgti į keletą iššūkių. Vienas jų – rizika, kad modelis pernelyg prisitaikys prie bendro pobūdžio duomenų rinkinio ir praras gebėjimą prisitaikyti prie specifinių savybių galutinėje pakopoje – įvyks katastrofinis užmiršimas (angl. catastrophic forgetting). Darbo [HR18] autoriai pažymi, kad svarbu pasirinkti tinkamus adaptavimo hiperparametrus, kadangi per dideli pasirinkti adaptavimo hiperparametrai gali perrašyti naudingą informaciją, įgytą per išankstinį kalbos modeliavimą, tuo tarpu per maži hiperparametrai gali lemti per didelį prisitaikymą. Be to, kiekviena papildoma modelio adaptavimo pakopa reikalauja papildomų skaičiavimo išteklių, todėl parinkus netinkamus parametrus, šis metodas gali būti brangesnis už klasikinį vienos pakopos adaptavimą.

Nepaisant to, tokia strategija ypač naudinga mažiau išteklių turinčioms kalboms ar specifinėms užduotims, kurioms duomenų kiekis yra ribotas – pavyzdžiui, lengvai suprantamos kalbos (ETR) sintezėje lietuvių kalbai. Šiuo atveju, tarpiniu etapu galima naudoti į paprastą kalbą (angl. plain language) supaprastintus tekstus, taip suteikiant modeliui papildomų žinių, kurios gali padėti kokybiškiau išmokti lietuvių kalbos supaprastinimo į ETR užduočiai būdingus teksto požymius.

1.5.2. Išteklių optimizavimas naudojant QLoRA

Kiekybinis mažo rango adaptavimas (angl. Quantized Low-Rank Adapter, toliau trumpinama kaip QLora) tai metodas, leidžiantis sutaupyti atminties resursų atliekant LLM adaptavimo užduotį, pristatytas [DPH+] straipsnyje. QLoRA sujungia dvi pažangias technologijas: 4 bitų svorių kvantavimą ir žemo rango adaptacijos metodą LoRA. Vietoje to, kad būtų atnaujinami visi modelio parametrai, QLoRA palieka bazinius svorius užšaldytus ir treniruoja tik nedidelius, specialiai pridėtus adapterius. Tokiu būdu gerokai sumažinamas atminties poreikis bei padidinamas treniravimo

greitis. [DPH+] autoriai pabrėžia, kad naudojant QLora adaptuojant skirtingo dydžio LLama modelius, pasiekiama MMLU metrikos rezultatų vidurkio reikšmė lygi 53,1%. Ši reikšmė nestipriai pralenkia pilno 16 bitų *(bfloat16* adaptavimo proceso rezultato reikšmę, kuri lygi 53,0%. Tai parodo, kad QLoRA leidžia pasiekti panašius rezultatus, sumažinant atminties panaudojimą, leidžiant adaptuoti net labai didelį LLaMA 65B modelį su 48 GB GPU.

1.5.3. Užklausų kūrimas

Užklausų kūrimas (angl. prompt engineering) yra svarbus metodas, padedantis efektyviai pritaikyti didžiuosius kalbos modelius (toliau trumpinama kaip LLM) specifinėms užduotims be papildomo modelio adaptavimo. Šis metodas paremtas suformuluotomis instrukcijomis, kurios modeliui nurodo norimą užduoties rezultatą. Užklausų kūrimas yra ypač naudingas, kai reikia greitai ir efektyviai pritaikyti modelį mažai išteklių turinčioms užduotims, pavyzdžiui, lietuviškų tekstų supaprastinimui į lengvai suprantamą kalbą.

Remiantis sistemine užklausų rengimo metodų apžvalga [SIB+25], struktūruotos instrukcijos, kuriose aiškiai nurodomas užduoties tiklas ir kokybės kriterijai, leidžia reikšmingai pagerinti užduoties rezultatų kokybę. Ypač efektyvios yra nurodymais grįstos užklausos (angl. instruction-based prompts), kai modeliui pateikiamoje įvestyke aiškiai nurodoma, ko tikimasi iš modelio. Supaprastinant tekstą į lengvai suprantamą kalbą, užklausoje galima pateikti teksto lengvai suprantama kalba gairėse nurodytas rekomendacijas, taip užklausoje nurodant konkrečias instrukcijas ir pagerinant rezultatų kokybę. Darbo utorius taip pat pabrėžia, kad šiuo metu vis dar dažniausiai užklausų kūrimo metodai pritaikyti vykdant užduotis anglų kalba, todėl norint pasiekti panašios kokybės rezultatus užsienio kalbomis, dažnai reikia naudoti specialius metodus.

Vienas populiariausių ir lengviausių naudoti, užduotims ne anglų kalba, metodų yra „mąstyk žingsnis po žingsnio“ (angl. Think Step by Step arba Chain-of-Thought, toliau trumpinama kaip CoT). Naudojant šį metodą, modeliui pateiktos užklausos pabaigoje pridedama frazė „mąstyk žingsnis po žingsnio“. Tai leidžia modeliui geriau apdoroti ir vykdyti sudėtingesnes, kelių žingsnių reikalaujančias užduotis. Remiantis [KGR+23] atliktu tyrimu, pridėjus „mąstyk žingsnis po žingsnio“ prie modeliui pateiktos užklausos, MultiArith duomenų rinkinyje modelio tikslumas išaugo nuo 17,7% iki 78,7%, o GSM8K duomenų rinkinyje – nuo 10,4% iki 40,7%. Norint užklausos pabaigoje pridėti „mąstyk žingsnis po žingsnio“, taip pat svarbu paruošti ir tinkamą užklausą. Dažnai naudojami užklausų tipai yra nulinio mokymo (angl. zero-shot) ir kelių pavyzdžių (angl. few-shot) metodai. Nulinio mokymo (angl. zero-shot) užklausos sudaromos, kai pateikiamas tik užduoties aprašymas be pavyzdžių, ir modelis generuoja atsakymą remdamasis šiuo aprašymu. Pavyzdžiui, užduoties formulavimas gali būti toks: „Supaprastink šį tekstą į lengvai suprantamą kalbą, naudok trumpus sakinius ir paprastus žodžius.“ Jei norima modeliui pateikti užduoties pavyzdžius, naudojamos kelių pavyzdžių (angl. few-shot) užklausos.

Taip pat užduotims ne anglų kalba dažnai naudojama užklausos išvertimo (angl. translate first prompting) strategija, kai ne anglų kalba įvesta užklausa pirmiausia išverčiama į anglų kalbą, kad modelis galėtų pritaikyti savo stiprybes atliekant užduotis anglų kalba [SIB+25].

Didesniuose modeliuose, tokiuose kaip „Llama-2“, užklausų kūrimas leidžia išnaudoti modelio išankstinio mokymo metu įgytas žinias. Pasitelkiant tokius metodus, galima pasiekti aukštą supaprastinto teksto kokybę, nepriklausomai nuo kalbos ar turimų duomenų kiekio.

1.6. Vertinimo metrikos

Apmokytų modelių, naudojančių natūralios kalbos apdorojimo technologijas, vertinimas yra svarbus ne tik dėl našumo, bet ir dėl sugeneruotų rezultatų kokybės. Norint tiksliai įvertinti modelio gebėjimus, būtina pasirinkti tinkamas metrikas, kurios atitiktų atliekamą NLP užduotį ir kalbą, su kuria modelis dirba.

Vertinant dekoduotojo transformerių modelius, tokius kaip Llama-2 ar GPT-3, naudojami tiek automatiniai, tiek ekspertiniai vertinimo metodai. Viena populiariausių kalbos modelių kokybės vertinimo metrikų yra BLEU (angl. Bilingual Evaluation Understudy), kuri buvo pristatyta dar 2002 metais, ir iki šiol plačiai naudojama įvairioms NLP užduotims vertinti. BLEU metrika ypač naudinga automatizuoto vertimo užduočių vertinimui, nes ji vertina, kiek sugeneruotas tekstas sutampa su originaliu tekstu pagal žodžių ar jų junginių atitikimą. BLEU metrika vertina sugeneruotą tekstą, lygindama jį su viena ar keliomis žodžio dalimis, apskaičiuojant, kiek dalių sugeneruotame tekste randama originaliame. Nepaisant plataus naudojimo NLP užduotims vertinti, BLEU netinka semantikos vertinimui ir dažnai yra kritikuojama dėl riboto efektyvumo vertinant konteksto išlaikymą, pavyzdžiui, teksto supaprastinimo užduotims. Kita populiari vertinimo metrika – ROUGE (angl. Recall-Oriented Understudy for Gisting Evaluation), dažniausiai naudojama vertinant teksto santraukas. Ji vertina, kiek sugeneruotas tekstas atitinka originalias teksto sekas. Taip pat, vertinant supaprastintą tekstą, dažnai naudojama metrika SARI (angl. System output against reference sentences for text simplification). Šis vertinimo būdas tikrina tris esminius aspektus: žodžių pridėjimą, pašalinimą ir išlaikymą. Remiantis [XNP+16], SARI metrika geriau vertina supaprastintą tekstą nei BLEU, nes ji tiksliau atspindi konteksto išlaikymą ir žmogaus skaitomumą. Be to, SARI geriau sutampa su ekspertiniu vertinimu dėl teksto paprastumo ir pasiekia aukštesnius rezultatus vertinant teksto supaprastinimo sistemas, palyginti su kitomis metrikomis, tokiomis kaip BLEU, FKBLEU ir kitos. Dar viena plačiai naudojama teksto vertinimo metrika – BERTScore [ZKW+20]. Ši metrika, naudojanti BERT ar kitus transformerių modelius, vertina semantinį tekstų panašumą ir dažnai pasitelkiama vertinant teksto kokybę, ypač kai svarbu išlaikyti prasmę. Tačiau BERTScore gali būti mažiau tinkama vertinant dekoduotojo modelius, nes ji labiau orientuota į koduotojo-dekoduotojo architektūras. Dar viena svarbi metrika, ypač dažnai naudojama kalbos modelių treniravimo ir adaptavimo metu, yra „Perplexity“ (toliau trumpinama kaip PPL). Ji matuoja modelio gebėjimą prognozuoti sekantį žodį tekste. Kuo mažesnė PPL reikšmė, tuo labiau modelis „pasitiki“ savo prognozėmis, kadangi turi mažesnį kiekį alternatyvių pasirinkimų. 1 lentelėje pavaizudotos šiame skyriuje pristatytos vertinimo metrikos ir jų palyginimas pagal pagrindinį tikslą.

1 lentelė. Automatinių teksto vertinimo metrikų palyginimas

Metrikos pavadinimas	Pagrindinis tiklas
BLEU	Vertimų vertinimas
SARI	Santraukų vertinimas
ROUGE	Supaprastintų tekstų vertinimas
BERTScore	Semantinio tekstų panašumo vertinimas
Perplexity	Modelio gebėjimas prognozuoti kitą žodį tekste

Skirtingi vertinimo metodai gali geriau pasirodyti vertinant skirtingas užduotis, taip pat gali skirtis jų kokybė skirtingoms kalboms. Ekspertinis vertinimas taip pat lieka neatsiejama vertinimo dalimi, nes žmonės gali tiksliai įvertinti, kaip iš tiesų suprantami, pritaikomi ir kokie yra sugeneruotų tekstų kokybiniai aspektai. Tačiau žmogiškas vertinimas reikalauja didelių išteklių, specialistų ir yra daug brangesnis laiko atžvilgiu.

1.6.1. Dideli kalbos modeliai kaip vertintojai

Vertinant didelių kalbos modelių užduotis pasitelkti galima ne tik automatinėmis metrikomis ar specialistų, suinteresuotų šalių vertinimais. Šioms vertinimo užduotims vis dažniau naudojami didieji kalbos modeliai ne tik užduočių, tokių kaip teksto generavimas, supaprastinimas, vertimas, bet ir kaip šių užduočių vertinimo įrankiai. Šis metodas pagrįstas tuo, kad iš anksto apmokyti kalbos modeliai geba ne tik suprasti užduoties specifiką, bet ir įvertinti sugeneruotų tekstų kokybę pagal kelis aspektus, tokius kaip gramatinis ir sintaksės korektiškumas, konteksto ir prasmės išlaikymas, skaitomumas ir paprastumas ir kiti.

Pavyzdžiui, Llama-2 arba GPT modelius galima pritaikyti kaip vertintojus, pateikiant jiems originalų ir sugeneruotą tekstą bei prašant palyginti jų kokybę pagal nurodytus kriterijus. Tokios sistemos pranašumas yra gebėjimas greitai įvertinti didelį tekstų kiekį ir pateikti nuoseklius vertinimus, tačiau svarbu užtikrinti, kad vertinimo procesas būtų patikimas ir atitiktų užduoties tikslus. Siekiant gauti kokybiškesnius vertinimo rezultatus, svarbu pateikti gerai sudarytą užklausą modeliui, naudojamam kaip vertintojui [SIB+25].

Didelių kalbos modelių, kaip vertinimo sitemų įvairioms darbo su tekstu užduotims, naudojimas gali sumažinti žmogiškojo vertinimo poreikį, taupant išteklius, specialistų laiką. Taip pat tai gali padėti užtikrinti greitesnį rezultatų grįžtamąjį ryšį didelių kalbos modelių adaptavimo įvairioms užduotims procese. Nors LLM kaip vertintojai gali padėti adaptuojant modelius atlikti užduotis su tekstu, šis metodas turi ir savų iššūkių. Didžiausias iš jų – modelio šališkumas. LLM kaip vertintojai gali pateikti neteisingus įvertinimus, jei duomenys, kuriais jie buvo apmokyti, neatspindi užduoties specifikos.

2. Metodologija

2.1. Duomenų rikiniai

Kelių pakopų modelio adaptavimui buvo pasirinkti du duomenų rinkiniai: vienas didesnės apimties ir bendresnio pobūdžio, kitas – mažesnis ir specializuotas, pritaikytas tekstų supaprastinimui į lengvai suprantamą kalbą. 2 lentelėje pavaizduotas duomenų kiekis bendrojo pobūdžio ir specializuotame duomenų rinkiniuose.

2 lentelė. Duomenų kiekis bendrojo pobūdžio ir specializuotame duomenų rinkiniuose

Duomenų rinkinys	Įrašų skaičius	Žodžių skaičius originaliame tekste	Žodžių skaičius supaprastintame tekste
Tekstų paprasta kalba duomenų rinkinys	2091	36439	34529
Tekstų lengvai suprantama kalba duomenų rinkinys	304	5677	5534

Pirmasis duomenų rinkinys buvo surinktas iš [MRK+25] tyrimui paruošto duomenų rinkinio, kuriame kiekvienas originalus sakinys, paimtas iš vyriausybinių viešųjų institucijų interneto svetainių, turėjo supaprastintą atitikmenį paprasta kalba (angl. plain language). Šis rinkinys buvo naudojamas pirmajai modelio adaptavimo pakopai, kurios tikslas – išmokyti modelį bendrai supaprastinti tekstą. Buvo pasirinkta naudoti būtent šį duomenų rinkinį, kadangi tai didesnės apimties duomenų rinkinys lietuvių kalba, kuris sudarytas iš 2091 originalių ir supaprastintų sakinių porų, apimančių 36439 žodžius originaliuose ir 34529 žodžius supaprastintuose tekstuose.

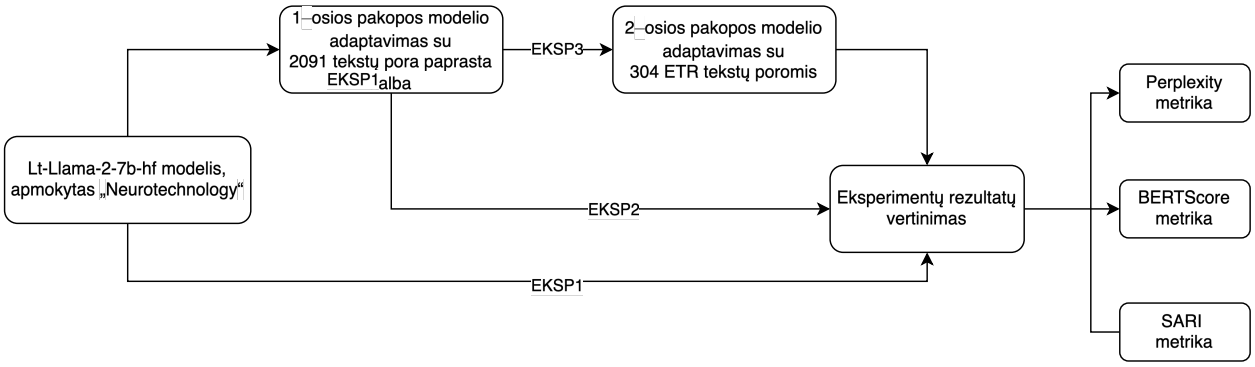
Antrasis duomenų rinkinys sudarytas iš į lengvai suprantamą kalbą supaprastintų tekstų. Lietuvių kalba kai kurie tekstai jau yra specialistų verčiami į lengvai suprantamą kalbą. Dalis šių tekstų yra laisvai prieinami internete. Šie supaprastinti tekstai tai dokumentai, dažniausiai pateikiami pdf formatu, o jų atitikmenys įvairiais formatais taip pat paskelbti internete. Šie dokumentai buvo naudojami kaip pagrindinis šaltinis lengvai suprantamos kalbos generavimo užduočiai atlikti.

Eksperimentams pasirinkti 2-ojo lygio lengvai suprantamos kalbos tekstai. Šie tekstai buvo lyginami su jų originaliais atitikmenimis, sudarant duomenų rinkinius. Kuriant duomenų rinkinius, buvo siekiama užtikrinti, kad visi sakiniai originalia kalba turėtų savo tiesioginius atitikmenis lengvai suprantama kalba. Taip buvo išvengta tekstų, kuriuose lengvai suprantamos kalbos versijos pateikia papildomą kontekstą ar pašalina papildomą informaciją, kuri pateikiama originaliuose tekstuose.

Galutiniame ETR duomenų rinkinyje originalių tekstų apimtis sudaro 304 įrašus, sudarytus iš bent vieno sakinio. Šie originalūs įrašai iš viso sudaro 5677 žodžius. Supaprastinti tekstai sudaryti iš 5534 žodžių. Šie duomenys sudaro pagrindą tolimesniems eksperimentams, kuriuose analizuojama LLM galimybė supaprastinti tekstus į lengvai suprantamą kalbą ir įvertinama supaprastintų tekstų kokybė.

2.2. Kalbos supaprastinimas į lengvai suprantamą kalbą

Visų eksperimentų vykdymo tvarka bei pasirinkti eksperimentų rezultatų vertinimo būdai pavaizduoti 1 pav. schemoje.



1 pav. Eksperimentų srauto schema

Kalbos supaprastinimo į lengvai suprantamą kalbą užduočiai atlikti buvo pasirinktas Neurotechnology pristatytas, su daug duomenų lietuvių kalba apmokytas, transformerių modelis Lt-Llama-2-7b-hf [NDM24]. Šis modelis pasirinktas dėl to, kad Lietuvoje dar trūksta stiprių didelių kalbos modelių, o šio modelio pagrindas – efektyvi ir plačiai paplitusi Llama-2 architektūra.

3 lentelė. Su Lt-Llama-2-7b-hf modeliu atlikti eksperimentai

Eksperimento nr.	Eksperimento aprašymas
EKSP1	Su daug duomenų lietuvių kalba apmokytas Lt-Llama-2-7b-hf modelis
EKSP2	Teksto lietuvių kalba supaprastinimo į lengvai suprantamą kalbą užduočiai adaptuotas Lt-Llama-2-7b-hf modelis, adaptuojant tik su ETR duomenimis
EKSP3	Teksto lietuvių kalba supaprastinimo į lengvai suprantamą kalbą užduočiai adaptuotas Lt-Llama-2-7b-hf modelis, adaptuojant dvejomis pakopomis: su paprastos kalbos duomenų rinkiniu ir su ETR duomenų rinkiniu

Tam, kad būtų galima įvertinti Lt-Llama-2-7b-hf modelio adaptavimo poreikį ir kokybę, buvo atlikti eksperimentai, naudojant tik apmokytą, viena pakopa adaptuotą bei dviem epochomis adaptuotą Lt-Llama-2-7b-hf modelį. Šie eksperimentai ir identifikatoriai pavaizduoti 3 lentelėje.

2.2.1. Išteklių optimizavimas naudojant QLoRA

Siekiant sutaupyti skaičiavimo išteklių, modelio adaptavimui planuojama naudoti kiekybinį mažo rango adaptavimo metodą QLoRA. Šis metodas leidžia efektyviai atlikti modelio adaptavimą net naudojant ribotus GPU išteklius, išsaugant aukštą rezultatų kokybę.

Eksperimentų metu modelis bus kvantuotas (angl. quantized) 4 bitų tikslumu, pasitelkiant NF4 kvantavimo metodą, kuris leidžia efektyviai išlaikyti svarbiausią modelio parametrų informaciją. informaciją apie modelio parametrus. Taip pat mokymo metu planuojama įjungti sluoksnių perkėlimą į CPU, taip efektyviau paskirstant apkrovą tarp sistemos GPU ir procesoriaus.

Adapterių (LoRA) konfigūracijai pasirinkti šie pagrindiniai nustatymai: mažo rango matricos dydis r=8, LoRA stiprumo koeficientas α=16, o dropout reikšmė – 0,05. Adapteriai bus taikomi pagrindinėse savaimio dėmesio mechanizmo vietose: q_proj ir v_proj sluoksniuose, kurie, turi didžiausią poveikį modelio prisitaikymui prie mokomos užduoties.

QLora konfigūracija modelio adaptavimo metu bus naudojama naudojantis PEFT biblioteka.

2.2.2. Mokymo parametrai

Prieš atliekant eksperimentus su skirtingai apmokytu modeliu ir skirtingomis užklausomis, planuojama atlikti tinkamiausių modelio adaptavimo užduočiai hiperparametrų paieškos procesą. Planuojama įvertinti modelio našumą bei nuostolio reikšmių pokytį modelio adaptavimo metu, keičiant šiuos hiperparametrus:

- Mokymosi žigsnių dydį. Planuojama išbandyti modelio adaptavimą su mokymosi žingsnių reikšmėmis nuo 5×10^{-6} iki 5×10^{-5}.
  - Atliekant pirmos pakopos adaptavimą, kai modelis apmokomas bendro pobūdžio duomenimis, bus taikomas didesnis mokymosi žingsnis (nuo 5×10^{-6} iki 3×10^{-5}), siekiant greičiau prisitaikyti prie lietuviško teksto supaprastinimo užduoties pobūdžio.
  - Atliekant antros pakopos adapvatimą su teksto supaprastinimo į ETR užduočiai specifiniais duomenimis, planuojama naudoti mažesnį žingsnio dydį (3×10^{-6}, 2×10^{-5}), kad būtų išvengta katastrofiško užmiršimo.
- Adaptavimo pradžioje planuojama taikyti žingsnių sušilimo (angl. warmup) etapą, siekiant stabilizuoti svorių atnaujinimą pirmosiomis adaptavimo fazėmis. Planuojama išbandyti 5% ir 10% visų žingsnių sušilimo etapą.
- Optimizavimo algoritmas. Modelio adaptavimui pasirinktas „AdamW“ optimizavimo algoritmas. Šis algoritmas leidžia efektyviai naudoti atmintį naudojant QLoRA metodą, palaikant stabilų svorių atnaujinimą.
- Duomenų rinkiniai (angl. batches). Bus testuojami duomenų rinkinių dydžiai nuo 4 iki 8 įrašų, atsižvelgiant į atminties sunaudojimą ir gradientų kaupimo suderinimą.
- Gradientų kaupimas (angl. gradient accumulation). Planuojama naudoti 2 žingsnių gradientų kaupimą, siekiant efektyviau išnaudoti GPU atmintį ir apdoroti didesnes duomenų rinkinio (angl. batch) reikšmes. Tokiu būdu modelis atnaujina svorius tik kas antrą žingsnį, tarsi būtų mokomas su dvigubai didesniu duomenų rinkiniu. Tai leidžia simuliuoti didesnį įrašų kiekį be papildomų atminties sąnaudų.

- Sluoksnių užšaldymas (angl. freezing). Eksperimentuose bus lyginami du variantai:
  - Visi modelio sluoksniai atviri mokymui.
  - Pirmieji sluoksniai užšaldyti. Užšaldyti sluoksniai yra tuomet, kai sluoksnio svoriai nėra atnaujinami mokymo ar adaptavimo metu.
- Modelio persimokymo (angl. overfitting) prevencija. Adaptavimo metu planuojama stebėti mokymosi validacijos duomenų rinkinio nuostolių reikšmes, siekiant nuspręsti kada modelis pradeda persimokyti ir kada reikėtų iš anksto sustabdyti (angl. early stopping) adaptavimo procesą.

2.2.3. Užklausų kūrimas

Adaptuotas modelis bus testuojamas su anksčiau nematytais tekstais. Tie patys tekstai bus paduodami modeliui kelis kartus, pateikiant juos su vis kitokia užklausa, taip patikrinant modelio rezultatų priklausomybę nuo užklausų formuluotės.

4 lentelėje pateikiamos modelio testavimui naudojamų užklausų formuluotės ir jų identifikacijos numeriai.

4 lentelė. Modelio testavimui pritaikytų užklausų identifikatoriai ir formuluotės

Užklausos ID	Užklausos formuluotė
1	Supaprastink šį tekstą į lengvai suprantamą kalbą žmonėms, turintiems lengvesnę intelekto negalią ar besimokantiems lietuvių kalbos.
2	Supaprastink šį tekstą į lengvai suprantamą kalbą žmonėms, turintiems lengvesnę intelekto negalią ar besimokantiems lietuvių kalbos. Naudok paprastus žodžius, trumpus sakinius, sudarytus iš vidutiniškai 7 žodžių. Mąstyk žingsnis po žingsnio.
3	Supaprastink šį tekstą į lengvai suprantamą kalbą žmonėms, turintiems lengvesnę intelekto negalią ar besimokantiems lietuvių kalbos. Pavyzdys: Q: „Noriu nutraukti paslaugų teikimo sutartį. Ar gali paslaugų teikėjas iš manęs pareikalauti netesybų?“ A: „Sutartį noriu nutraukti anksčiau. Ar man reikės sumokėti daugiau pinigų, nes sutartį nutraukiau anksčiau?“

4	Supaprastink šį tekstą taip, kad jis atitiktų lengvai suprantamos kalbos antrąjį lygį (A2), skirtą žmonėms, turintiems lengvesnę intelekto negalią arba besimokantiems lietuvių kalbos. Vadovaukis toliau pateiktomis gairėmis: - Išrink tik svarbiausią informaciją, atsisakyk detalių, kurios nėra būtinos pagrindinei minčiai. - Naudok paprastus sakinius. Sakinyje turi būti vidutiniškai 7 žodžiai. - Vartok tik dažniausiai vartojamus žodžius. Jei reikia panaudoti rečiau naudojamus žodžius, juos paaiškink. - Pakeisk sudėtingus arba tarptautinius žodžius paprastais sinonimais. - Pašalink vardus, pavadinimus, terminus, kurie nėra būtini. - Iliustruok mintis paprastais kasdieniais pavyzdžiais. Mąstyk žingsnis po žingsnio.

Modelio testavimui pasirinkti 4 užklausų kūrimo metodai:

- Paprasta instrukcija. Tai paprasta nulinio mokymo (angl. zero-shot) užklausa be pavyzdžių, skirta įvertinti, kaip modelis geba atlikti užduotį vien tik pagal aiškiai pateiktą instrukciją.
- Instrukcija ir „mąstyk žingsnis po žingsnio“ metodas. Papildoma instrukcija skatina modelį atlikti užduotį analizuojant ją nuosekliai, žingsnis po žingsnio. Tokia strategija padeda pagerinti modelio rezultatų kokybę, ypač kai tekstas sudėtingesnis.
- Instrukcija su pavyzdžiu. Kelių pavyzdžių (angl. few-shot) užklausa suteikia modeliui papildomos informacijos apie tai, koks yra pageidaujamas rezultatas.
- Instrukcija, paremta 2 lygio ETR rengimo gairėmis, pateikiant svarbiausius punktus, vertus atsižvelgti, kad informacija būtų tikrai suprantama. Šios instrukcijos pabaigoje pateikiamas „mąstyk žingsnis po žingsnio“ metodas, kad modelis nuosekliai pritaikytų kiekvieną gairių punktą supaprastinamam tekstui.

2.3. Supaprastintų tekstų vertinimas

Supaprastintų tekstų vertinimui pasirinkta naudoti 10% testavimo duomenų aibės poaibį.

5 lentelė. Originalių ir specialistų išverstų tekstų porų pavyzdžiai

Pavyzdžio ID	Originalus tekstas	Specialistų supaprastintas tekstas
1	Siunčiant pašto siuntą į užsienį, svarbu žinoti, kad kiekviena šalis nustato, kokius daiktus į jas yra draudžiama siųsti, todėl siuntėjui rekomenduojama dėl detalios informacijos kreiptis į pašto paslaugos teikėją, kuriam bus atiduota siųsti pašto siunta.	Jei siunčiate daiktus į užsienį, žinokite, kad kiekviena šalis turi savo taisykles.
2	Kiekvienas turi teisę į pilietybę.	Mes visi galime būti kurios nors šalies piliečiais.

5 lentelėje pavaizduoti keli originalių tekstų ir specialistų parengtų supaprastintų tekstų pavyzdžiai iš testavimui skirtos duomenų aibės.

2.3.1. Kokybės vertinimas naudojant SARI metriką

Vertinant supaprastintą tekstą buvo naudojama SARI metrika, skirta supaprastintų tekstų kokybei įvertinti, juos lyginant su originaliais tekstais. Ši metrika vertina supaprastinimo kokybę pagal tris pagrindines operacijas: žodžių pridėjimą (angl. add), žodžių išlaikymą (angl. keep) ir žodžių pašalinimą (angl. delete). Ji apskaičiuojama kaip trijų operacijų vidurkis:

$$\text{SARI} = d_1 F_{\text{pridėjimas}} + d_2 F_{\text{išlaikymas}} + d_3 P_{\text{pašalinimas}} \tag{1}$$

kur $d_1 = d_2 = d_3 = \frac{1}{3}$, o $F_{\text{pridėjimas}}$, $F_{\text{išlaikymas}}$ ir $P_{\text{pašalinimas}}$ apibrėžti kaip tikslumo ir atkūrimo rodikliai.

Operacijos tikslumas:

$$P_{\text{operacija}} = \frac{1}{k} \sum_{n=1}^{k} p_{\text{operacija}}(n) \tag{2}$$

Operacijos atkūrimas:

$$R_{\text{operacija}} = \frac{1}{k} \sum_{n=1}^{k} r_{\text{operacija}}(n) \tag{3}$$

Operacijos F1 rodiklis:

$$F_{\text{operacija}} = \frac{2 \cdot P_{\text{operacija}} \cdot R_{\text{operacija}}}{P_{\text{operacija}} + R_{\text{operacija}}} \tag{4}$$

Čia:

- operacija $\in \{\text{pridėjimas}, \text{išlaikymas}, \text{pašalinimas}\}$ – operacijos tipas (pridėjimas, išlaikymas arba pašalinimas);
- k – maksimalus n-gramų eilės lygis;
- $p_{\text{operacija}}(n)$ – operacijos n-gramų tikslumas;
- $r_{\text{operacija}}(n)$ – operacijos n-gramų atkūrimas.

Atliekant modelio vertinimą, SARI metrika apskaičiuota naudojantis „evaluate-metric/sari“ iš „HuggingFace“.

2.3.2. Kokybės vertinimas naudojant BERTScore metriką

Vertinant supaprastinto teksto kokybę taip pat buvo naudojama BERTScore metrika. BERTScore vertina semantinį panašumą tarp sugeneruoto ir originalaus (angl. reference) teksto, naudojant kontekstinius įterpinius (angl. contextual embeddings) ir žodžių palyginimą. Ši metrika paremta transformerių kodotojo architektūra, pritaikant savaimio mokymosi mechanizmą.

Kiekvienam sugeneruoto teksto žodžiui ieškomas labiausiai panašus žodis originaliame tekste, pagal kosinuso panašumo skaičiavimą tarp jų vektorių. Tuomet skaičiuojami trys pagrindiniai rodikliai:

- tikslumas (angl. precision) – kiek sugeneruotame tekste yra semantiškai panašių žodžių į žodžius originaliame tekste,
- atkūrimas (angl. recall) – kiek originalaus teksto žodžių rasta sugeneruotame tekste,
- F1 rodiklis – galutinis vertinimo rezultatas, apskaičiuojamas formule $F_{\text{BERT}} = 2 \cdot \frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}$, kur R_{BERT} atkūrimas, o P_{BERT} – tikslumas.

2.3.3. Modelio rezultatų vertinimas naudojant PPL metriką

Optimalių hiperparametrų paieškos metu, modelio kokybei įvertinti buvo naudojama PPL metrika. Ši metrika dažnai taikoma didelių kalbos modelių vertinimui, siekiant sužinoti, kaip gerai modelis numato kitą žodį tekste. Mažesnė PPL reikšmė reiškia, kad modelis yra geriau prisitaikęs atliekamai užduočiai ir generuoja tekstą nuosekliau.

PPL skaičiuojama remiantis tikimybių paskirstymu, kurį modelis priskiria žodžių sekai. Ji apibrėžiama kaip eksponentinė vidutinio žodžio entropijos reikšmė:

$$\text{PPL} = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log p(w_i)\right) \tag{5}$$

kur N – žodžių skaičius tekste, o $p(w_i)$ – modelio priskirta tikimybė žodžiui w_i.

2.3.4. Kokybės vertinimas naudojant didelį kalbos modelį kaip vertintoją

Siekiant įvertinti supaprastinto teksto kokybę ir aiškumą buvo naudojamas OpenAI GPT-4o-mini didelis kalbos modelis. Šis modelis buvo naudojamas kaip papildomas kokybinis vertintojas, siekiant įvertinti eksperimentų metu Lt-Llama-2-7b-hf modelio sugeneruotų tekstų aiškumą, konteksto išlaikymą ir paprastumą. Šis metodas leido įvertinti tekstus pagal konkrečius kriterijus, kuriuos sunku kiekybiškai išreikšti naudojant tik automatines metrikas, tokias kaip SARI, nereikalaujant žmogiškųjų išteklių.

Supaprastintų tekstų vertinimas, naudojant OpenAI GPT-4o-mini modelį, buvo atliktas „Google Colab“ aplinkoje, siunčiant API užklausą į OpenAI serverius. API užklausoje buvo pateikta specialiai suformuluota užklausa (angl. prompt), kurioje buvo nurodyti trys tekstai

- originalus tekstas;
- specialistų, į lengvai suprantamą kalbą, supaprastintas tekstas;
- modelio teksto supaprastinimo į lengvai suprantamą kalbą rezultatas.

Pateikus šiuos tekstus, užklausoje nurodyti trys rodikliai, pagal kuriuos skalėje nuo 0 iki 10 prašoma įvertinti modelio supaprastintą tekstą. Pasirinkti kriterijai buvo šie:

- Aiškumas – įvertinti, kaip lengvai tekstas gali būti suprantamas žmonėms su proto negalia ar ribotais skaitymo įgūdžiais.
- Kontekstas – įvertinti, ar supaprastintas tekstas išlaiko originalaus teksto prasmę ir ar nėra prarastos svarbios detalės.
- Paprastumas – įvertinti, ar tekstas parašytas aiškiais, trumpais sakiniais ir paprastais žodžiais.

Taip pat, užklausos pabaigoje, paprašyta pateikti bendrą supaprastinto teksto įvertinimą. Šiam įvertinimui bus pateikti 5 geriausiai užduotį atliekančio modelio rezultatai, gauti pateikus testavimo aibės tekstus su 4 lentelėje pavaizduotomis užklausomis.

3. Eksperimentai ir rezultatai

3.1. Modelio adaptavimas

Adaptuojant didelį kalbos modelį Lt-Llama-2-7b-hf teksto lietuvių kalba supaprastinimo į lengvai suprantamą kalbą užduočiai atlikti, buvo atliekami eksperimentai, siekiant rasti geriausius modelio adaptavimo hiperparametrus, leisiančius modeliui geriausiai prisitaikyti prie atliekamos užduoties. Pirmiausia buvo ištestuotas neadaptuotas modelis, siekiant įvertinti, ar modelio adaptavimas pagerina jo kokybę atliekant teksto supaprastinimo į ETR užduotį. 6 lentelėje pavaizduoti SARI, BERTScore bei PPL vertinimai EKSP1 eksperimentui. SARI vertinimo reikšmė 55,291 ir BERTScore reikšmė 0,609 parodė, kad modelis gali vidutiniškai supaprastinti tekstą, o PPL reikšmė lygi 4,528 parodė, kad modelis nesunkiai generuoja tekstą, vidutiniškai renkantis kitą žodį iš keturių su puse alternatyvų.

6 lentelė. EKSP1 eksperimento rezultatai

ID	SARI vertinimas	BERTScore vertinimas	PPL vertinimas
1	55,291	0,609	4,528

7 lentelėje pavaizduoti EKSP2 eksperimentui išbandyti hiperparametrai bei modelio rezultatų vertinimas SARI, BERTScore bei PPL metrikomis. Pradiniai hiperparametrai buvo pasirinkti remiantis hiperparametrais, naudotais apmokyti Lt-Llama-2-7b-hf modelį. Pasirinkti parametrai buvo paredaguoti siekiant atsižvelgti į mažesnį duomenų rinkinį modelio adaptavimui ir dėl to galimas grėsmes. Atliekant eksperimentus buvo keičiamas mokymosi žingsnis, renkantis reikšmes nuo 5×10^{-6}, iki 3×10^{-5}. Žemesnis mokymosi žingsnių dydis buvo pasirinktas, nes jis apsaugo modelį nuo persimokymo, bet gali sulėtinti mokymąsį. Pirmiausia pritaikytas 5% žingsnių sušilimo etapas, kad būtų išvengta per staigių atnaujinimų adaptavimo pradžioje, kai modelis dar nėra tinkamai prisitaikęs konkrečios užduoties atlikimui. Pasirinkus didesnį mokymosi žingsnių kiekį buvo pasirinkta padidinti ir apšilimo žingsnių dydį iki 10%, kad sumažėtų staigaus modelio persimokymo rizika.

Atsižvelgiant į nedidelį ETR duomenų rinkinio dydį, modelio adaptavimo metu buvo pasirinktas 4 įrašų duomenų rinkinių dydis. Siekiant simuliuoti didesnį duomenų rinkinį, visiems bandymams buvo pritaikytas 2 žingsnių gradientų kaupimas. Modelis buvo mokomas 10 epochų, naudojant ankstyvo sustabdymo mechanizmą, jei adaptavimo metu nuostolių reikšmės nustotų kisti. Atliekant bandymus, kurių ID = 4 ir 5, buvo užšaldyti pirmieji 8 modelio sluoksniai, kad pagrindinės lietuvių kalbos žinios liktų nepakitusios, o likę sluoksniai būtų pritaikyti ETR kalbos struktūrai. 7 lentelėje galima pamatyti, kad modelio adaptavimui geriausiai pasiteisino bandymas, kurio ID = 4. Nors visų bandymų vertinimo reikšmės buvo panašios, 4–ojo bandymo rezultatų visuma atrodo geriausia. Nors 7-ojo bandymo rezultatai pasiekė dar aukštesnį SARI 55,511 rezultatą, reikšmingai didesnė PPL reikšmė indikuoja, kad modelio adaptavimo metu įvyko persimokymas.

7 lentelė. EKSP2 eksperimento hiperparametrų paieškos rezultatai

ID	Mokymosi žingsnis	Žingsnių sušilimo etapas	Įrašų dydis	Sluoksnių užšaldymas	SARI vertinimas	BERTScore vertinimas	PPL vertinimas
1	1×10^{-5}	0,05	4	-	55,069	0,609	4,44
2	1×10^{-5}	0,1	4	-	55,081	0,606	4,50
3	2×10^{-5}	0,1	4	-	55,172	0,604	5,10
4	$\mathbf{1 \times 10^{-5}}$	**0,05**	**4**	**pirmi 8 sluoksniai**	**55,090**	**0,610**	**4,54**
5	5×10^{-6}	0,05	4	pirmi 8 sluoksniai	54,340	0,612	4,26
7	3×10^{-5}	0,1	6	-	55,511	0,608	10,42

Atlikus EKSP2 eksperimento hiperparametrų paiešką ir nustačius optimaliausius hiparametrus, buvo atliktas EKSP3 eksperimentas, kurio tikslas – atlikti dviejų pakopų modelio adaptavimą, pritaikant modelį lietuvių kalbos tekstų supaprastinimo užduočiai. Pirmoje pakopoje modelis buvo adaptuojamas naudojant bendro pobūdžio, supaprastinta lietuvių kalba parašytus tekstus. Kadangi ši pakopa skirta bendram modelio prisitaikymui lietuviško teksto supaprastinimo užduočiai, modelio sluoksniai buvo nešaldomi, leidžiant visam modeliui adaptuotis prie užduoties.

Pradiniai hiperparametrai buvo parinkti remiantis ankstesnio EKSP2 eksperimento metu gautais rezultatais. Taip pat, buvo atlikti papildomi bandymai siekiant optimizuoti šiuos hiparametrus bendro pobūdžio užduočiai bei modelio adaptavimui su didesniu duomenų rinkiniu. Eksperimentai buvo atliekami keičiant mokymosi žingsnio dydį, mokymosi žingsnio dydžio šildymo etapus ir duomenų rinkinio įrašų dydį. Bandymų, keičiant hiperparametrus modelio adaptavimo metu, rezultatai pateikti 8 lentelėje. Remiantis lentelės duomenimis, geriausius rezultatus pasiekė eksperimentai, kurių ID = 3 ir 4. Pastebėta, kad padidintas įrašų dydis iki 8, turėjo teigiamos įtakos modelio teksto supaprastinimo gebėjimams. Tai indikuoja aukštesnis SARI metrikos rezultatas bei mažesnė PPL reikšmė. Atsižvelgiant į šiuos rezultatus, antros pakopos modelio adaptavimui buvo pasirinkta konfigūracija su didesniu įrašų dydžiu ir mokymosi žingsniu. Šie hiperparametrai pasirinkti siekiant leisti modeliui efektyviau prisitaikyti prie supaprastintų lietuviškų tekstų. Antros pakopos adaptavimui pasirinkta naudoti modelį, gautą atlikus bandymą, kurio ID = 4. Nors šio bandymo SARI metrikos rezultatas šiek tiek mažesnis už eksperimento, kurio ID = 1, o BERTScore rezultatas nėra reikšmingai mažesnis už eksperimento, kurio ID = 1 rezultatą, bendra rezultatų visuma ir mažesnė PPL reikšmė leidžia manyti, kad šio eksperimento metu modelis geriausiai prisitaikė prie teksto paprasta kalba supaprastinimo užduoties.

8 lentelė. EKSP3 pirmosios pakopos adaptavimo hiperparametrų paieškos rezultatai

ID	Mokymosi žingsnis	Žingsnių sušilimo etapas	Epochų skaičius	Įrašų dydis	SARI vertinimas	BERTScore vertinimas	PPL vertinimas
1	1×10^{-5}	0,05	1	4	54,357	0,615	4,27
2	7×10^{-6}	0,02	1	4	54,358	0,612	4,29
3	1×10^{-5}	0,05	1	8	54,376	0,613	4,24
4	$\mathbf{2 \times 10^{-5}}$	**0,05**	**1**	**8**	**54,375**	**0,613**	**4,20**
5	2×10^{-5}	0,05	2	8	55,186	0,607	4,64

Antroje adaptavimo pakopoje modelis buvo toliau pritaikomas lietuvių kalbos supaprastinimo į lengvai suprantamą kalbą užduočiai atlikti. Antrosios pakopos metu buvo siekiama išlaikyti pirmoje pakopoje išmoktas bendras teksto lietuvių kalba supaprastinimo savybes bei leisti modeliui geriau prisitaikyti prie antrojo lygio ETR tekstų. Dėl šių priežasčių buvo pasirinkta šaldyti dalį modelio sluoksnių, kad būtų apsaugoti pagrindiniai lietuvių kalbos supaprastinimo gebėjimai, leidžiant kitiems sluoksniams prisitaikyti prie sudetingesnės užduoties.

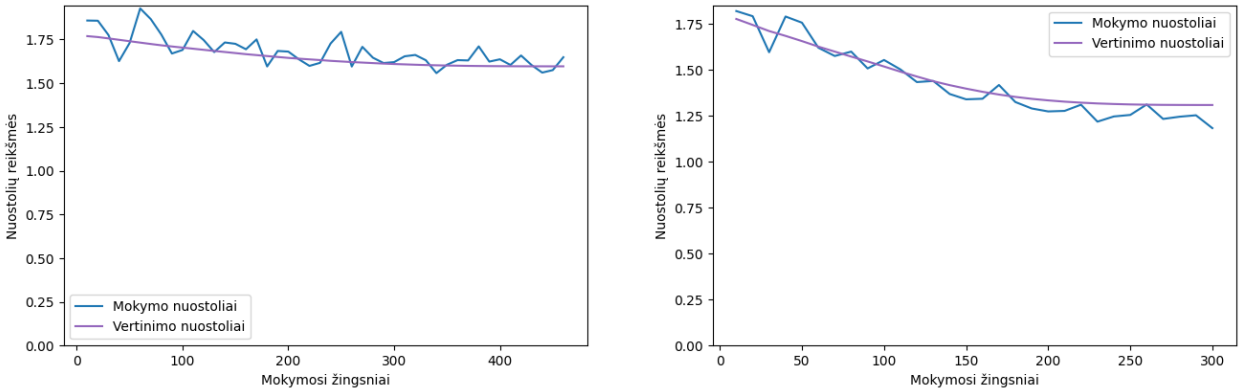
Antros pakopos optimalių hiperparametrų parinkimui taip pat buvo atlikti bandymai, kurių rezultatai pateikti 9 lentelėje. Antroje pakopoje geriausi rezultatai (SARI 55,303, BERTScore 0,613) buvo gauti atliekant eksperimentą, kurio ID = 6. Šiam eksperimentui altikti buvo pasirinktas 2×10^{-5} mokymosi žingsnis, 6 įrašų duomenų rinkinio dydis ir buvo užšaldant pirmi 4 modelio sluoksniai. Nors automatinių metrikų rezultatai nė vienam bandymui nebuvo reikšmingai pagerėję, SARI ir BERTScore metrikų rezultatai buvo didžiausi iš visų atliktų bandymų, taip pat parodymami truputį geresnį modelio gebėjimą atlikti teksto supaprastinimo į ETR užduotį nei EKSP1 modelis.

9 lentelė. EKSP3 antrosios pakopos adaptavimo hiperparametrų paieškos rezultatai

ID	Mokymosi žingsnis	Žingsnių sušilimo etapas	Įrašų dydis	Sluoksnių užšaldymas	SARI vertinimas	BERTScore vertinimas	PPL vertinimas
1	5×10^{-6}	0,05	4	pirmi 8 sluoksniai	54,350	0,612	4,30
2	5×10^{-6}	0,05	8	pirmi 8 sluoksniai	54,352	0,612	4,26
3	1×10^{-5}	0,05	8	pirmi 8 sluoksniai	55,201	0,608	4,45
4	1×10^{-5}	0,05	8	-	55,133	0,608	4,37
5	3×10^{-5}	0,05	8	-	55,163	0,608	4,96

6	$\mathbf{2 \times 10^{-5}}$	**0,05**	**6**	**pirmi 4 sluoksniai**	**55,303**	**0,613**	**5,21**

2 pav. palygintos nuostolių reikšmės modelio adaptavimo EKSP3 eksperimento antrosios pakopos metu.



(a) EKSP3 bandymas, kurio ID = 1

(b) EKSP3 bandymas, kurio ID = 6

2 pav. EKSP3 antrosios pakopos modelio nuostolių kitimo mokymo ir validavimo duomenų aibėms mokymosi eigoje palyginimas tarp eksperimentų

2a pav. pavaizduotos bandymo su pradiniais hiperparametrais, kurio ID = 1, nuostolių reikšmės mokymo ir validavimo duomenų aibėms. 2b pav. pavaizduotos geriausius rezultatus pasiekusio bandymo, kurio ID = 6, nuostolių reikšmės mokymo ir validavimo duomenų aibėms. Kaip galima pamatyti iš paveikslėlių, su pradiniais hiperparametrais modelis mokėsi, nes jo validacijos duomenų aibės nuostolių vertės mažėjo, bet mokymosi duomenų aibės nuostolių reikšmės keitėsi labai netolygiai. Geriausius rezultatus pasiekusio modelio nuostolių reikšmės mokymosi ir validavimo duomenų aibėms mažėjo nuosekliau ir pasiekė žemesnę nuostolių vertę. Taip pat iš 2b pav. galima pastebėti, kad adaptavimo proceso pabaigoje, ties 250 žingsniu validacijos duomenų aibės nuostolių vertė pradėjo mažėti lėčiau, o mokymosi duomenų aibės vertė toliau mažėjo, indikuojant, kad modelis pradėjo persimokyti. Atliekant eksperimentus, buvo naudojama modelio versija prieš įvykstant perisimokymui.

3.2. Užklausų kūrimas

Radus geriausiai teksto lietuvių kalba supaprastinimo į ETR užduočiai prisitaikiusį modelį ir tinkamiausius hiperparametrus jo adaptavimui, buvo pradėtas modelio testavimas pasitelkiant užklausų kūrimą. Testavimui buvo pasirinktas EKSP3 eksperimento modelis, adaptuotas dvejomis pakopomis. Šis modelis 9 lentelėje pažymėtas identifikaciniu numeriu ID = 6. Adaptuotam modeliui kaip įvestys buvo paduodama testavimo duomenų rinkinio aibė su skirtingomis užklausomis, pavaizduotomis 4 lentelėje. Gautas modelio rezultatas įvertintas SARI bei BERTScore kiekybinėmis metrikomis. Siekiant įvertinti modelio supaprastintų tekstų kokybę, modelio rezultatai taip pat

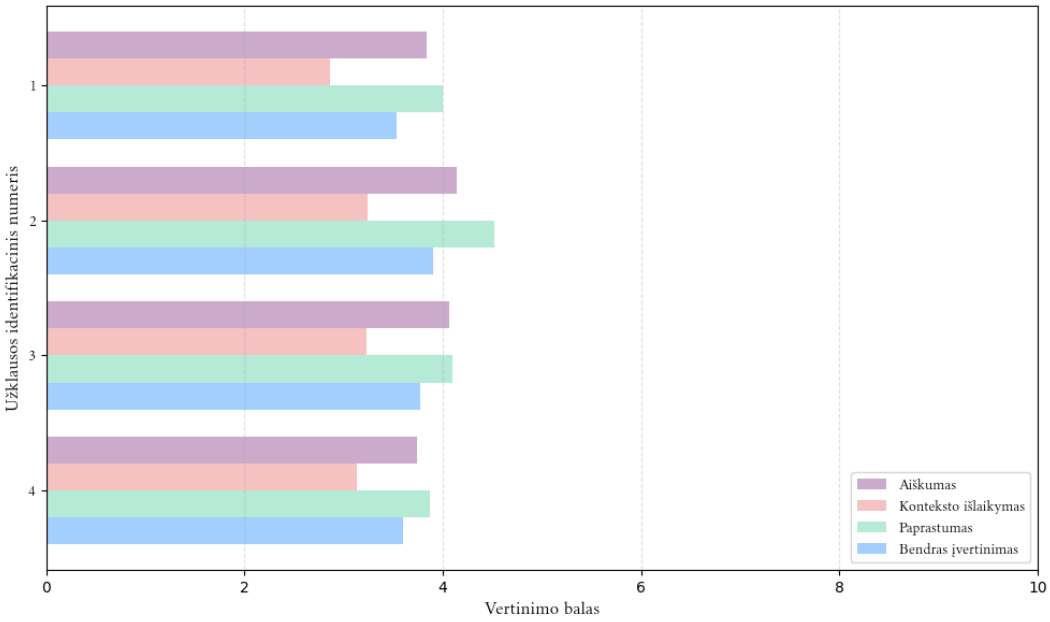
įvertinti didelio kalbos modelio GPT-4o-mini kaip vertintojo. 10 lentelėje pateikiami kiekvienai užklausai gauti rezultatai.

10 lentelė. Užklausų kūrimo rezultatai

Užklausos ID	SARI vertinimas	BERTScore vertinimas	GPT-4o-mini vertinimas
1	54,19	0,646	3,533 / 10
2	51,27	0,652	3,905 / 10
3	49,74	0,642	3,767 / 10
4	49,51	0,643	3,6 / 10

Vertinant užduočiai adaptuoto modelio rezultatus automatizuotomis metrikomis galima pastebėti, kad SARI vertinimo metrikos rezultatai ganėtinai maži. Aukščiausi SARI metrikos rezultatai (54,19) pasiekti naudojant pirmąją užklausą. BERTScore reikšmės vertinant modelio rezultatus buvo kiek didesnės ir svyravo nuo 0,642 iki 0,652. Didžiausia BERTScore = 0,652 reikšmė buvo gauta naudojant užklausą, kurios ID = 2. Kokybinis modelio rezultatų vertinimas pasitelkiant API užklausa kviečiant GPT-4o-mini didelį kalbos modelį kaip vertintoją, atskleidė papildomų įžvalgų. Bendras modelio vertinimas ne visada sutapo su automatinėmis vertinimo metrikomis ir bendrai rezultatus su visomis užklausomis įvertino praščiau nei automatinės metrikos. Aukščiausiu rezultatu GPT-4o-mini įvertino bandymą su užklausa, kurios ID = 2. Eksperimentų su šia užklausa SARI rezultatas nebuvo aukščiausias, palyginus su kitais eksperimentais.

3 pav. pavaizduotas GPT-4o-mini rezultatų, pavaizduotų 10 lentelėje grafikas, vertinant pagal keturis skirtingus kriterijus: supaprastinto teksto aiškumą, konkteksto išlaikymą, teksto paprastumą ir bendrą įvertinimą. Grafike galima pastebėti, kad su visomis užklausomis gauti rezultatai buvo labai panašūs. Pagal skirtingus kriterijus galima matyti, kad modelis eksperimentų su visomis užklausomis rezultatus geriausiai įvertino pagal paprastumą, o praščiausiai – pagal konteksto išlaikymą.



3 pav. Supaprastintų tekstų GPT-4o-mini modelio vertinimai pagal vertinimo kriterijus

10 lentelėje ir 3 pav. pateikti rezultatai parodo, kad sudėtingų užklausų kūrimas ne visada gali padėti modelio rezultatų pagerinimui ir gali net pabloginti modelio rezultatus.

Rezultatai

Šiame darbe buvo atlikta literatūrinė analizė apie teksto lietuvių kalba supaprastinimą į lengvai suprantamą kalbą pasitelkiant didelius kalbos modelius. Taip pat buvo atlikti eksperimentai su duomenimis lietuvių kalba apmokytu dideliu kalbos modeliu Lt–Llama–2, siekiant pritaikyti modelį lietuviškų tekstų supaprastinimo į ETR užduočiai atlikti.

Pirmiausia darbe buvo išanalizuotos šios temos:

- Lengvai suprantama kalba ir dirbtinio intelekto taikymas žmonėms, turintiems negalią.
- Natūralios kalbos technologijos ir jų pritaikymas teksto generavimo bei supaprastinimo užduotims.
- Didelių kalbos modelių adaptavimo procesas, efektyvumo užtikrinimas pritaikant skirtingus adaptavimo metodus.
- Kiekybiniss ir kokybinis didelių kalbos modelių sugeneruotų tekstų vertinimas.

Atlikus literatūrinę analizę buvo atlikti šie darbai:

- Duomenų rinkinių paruošimas didelio kalbos modelio Lt–Llama–2 adaptavimo procesui. Buvo surinkti ir sutvarkyti duomenys dviejų pakopų adaptavimo procesui. Duomenys pirmąjai pakopai buvo gauti iš ankščiau atliktų tyrimų su supaprastinta lietuvių kalba, o ETR duomenų rinkinys buvo parengtas iš ekspertų pateiktų tekstų lengvai suprantama kalba.
- Optimalių hiperparametrų modelio adaptavimui paieška, vykdant skirtingus eksperimentus adaptuojant Lt–Llama–2 modelį.
- Užklausų parengimas ir tekstų supaprastinimo pateikiant modeliui skirtingas užklausas eksperimentų atlikimas.

Atlikus eksperimentus ir radus geriausiai teksto lietuvių kalba supaprastinimo į ETR užduočiai prisitaikiusį modelį, buvo atliktas kiekybinis ir kokybinis modelio vertinimas:

- Modelio kiekybiniai rezultatai buvo įvertinti naudojant SARI ir BERTScore metrikas.
- Modelio kokybiniai rezultatai buvo įvertinti siunčiant API užklausą į OpenAi GPT-4o-mini LLM, taip pasitelkiant dideliu kalbos modeliu kaip vertintoju.

Išvados

1. Teksto supaprastinimas į lengvai suprantamą kalbą yra svarbi tema socialinės įtraukties srityje. Dideli kalbos modeliai gali būti efektyviai pritaikyti teksto supaprastinimo užduočiai atlikti, tačiau daugeliu atvejų, kaip ir kitose natūralios kalbos užduotyse, jų rezultatai priklauso nuo tinkamo modelio adaptavimo proceso bei tinkamų hiperparametrų pasirinkimo.

2. Kelių pakopų modelio adaptavimas gali pagerinti modelio gebėjimą prisitaikyti prie naujos užduoties. Taip pat svarbu parinkti optimalius mokymo hiperparametrus. Tai ypač gali padėti turint nedidelius duomenų rinkinius ar pritaikant modelį sudėtingai užduočiai, prie kurios modelis sunkiai prisitaiko.

3. Gerai suformuluotos užklausos gali padėti modeliui pasiekti geresnių rezultatų be papildomo adaptavimo. Tačiau, per sudėtingos užklausos gali rezultatus reikšmingai pabloginti.

4. Dideli kalbos modeliai gali būti efektyviai naudojami kaip kokybiniai vertintojai.

Šis tyrimas gali būti tęsiamas, siekiant gauti geresnius rezultatus supaprastinant lietuvišką tekstą į lengvai suprantamą kalbą. Šis tyrimas turi keletą apribojimų:

- Mažus duomenų rinkinius, prie kurių modelis gali greitai prisitaikyti ir persimokyti.
- Eksperimentai atlikti turint ribotus išteklius.

Tęsiant tyrimą vertėtų paruošti didesnius ir daugiau skirtingų temų apimančius duomenų rinkinius bei atlikti modelio adaptavimo procesą aplinkoje, turinčioje daugiau išteklių.

Kadangi adaptuoto modelio testavimas pasitelkiant užklausų kūrimą parodė, kad užklausos gali pabloginti modelio rezultatus, ateities tyrimuose galima bandyti pritaikyti Lt-Llama-2 modelį geresniam užklausų supratimui naudojant nurodymų derinimą (angl. instruction tuning). [CHL+22] autoriai pabrėžia, kad nurodymų derinimas gali reikšmingai pagerinti su daug duomenų apmokytų didelių kalbos modelių našumą bei rezultatų kokybę.

Lyginant šio darbo rezultatus su panašiais anksčiau atliktais darbais, nors tekstų supaprastinimas lietuvių kalba pasitelkiant LLM nėra plačiai ištirtas, tyrime [MRK+25] atliekamas LLM adaptavimas administracinių tekstų supaprastinimo į paprastą kalbą užduočiai atlikti. Tyrimo metu buvo adaptuojami transformerių architektūra grįsti modeliai T5, mBART ir Lt-Llama-2. Tyrimo rezultatai parodė, kad dažnai modeliai sakinius ne supaprastina, o praplėčia, paaiškina. Nors T5 ir mBART modeliai parodė geresnius rezultatus, kurių SARI rezultatai svyravo nuo 54,12 iki 72,98, užduočiai adaptuotas Lt-Llama-2 modelis nepasiekė reikšmingų rezultatų. Atsižvelgiant į tai, tolimesniuose tyrimuose būtų galima išbandyti tekstų lietuvių kalba supaprastinimo į ETR užduočiai pritaikyti kitus LLM.

Panašūs tyrimai buvo atlikti ir kitomis kalbomis. [MRM24] buvo tiriamas ispaniškų tekstų supaprastinimas į ETR pasitelkiant Llama-2 modelį. Tyrime aprašytas metodas paremtas Llama-2 modelio adaptavimu siekiant modelį pritaikyti tekstų supaprastinimo į ETR kalbą užduočiia atlikti.

Tyrimo metu buvo išbandyta išversti ispaniškus tekstus į anglų kalbą, juos supaprastinti ir išversti atgal į ispanų kalbą. Rezultatai parodė, kad supaprastinti tekstai buvo lengviau suprantami, o kokybiniai rezultatai patvirtino modelio gebėjimą supaprastinti turinį išsaugant jo prasmę.

Nors šiame darbe daugiausia dėmesio skiriama lietuvių kalbai, sėkmingas Llama-2 taikymas teksto supaprastinimui ispanų kalba parodo modelio lankstumą. Nors duomenų rinkiniai ir kalbos skiriasi, gauti rezultatai leidžia manyti, kad, tinkamai apmokius ir adaptavus Llama-2 modelį, jis galėtų būti pritaikytas lietuviško teksto supaprastinimo užduočiai atlikti.

Kitame tyrime [BMM25] buvo tiriamas estų kalbos teksto supaprastinimas naudojant LLM. Šis tyrimas aktualus, nes estų, kaip ir lietuvių kalba, turi mažiau pritaikytų LLM įrankių bei turi ribotus supaprastintos kalbos tekstų išteklius. Tyrimo metu Llama modelis buvo adaptuojamas naudojant duomenų rinkinį, kuris buvo sudarytas iš išverstų supaprastintų tekstų bei GPT-4 modelio supaprastintų tekstų. Llama modelio rezultatai buvo lyginami su kitų modelių, pavyzdžiui, DRESS, OpenNMT ir T5 rezultatais. Llama 3.1 ir OpenNMT modelių palyginimas parodė, kad nors OpenNMT pasiekė šiek tiek aukštesnį BLEU balą (30,05 ir 27,04), Llama 3.1 pranoko OpenNMT pagal SARI metriką (49,72 ir 47,43). Tai parodo, kad tekstai buvo supaprastnti kokybiškiau. Be to, Llama 3.1 turėjo šiek tiek mažesnį FKGL rezultatą (8,71 prieš 9,02), o tai parodo, kad Llama 3.1 modelio supaprastinti tekstai buvo šiek tiek lengviau suprantami. Taip pat supaprastinti tekstai buvo įvertinti trijų žmonių, kuriems estų kalba yra gimtoji. Ekspertinio vertinimo metu Llama 3.1 modelio supaprastinti tekstai buvo įvertinti reikšmingai geriau (3,03 ir 1,6 balais, vertinant 4 balų skalėje), taip parodant geresnius modelio gebėjimus supaprastinti tekstą į ETR.

Literatūra

[22] *Global Report on Health Equity for Persons with Disabilities*. en. World Health Organization, Geneva, 1st ed leid., 2022. ISBN: 978-92-4-006360-0.

[24] Disability and Development Report 2024. en, 2024.

[BDV] Justina Bružaitė-Liseckienė, Inga Daraškienė ir Laura Vilkaitė-Lozdienė. Teksto lengvai suprantama kalba rengimo gairės. lt.

[BMM25] Eduard Barbu, Meeri-Ly Muru ir Sten Marcus Malva. Improving Estonian Text Simplification through Pretrained Language Models and Custom Datasets. en, 2025-01. DOI: 10.48550/arXiv.2501.15624. URL: http://arxiv.org/abs/2501.15624 (tikrinta 2025-03-09). Citation-key: simplifying-estonian-text arXiv:2501.15624 [cs].

[BMR+20] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah ir k.t. Language Models are Few-Shot Learners. en, 2020-07. DOI: 10.48550/arXiv.2005.14165. URL: http://arxiv.org/abs/2005.14165 (tikrinta 2025-01-01). arXiv:2005.14165 [cs].

[CHL+22] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph ir k.t. Scaling Instruction-Finetuned Language Models. en, 2022-12. DOI: 10.48550/arXiv.2210.11416. URL: http://arxiv.org/abs/2210.11416 (tikrinta 2025-05-21). arXiv:2210.11416 [cs].

[DCL+19] Jacob Devlin, Ming-Wei Chang, Kenton Lee ir Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. en, 2019-05. DOI: 10.48550/arXiv.1810.04805. URL: http://arxiv.org/abs/1810.04805 (tikrinta 2025-01-01). arXiv:1810.04805 [cs].

[DPH+] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman ir Luke Zettlemoyer. QLORA: Efficient Finetuning of Quantized LLMs. en. Citation-key: QLORA.

[GMS+20] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey ir Noah A. Smith. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. en, 2020-05. DOI: 10.48550/arXiv.2004.10964. URL: http://arxiv.org/abs/2004.10964 (tikrinta 2025-05-04). arXiv:2004.10964 [cs].

[HR18] Jeremy Howard ir Sebastian Ruder. Universal Language Model Fine-tuning for Text Classification. en, 2018-05. DOI: 10.48550/arXiv.1801.06146. URL: http://arxiv.org/abs/1801.06146 (tikrinta 2025-05-04). arXiv:1801.06146 [cs].

[KGR+23] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo ir Yusuke Iwasawa. Large Language Models are Zero-Shot Reasoners. en, 2023-01. DOI: 10.48550/arXiv.2205.11916. URL: http://arxiv.org/abs/2205.11916 (tikrinta 2025-03-09). arXiv:2205.11916 [cs].

[KKK+23] Diksha Khurana, Aditya Koli, Kiran Khatter ir Sukhdev Singh. Natural language processing: state of the art, current trends and challenges. en. *Multimedia Tools and Applications*, 82(3):3713–3744, 2023-01. ISSN: 1380-7501, 1573-7721. DOI: 10.1007/s11042-022-13428-4. URL: https://link.springer.com/10.1007/s11042-022-13428-4 (tikrinta 2024-12-30).

[LLA22] Ivano Lauriola, Alberto Lavelli ir Fabio Aiolli. An introduction to Deep Learning in Natural Language Processing: Models, techniques, and tools. en. *Neurocomputing*, 470:443–456, 2022-01. ISSN: 09252312. DOI: 10.1016/j.neucom.2021.05.103. URL: https://linkinghub.elsevier.com/retrieve/pii/S0925231221010997 (tikrinta 2024-12-30).

[MP14] Klaus Miesenberger ir Andrea Petz. Easy to Read on the Web – State of the Art and Research Directions. *Procedia Computer Science*. 5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, DSAI 2013, 27:318–326, 2014-01. ISSN: 1877-0509. DOI: 10.1016/j.procs.2014.02.035. URL: https://www.sciencedirect.com/science/article/pii/S1877050914000374 (tikrinta 2024-12-22).

[MRK+25] Justina Mandravickaitė, Eglė Rimkienė, Danguolė Kotryna Kapkan, Danguolė Kalinauskaitė, Antanas Čenys ir Tomas Krilavičius. Automatic Text Simplification for Lithuanian: Transforming Administrative Texts into Plain Language. en. *Mathematics*, 13(3):465, 2025-01. ISSN: 2227-7390. DOI: 10.3390/math13030465. URL: https://www.mdpi.com/2227-7390/13/3/465 (tikrinta 2025-03-09). Citation-key: text-simplification-in-Lithuanian.

[MRM24] Paloma Martínez, Alberto Ramos ir Lourdes Moreno. Exploring Large Language Models to generate Easy to Read content. en. *Frontiers in Computer Science*, 6:1394705, 2024-10. ISSN: 2624-9898. DOI: 10.3389/fcomp.2024.1394705. URL: https://www.frontiersin.org/articles/10.3389/fcomp.2024.1394705/full (tikrinta 2025-03-09). Citation-key: simplifying-in-spanish.

[NDM24] Artūras Nakvosas, Povilas Daniušis ir Vytas Mulevičius. Open Llama2 Model for the Lithuanian Language, 2024-08. DOI: 10.48550/arXiv.2408.12963. URL: http://arxiv.org/abs/2408.12963 (tikrinta 2025-01-12). arXiv:2408.12963 [cs].

[RSR+23] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li ir Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. en, 2023-09. DOI: 10.48550/arXiv.1910.10683. URL: http://arxiv.org/abs/1910.10683 (tikrinta 2025-01-01). arXiv:1910.10683 [cs].

[SIB+25] Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze ir k.t. The Prompt Report: A Systematic Survey of Prompt Engineering Techniques. en, 2025-02. DOI: 10.48550/arXiv.2406.06608. URL: http://arxiv.org/abs/2406.06608 (tikrinta 2025-05-04). arXiv:2406.06608 [cs].

[SKY+18] Samuel L. Smith, Pieter-Jan Kindermans, Chris Ying ir Quoc V. Le. Don't Decay the Learning Rate, Increase the Batch Size, 2018-02. DOI: 10.48550/arXiv.1711.00489. URL: http://arxiv.org/abs/1711.00489 (tikrinta 2025-01-05). arXiv:1711.00489 [cs].

[TMS+23] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert ir k.t. Llama 2: Open Foundation and Fine-Tuned Chat Models. en, 2023-07. DOI: 10.48550/arXiv.2307.09288. URL: http://arxiv.org/abs/2307.09288 (tikrinta 2025-01-01). arXiv:2307.09288 [cs].

[VSP+] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser ir Illia Polosukhin. Attention is All you Need. en.

[XNP+16] Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen ir Chris Callison-Burch. Optimizing Statistical Machine Translation for Text Simplification. en. *Transactions of the Association for Computational Linguistics*, 4:401–415, 2016-12. ISSN: 2307-387X. DOI: 10.1162/tacl_a_00107. URL: https://direct.mit.edu/tacl/article/43364 (tikrinta 2025-01-04).

[ZKW+20] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger ir Yoav Artzi. BERTScore: Evaluating Text Generation with BERT. en, 2020-02. DOI: 10.48550/arXiv.1904.09675. URL: http://arxiv.org/abs/1904.09675 (tikrinta 2025-05-10). arXiv:1904.09675 [cs].