

TARTU ÜLIKOOL
Arvutiteaduse instituut
Andmeteaduse õppekava

Gervin Ilus
Eestikeelsete tekstülesannete genereerija ja
lahendaja
Magistritöö (15 EAP)

Juhendaja: Sven Aller, MSc

Tartu 2024

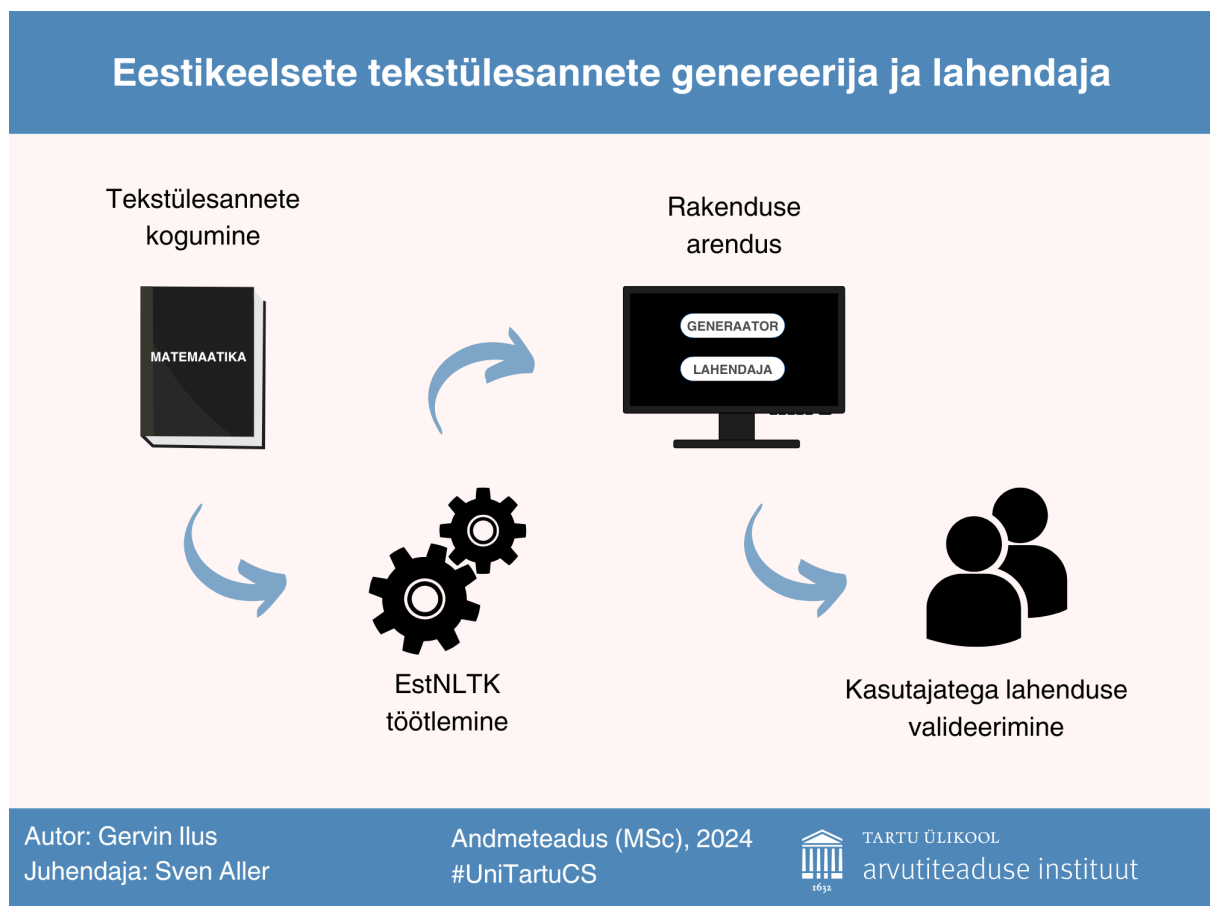
Eestikeelsete tekstülesannete genereerija ja lahendaja

Lühikokkuvõte: Magistritöö eesmärk on eestikeelse tekstipõhiste matemaatiliste ülesannete genereerija ja lahendaja arendamine, mis kasutaks töötluks EstNLTK teekide kogumit. Näidisprogrammi eesmärk on pakkuda toetust nii õppeprotsessile kui ka näidata eestikeelse loomuliku keele töötlu (ingl *NLP*) võimekust tekstülesannete vallas. Rakendus võimaldab kasutajatel luua ja lahendada matemaatilisi tekstülesandeid, pakudes lisaks vastusele ka detailset lahenduskäiku. Töö hõlmab ka kasutajatega rakenduse testimist ja tulemuste analüüsi, et saada ülevaadet loodud süsteemi tugevustest ja kitsaskohtadest. Töös analüüsitakse ka varasemaid sarnaseid lahendusi, et mõista olemasolevate süsteemide võimalusi ja piiranguid.

Võtmesõnad: keeletehnoloogia, tekstülesanded, EstNLTK

CERCS: P175 Informaatika, süsteemiteooria

Visuaalne kokkuvõte:



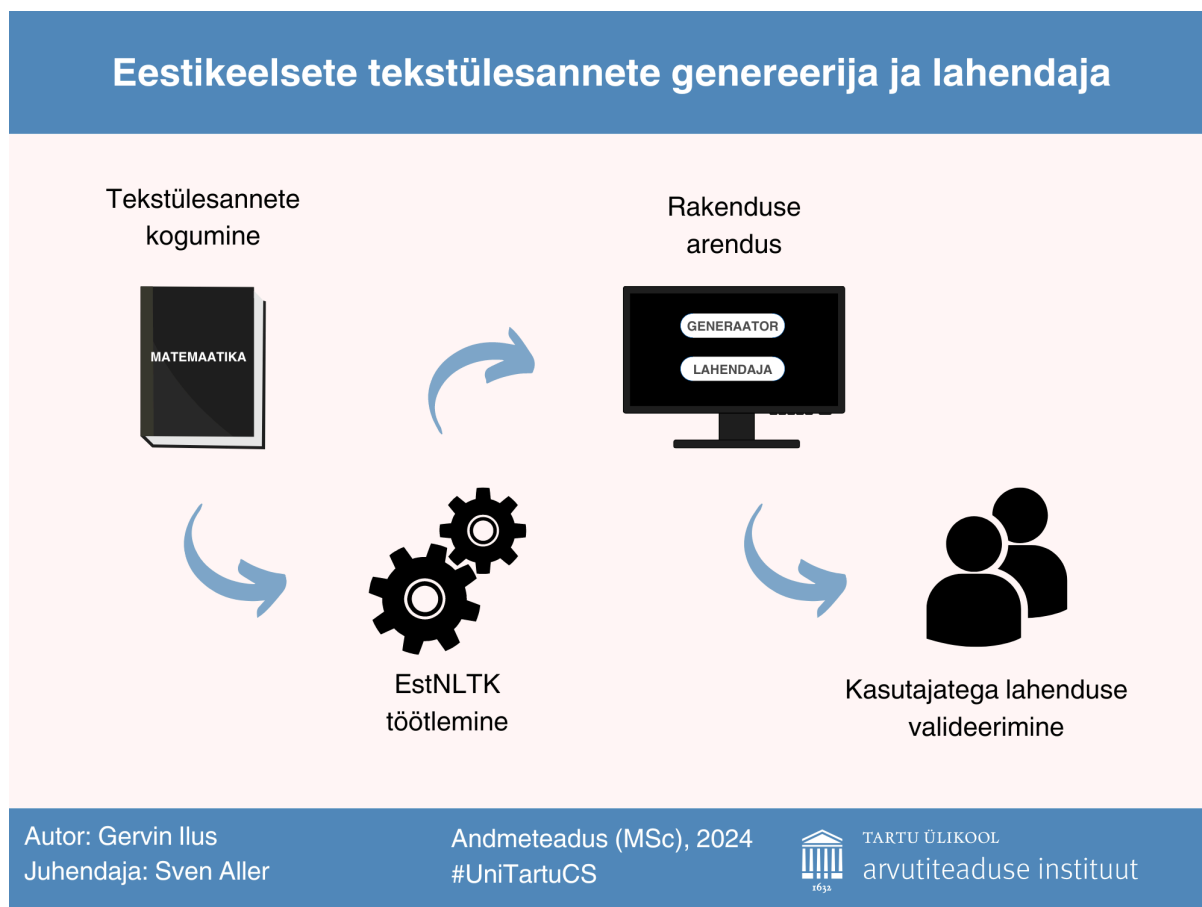
Estonian Text Problem Generator and Solver

Abstract: The objective of this master's thesis is to develop a generator and solver for Estonian text-based mathematical problems, utilizing the Estonian Natural Language Toolkit (EstNLTK) toolkit for processing. The application aims to support both the educational process and show capabilities of Estonian natural language processing in the education domain. The application allows users to create and solve mathematical text problems, providing not only the solutions but also a detailed step-by-step explanation. The project also includes user testing of the application and an analysis of the results to gain insight into the strengths and limitations of the created system. Additionally, the thesis analyzes previous similar solutions to understand the capabilities and limitations of currently existing systems.

Keywords: language technology, text problems, EstNLTK

CERCS: P175 Informatics, systems theory

Visual abstract:



Sisukord

Sissejuhatus	6
1. Teoreetiline taust	7
1.1 Matemaatikaõpe ja tekstülesanded	7
1.1.1 Tekstülesannete ülevaade	7
1.1.2 Tekstülesannete roll hariduses	8
1.1.3 Probleemid tekstülesannete lahendamisel	9
1.1.4 Kooliastmete vahelised erinevused	10
1.2 Loomuliku keele töötlus	10
1.3. EstNLTK	11
2. Olemasolevad lahendused	13
2.1. Tekstülesannete generaator	13
2.2. Tekster	14
2.3. Tekstülesannete automaatne lahendaja	15
2.4. NutiSport	16
2.5. Kokkuvõte praegustest lahendustest	17
3. Programmi ülevaade	19
3.1. Ülesehitus	19
3.2. Arenduse tööriistad ja tehnoloogiad	20
3.3 Nõuded rakendusele	21
3.3.1 Funktsionaalsed nõuded	21
3.3.2 Mittefunktsionaalsed nõuded	21
4. Valminud lahendus	22
4.1. Funktsionaalsuse kirjeldus	23
4.1.1. Generaator	23
4.1.2. Lahendaja	25
4.1.2.1 Parser	26
4.1.2.2. Liitmisülesanne	28
4.1.2.3. Lahutamisülesanne	29
4.1.2.4. Äraandmisülesanne	31
4.1.2.5. Korrutamine ja jagamine	31
4.1.3. Ajalugu	32
5. Valminud lahenduse analüüs	33
5.1. Rakenduse testimine	33
5.1.1. Testimise metoodika	33
5.1.2. Testimise tulemused	33
5.2. Tulemuste kokkuvõte	34
6. Võimalikud edasiarendused	35
6.1 Süvaõpe	35
6.2 Ülesannete lisamine	35

6.3 Keerukuse tõstmine	35
6.4 Kasutajaliidese täiustamine	35
Kokkuvõte	37
Kasutatud kirjandus	38
Lisad	40
1. Rakenduse lähtekood	40
2. Litsents	41

Sissejuhatus

Matemaatika on oluline õppeaine kogu koolitee vältel alates 1. klassist kuni 12. klassini. Erilist tähelepanu pööratakse tekstülesannetele esimeses ja teises kooliastmes (1.-6. klass), kus need ülesanded aitavad arendada õpilaste loogilist mõtlemist ja probleemi lahendamise oskust [1]. Traditsioonilised õpikutes leiduvad tekstülesanded on liiga ühesugused, mistõttu õpetajad peavad koostama ise klassile vastavaid ülesandeid, mis on õpetaja jaoks aga aeganõudev lisatöö [2].

Viimase aastakümne Eesti keeletehnoloogia ja täpsemalt EstNLTK (ingl *Estonian Natural Language Toolkit*) areng on avanud uusi võimalusi haridussektorile, eriti loomuliku keele töötamise (ingl *Natural Language Processing* ehk *NLP*) rakendamisel õppesisu kohandamisel ja õppeprotsessi interaktiivsemaks muutmisel [3]. Loomuliku keele töötamise rakendamine haridustehnoloogias võimaldab muuta õppetööd isikupärasemaks ja tõhusamaks, automatiseerides ja personaliseerides erinevaid õppeülesandeid, sealhulgas matemaatiliste tekstülesannete genereerimist ja lahendamist.

Eestikeelsete tekstülesannete jaoks on seni olnud piiratud arv tööriistu, mistõttu selle magistritöö fookuses on esimesele ja teisele kooliastmele mõeldud matemaatiliste tekstülesannete genereerija ja lahendaja arendamine, kasutades selleks EstNLTK teekide kogumit. Projekti eesmärk on demonstreerida keeletehnoloogia potentsiaali matemaatikaõppe tõhustamisel, pakkudes õpetajatele, õpilastele ja vanematele tööriista, mis automatiseerib tekstülesannete loomise ja lahendamise ning esitab ka detailse lahenduskäigu.

Lõputöö koosneb kuuest peatükist. Esimeses etapis keskendutakse tekstülesannete ja keeletehnoloogia tausta avamisele. Teises peatükis antakse ülevaade olemasolevatest lahendustest ja analüüsitakse nende tugevusi ja puudujääke. Järgneb praktiline osa, kus kirjeldatakse veebiprogrammi loomist arendusprotsessi arhitektuurist kasutajaliideseni. Neljandas peatükis esitletakse valminud rakendust, selgitakse selle funktsionaalsusi ja tehnilisi aspekte. Seejärel viiakse läbi rakenduse testimine ja analüüs, et hinnata süsteemi efektiivsust ja kasutajasõbralikkust. Lõputöö lõpeb võimalike edasiarenduste väljatoomisega.

1. Teoreetiline taust

1.1 Matemaatikaõpe ja tekstülesanded

Matemaatikaõpe on oluline hariduse komponent, mis keskendub mõistete ja seoste tundmisele, probleemide lahendamisele ning loogilisele arutlemisele [4]. Algkoolist kuni õpingute lõpuni on matemaatika keskne aine, mis peaks julgustama õpilasi aktiivselt uusi teadmisi omandama probleemide lahendamise ja seoste loomise kaudu. Kriitiliseks osaks selles protsessis on matemaatika tekstülesanded, mis nõuavad õpilastelt faktide eristamist ning oluliste seoste märkamist ja rakendamist.

1.1.1 Tekstülesannete ülevaade

Tekstülesanne on kirjalik ülesanne, mis esitab jutustava situatsiooni koos konkreetsete numbriliste andmetega ja esitab ühe või mitu küsimust, mille lahendus nõuab arvutuslikku tegevust. Sellise ülesande eesmärk on leida seos esitatud andmete (tingimuste) ja küsimuses oleva otsitava suuruse vahel ning moodustada nende põhjal matemaatiline võrdus või valem. Seejärel kasutatakse sobivaid aritmeetilisi operatsioone, et leida otsitav suurus, mille abil formuleeritakse täpne vastus küsimusele [5].

Tekstülesande võib jagada kaheks põhiosaks: andmed ja küsimus(ed). Lahendusprotsess hõlmab andmetest oluliste seoste märkamist ning nende põhjal õigete aritmeetiliste toimingute valimist ja teostamist. Kogu protsessi lõpptulemuseks on arvuline vastus, mis lahendab algselt esitatud küsimuse.

Matemaatika tekstülesanded on mitmeti liigitatavad. Levinumad liigitused on järgnevad [4]:

1. Liht- ja liitülesanded:

- **Lihtülesanded** nõuavad lahendamiseks üht tehet ja neil on selge ja otsene seos andmete ning küsimuste vahel.
- **Liitülesanded** nõuavad lahendamiseks kahte või enamat tehet ning nende ülesannete puhul on sageli vajalik leida omavahelised esma- ja teisejärgulised seosed, mis muudavad need keerukamaks.

2. Konkreetsed ja abstraktsed tekstülesanded:

- **Konkreetsed** ülesanded käsitlevad reaalelus esinevaid hulkade või suuruste arvulisi väärtusi ja nende vahelisi seoseid.

- **Abstraktsed** ülesanded keskenduvad arvuliste väärtuste omavahelistele seostele, pakkumata seoseid reaalsusega.

3. Avatud ja suletud tekstülesanded:

- **Suletud ülesannetel** on määratletud selgelt alg- ja lõpitingimused ning on need tavaliselt suunatud spetsiifilise vastuse leidmisele.
- **Avatud ülesanded** võimaldavad alg- ja lõpitingimuste osas paindlikkust, andes võimaluse lahenduse leidmisel lähtuda mitmest erinevast vaatenurgast ja tõlgendusest, mis võib viia mitme sobiva lahenduseni.

Tekstülesanded on liigitatavad nende kasutatud matemaatiliste tehete järgi (liitmine, lahutamine, korrutamine, jagamine). Ühetehtelistes tekstülesannetes kasutatakse ühte neljast põhilisest aritmeetika operatsioonist: liitmist, lahutamist, korrutamist või jagamist.

Olenevalt teksti sisust võib olla tegemist kuue võimalusega [6]:

- 1) ühendi leidmine liitmise teel, nt *Jukul on 2 õuna ja 3 pirni. Mitu puuvilja on Jukul kokku?*
- 2) ühendi leidmine korrutamise teel, nt *Jukul on 2 korvi õunu ja igas korvis on 5 õuna . Mitu õuna on Jukul kokku?*
- 3) osa leidmine lahutamise teel, nt. *Jukul oli 5 õuna, aga ta sõi 2 ära. Mitu õuna on Jukul alles?*
- 4) jaotamine või mahutamine jagamise teel, *Jukul on 6 õuna. Ta jagab need võrdselt kolmele sõbrale. Mitu pirni saab iga sõber?*
- 5) võrdlemine kasutades liitmist või lahutamist, nt *Jukul on 5 õuna ja Maril on 3 õuna. Peale selle leidis Mari veel 2 õuna. Kummal on rohkem õunu?*
- 6) võrdlemine kasutades korrutamist või jagamist, nt *Jukul on 5 õuna ja Maril on õunu 3 korda rohkem. Kummal on rohkem õunu?*

1.1.2 Tekstülesannete roll hariduses

Tekstülesannete roll hariduses on mitmetahuline ja oluline, pakkudes õpilastele olulisi oskusi, mis on vajalikud nii akadeemilises kui ka igapäevaelus. Eesti põhikooli riiklikus õppekavas rõhutatakse tekstülesannete lahendamise skeemi oskust, mis on alustalaks kriitilisele mõtlemisele ja loogilisele analüüsile [7]. Tekstülesanded õpetavad õpilasi mõistma ja tõlgendama keerulist informatsiooni, arendades samal ajal nende võimet teha matemaatilisi järeldusi ja rakendada teoreetilisi teadmisi praktilistes situatsioonides. Need oskused on

hädavajalikud ühiskonnas toimimiseks, kus infotehnoloogia ja digipädevus mängivad üha suuremat rolli. Läbi tekstülesannete lahendamise saavad õpilased paremini aru numbriliste andmete tähendusest pärismaailma kontekstis, õppides samas digitaalseid vahendeid kasutama ja tõhusamalt uut infot omastama. Tänu oma praktilisusele ja interdistsiplinaarsusele on tekstülesanded hariduse keskmes, aidates kaasa õpilaste arengule ja valmistades neid ette tulevikuks.

1.1.3 Probleemid tekstülesannete lahendamisel

Matemaatika tekstülesanded on õppeprotsessis vältimatud, kuid nende lahendamine esitab õpilastele mitmeid väljakutseid. Maarit Meola 2023. aasta lõputöös käsitletavat probleemi annavad hea ülevaate sellest, miks tekstülesanded võivad olla õpilastele keerulised ja millised on potentsiaalsed lahendused [2].

Üks peamisi tõstatatud probleeme on õpetajate mure standardsete õpikute tekstülesannete ühetaolisuse ja eluvõõruse pärast [6]. See võib oluliselt mõjutada õpilaste motivatsiooni ja huvi nende ülesannete lahendamise vastu. Õpetajad on märkinud, et elulähedasemad ja rohkem igapäevaeluga seotud ülesanded oleksid kaasahaaravamad ning aitaksid paremini õpilastel mõista matemaatika olulisust ja rakendusi reaalses maailmas [4]. Sellise probleemi leevendamiseks soovitakse, et õpetajad ise looksid ülesandeid. Siiski toob see kaasa lisakoormuse õpetajatele, mis omakorda nõuab aega ja ressursse.

Teine suur takistus on õpilaste kalduvus teksti mitte põhjalikult lugeda, vaid kohe numbritega katsetama hakata. See viib tihti valede lahendusteni, mis näitab, et olulisemaks probleemiks on lugemisoskuse ja teksti mõistmise puudulikkus. Paljud õpilased võivad olla võimelised lugema sõnu, kuid neil on raskusi tekstis esitatud probleemi mõistmisega. Lisaks kardetakse hääbisse jäämist, mistõttu ollakse pigem valmis pakkuma kiiret, kuid ebakorrektset lahendust, kui tunnistama probleemi mõistmise raskust [8].

Kolmandaks probleemiks on tekstides olev segav lisainformatsioon. Mõned õpetajad näevad selles aga hoopis võimalust õpetada õpilastele olulise info välja sõelumist müra. Sellised ülesanded toovad aga sageli kaasa segaduse, eriti neile õpilastele, kes alles arendavad oma lugemis- ja analüüsioskust.

Vormistuspõuete erinevused koolide ja õpetajate vahel võivad samuti põhjustada segadust ja raskusi tekstülesannete standardiseeritud lahenduskäigus. Üks õpetaja tegi ettepaneku, et

matemaatika tekstülesannete vormistus võiks olla riiklikult standardiseeritud, aidates õpilastel kohaneda näiteks kooli vahetades.

1.1.4 Kooliastmete vahelised erinevused

Erinevatel kooliastmetel on matemaatiliste tekstülesannete koostamisel ja lahendamisel olulisi erinevusi. Alates esimesest klassist, kus õpilased tutvuvad põhiliste aritmeetiliste toimingutega, kuni kuuenda klassini, kus matemaatilised probleemid muutuvad märkimisväärselt keerukamaks, on iga klassi jaoks oluline mõista nende arengulisi vajadusi ja võimekust. See analüüs aitab meil ülesannete genereerimisel piiritleda iga klassi tekstülesannete taseme, täpsemad piiritletud peatükis 4.1.1.

I kooliaste (1.-3. klass) katab Eesti haridussüsteemis esialgsed matemaatilised põhioskused ja selle raames on tekstülesanded suunatud põhiliste aritmeetiliste tehteid õpetamisele ning arvutamise harjutamisele. Riikliku õppekava kohaselt peab õpilane I kooliaste lõpuks oskama lahendada lihtsaid ühe- ja kahetehtelisi tekstülesandeid, mis hõlmavad peamiselt liitmist, lahutamist ning lihtsamat korrutamist ja jagamist [9]. Õpilased arendavad oskust tekstist matemaatilist olulisi andmeid eristada ja neid arvutustes kasutada. Tekstülesanded siduvad matemaatikat igapäevaelu olukordadega: näiteks võib ülesanne kirjeldada poeskäiku, kus tuleb arvutada raha kasutamist, või olukordi, kus on vaja mõõta pikkusi, kaalusid ja muid mõõtühikuid kasutades lihtsamaid geomeetrilisi kujundeid. Õpilane liidab ja lahutab peast 100 piires ja korrutab ja jagab 100 piires ühekohalise arvuga [9].

II kooliastmes (4.-6. klass) läheb tekstülesannete lahendamine sügavamaks ja keerukamaks, kattes mitte ainult aritmeetilisi tehteid, vaid laiendades ka algebra, geomeetria, mõõtmise ja andmete analüüsi oskusi [9]. Õppekava järgi peaksid õpilased oskama lahendada mitmetehtelisi tekstülesandeid, mis nõuavad võrrandite koostamist ja lahendamist [9]. Õpilane liidab ja lahutab peast 1000 piires, korrutab ja jagab 100 piires ja arvutab positiivsete ratsionaalarvudega, sh harilike murdudega, mille vähim ühine nimetaja on kuni 100 [9].

1.2 Loomuliku keele töötlus

Loomuliku keele töötlus on arvutiteaduse, tehisintellekti ja arvutilingvistika valdkond, mis keskendub inimese (loomuliku) keele mõistmisele ja analüüsimisele arvutite poolt [10]. See hõlmab erinevaid tehnoloogiaid nagu masintõlge, kõnetuvastus, tekstist kõnesüntees, emotsioonide tuvastamine tekstist, teksti klassifitseerimine ja palju muud. Eesmärk on muuta arvutid võimeliseks suhtlema inimestega inimkeelt kasutades ning mõistma ja genereerima

inimkeelseid tekste. Tänapäeva loomuliku keele töötlustes kasutatakse peamiselt masinõppel ja närvivõrkudel baseeruvaid keelemudeleid nagu GPT-4 ja BERT [11].

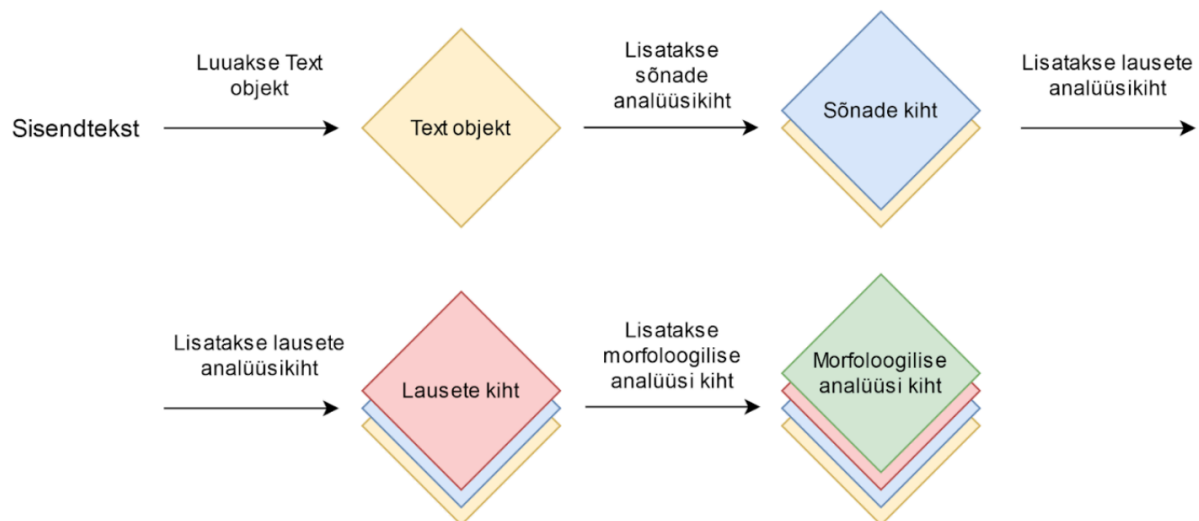
Tekstitötluse ülesannete lahendamiseks on erinevad tööriistakomplekte (ingl *toolboxes* või *toolkits*), mis on enamasti vabavarana kättesaadavad ja pakuvad võimsaid vahendeid keeleanalüüsi teostamiseks. Üks populaarsemaid tööriistu selles suunas on NLTK (ingl *Natural Language Toolkit*), mis tuleb vaikinisi Pythoni programmeerimiskeelega kaasa [12]. Eesti keele jaoks on spetsiaalne teekide kogum EstNLTK.

1.3. EstNLTK

EstNLTK (ingl *Estonian Natural Language Toolkit*) on spetsiaalselt eesti keele töötlusteks välja töötatud avaliku lähtekoodiga teekide kogum, mis ühendab endas mitmeid NLP tööriistu ja meetodeid [3]. Projekti rahastab EKT (Eesti Keeletehnoloogia Riiklik Programm), mis näitab riiklikku pühendumust keeletehnoloogia valdkonna arendamisele [3].

EstNLTK abil saab teha palju erinevaid keeletötluse ülesandeid nagu näiteks sõnade ja lausete tokeniseerimine, morfoloogiline analüüs ja disambiguatsioon, süntaktiline analüüs, verbiahelate tuvastaja, ajaväljendite märkimine, nimeüksuste äratundmine ja aadresside tuvastamine [3].

Enda lõputöös kasutan EstNLTK viimast versiooni 1.7.2, mis avaldati 2023. aasta augustis.



Joonis 1. EstNLTK kihiline struktuur [13].

EstNLTK tötluse keskmes on *Text* objekt, mis toimib ankruna, millele saab juurde lisada erinevaid analüüsikihte vastavalt vajadusele. Seda protsessi illustreerib hästi joonis 1, mis näitab *Text* objektile kihtide juurde lisamist. Esimesena lisatakse sõnade analüüsikiht, mis

töötleb teksti sõnastruktuurilisel tasemel, eraldades sõnad ja määrates nende piirid. Järgnevalt lisatakse lausete analüüsikiht, mis töötleb *Text* objektis olevat sõnade kihti, et tuvastada lausete piirid ja struktuur. Seejärel lisatakse morfoloogilise analüüsi kiht, mis kasutab sõnade ja lausete struktuurilist infot, et teostada sõnade morfoloogiline analüüs. Iga kihi lisamine EstNLTK-s on samm edasi teksti süvitsi mõistmisel, võimaldades kasutada saadud kihte edasistes NLP ülesannetes.

2. Olemasolevad lahendused

Eestikeelsete tekstülesannete genereerimise ja lahendamise valdkonnas on mitmeid olemasolevaid lahendusi, mis pakuvad eri lähenemisi ja tehnilisi lahendusi. Järgnevalt antakse ülevaade mõnedest märkimisväärsimatest projektidest, mis toetavad laiemat arusaama selles valdkonnas toimuvast.

2.1. Tekstülesannete generaator

Eestikeelsete tekstülesannete generaatoreid on loodud ka varem, üks selline asub veebiaadressil <https://www.filosoft.ee/henri/mate/> [14]. See rakendus loodi 2016. aastal ja suudab luua tekstülesandeid vastavalt kasutaja poolt määratud parameetritele [15]. Kasutajad saavad määrata, mis tehet on vastuse saamiseks vaja, valida arvude vahemiku, mida tehted peaks hõlmama ja kontrollida oma vastuse õigsust. Vale vastuse korral saab kasutaja uuesti arvata, kuni õige vastuse sisestamiseni.

Tekstülesannete generaator

Millist ülesannet soovid?

- ☐ kahe tehtega ülesanne
☒ juhuslik ülesanne

Vastuse saamiseks on vaja

- ☐ liita
☐ lahutada
☒ korrutada
☐ jagada

Ülesande tekstis on sõna

- ☒ rohkem
☐ vähem

Arvutamine piires

- ☐ tehted ühekohalise arvuga
☐ tehted korrutustabelist

Mõtle ülesanne!

Koolis elab 26 ema. Isasid on koolis 2. Mitu korda on isasid vähem kui emasid?

Vastus:

0	Vastan	
1	2	3
4	5	6
7	8	9
Kustuta		

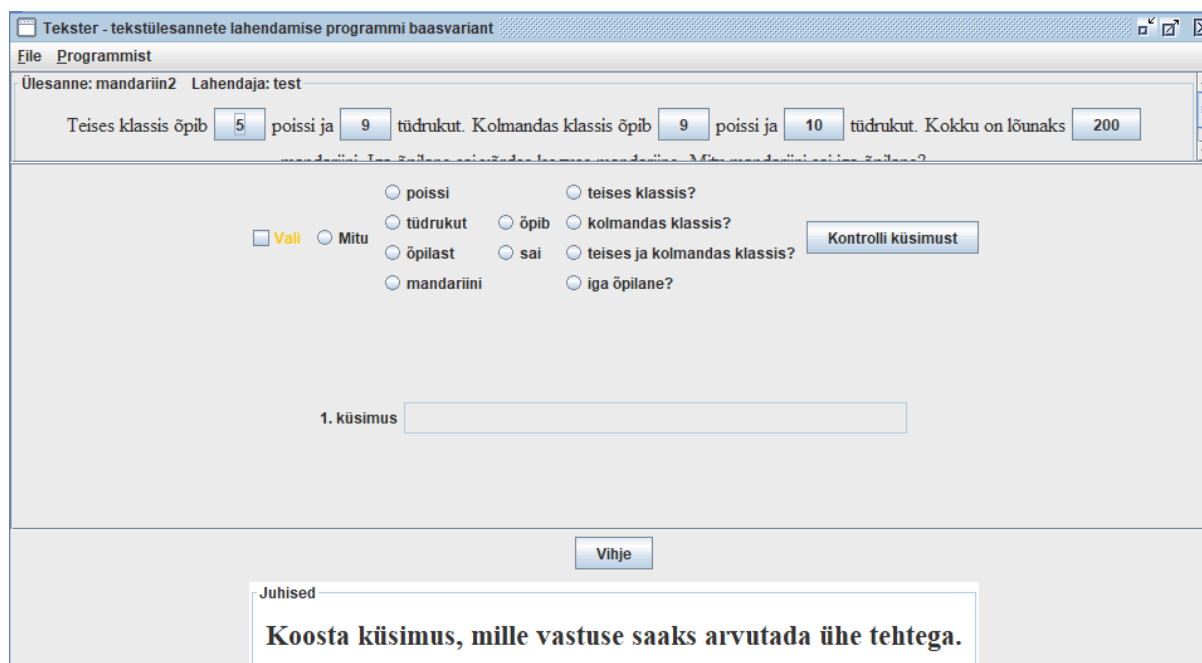
Pizza! Pizza!
Zone Counter

Joonis 2. Ekraanitõmmis Filosoft tekstülesannete generaatori praegusest veebilehest.

Üksikasjalikum analüüs näitas, et rakendus kasutab tekstülesannete loomiseks eeldefineeritud literaale [15]. Kasutatakse 14 erinevat malli, mida rakendus vastavalt kasutaja poolt valitud parameetritele juhuslike arvudega täidab. Kuigi süsteem on suuteline genereerima erinevaid ülesandeid, võib nende elulisus jääda sageli küsimärgi alla. Näiteks võivad tekkida situatsioonid, kus "liivakastis tantsib balletti 23 vampiiri" või "koolis elab 26 ema". Sellised ebareaalsed stsenaariumid võivad kasutajatele mõjuda eemaletõukavalt.

2.2. Tekster

Tekster on õpitarkvara, mis võimaldab koostada ja lahendada eestikeelseid tekstülesandeid. See loodi esmakordselt 2006. aastal Evari Koppeli bakalaureusetöö raames [16]. Hiljem on Teksterit kolme eraldi bakalaureusetöö käigus täiendatud: Joosep Kibal (2013), Katrin Valdson (2014) ja Joosep Norma (2015) [17, 18, 19]. Tarkvara koosneb kahest peamisest komponendist: ülesannete lahendamise rakendus õpilastele ja ülesannete koostamise programm õpetajatele. Ülesannete lahendamiseks on kasutajaliideses nupud ning enne küsimusele vastamist peab ise sobiva küsimuse kokku panema. Ülesandele vastamiseks peab moodustama avaldise, misjärel on võimalik enda vastuse õigsust kontrollida. Rakendus toetab vaid küsimusi, millele saab ühe sammuga vastata.



Joonis 3. Ekraanitõmmis Teksteri küsimuse koostamisest.

Tekster on arendatud Java programmeerimiskeeles, kasutades graafiliseks kasutajaliideseks Java Swingi, mis võimaldab luua kergekaalulisi kasutajaliideseid. Rakenduse kasutamiseks

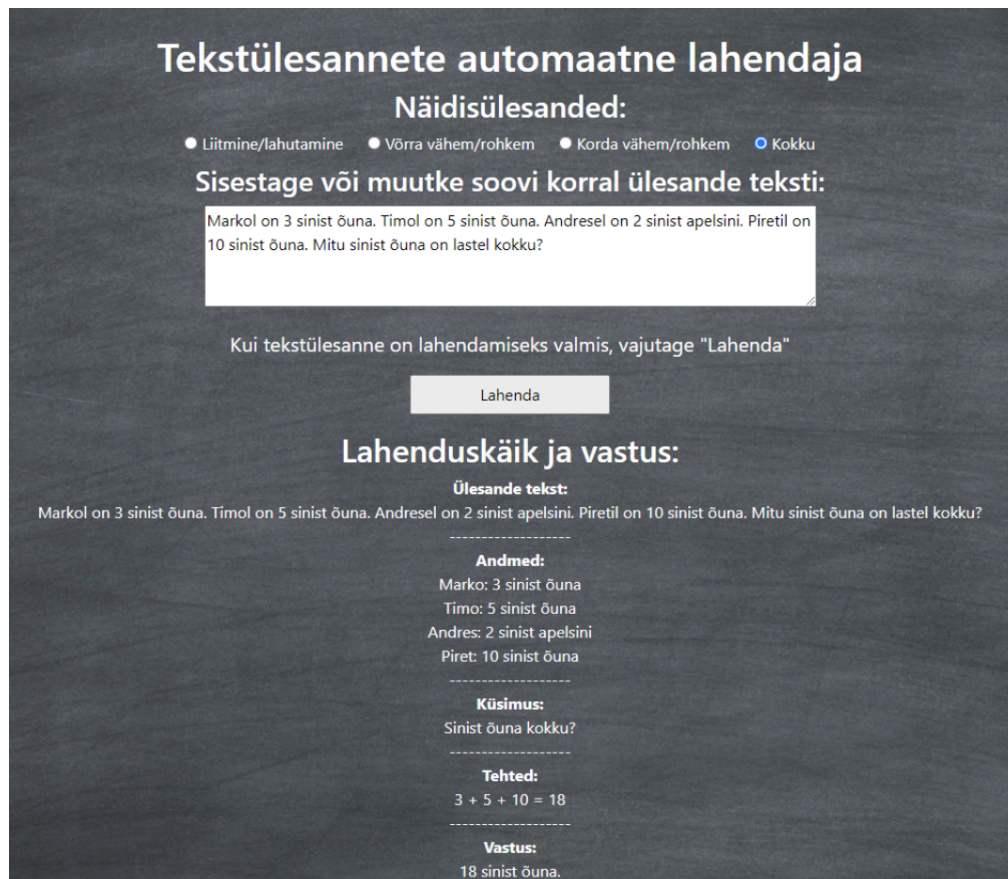
on vajalik Java runtime keskkonna olemasolu süsteemis. Kuigi Tekster on aja jooksul saanud täiendusi, on see siiski jäänud suures osas samaks ning kasutajakogemus ei pruugi vastata tänapäevastele ootustele, kuna praeguseks on enamik kasutajaid harjunud veebipõhiste lahendustega.

Tekster sisaldab eraldi rakendust uute tekstülesannete lisamiseks, mis on peamiselt mõeldud õpetajatele. Lisatavad ülesanded peavad järgima Teksteri kindlaksmääratud süntaksit ning õpetajal tuleb määratleda nii muutujad kui ka võimalikud küsimused. Kuigi Tekster sisaldab erinevaid eelnevalt tehtud ülesandeid, võib nende vähesus osutada pikema kasutamise jooksul probleemiks. Rakendus tundub vananenud ja kasutajasõbralikkus jääb alla tänapäevastele haridustehnoloogilistele lahendustele ja standarditele.

2.3. Tekstülesannete automaatne lahendaja

Tartu Ülikoolis kaitses Priit Vallapi 2022. aastal bakalaureusetöö "Tekstülesannete automaatne lahendaja". Tema lõputöö keskendub eestikeelsete matemaatiliste tekstülesannete automaatsele lahendamisele, kasutades selleks EstNLTK keeletöötluste teekide kogumit [20]. Töö põhirõhk on oskusel korrektselt tuvastada ja analüüsida esitatud tekstülesannetes olevaid sõnu, et mõista ülesande püstitust ja välja töötada korrektne lahenduskäik.

Vallapi süsteem eristub eelnevatest lahendustest tänu oma võimele määratleda sõnade algvorme ja tuvastada keelelist konteksti, mis on hädavajalik õigete matemaatiliste toimingute sooritamiseks. Süsteem oskab töödelda kasutajate vabas vormis esitatud tekstülesandeid, pakkudes sellega olulist paindlikkust võrreldes varasemate meetoditega.



Joonis 4. Ekraanitõmmis Priit Vallap tekstülesannete generaatori lahendusest.

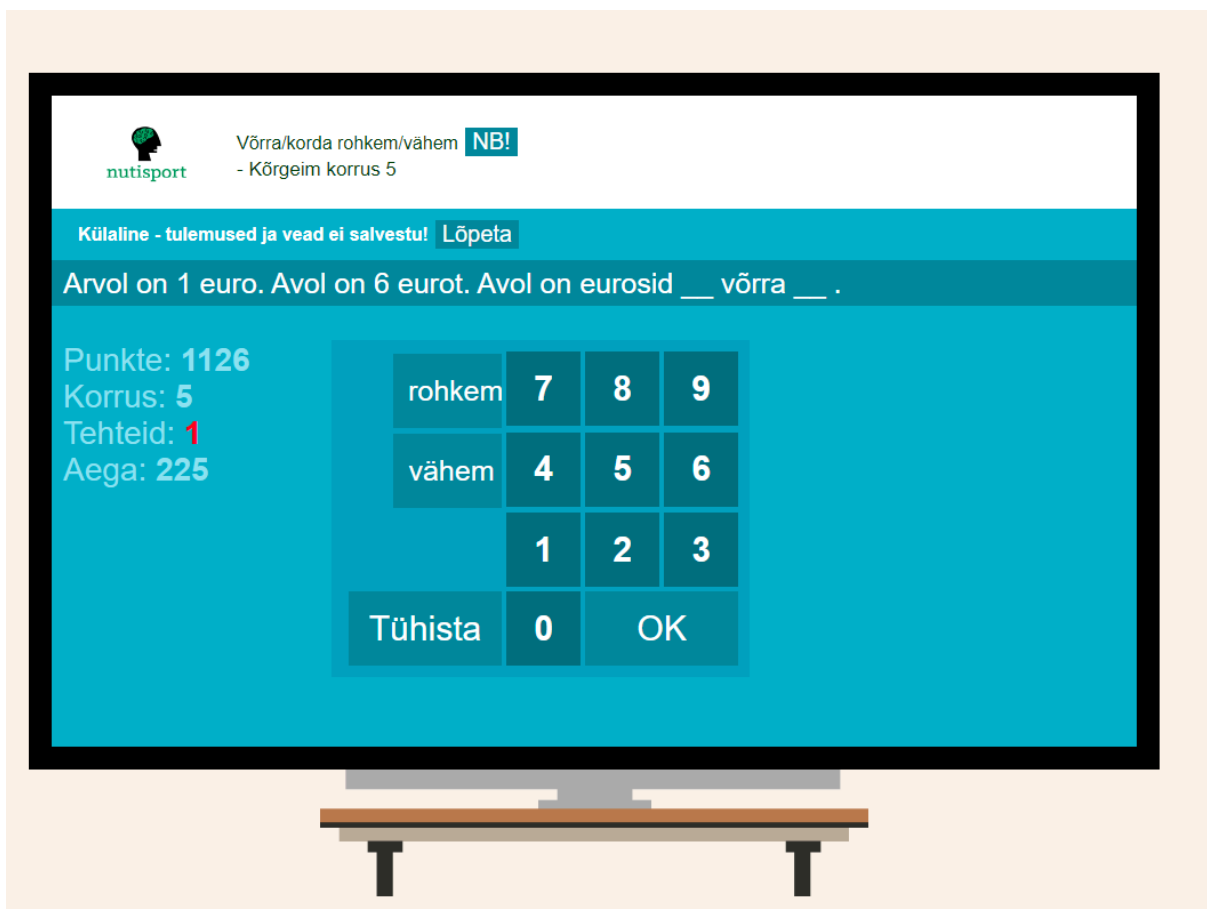
Rakendus demonstreerib stabiilsust standardsete näidisülesannetega, kuid kohandatud või tavapärasest erinevate tekstülesannete puhul esineb probleeme, kus programm ei suuda tuvastada või õigesti töödelda sisendeid, viies ekslike lahenduste ja veateadetení. Originaalne veebileht ei ole enam kättesaadav, mistõttu kasutajad peavad lahendust lokaalselt jooksutama.

Vallap nendib töö kitsaskohti ja näeb potentsiaali süsteemi edasiarenduseks. Ta soovitab muuta muutujate väärtuste leidmise protsess tehteks, mis suurendab lahendaja paindlikkust ja täpsust. Arengukohana toob ta ka välja laiendada süsteemi funktsionaalsust, lisades tuge teist tüüpi matemaatilistele ülesannetele nagu trigonomeetria ja algebra. See laiendaks rakenduse kasutusvõimalusi ja tõstaks selle akadeemilist ja praktilist väärtust.

2.4. NutiSport

Nutisport on interaktiivne e-õppe platvorm, mis pakub sportliku võistluse vormis interaktiivset matemaatikaõpet [21]. Rakendus imiteerib linna, kus matemaatika mõisted on visualiseeritud tänavate, majade ja korrustena. Rakendus katab kuut teemat: arvud, arvutamine, algebra, mõõtmine, teljestik ja geomeetria. Lisaks on igal ülesandel edetabel ning

saab enda tulemust teistega võrrelda nii riiklikul, piirkondliku kui ka kooli ja klassi tasemel. Kõigile ülesannetele on lisatud ka ajaline komponent, mida võetakse aluseks punktide arvutamisel. Tasuta on igas majas ehk teemas kättesaadav demo versioon, mis sisaldab viit korrust ehk harjutust. Täisversioon, mis sisaldab 10-30 ülesannet igas majas, ligipääsuks peab maksma õppeaasta tasu. Tekstülesannete osakaal rakenduses piirdub aga kahe majaga - tähtsavaldisel väärtuse arvutamine ja võrra/korda rohkem/vähem maja. Demo versioonis loodavad tekstülesanded põhinevad mallidel, mis valivad eelnevalt kindlaks määratud nimekirjast nimed ja objektid.



Joonis 5. Ekraanitõmmis nutisport ülesande lahendamise vaatest.

2.5. Kokkuvõte praegustest lahendustest

2023. aastal on põhikooli riiklikus õppekavas eriline rõhk asetatud digipädevusele ja digitehnoloogia kasutamisele õppeprotsessis [7]. Analüüs olemasolevatest lahendustest näitab, et kuigi on tehtud märkimisväärsed samme tekstülesannete genereerimise ja lahendamise automatiseerimise suunas, on tänaseks paljud neist lahendustest vananenud või funktsionaalselt piiratud. On selge vajadus täpsema ja kasutajasõbralikuma lahenduse järele.

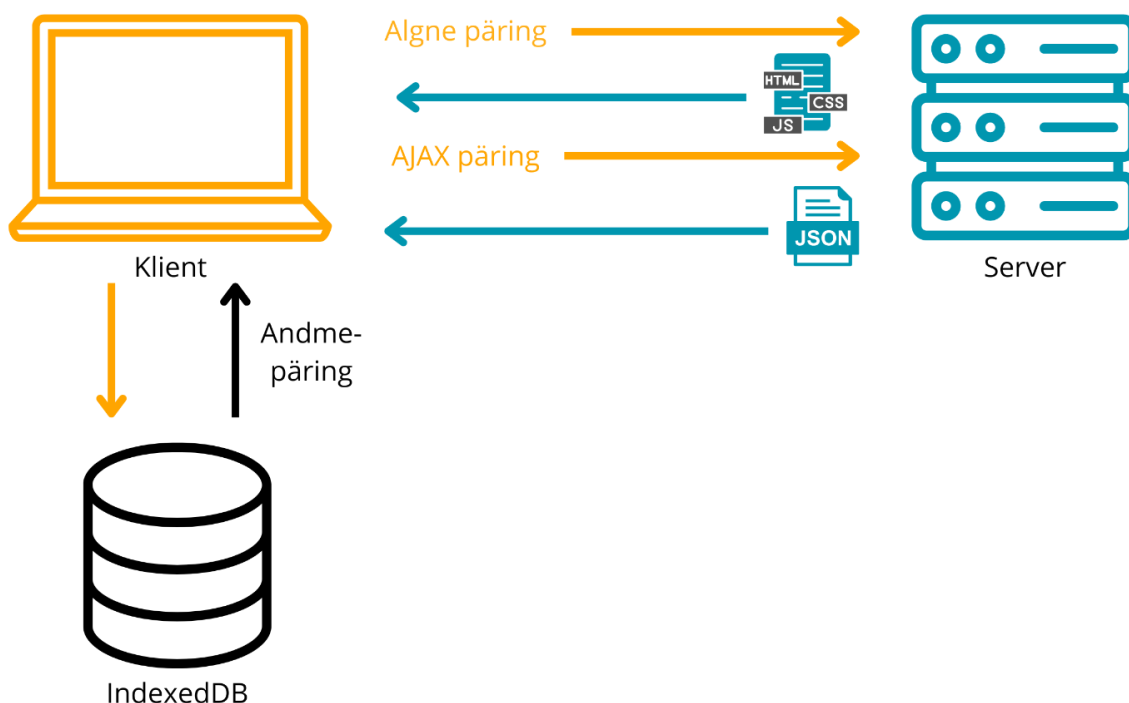
Loodav süsteem peab suutma genereerida ja lahendada mitmekülgseid eestikeelseid tekstülesandeid, esitama usutavaid stsenaariume ning kasutama kaasaegseid keeletehnoloogilisi lahendusi.

3. Programmi ülevaade

See peatükk tutvustab programmi ülesehitust, suhtlust komponentide vahel ja kasutatud tehnoloogiaid.

3.1. Ülesehitus

Loodud lahendus on üheleherakendus (ingl *single-page application*), mis koosneb kliendi- ja serveripoolsetest komponentidest. Üheleherakendus suhtleb taustal veebiserveriga ning muutuste korral kirjutatakse veebileht dünaamiliselt üle, ilma seda uuesti laadimata [22]. Tagaliides on kirjutatud Pythoni programmeerimiskeeles ja Flaski veebiraamistikku (vt. peatükk 3.2) kasutades, mis on oma lihtsuse poolest siin väga sobiv.



Joonis 6. Rakenduse ülesehitus.

Veebirakendus laeb kasutajale esilehe *matemaatika.html*, mis on stiilistatud CSS-i abil ja sisaldab dünaamilist käitumist JavaScripti kaudu. Edasi suhtlus serveriga toimub läbi AJAX päringute. Kasutades AJAX-i, saavad JavaScripti skriptid tegelevad kasutaja sisendi kogumise, päringute asünkroonselt serverisse saatmise ja serveri vastuste tagasisaatisega ilma lehekülge uuesti laadimata [23]. See võimaldab veebirakendusel dünaamiliselt ja viivitamatult uuendada sisu vastavalt serverilt saadud andmetele, pakkudes kasutajale sujuvat ja katkematut kasutajakogemust.

Lisaks on kasutusel IndexedDB andmebaas (vt. peatükk 3.2), mida kasutatakse lahenduste ajaloo salvestamiseks lokaalselt kasutaja seadmes. See tagab andmete kiire kättesaadavuse ja vähendab vajadust pideva võrguühenduse järele. Loodud andmebaas võimaldab kasutajal vaadata tema poolt tehtud päringuid ja saadud vastuseid ilma serverita, võimaldades neid isegi vaadata ilma internetiühendusest.

3.2. Arenduse tööriistad ja tehnoloogiad

Projektis kasutatakse järgmisi peamisi arendusvahendeid ja tehnoloogiaid:

Python - Python on programmeerimiskeel, mida on NLP valdkonnas laialdaselt tunnustatud tänu selle paindlikkusele ja ulatuslikule teekide valikule. Kuna EstNLTK (vt peatükk 1.3.) on kasutatav Pythonis, on see kõige sobivam valik selle rakenduse jaoks. Lisaks on Python tuntud oma lihtsuse ja kiire arendustempo poolest.

Flask - Flask on Pythoni mikro-veebiraamistik, mis on mõeldud lihtsate veebirakenduste kiireks arendamiseks [24]. Flask võimaldab arendajatel luua hõlpsasti skaleeritavaid ja interaktiivseid veebirakendusi. Flaski lihtsus ja paindlikkus teeb selle sobivaks platvormiks, millele ehitada meie rakendus, kuna see võimaldab kerge vaevaga ühendada ees- ja tagaliidese.

HTML, CSS, JavaScript - Projekti veebiliidese loomisel on kasutatud HTML-i, CSS-i ja JavaScripti. See valik võimaldas vältida raamistiku kasutuselevõttu ja sellega seonduvat lisakoodi, mis oleks muutnud süsteemi keerulisemaks. HTML-i kasutatakse veebilehtede struktuuri loomiseks, CSS-i abil saavutatakse järjepidev stiil ja kujundus ning JavaScript lisab lehtedele dünaamilisust ja interaktiivsust.

IndexedDB - Andmebaasi alternatiivina on kasutusel IndexedDB, mis on madala latentsusega kliendipoolne andmebaas, mis võimaldab salvestada suures mahus struktureeritud andmeid otse kasutaja brauseris [25]. See võimaldab rakendusel kiiremini reageerida ja pakkuda paremat kasutajakogemust, vähendades serveriga suhtlemise vajadust. Loodavas lahenduses on IndexedDB tähtsal kohal, talletades varasemalt lahendatud tekstülesannete lahendusi, mis võimaldab kasutajatele vaadata varasemalt lahendatud ülesandeid ilma välisesse andmebaasi päringut tegemata. Selle peamiseks puuduseks on see, et andmed asuvad lokaalselt kasutaja seadmes, mis piirab neile ligipääsu teistest seadmetest.

3.3 Nõuded rakendusele

3.3.1 Funktsionaalsed nõuded

Funktsionaalsed nõuded määratlevad, milliseid konkreetseid funktsioone ja teenuseid süsteem peab pakkuma.

Ülesannete genereerimine - Kasutaja saab valida klassi ja teema, mille alusel ülesanne genereeritakse. Kasutaja valikul genereeritakse matemaatiline tekstülesanne, mis vastab valitud klassi ja teema nõuetele. Genereeritud ülesanne kuvab konteksti ja vajalikud arvud vastavalt valitud parameetritele.

Ülesannete lahendamine - Kasutaja saab sisestada tekstülesande ja süsteem tuvastab olulised andmed. Kasutaja sisestab tekstülesande käsitsi või kasutab genereeritud ülesannet. Lahendaja analüüsib teksti, tuvastab numbrilised andmed ja määrab vajaliku matemaatilise operatsiooni(d). Süsteem kuvab andmed, lahenduskäigu ja vastuse.

Lahenduste ajalugu - Kasutaja saab vaadata eelnevalt lahendatud ülesandeid. Lahenduste ajalugu salvestatakse lokaalsele seadmele kasutades IndexedDB andmebaasi. Kasutaja saab valida eelneva ülesande ajaloost ning seda uuesti vaadata ja lahendada.

3.3.2 Mittefunktsionaalsed nõuded

Mittefunktsionaalsed nõuded määravad süsteemi toimivuse ja kasutusomadused, mille hulka kuuluvad jõudlus, töökindlus, kasutusmugavus jne.

Kasutatavus - Rakendust peab olema lihtne kasutada ilma põhjaliku väljaõppeta. Kasutajaliides peab olema loogiline ja intuitiivne, tagades sujuva kasutajakogemuse.

Jõudlus - Veebirakendus peab kiirelt reageerima kasutaja sisendile ja päringutele. Süsteem peab suutma käsitleda genereerimis- ja lahendamise päringuid mõistliku aja jooksul (mitte rohkem kui mõne sekundi).

Erinevate seadmete tugi - Rakendus peab olema kasutatav erinevates seadmetes (lauaarvutid, nutitelefonid, tahvelarvutid). Kohanduv disain peab tagama parima võimaliku kasutajakogemuse kõigil seadmetel.

4. Valminud lahendus

Käesoleva lõputööna valmis veebirakendus “Eestikeelsete tekstülesannete genereerija ja lahendaja”. Valminud rakendus on kättesaadav veebiaadressil www.gerv.in/matemaatika/. Tegu on üheleherakendusega, mis sisaldab kolme peamist komponenti: genereerija, lahendaja ja ajalugu.

The screenshot shows the web application interface. At the top, the title "Eestikeelsete tekstülesannete genereerija ja lahendaja" is displayed. Below it, a descriptive paragraph states: "See veebileht on osa Gervin Ilusa magistritööst, mis võimaldab eestikeelseid tekstülesandeid genereerida ja lahendada. Rakendus võimaldab genereerida klass ja raskusastmele tekstülesandeid ning neid lahendada, kuvades nii õige vastuse kui ka detailse lahenduskäigu." Below this, a prompt "Sisesta enda tekstülesanne või lase see genereerida." is shown. The interface includes two input fields: "Vali klass:" with a dropdown menu showing "1. klass", and "Vali teema:" with a dropdown menu showing "Liitmine". A blue button labeled "Genereeri" is positioned below these fields. Below the buttons is a large text area containing the placeholder text "Genereeri ülesanne või sisesta siia manuaalselt". At the bottom of the interface are two blue buttons: "Lahenda" and "Ajalugu".

Joonis 6. Kuvatõmmis loodud veebirakendusest sisendita.

Kasutajaliide on disainitud arvestades lihtsust ja kasutatavust. Veebilehe ülemises otsas on rakendust kirjeldav tekst, millele järgneb genereerija sektsioon. Kasutaja saab valida klassi ja teema (vt peatükk 4.1.1), mispeale tekib sisendkasti genereeritud ülesanne. Kui kasutajal on juba ülesanne olemas ning ta soovib ainult seda lahendada, on tal võimalik see otse sisestada tekstialasse. Kui tekstiala on tühi, saab kasutaja veateate. Kui tekstialas on sisu, saab kasutaja kasutada lahendaja funktsionaalsust. Vajutades nuppu “Lahenda” saadetakse päring serveri ning saadakse vastus (vt peatükk 4.1.2). Nuppudele “Genereeri” ja “Lahenda” vajutamine annab kasutajale tagasisidet, et süsteem on nende sisendiga tegelemas. Vastavalt päringu vastusele kuvatakse ülesande andmed, lahenduskäik ja vastus. Lisaks salvestatakse ülesanne ja selle vastus lokaalsesse andmebaasi. Vajutades nuppu “Ajalugu” kuvatakse varasemad

kasutaja sisendid. Valides varasema ülesande, täidetakse tekstiala varasema ülesandega ning kuvatakse varasema ülesande andmed, lahenduskäik ja vastus.

Veebileht on kohanev erinevate seadmetega nagu nutitelefonid ja tahvelarvutid, et tagada parim kasutajakogemus sõltumata kasutaja seadmest.

4.1. Funktsionaalsuse kirjeldus

Rakenduse peamiseks funktsioonideks on eestikeelsete matemaatiliste tekstülesannete genereerimine ja lahendamine. Lisaks sellele sisaldab rakendus ka ajaloo funktsiooni. Teksti genereerimise ja ülesannete lahendamise süsteemi keskmes on EstNLTK teekide kogum.

4.1.1. Generaator

Selleks, et ise tekstülesandeid genereerida, uuriti õpikutes ja veebis leiduvaid materjale ja tekstülesandeid [26, 27, 28]. Vaadeldi nendes esitatud tekstülesandeid, nende struktuure, sõnavara ning kasutatud kontekste. Generaator võimaldab genereerida ühetehtelisi tekstülesandeid igale klassile esimeses ja teises kooliastmes (1.-6. klass) vastavalt klassis omandatavale teemadele. Kõik klassid sisaldavad aritmeetilisi ülesandeid, keskendudes peamiselt liitmis-, lahutamise-, korrutamise- ja jagamistehtele. Generaatori kahest rippmenüüst esimene võimaldab klassi valimist ning teine selles käsitletavat teemat, mille alusel uus ülesanne genereeritakse. See võimaldab igal õpilasel enda õppekava nõuetele vastav tekstülesanne genereerida. Lisaks on rippmenüüs valik "Juhuslik", mis klassi puhul valib nii juhusliku klassi kui ka teema. Kui klass on määratud, saab ka teemana valida "Juhuslik", mis valib juhusliku teema klassi teemade hulgast.

Eestikeelsete tekstülesannete genereerija ja lahendaja

See veebileht on osa Gervin Ilusa magistritööst, mis võimaldab eestikeelseid tekstülesandeid genereerida ja lahendada. Rakendus võimaldab genereerida klass ja raskusastmele tekstülesandeid ning neid lahendada, kuvades nii õige vastuse kui ka detailse lahenduskäigu.

Sisesta enda tekstülesanne või lase see genereerida.

Vali klass:

1. klass

Vali teema:

Liitmine

Genereeri

Toomasel on 2 magusat õuna ja 3 mahlast pirni. Mitu puuvilja on Toomasel kokku?

Lahenda

Ajalugu

Joonis 7. Kuvatõmmis loodud veebirakendusest koos esimese klassi liitmisülesandega.

Tekstülesanded on piiritletud klasside kaupa järgnevalt, aluseks võeti 2022. aasta põhikooli matemaatika õppekava kirjeldusi, kus on iga klassi kohta õppe kirjeldus [29]:

1. klass - liitmine ja lahutamine 20 piires
2. klass - liitmine ja lahutamine 100 piires, korrutamine ja jagamine 10 piires
3. klass - liitmine ja lahutamine 100 piires, korrutamine ja jagamine ühekohalise arvuga 100
4. klass - kõik eelnev ja liitmine ja lahutamine 1000 piires, korrutamine ja jagamine 100 piires
5. klass - kõik eelnev ja liitmine, lahutamine, korrutamine, aga keerulisemad
6. klass - kõik eelnevalt mainitud tegevused, suurema arvulise piiranguga tehted.

Generaator saadab valitud klassi ja teemaga päringu tagaliidesele, kus toimub esmalt funktsioonis *generate_problem* sisendi valideerimine, kas sisestatud klass ja teema on lubatud. Kui sisend on korrektne, valib funktsioon *generate_topic_problem* juhusliku malli vastavalt valitud klassile ja teemale sõnastikust.

Arvude genereerimiseks on funktsioon `generate_numbers`, mis loob kaks sobivat arvu vastavalt klassile. Nimede ja asukohtade valikuks on tehtud loendid *estonian_names* ja *estonian_locations*. Esemete ja omaduste jaoks on funktsioon `get_random_item_attribute`, mis kasutab loodud sõnastikke *item_categories* esemete kategooriateks ja *attribute_categories* omaduste kategooriateks. Seejärel kasutatakse Wordneti, et leida esemetele ja omadustele hüponüüme ehk alammõistelisi sõnu, nt maius hüponüümid on suhkruvatt, komm, šokolaad jne. Wordnetist võib sattuda ka õppuritele vähem tuttavaid sõnu, mis võib nende sõnavara aga hoopis rikastada või uue ülesande genereerida.

Järgmise sammuna on funktsioon `inflect_word`, mis pöörab valitud eseme, omaduse, nime ja asukoha sõnad vastavalt malli nõuetele (nt ainsus, mitmus, omastav kääne). Kõik genereeritud sõnad asendatakse juhuslikult valitud mallis ja ülesanne tagastatakse kasutajale.

4.1.2. Lahendaja

Lahendaja eesmärk on automaatselt analüüsida ja lahendada matemaatika tekstülesandeid eesti keeles. Protsess koosneb järgmistest põhisammudest:

1. **Tekstülesande struktureerimine:** Tekst jagatakse lauseteks, teostatakse morfoloogiline analüüs ja süntaktiline sõltuvuste analüüs.
2. **Faktide ja küsimuse eraldamine:** Tuvastatakse ülesandes esitatud faktid ja küsimus.
3. **Aritmeetilise tehte tuvastamine:** Leitud faktide ja sõnastuse põhjal tehakse kindlaks vajalik tehe.
4. **Lahenduse leidmine:** Rakendatakse sobivat aritmeetilist tehet faktidele.
5. **Lahenduse esitamine:** Genereeritakse loetav lahenduskäik koos vastusega.

Lõpptulemusena esitatakse kasutajale andmed (ülesandes esitatud faktid), lahenduskäik (aritmeetilised tehted ja nende tulemused) ning lõplik vastus. Kui rakendus ei suuda tekstülesannet edukalt töödelda, antakse kasutajale veateade koos täpsema kirjeldusega probleemi kohta, nt ebapiisavad andmed või küsimuse puudumine.

Eestikeelsete tekstülesannete genereerija ja lahendaja

See veebileht on osa Gervin Ilusa magistritööst, mis võimaldab eestikeelseid tekstülesandeid genereerida ja lahendada. Rakendus võimaldab genereerida klass ja raskusastmele tekstülesandeid ning neid lahendada, kuvades nii õige vastuse kui ka detailse lahenduskäigu.

Sisesta enda tekstülesanne või lase see genereerida.

Vali klass:

1. klass

Vali teema:

Liitmine

Genereeri

Toomasel on 2 magusat õuna ja 3 mahlast pirni. Mitu puuvilja on Toomasel kokku?

Lahenda

Toomas magusaid õunu: 2, Toomas mahlanemahseid pirne: 3
Lahenduskäik:
Liidame arvud 2 ja 3
Vastus: 5

Ajalugu

Joonis 8. Kuvatõmmis loodud veebirakendusest koos lahendajaga.

Matemaatika tekstülesannete lahendamiseks on välja töötatud mitmeastmeline süsteem (toru), mis kasutab erinevaid algoritme ja keeletehnoloogilisi tööriistu.

4.1.2.1 Parser

Tekstülesannete lahendaja tuumiku moodustab parser, mis vastutab ülesande sisu mõistmise ja struktureerimise eest. Parseri ülesanne on eraldada ülesandest olulised faktid, tuvastada küsimus ning määrata, millist matemaatilist tehet tuleb lahenduse leidmiseks rakendada.

Parser alustab tööd lausete tuvastamisega ning seejärel analüüsib iga lauset eraldi. Küsimuse leidmiseks otsitakse lauset, mis lõpeb küsimärgiga. Faktide leidmiseks kasutatakse süntaktilist analüüsi, et tuvastada lauses subjekt, tegusõna ja objekt. Selleks rakendatakse

morfoloogilist analüüsi, mis määrab sõnade põhivormid, osasõnad ja grammatilised omadused. Morfoloogiline analüüs on oluline järgnevatele etappidele, kuna see annab infot sõnade tähenduse ja funktsiooni kohta lauses. Sõltuvusanalüüsiks kasutatakse *MaltParserTagger*, mis loob sõnade vahel grammatilistel seostel põhineva sõltuvuspuu.

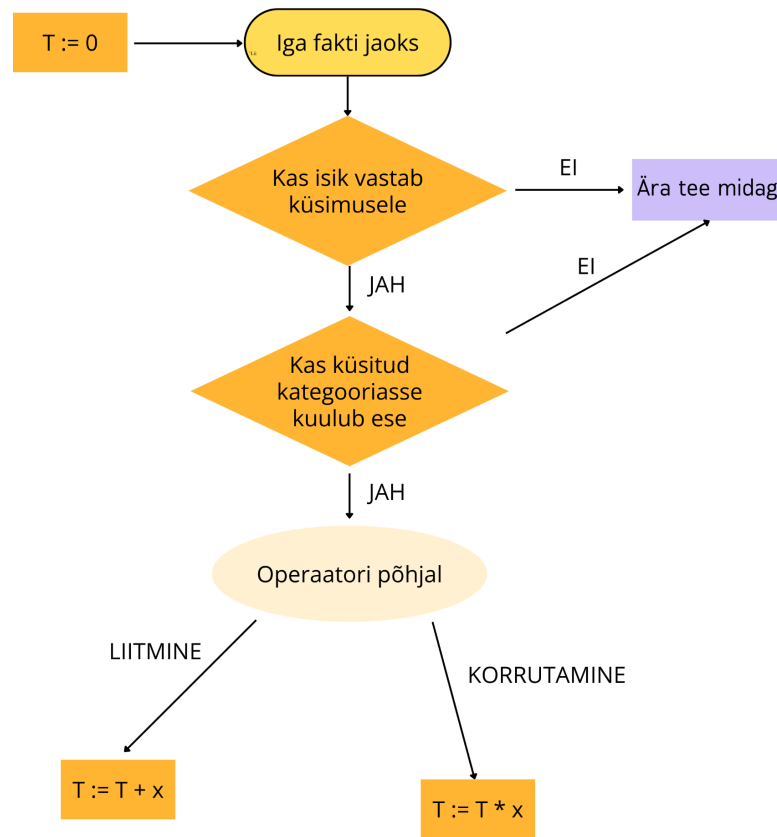


Joonis 9. Lahendaja parser

Parser kasutab Eesti Wordneti, mis aitab mõista objektide vahelisi seoseid. Näiteks kui küsimuses küsitakse "Mitu puuvilja on Jukul?", siis Wordneti abil kontrollitakse, kas faktides mainitud "õunad" ja "pirnid" kuuluvad tõepoolest puuviljade kategooriasse. See tagab, et lahenduses arvestatakse ainult relevantseid objekte.

Parser kasutab ka loodud operaatori märgendajat (*MathOperationTagger*), mis tuvastab matemaatilisi operatsioone. See märgendaja otsib tekstist sõnu ja fraase, mis viitavad liitmisele, lahutamisele, korrutamisele või jagamisele. Tuvastatud operatsioon on oluline, et valida õige aritmeetiline tehe ülesande lahendamiseks.

4.1.2.2. Liitmisülesanne



Joonis 10. Lahendaja liitmise algoritm

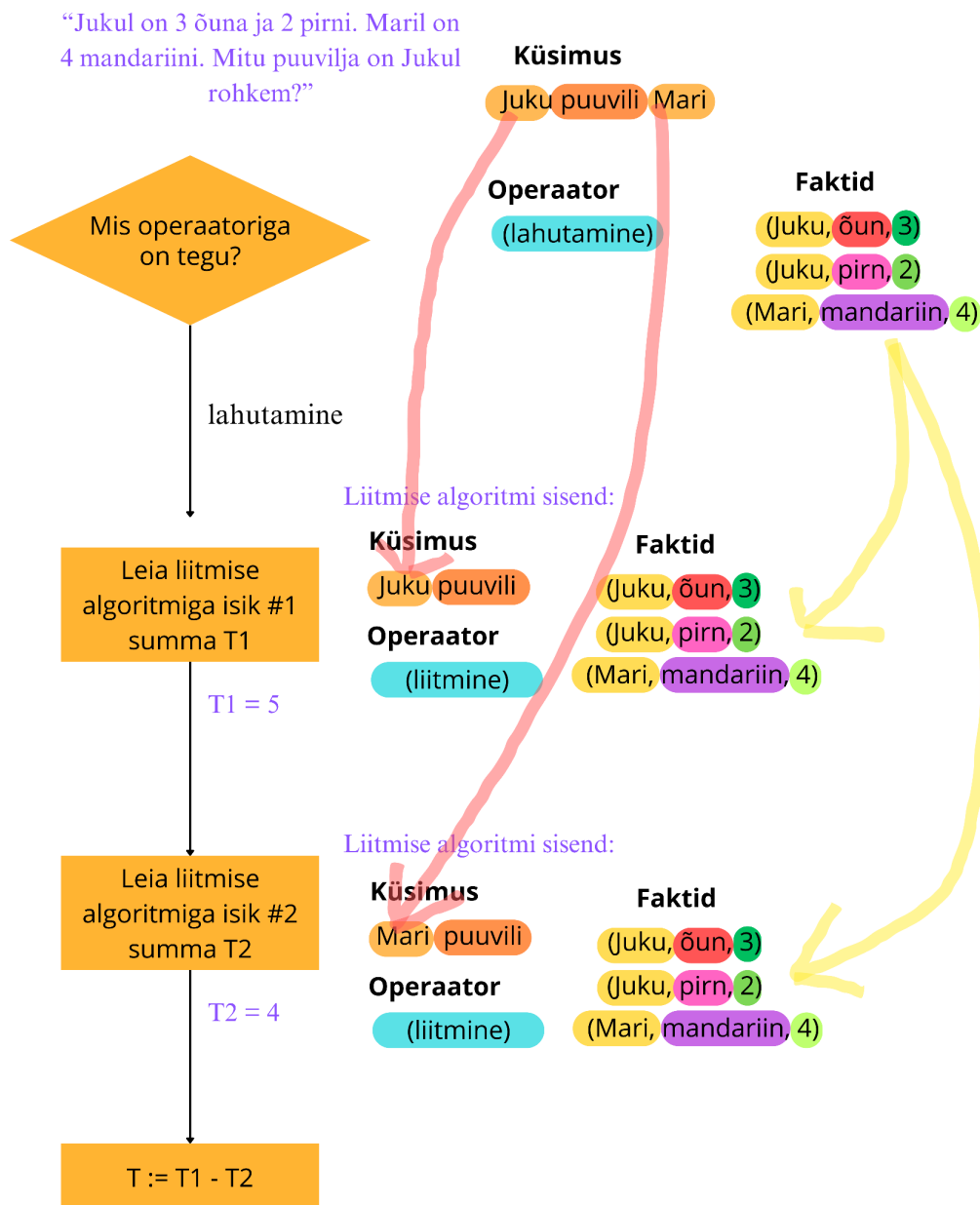
Iga parseri poolt tuvastatud fakti kohta tehakse järgmised:

- 1) **Isiku kontrollimine** - Kui ülesandes on määratud konkreetne isik ("nt mitu õuna on Jukul?"), siis kontrollitakse, kas fakt on seotud selle isikuga. Kui ei ole, jäetakse fakt vahele.
- 2) **Kategooria kontrollimine** - Kui küsimuses on määratud konkreetne kategooria (nt "Mitu puuvilja on kokku?"), siis kontrollitakse, kas fakti objekt kuulub sellesse kategooriasse Wordneti abil. Kui ei kuulu, siis fakt jäetakse vahele.
- 3) **Liitmine** - Kui fakt vastab nii isiku kui ka kategooria tingimustele, siis faktis sisalduv arv liidetakse tulemusele.

4.1.2.3. Lahutamisülesanne

Lahutamisülesannete lahendamiseks kasutatakse järgmist algoritmi:

1. **Operaatori kinnitamine:** MathOperationTagger klassi abil otsitakse tekstist sõnu või fraase, mis viitavad lahutamisele.
2. **Esimese isiku summa leidmine (T1):** Kasutades liitmise algoritmi (vt peatükk 4.1.2.2), leitakse esimese isiku (nt Juku) poolt omadud objektide koguarv. Selleks edastatakse liitmise algoritmile küsimus kujul "(Esimene isik) (objekti tüüp)" ja operatsioon "liitmine".
3. **Teise isiku summa leidmine (T2):** Sarnaselt eelmise sammuga leitakse teise isiku (nt Mari) poolt omadud objektide koguarv, kasutades liitmise algoritmi ja küsimust kujul "(Teine isik) (objekti tüüp)".
4. **Lahutamine:** Esimese isiku summast (T1) lahutatakse teise isiku summa (T2). Tulemuseks on esimese isiku poolt rohkem omatud objektide arv.



Joonis 11. Lahendaja ära andmise algoritm

Vaatleme lauset: "Jukul on 3 õuna ja 2 pirni. Maril on 4 mandariini. Mitu puuvilja on Jukul rohkem?". Küsimuseks on Juku puuvili Mari.

Esimese isiku summa leidmiseks (T1) edastatakse liitmise algoritmile küsimus (Juku, puuvili) ja operatsioon liitmine. Liitmise algoritmiga leiame, et Jukul on 5 puuvilja (3 õuna ja 2 pirni). Seega T1 = 5.

Teise isiku summa leidmiseks (T2) edastatakse liitmise algoritmile küsimus (Mari, puuvili) ja operatsioon liitmine. Liitmise algoritm leiab, et Maril on 4 puuvilja (4 mandariini). Saame et T2 = 4.

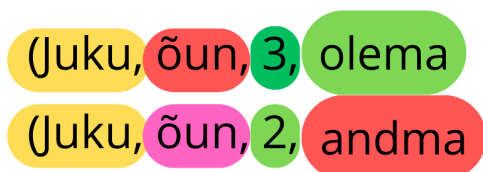
Teades esimese ja teise isiku summat, leiame T lahutades T1st T2. $T = T1 - T2 = 5 - 4 = 1$.

4.1.2.4. Äraandmisülesanne

Äraandmisülesannete eripäraks on see, et ülesandes kirjeldatakse olukorda, kus teatud koguses midagi üldiselt ära antakse või antakse teisele isikule. Kui fakti tegusõna kuulub äraandmist märkivate sõnade hulka (nt "andma"), siis faktis sisalduv arv muudetakse negatiivseks selle isiku jaoks, kes objekti ära annab. See hulk ei tule Wordnetist, vaid on manuaalselt koostatud. Kasutame sõltuvuspuid, et määrata, kas äraandmist märkivale tegusõnale (nt "andma", "kaotama") järgneb mõni muu nimisõna või asesõna, mis tähistab saajat. Kui saaja on olemas, siis faktis sisalduv arv muudetakse negatiivseks andja jaoks ja positiivseks saaja jaoks.

“Jukul on 3 õuna. Ta andis 2 õuna
ära. Mitu õuna on tal nüüd?”

Faktid



Joonis 12. Lahendaja ära andmise algoritm

4.1.2.5. Korrutamine ja jagamine

Sarnaselt liitmise ja lahutamisega algab korrutamise ja jagamise ülesannete lahendamine nende tuvastamisega. Operaatori märgendaja abil otsitakse tekstist sõnu, mis viitaksid vastavale tehteliigile. Tüüpilised märksõnad korrutamise jaoks on "korda", "korrutatud" või "igaühele", samas kui jagamise puhul võivad esineda sõnad nagu "jagatud", "võrdselt" või "igaüks saab".

Korrutamise algoritm käitub sarnaselt liitmise algoritmile (vt peatükk 4.1.2.2). Jagamine käitub sarnaselt lahutamisega (vt peatükk 4.1.2.3). Korrutamise ja jagamise järjekorra määramiseks kasutame sõltuvuspuid analüüsi.

4.1.3. Ajalugu

Ajaloo funktsioon on oluline osa rakendusest, mis võimaldab kasutajatel vaadata varasemalt lahendatud tekstülesandeid ja nende lahendusi. Ajaloo andmed salvestatakse IndexedDB andmebaasi, mis on kõrge jõudlusega kliendipõhine andmebaasisüsteem. Kasutamine IndexedDB andmebaasi võimaldab andmete kiiret kättesaadavust ja vähendab serveripoolsete päringute vajadust, mis tagab rakenduse sujuvama toimimise isegi piiratud või aeglase internetiühenduse korral. Kuna andmed salvestatakse lokaalselt kasutaja seadmes, on kasutaja ülesannete andmed kaitstud.

5. Valminud lahenduse analüüs

Järgnevas peatükis analüüsitakse valminud lahendust manuaalse testimise ja kasutajate tagasiside põhjal. Testimise protsess hõlmas nii funktsionaalsuste kontrollimist kui ka kasutajakogemust.

5.1. Rakenduse testimine

Rakenduse testimine on oluline samm, et veenduda selle toimivuses ja kasutajakogemuses. Läbiviidud testimine jagunes kahte ossa: funktsionaalne testimine ja kasutajate tagasiside kogumine.

5.1.1. Testimise metoodika

Testimise protsessis osales kokku seitse inimest, kelleks olid sõbrad ja tuttavad. Testijatele anti ülevaade lühiülevaade rakenduse funktsioonidest (ülesannete genereerimine, lahendamine ja ajaloo vaatamine) ning nad pidid esitama tagasisidet kogemuse kohta. Testimine toimus kahel etapil:

1. **Funktsionaalne testimine** - kontrolliti, kas rakenduse kõik funktsionaalsused töötavad ootuspäraselt. Kasutajad pidid genereerima ja lahendama tekstülesandeid, kasutama ajaloo vaatamise funktsiooni ning hindama rakenduse stabiilsust ja jõudlust.

2. **Kasutajate tagasiside** - osalejatel paluti kommenteerida rakenduse kasutajasõbralikkust, kujundust ja üldist rahulolu. Kogutud tagasisidet analüüsiti, et selgitada välja võimalikud arendusvajadused ja täiustamise võimalused.

5.1.2. Testimise tulemused

Testimise tulemused olid üldjoontes positiivsed. Alljärgnevalt mõned peamised tähelepanekud:

- **Kasutusmugavus:** Enamik kasutajaid leidis, et rakendus on intuitiivne ja kasutajasõbralik. Nad hindasid kõrgelt lihtsat ja selget kasutajaliidest ning leidsid, et rakendus ei vaja spetsiaalset väljaõpet kasutamiseks.
- **Funktsionaalsus:** Generaatori ja lahendaja funktsioonid töötasid ootuspäraselt.
- **Ajaloo funktsionaalsus:** Rakenduse ajaloo funktsionaalsus, mis võimaldab kasutajal vaadata varasemaid ülesandeid ja lahendusi, oli kasutajate seas populaarne.
- **Kujundus:** Enamik kasutajaid pidas rakenduse kujundust meeldivaks ja kaasaegseks. Katsetused ka telefonis kasutada olid edukad.

- **Veateated:** Kasutajad, kes proovisid poolikuid ülesandeid sisestada, said ootuspärased veateated. Hinnati täpsete veateadete olemasolu, mis aitasid kiiresti aru saada, milles probleem seisneb.

5.2. Tulemuste kokkuvõte

Testijate arvamused ja tähelepanekud olid üldjoontes positiivsed. Enamik funktsionaalsusi töötasid ootuspäraselt ja rakendus pälvis kiitusi oma kasutajasõbralikkuse eest. Mõned soovitusel edasiseks arenduseks hõlmasid:

- Laiendada ülesannete valiku - kuigi ülesanded varieerusid, leidis mõni kasutaja, et võiks olla rohkem erinevaid ja keerukamaid ülesandeid, eriti teise kooliastme jaoks.
- Täiendavad juhised - kuigi rakendus oli üldiselt lihtne kasutada, leiti, et mõned täiendavad juhendid võiksid kasuks tulla.

6. Võimalikud edasiarendused

Nii isiklikud katsetused ning kasutajatega testimise tulemused paljastasid mitmeid edasiarendamise võimalusi, millest mõned on välja toodud järgnevas peatükis.

6.1 Süvaõpe

Kuigi praegune rakendus on suuteline tuvastama ja lahendama lihtsamaid matemaatilisi tekstülesandeid esimese ja teise kooliastme jaoks, on üheks potentsiaalseks edasiarenduseks süvaõppe tehnoloogiate integreerimine. Süvaõppe võimaldaks luua keerukamaid ja realistlikumaid ülesandeid ning pakkuda täpsemaid lahendusi. Selle rakendamisega oleks võimalik

6.2 Ülesannete lisamine

Kasutajate testimise käigus selgus, et kuigi olemasolevad ülesannete tüübid katavad põhilised matemaatilised mõisted ja tehnikad, on mitmeid täiendavaid matemaatilisi teemasid, mida rakendus võiks käsitleda. Edasised arendused võiksid sisaldada:

- Täiendada olemasolevaid teemasid.
- Lisada uusi teemasid nagu geomeetria, trigonomeetria ja algebra.

6.3 Keerukuse tõstmine

Praegu lahendus hõlmab ühetehtelisi aritmeetilisi ülesandeid ja ei pruugi piisavalt väljakutseid pakkuda kõrgema taseme õpilastele. Keerukuse tõstmine võiks sisaldada järgmist:

- Mitmetasemeliste ülesannete lisamine, mis eeldavad keerukamaid arvutusi ja loogilist mõtlemist.
- Ülesannete dünaamiline kohandamine vastavalt kasutaja õppe edukusele – rakendus võiks analüüsida kasutaja varasemat sooritust ja pakkuda järgnevalt keerukamaid ülesandeid.

6.4 Kasutajaliidese täiustamine

Lahendaja kasutajaliides on praegu minimaalne ning keskendub peamiselt tekstilisele sisendile ja väljundile. Edaspidi võiks luua interaktiivsema ja kasutajasõbralikuma liidese,

mis võimaldaks kasutajal ülesandeid sisestada erinevatel viisidel (nt pildi üleslaadimine) ning saaks visuaalselt kuvada lahenduskäiku ja selgitusi. Samuti võiks lisada võimaluse anda kasutajale tagasisidet ja vihjeid, kuidas ülesande lahendamisele läheneda.

Kokkuvõte

Käesoleva magistritöö raames arendati eestikeelsete tekstipõhiste matemaatiliste ülesannete genereerija ja lahendaja, kasutades selleks EstNLTK teekide kogumit. Töö eesmärk oli demonstreerida keele tehnoloogia potentsiaali õppeprotsessi toetamisel ja matemaatikaõppe tõhustamisel. Loodud rakendus pakub õpetajatele, õpilastele ja vanematele tööriista, mis automatiseerib nii ülesannete loomise kui ka lahendamise, esitades lisaks ka detailse lahenduskäigu.

Projekt hõlmas põhjalikku analüüsi olemasolevate lahenduste kohta, tuues esile nende tugevused ja puudujäägid. Sellele järgnes uue süsteemi planeerimine ja arendus, kus kasutati mitmeid loomuliku keele töötlus (NLP) vahendeid ja tehnikaid, sealhulgas morfoloogilist analüüsi, sõltuvusanalüüsi ning WordNeti semantilist võrgustikku. Arendatud süsteem võimaldab kasutajatel genereerida erineva raskusastmega matemaatikaülesandeid, lahendada neid automaatselt ning saada üksikasjalikku tagasisidet lahenduskäigu kohta. Süsteem on võimeline lahendama ka ise sisestatud tekstülesandeid.

Rakenduse testimine kinnitas selle funktsionaalsust ja kasutajate rahulolu. Samuti tuvastati testimisel mitmeid võimalusi süsteemi edasiseks parendamiseks, näiteks ülesannete keerukuse tõstmine, teemade laiendamine ning kasutajaliidese täiustamine.

Käesolev töö näitas, et EstNLTK ja kaasaegsed NLP tehnoloogiad pakuvad võimsaid vahendeid haridustehnoloogia valdkonnas, eriti matemaatiliste tekstülesannete kontekstis. Loodud "Eestikeelsete tekstülesannete genereerija ja lahendaja" on väärtuslik lisand Eesti haridustehnoloogia maastikule, laiendades loomuliku keele töötlusvõimalusi hariduses. Edasised uuringud ja arendused aitavad süsteemi veelgi täiustada, muutes selle veelgi kasulikumaks ja mitmekülgsemaks tööriistaks õppetöös.

Kasutatud kirjandus

- [1] Palu, A. (2010) *Algklassiõpilaste matemaatikaalased teadmised, nende areng ja mõjutavad tegurid* <http://hdl.handle.net/10062/15295> (15.02.2024)
- [2] Meola, M. (2023) *Probleemid matemaatika tekstülesannete lahendamisel ja võimalikud lahendused* <http://hdl.handle.net/10062/89173> (15.02.2024)
- [3] Laur, S., Orasmaa, S., Särg, D., Tammo, P. (2020) *EstNLTK 1.6: Remastered Estonian NLP Pipeline*. <https://estnlk.github.io/> (17.12.2023)
- [4] Piht, S. (2010) *Matemaatika õpetamisest esimeses kooliastmes* <https://www.digar.ee/arhiiv/et/raamatud/21447> (15.02.2024)
- [5] Kuusk, R. (2006) *Matemaatika tekstülesannete mõistmisraskused põhikooli lihtsustatud õppekava järgi õppivatel lastel* <http://hdl.handle.net/10062/246> (15.02.2024)
- [6] Kikas, E. (2010) *Õppimine ja õpetamine esimeses ja teises kooliastmes* <http://hdl.handle.net/10062/40579> (18.03.2024)
- [7] Haridus- ja Noorteamet (2023) *Põhikooli riiklik õppekava 2023* <https://www.riigiteataja.ee/akt/129082014020?leiaKehtiv> (15.03.2024)
- [8] Ōunap, K. (2020) *Vead teise kooliastme matemaatika ülesannete lahendamisel* (06.04.2024)
- [9] Vabariigi Valitsus (2023) *Põhikooli riiklik õppekava ainevaldkond "Matemaatika"* https://www.riigiteataja.ee/akt/1080/3202/3005/18m_pohi_lisa5.pdf# (18.03.2024)
- [10] IBM *What is Natural Language Processing (NLP)?* <https://www.ibm.com/topics/natural-language-processing> (05.04.2024)
- [11] Saadani, T. (2022) *6 Natural Language Processing Models you should know* <https://medium.com/ubiai-nlp/5-natural-language-processing-models-you-should-know-836958303ce3> (05.04.2024)
- [12] Bird, S., Klein, E., Loper, E. (2009) *Natural language processing with Python: analyzing text with the natural language toolkit*. <https://www.nltk.org/> (18.12.2023)
- [13] Karindi, K. E. (2020) *Lõputöö teksti analüsaator* <https://hdl.handle.net/10062/93338> (15.02.2024)
- [14] Filosoft <https://www.filosoft.ee/henri/mate/> (01.02.2024)
- [15] Filosoft <https://www.filosoft.ee/henri/mate/textylesanded.js> (01.02.2024)

- [16] Koppel, E. (2006) *Tekstülesannete lahendamise programmi baasvariant*
<http://hdl.handle.net/10062/73272> (02.02.2024)
- [17] Kibal, J. (2013) *Tekstülesannete lahendamise programmi funktsionaalsuse täiendamine* <http://hdl.handle.net/10062/32948> (02.02.2024)
- [18] Valdson, K. (2014) *Tekstülesannete lahendamise programmi täiendamine*
<http://hdl.handle.net/10062/55998> (02.02.2024)
- [19] Norma, J. (2015) *Eestikeelsete matemaatika õpiprogrammide ülevaade ja tekstülesannete lahendamise programmi täiendamine*
<http://hdl.handle.net/10062/56125> (02.02.2024)
- [20] Vallap, P. (2022) *Tekstülesannete automaatne lahendaja*
<https://hdl.handle.net/10062/91669> (02.02.2024)
- [21] Nutisport <https://nutisport.eu/> (14.02.2024)
- [22] Mozilla Developer Network *SPA (Single-page application)*
<https://developer.mozilla.org/en-US/docs/Glossary/SPA> (15.03.2024)
- [23] IBM (2021) *What is Ajax?*
<https://www.ibm.com/docs/en/rational-soft-arch/9.7.0?topic=page-asynchronous-javascript-xml-ajax-overview> (16.03.2024)
- [24] Flask documentation <https://flask.palletsprojects.com/en/3.0.x/> (16.03.2024)
- [25] Bell, J. (2023) *Indexed Database API 3.0* <https://w3c.github.io/IndexedDB/> (15.03.2024)
- [26] Belials, K. (2021) *Matemaatika töövihik 3. klassile I osa* (21.02.2024)
- [27] Viil, T. (2022) *Matemaatika tööraamat 6. klassile II osa* (21.02.2024)
- [28] Jürgenstein, T., Varik, K. (2013) *Aritmeetika tekstülesanded*
<http://tekstylesanded.weebly.com/i-osa1.html> (21.02.2024)
- [29] Haridus- ja teadusministeerium (2023) *Põhikooli ainevaldkond "Matemaatika"*
<https://oppekava.ee/wp-content/uploads/2015/11/Matemaatika-t%C3%B6%C3%B6kva-n%C3%A4idis-1.-klassile.pdf> (15.02.2024)
- [30] Tartu Ülikooli arvutilingvistika uurimisrühm. *Eesti Wordnet*
<https://www.cl.ut.ee/ressursid/teksaurus> (20.02.2024)

Lisad

1. Rakenduse lähtekood

Rakenduse lähtekood on kättesaadav <https://github.com/Gervin1/matemaatika>.

2. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Gervin Ilus**,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose **Eestikeelsete tekstülesannete genereerija ja lahendaja**, mille juhendaja on **Sven Aller**, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni;
2. annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 4.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni;
3. olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile;
4. kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Gervin Ilus

15.05.2024