TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Markus Mikko

Kõnesünteesi kasutav rakendus ettevõttele Elisa Eesti AS

Bakalaureusetöö (9 EAP)

Juhendaja: Sven Aller

Tartu 2021

Kõnesünteesi kasutav rakendus ettevõttele Elisa Eesti AS

Lühikokkuvõte:

Elisa Eesti ASi kui suure telekomi ettevõtte teenust kasutavad sajad tuhanded kliendid, kes soovivad probleemide korral koheselt infot. Töö kirjutamise hetkel nad saavad seda väga valitud teenuste puhul ja saadavas vastuses puudub detailsus. Käesoleva bakalaureuse töö tulemusena valmis rakendus, mille abil on võimalik soovitud tekst sünteesida kõneks, kasutades Tartu Ülikooli kõnesünteesiliidest ja seda ettevõttele sissetulevate kõnede puhul automaatvastusena esitada.

Võtmesõnad:

Kõnesüntees, Elisa Eesti AS, rakendus

CERCS: P175 Informaatika, süsteemiteooria

Text-to-speech application for Elisa Eesti AS

Abstract:

Elisa Eesti AS is one of the biggest telecommunication companies in Estonia and has many customers, who, in the event of technological failure, want some clarification immediately. At the time of writing there are only handful of prerecorded prompts in the company, which, in addition, are not that informative. As a result of the thesis, an application, where you can synthesize desired text to artificial speech, was created. The application uses an API created by University of Tartu.

Keywords: Text-to-speech, Elisa Eesti AS, application

CERCS: P175 Informatics, systems theory

Sisukord

Sissejuhatus

Elisa Eesti AS on üks suuremaid telekomiettevõtteid Eestis, mis pakub mitmeid erinevaid teenuseid erinevate lahendustega. Elutähtsa teenuse pakkujana võib esineda ka probleeme, mis pole alati sõltuvad neist endast ja mille kohta kliendid soovivad koheselt infot. Kui nad seda mõistliku ajaga ei saa, siis on tavapäraks helistada ettevõtte infotelefonile ja sealt lisainfot küsida. See aga on ettevõttele ebavajalik, sest kliendid helistavad sama murega, mistõttu jääb töötajal muude tegevuste jaoks vähem aega. Hetkel kasutatakse automaatvastustena ettesalvestatud helifaile. Nende miinuseks on vastuste salvestamise aeglus ja saadava vastuse üldisus. Samuti pole neid võimalik peale salvestust enam muuta.

Käesoleva töö eesmärk on luua rakendus, mille abil on võimalik soovitud teksti kõneks sünteesida ja loodud helifaili kasutada sellist tüüpi olukordadeks, mida on ülalpool kirjeldatud. Selle töö käigus valminud rakendus võimaldab ettevõtte töötajal sünteesida soovitud vastus sünteeskõneks, kasutades Tartu Ülikooli teadlaste poolt loodud liidest, ja seda sissetulevate kõnede puhul automaatvastusena ette mängida. Antud rakendusega on võimalik muuta ettevõtte töötajate tööd efektiivsemaks ja samuti anda mõjutatud klientidele kiiremalt vajalikku detailset infot. Lisaks annab rakendus ka ettevõttele suurema vabaduse, mis probleemide puhul automaatvastust mängida, ja ka kiiruse, millega saab detailset vastust sünteesida.

Töö valmis ettevõtte Elisa Eesti AS jaoks, mis kasutab antud rakendust selleks, et oleks võimalik olla info jagamises kiirem ning vabastada ettevõtte töötajad ühetaolistest kõnedest, kus töötajat reaalsuses vaja ei olegi.

Peatükis 1 tutvustatakse lähemalt kõnesünteesi ajalugu, põgusat vaadet tulevikku ja kõnesünteesi seisu Eestis. Peatükis 2 kirjeldatakse erinevaid mudeleid, mida tänapäeval kasutatakse, koos oma tugevuste ja nõrkustega. Peatükis 3 tutvustatakse kõnesünteesiga seotud probleeme, mis võivad ka antud töö tulemust häirida ja millega peab töö kirjutamise hetkel arvestama. Peatükis 4 kirjeldatakse, milliseid tehnoloogiad praktilise rakenduse jaoks kasutati ja mis on nende eelised. Viiendas peatükis on võimalik lugeda valminud lahenduse kohta, mida illustreerivad ka kujutused. Peatükis 6 on ülevaatlik kokkuvõte valminud tööst.

1. Kõnesünteesi ajalugu

Inimkõne sünteesimine ehk võime teisendada suvalist teksti kõneks ilma inimeseta on olnud mitmeid sajandeid teadlaste unistuseks. Käesolevas peatükis kirjeldatakse põgusalt kõnesünteesi ajalugu ja erinevaid ajaloolise tähtsusega seadmeid ja teooriaid, kirjeldatakse seisu Eestis ning heidetakse pilk tulevikku.

1.1 Mehaaniline kõnesüntees

Schroederi väitel [1] tehti esimesed sammud sünteesimiseks juba üle 200 aasta tagasi meile väga lähedal, nimelt Peterburis, kui 1779. aastal saksa professor Christian Kratzenstein põhjendas viie peamise vokaali füsioloogilisi erinevusi. Nendeks vokaalideks olid /a/, /e/, /i/, /u/ ja /o/. Lisaks erinevuste põhjendamisele suutis ta ehitada ka kõige esimese aparaadi, mille abil oli võimalik neid vokaale esitada. Aparaat koosnes resonaatoritest, mis on sarnased inimese kõnetraktile ja mis aktiveeriti vibreerivate pillirooogudega, näiteks kasutati sama tehnoloogiat tollal ka muusikainstrumentidel. Story [2] täiendab, et Kratzenstein osales aparaadiga võistlusel, mille kutsus välja tema kolleeg ja sõber, kuulus matemaatik Leonhard Euler. Euler oskas ennustada, et sellistest masinatest on kasu peamiselt kõnevõimetutel inimestel. Allika [1] järgi tutvustas 12 aastat hiljem Wolfgang von Kempelen Viinis akustilis-mehaanilist kõneseadet (ingl *Acoustic-Mechanical Speech Machine*). Üllataval kombel alustas von Kempelen selle masina ehitamist mitmeid aastaid varem, kui seda oli teinud Kratzenstein, aga otsustas 20 aastat koguda andmeid. Andmete tulemuste abil kirjutas raamatu, kus kirjeldas, kuidas saab inimkõne sünteesida ja seletas lahti mõned eksperimendid tema masinaga. Von Kempeleni uuringu tulemusena tuli välja, et kõnesünteesis on kõige tähtsam osa kõnetrakt, kuigi varasemalt peeti pikalt selleks kõri. Allika sõnul ei võetud kahjuks tema masinat kuigi tõsiselt, sest mees oli varasemalt teinud ühe masina veel – rääkiva ja malet mängiva aparaadi. Kahjuks antud seade sellist revolutsiooni reaalsuses teha ei suutnud ja „aparaadiks“ osutus hoopis peidus olnud mees. Von Kempeleni masinaga oli võimalik sünteesida mõnda sõna, kuid mitte lauseid [2]. Põhjusteks võis lugeda nii aparaadis tekkivat õhupuudust kui ka masina kasutamise keerukust. Tekkinud heli sarnanes lapsehäälele, sest lapsehäält kritiseeriti palju vähem ja masinat demonstreerides võis seda saata suurem edu. Allika sõnul tegi autor seda meelega, sest ta kartis kriitikat, mida ta oli varasema masinaga kogenud, aga kahjuks arvasid siiski mõned huvilised, et ka see masin on võlts. Seetõttu ei olnud masin nii kuulus ja edukas kui oleks pidanud olema [1].

Schroeder jätkab, et peale seda ehitas Charles Wheatstone enda versiooni eelnimetatud von Kempeleni masinast, mis sai samuti kuulsaks. Masin oli oma olemuselt keerulisem ja selle tulemusena oli võimalik produtseerida täishäälikuid, mõnda kaashäälikut ning isegi mõningat sõna. Kõik varasemad masinad olid eelkäijaks Dayton Milleri aparaadile Phonodeik, mida kasutas mees peamiselt muusikainstrumentide ja täishäälikute lainekujude uurimiseks [2]. 1914. aastal avaldas ta oma uuringud selle masinaga ja kirjeldas tekkinud protsessi harmoonilise sünteesina. See on esimene kord, kus keegi kirjeldab oma protsessi kasutades sõna „süntees“, väidab allikas. Kõiki varasemaid aparaate kutsuti kas kõnemasinateks, automaatideks või lihtsalt süsteemideks, mille abil sai inimkõne produtseerida [2]. Selleks ajahetkeks oli mehaaniline ajastu juba lõppemas. Von Kempeleni ja eelkäijate masinad olid selleks hetkeks juba ammu unustatud, sest need olid asendatud elektromehaaniliste seadmetega. Erinevad katsetused mehaaniliste masinatega jätkusid 1960. aastateni, kuid tol hetkel mingit suuremat edu ei saavutanud ja seetõttu oli aeg uueks ajastuks [1].

1.2 Elektrooniline kõnesüntees

Hüpe mehaanilistest ja elektromehaanilistest masinatest elektrisüsteemidele sai alguse 1922. aastal, kui noor füüsik John Stewart avaldas artikli „Vokaalorganite elektriline analoog“ (ingl „*An Electrical Analogue of the Vocal Organs*“) [2]. Seal avaldas ta joonise lihtsast elektriskeemist ja kommentaari, kus mees väitis, et tähelepanuta on jäänud võimalus, kus hääleorganeid saab imiteerida helisageduste võnkumistega elektriahelas. Masin suutis lihtsa vaevaga genereerida erinevaid täishäälikuid. Stewart oskas ka ette näha väga suurt probleemkohta kõnesünteesi valdkonnas. Ta kirjutas oma töö lõppu, et keeruline ei ole ehitada seadet, millega saab helisid produtseerida, vaid tekkiva kõne arusaadavus ja sellest tulenevalt kõnes esinevate reeglite järgimine.

Kõige esimene masin, mida võib täielikult kõnesüntesaatoriks pidada, oli „VODER“ (ingl „*Voice demonstrating operator“*) [3]. Seda tutvustas Homer Dudley New Yorgi maailmamessil aastal 1939. Masinalt heli kättesaamiseks oli vaja teatud osavust. Kõnekvaliteet ja arusaadavus jätsid väga palju soovida, kuid masina olemasolu näitas ala lõputut potentsiaali: peale masina demonstratsiooni kasvas teadusmaailma huvi kõnesünteesi vastu koheselt, sest lõpuks suudeti tõestada, et reaalsuses on võimalik arusaadavat inimkõne sünteesida. Masina ainuke eesmärk ei olnud rahvale demonstreerimiseks, vaid seda kasutati ka näiteks turvalisuse eesmärkidel, näiteks tähtsate vestluste šifreerimiseks Churchilli ja Roosevelti vahel II maailmasõja ajal [2]. Järgmiste kümnendite jooksul, jätkab Klatt [3], valmis mitmeid erinevaid süntesaatoreid, kuid esimene täielik tekst-kõneks süsteem valmis Jaapanis, Noriko Umeda juhitud teadusrühma poolt 1968. aastal. Tekst oli küll arusaadav, kuid liialt monotoonne ning ebakvaliteetne võrreldes tänapäevaste süsteemidega. Allika sõnul esimene abivahend oli 1976. aastal loodud ettelugeja, mis kasutas optilist skännerit ja oskas lugeda üpris arusaadavalt. Kahjuks oli ta tavakasutajale liiga kallis – maksis ka kümneid aastaid peale valmimist umbes 30 000$. Siiski kasutati seda raamatukogudes, kus see oli mõeldud peamiselt vaegnägijatele.

Suured sammud kõnesünteesi arengus olid tehtud ja 1980ndateks oli valminud juba mitmeid kaubanduslikke tekst-kõneks süsteeme [3].

1.3 Kõnesüntees Eestis

Eestikeelne tekst-kõne süntees on kasutatav vabavarana juba 2002. aastast [6]. Meister ja Alumäe [7] väidavad, et kuna keeletehnoloogia on valdavalt keelespetsiifiline, siis on ta ka üpris kallis. Kui suure kõnelejate arvuga rakenduste arendust toetab nõudlus, siis eesti keele puhul seda loota ei saa. Allika sõnul tasub majanduslikult keeletehnoloogia areng ära alates 10 miljonist kõnelejast. Et eesti keele keeletehnoloogia oleks jätkusuutlik, oleks vaja, et selle arengut ja uurimist toetaks riik. 2004. aastal otsustas Vabariigi Valitsus heaks kiita strateegia “Eesti keele arendamise strateegia (2004-2010)”, kus on eraldi peatükk ka keeletehnoloogial. Selle põhjal käivitati ka 2006. aastal Haridus- ja Teadusministeeriumi haldusalas programm “Eesti keele keeletehnoloogiline tugi (2006-2010)”. Antud programmi ülesandeks oli uurida ja arendada eesti keele arvutitöötluseks sobivaid tehnoloogilisi lahendusi ning luua uuringuteks ja tehnoloogiaarenduseks vajalikke keeleressursse [7]. Tänasel päeval on samuti riik toetamas keeletehnoloogiat ja täiesti uue programmiga, mis kannab nime “Eesti keeletehnoloogia 2018-2027” [8]. Riiklik programm toetab keeletehnoloogiaalast teadus- ja arendustegevust, et luua uusi eesti keele keeletehnoloogilisi rakendusi, tõsta olemasolevate rakenduste kvaliteeti ning võtta neid kasutusele võimalikult paljudes valdkondades nii era-, avalikus kui ka kolmandas sektoris. Allika järgi on programmi eesmärk, et eesti keele keeletehnoloogia põhikomponendid oleks

rahvusvahelises võrdluses heal tasemel ja eesti keeletehnoloogiat kasutaks lai sihtgrupp. Programmi eesmärkide saavutamist toetatakse kolme meetme abil [8]:

1. baastehnoloogiate ja -ressursside arendamine, mida ei ole veel rakendatud mingi süsteemi osana ja mille tulemus on terviklik tarkvaramoodul või süstematiseeritud keeleressurss
2. keeletehnoloogia kasutuselevõtt, mis hõlmab tehnoloogia rakendamist mingi süsteemi osana või iseseisva rakenduse loomist ja mille tulemus on keeletehnoloogia toimimine mingi süsteemi osana või keeletehnoloogilise fookusega rakenduse valmimine
3. Eesti Keeleressursside Keskuse (EKRK) kui keeletehnoloogia keskse taristu ja infovahendaja toetamine ning rahvusvahelise koostöö edendamine

Vainu [9] uuris 2020. aasta märtsis erinevaid eestikeelseid tekst-kõneks süsteeme. Tollal oli olemas Eesti Keele Instituudi poolt neli erinevat süntesaatorit – DNN, Ossian, üksuste valikul põhinev süntees ja ka HTS, millele lisaks testiti ka Tartu Ülikooli Neurokõne ja Google'i pilve süntesaatorit. Kahjuks oli allika sõnul kõigis neis erinevaid puudusi: DNNi ja Ossiani puhul olid probleemiks numbrid, nt "100. aastapäev" loeti kui "üks null null aastapäev". Lisaks toodi välja, et tegu on aeglase süntesaatoriga, seega sai kannatada kasutajamugavus. HTSi puhul need vead olid parandatud, kuid siiski on võimalik allika sõnul parandusi teha. Üksuste valikul põhineval sünteesil oli probleemkohaks samuti sünteesimise aeg. See-eest numbritega sai hästi hakkama, aga kahjuks polnud häälekvaliteet ega tempo sujuvad. Allika väitel ei saa Google'i taset ja finantsilist võimekust Eesti süntesaatorite omadega võrrelda, kuid kahjuks pole seal hetkel veel eesti keele tuge. Katsetati soome keelega ja tulemused olid väga head, kuid arusaadavalt soomepärased. Viimasena testiti Tartu Ülikooli Neurokõne, mille tulemused olid head. Ainukeseks veaks loeti teksti pikkuse piiramist, sest töö kirjutamise hetkel saab sünteesida ainult kuni 200 sümboli pikkuseid tekste. Allika väitel on viimane kasutamiseks ka kõige parem variant, sest seda arendatakse edasi ja on teistega võrreldes kiire. Eelnevatele lisaks on kood mõeldud avalikuks kasutamiseks ja saab vajaduse korral teha erinevaid muudatusi.

1.4 Tänapäev ja tulevik

Tehnoloogiaplatvorm Euphony [4] oskas ennustada juba paar aastat tagasi, et kõnesünteesitehnoloogia hakkab meie eludes mängima suurt rolli. Tuuakse välja kolm peamist fookusala, mis viivad kõnesünteesi uuele tasemele: närvivõrgud, häälekogemus (VX ehk ingl *voice experience*) ja emotsionaalne kõnesüntees. Esimese puhul toob allikas näiteks Google'i arendatud WaveNeti, mis sõltub tuhandetest näidetest ja sünteesitud kõne on suurepärane, kuid see-eest nõuab väga suurt arvutuslikku jõudlust. Siiski suudeti järjekordselt illustreerida potentsiaali, kuhu võime närvivõrkudega jõuda. VX'i all mõeldakse terviklikku kogemust, kus inimene saab oma häälega suhelda inimeste, andmete ja tarkvaraga. Suured tehnoloogiagigandid Amazon ja Apple on suunanud oma fookuse just antud valdkonna peale, sest selle abil on võimalik suhelda otse oma klientidega. Hääl selle vahendamiseks on kõige loomulikum ja ka kiirem võimalus, sest igapäevaelus moodustab hääl suure osa. Viimasena mainib allikas emotsionaalse kõne, sest igal inimesel on täiesti eriline hääl ja ta oskab sellega väljendada oma tuju, suhtumist ning lausa isiksust. Ennustatakse, et enamikul süntesaatoritest tekib oskus analüüsida emotsiooni, mille pealt oskab ta enda tooni ja suhtumist muuta. Skinner [5] toob välja, et kasutusalasid on hetkel väga mitmeid: võimalus aidata inimesi õpiraskustega, nt inimesi, kellel on raskusi lugemisega; aidata inimesi nägemispuudega, kellel pole võimalust lugeda; aidata rääkida inimestel, kellel seda võimalust pole või on see võimalus raskendatud, nt tummad ja

viimasena saab näiteks kasutada ka hooldekodudes, kus seal viibivad inimesed võivad olla üksildased ja inimhäälega suhtlemine võiks tekitada parema tunde. Allikas väidab, et kõnesünteesi areng muudab suhtlust kui vormi ja kuhu me lõpuks jõuame, seda oskab näidata ainult aeg.

2. Peamised sünteesimudelid

Tekst-kõne sünteesi eesmärk on teisendada tekst võimalikult loomuliku kõlaga kõneks. Meister jt. [7] sõnul on selle jaoks vajalikud järgnevad etapid:

- Lingvistiline eeltöötlus: teisendatakse kirjasolev tekst häädustekstiks, mille raskusaste sõltub suuresti keelest.
- Kõneprosoodia modelleerimine: arvutatakse häälikude kestused ja meloodiakontuur vastavalt lausetüübile.
- Kõnesignaali tekitamine

Kõnesignaali on võimalik tekitada mitmeid erinevaid meetodeid pidi. Igaühel neist on nii positiivseid külgi kui ka probleemseid kohti. Mihkla [10] jagab need kolme peamisse gruppi:

- Artikulatoorne süntees
- Reegelsüntees (formantsüntees)
- Ahelsüntees (kompilatiivne süntees)

2.1 Artikulatoorne süntees

Krögeri [11] sõnul üritab artikulatoorne süntees elektrooniliselt modelleerida inimese hääleorganeid nii perfektselt kui võimalik, baseerudes kõneproduktsiooni füsioloogilisel kirjeldusel. Seetõttu on antud mudelil tohutu potentsiaal toota võimalikult ideaalilähedast kõnet. Teisest küljest on ta ka kõige keerulisem implementeerida, sest ta nõuab väga palju arvutuslikku jõudlust ja on seetõttu aeglane. Sama väidab ka Mihkla [10], et tegu on liialt töömahuka meetodiga arvutuslikus mõttes, mistõttu teda kasutatakse ainult uurimiseks ja reaalajas on tema kasutamine võimatu. Meister jt [7] samuti kinnitavad, et praktiliseks rakenduseks antud meetodit kasutada ei saa, sest sünteesitava häälelaine arvutamine pole suvalise teksti puhul reaalajas võimalik.

2.2 Reegelsüntees

Meister jt [7] sõnul tugineb formantmudel e. reegelmudel kõnesignaali akustilis-foneetilisele kirjeldusele. Baasmudel koosneb allikast ja filtrist, kus allikas modelleerib häälekurdude võnkumist ja filter kõnetrakti formante ehk resonantssagedusi. Mõlemat juhib väga suur hulk keelespetsiifilisi reegleid. Mihkla [10] kinnitab, et reegelsüntees põhineb reeglitel, kus on kirjeldatud häälikute mõju üksteisele. Läbi ajaloo on reegelsünteesi ehitatud põhiliselt formantsüntesaatorina [12]. Kõne kirjeldamiseks kasutatakse kuni 60 parameetrit, mis muutuvad ja mis on seotud formantide sageduste, nende ribalaiuste ja kõri lainekuju kirjeldusega.

Formantsüntees oli peamiseks meetodiks 1970. aastatel, kui loodi hulgaliselt rakendusi eri keeltele, s.h ka eesti keelele. Hilisemalt on mudelit kasutatud pigem uurimistöödes või piiratud arvutusvõimsusega seadmetes, näiteks mobiiltelefonides [7].

2.3 Ahelsüntees

Ahelsüntees põhineb tavakõnest välja lõigatud lõikude, näiteks difoonidele (kahe hääliku ühenditele), trifoonidele (kolme hääliku ühenditele) või silpide jms sobival ühendamisel [7]. Selle jaoks on allika järgi vaja koostada kõnesegmentidest koosnev andmebaas, kus sünteesi käigus valitakse vastavad segmendid. Spetsiaalne algoritm ühendab segmendid ühtseks lauseks ja tulemuseks on kõrgekvaliteediline sünteeskõne.

Ahelsünteesi eelduseks on see, et kõne ei ole lihtsalt reas olevad häälikud - pigem koosneb kõne häälikute üleminekutest [10]. Difoonidest koosnevad segmendid on ahelsünteesis kõige populaarsemad kõneühikud, sest neid on hästi vähe vaja, kui soovida suvalise teksti alusel kõne sünteesida. Eesti keele andmebaas sisaldab näiteks 1700 difooni [7]. Ahelsüntees on olnud lähiajaloos üks populaarsemaid mudeleid [13].

2.4 Markovi peitmudelid

Markovi peitmudelid (ingl *Hidden Markov Models*, HMM*)* on tõenäosuslikud olekuautomaadid, mida käsitletakse kui Markovi protsessi [14]. Markovi protsess on protsess, mis koosneb reast olekutest, kusjuures ühest olekust teise siirdumise tõenäosus sõltub ainult neist kahest olekust, mitte aga eelmistest olekutest. Kõnesünteesivaldkonnas on HMMe kasutatud valdava eduga. Esimesed kirjed mudelitest on 1960. aastatest, kuid praktikas alustati nende kasutamisega alles 1980. aastate lõpus. Tänapäeval kasutatakse neid mitmes eri valdkonnas ja populaarsus aina kasvab. Allikas jätkab, et HMMi võib kirjeldada lõpliku olekumasinana, mis genereerib vaatluste jada. Vaatluse genereerimiseks on vaja alguses teha otsus, millisesse olekusse edasi liikuda, ja peale seda genereeritakse järgmine vaatlus vastavalt hetkeoleku tihedusfunktsiooni tõenäosusele. Sabaliuskas [15] toob keerulisele selgitusele lihtsa näite: HMMil on kaks olekut – peidetud ja vaadeldav e. vaatlus. Peidetud olek on olek, mida eraldi ei jälgita, kuid ta on tõenäosuslikult tuletatavad vaadeldava oleku põhjal. Allikas toob näite elust enesest: inimesel ei ole võimalik teada, mis tujus ta elukaaslane on, kuid ta suudab selle tuletada tema käitumisest. Seega tuju on siin peidetud olek ja käitumine on vaadeldav olek, mille pealt võib teha ennustuse, mis tujus on elukaaslane. Sama loogikat kasutavad ka kõnesünteesi mudelid HMMi abil, kuid mõistetavalt on valemid hulgaliselt keerulisemad. Kõnesünteesis HMMi kasutamise protsess on järgnev: esmalt võetakse välja kõne parameetriline kujutus, millele rakendatakse filtrit ja põhiparameetreid kõneandmebaasist [16]. Peale seda modelleeritakse kõik kokku, kasutades alamsõnade kogumit. Allika järgi arvutatakse kõige tõenäolisemad kõne parameetrid järgmisena ja konstrueeritakse kõne lainekuju. HMMi mudelite tugevuseks on väikse andmebaasi vajalikkus, seetõttu on võimalik mudelitega ehitada mitut erinevat kõnetüüpi väga kiirelt, väidab allikas.

2.5 Tehisnärvivõrgud

Tehisnärvivõrgud on masinõppel põhinevad arvutussüsteemid, mis põhinevad andmepaaridel ja neile kaasa antud klassist [17]. Peale mudeli treenimist antakse närvivõrgule ette ainult andmete osa ilma vastava klassita. Tekst-kõneks sünteesi puhul on sisendiks tekst ja väljundiks helilaine. Närvivõrgud põhinevad pertseptronitel, millel on mitu sisendit. Nendele sisenditele antakse juhuslikud kaalud, mis omakorda summeeritakse ja mis omakorda läheb läbi aktiveerimisfunktsiooni, mis muudab kaale. Treenimisel võrreldakse mudelist tulnud väljundit kaasa antud andmestiku klassiga. Kui mudel on piisavalt osav, saab teda lõpuks kasutada reaalsete andmete peal. Mitmeid närvivõrke kasutavaid süsteeme on töö kirjutamise hetkel juba valmis, mõnedeks näideteks on Wavenet, SampleRNN ja Char2Wav [17].

3. Kõnesünteesi probleemid

Töö kirjutamise hetekl on kõne sünteesimisega seotud mitmeid probleeme [3, 11]:

- Teksti eeltöötlus ja prosoodia ehk tekstile kõne sünteesimine, probleeme tekitab tõik, et tekst ei edasta mingit infot sellest, kuidas teda peaks ette lugema ja/või hääldama
- Kõne loomulikkus
- Emotsioon – süntesaatorid tunduvad enamasti liialt monotoonsena

Huvitaval kombel on keerulisem sünteesida nais- ja lapshäälega kõnet, kuna nende hääletooni kõrgus on kaks-kolm korda kõrgem kui meeste oma [18]. Selle põhjuseks on formantsageduste lokaliseerimise hindamise keerukus kõrgemate sageduste korral. Probleemidest pikema ülevaate leiab alampeatükkidest 3.1, 3.2 ja 3.3.

3.1 Prosoodia

Prosoodia on üks võtmekohti kõnesünteesis, selle abil on võimalik rakendada emotsiooni ja foneetilisi aspekte [18]. Prosoodia otsustab, kuidas lauset hääldatakse ja millise tempo ning tooniga see on. Juurprobleemiks on sünteesimine kirja pandud tekstist, sest tekst ei edastada mingit infot kõneprosoodiast [13]. Seega rütmi, meloodia ja rõhuasetused peab algoritm ise sünteesima. Mõnel juhul erineb kõne tekstist drastiliselt ja kui sünteesitud kõnes puuduvad hingetõmbepausid või asuvad need vales kohas, kõlab sünteesitud kõne väga ebaloomulikuna või halvimal juhul kaotab sünteesitud lause oma mõtte. Murekohaks on ka lausete omavaheline seos – inimkõnes on kõneprosoodia seotud varasemalt öeldud lausetega [18]. Viimaste aastatega on selles vallas tehtud suuri samme edasi, kuid sünteesitav kõne ei ole siiski piisavalt sujuv ja on kohati liialt monotoonne. Allikas oskab veel välja tuua probleemi erinevate aktsentidega, kuid selles vallas ei ole veel ühtegi teaduslikku uuringut tehtud.

3.2 Loomulikkus

Kuliowska [18] jt väidavad, et üks tähtsamaid muresid on ehitada süsteem, mille sünteesitav kõne oleks ka naturaalne. Kuigi viimasel ajal sünteesitud kõne tulemus ei ole enam “robotlik”, vaid meenutab vägagi inimese häält, on siiski murekohti, mis vähendavad üldmuljet. Näiteks HMMi kasutavatel sünteesimudelitel on taustal kuulda häirivat müra.

3.3 Emotsioonid

Emotsioon on seotud kõne usutavusega [18]. Kui süsteem saab aru, et kasutaja on kuri või pahane, siis ta saaks automaatselt muuta oma tooni ja kasutaja poole pöördumist. Selleks, et kõnesüntesaator oskaks emotsiooni näidata, on vaja allika järgi lahendada probleem: tekstist peab mõistma emotsiooni. Teine probleem on emotsiooni hindamises – nimelt ei ole eriti head meetodit emotsiooni hindamiseks, näiteks saab kõnetuvastuses hinnata vigade arvu, kuid sarnast head mõõdikut emotsiooni hindamiseks ei leidu. Lisaks on emotsioon väga subjektiivne ja võib olla iga rääkija jaoks erinev, kuna see omakorda sõltub veel rääkija tujust, arvamustest ja kultuurilisest taustast [18]. Kaasatundvam emotsioon, näiteks mure korral kõnekeskusesse helistades, võiks anda positiivsema kogemuse Elisa Eesti kliendile.

4. Kasutatud tehnoloogia

Käesoleva töö rakendus koosneb kasutajaliidesest, mis peab suhtlema kõnesünteesi pakkuva liidesega ja viisi, kuidas rakendust ehitada ning kättesaadavaks teha. Valik langes kasutajaliidese ehitamise puhul Angularile, mida kirjeldatakse pikemalt alampeatükis 4.1. Rakenduse ehitamiseks kasutatakse Dockerit, mille kohta on võimalik lugeda alampeatükis 4.2. Sünteesitud helifail saadetakse Google'i pilve, mille tugevustest saab lugeda alampeatükis 4.3.

4.1 Angular

Angular on TypeScriptil põhinev avatud raamistik, millega saab ehitada kliendipoolseid veebirakendusi. TypeScript on JavaScripti ülemhulk ja viimast kasutatakse veebilehe disainimise ja sündmuste loogika ehitamiseks [19]. Tavaliselt kasutatakse Javascripti, et luua kasutajale interaktsioone ja komponente. See tegeleb ka kogu dünaamilise sisuga, mille all mõeldakse pidevas muutumises olevat sisu ja selle kohanemist. Näiteks saab JavaScriptiga teha kindlaks, kas veebisaiti kasutab mobiiltelefon või mõni muu seade. Tänu tuvastusele on võimalik näiteks kohandada ja kuvada veebilehti erinevalt, vastavalt kasutaja seadmele. See aga viis arendajaid tegema oma teeke, mis aitasid hoida koodi lühemana ja keeruliste protsesside implementeerimise viia lihtsamaks. Sealt kasvas välja teek nimega jQuery, mis on väga võimas ja populaarne, kuid mille miinuseks on struktuuripuudus [19]. Ilma struktuurita võib minna suurte projektide puhul koodi haldamine väga keeruliseks. Seda silmas pidades loodigi Angular, mis on kliendipoolne JavaScripti raamistik, mis aitaks luua SPA-sid (ingl *single page application*), kasutades veebiarenduse kõige paremaid tavasid. Angulariga tuleb kaasa struktureeritud keskkond, tänu millele on koodi haldamine ülimalt mugav ja lihtne, sõnab allikas. Käesoleva töö kasutajaliides on kirjutatud Angulariga, sest töö autor soovis õppida midagi uut ja koodi täiendamisel on Angulariga lihtne struktuuri säilitada.

4.2 Docker

Docker on tarkvaraplatvorm, mis on mõeldud rakenduste ehitamiseks, mis põhinevad konteineritel [20]. Konteinerid on väiksed ja lihtsad rakenduskeskkonnad, mis kasutavad operatsioonisüsteemituuma, kuid töötavad üksteisest eraldatuna. Konteinereid on kasutatud küll juba varem, kuid Docker alustas alles aastal 2013 ja peale mida on läinud konteinerite kasutamine väga populaarseks. Allika sõnul on kaasaegse tarkvaraarenduse üks eesmärk hoida ühes *host*is kasutatavaid rakendusi üksteisest isoleerituna, et nad ei segaks üksteist põhjendamatult. See aga on keeruline, kuna pakettide, teekide ja erinevate komponentide koosmõjul võib see siiski tahtmatult juhtuda. Üks lahendus on kasutada virtuaalmasinaid, mis hoiab programme samal riistvaral täiesti eraldi ja vähendab omavahelisi konflikte, kuid nende kasutamine on väga mahukas: igaüks vajab enda operatsioonisüsteemi ja seetõttu gigabaitides mälu. Lisaks on neid ka keeruline ja tüütu hooldada ning uuendada. Konteinerite suurused on samas megabaitides ja konteinerid kasutavad kordades vähem ressurssi, kui virtuaalmasinad seda teevad. Konteinerid pakuvad väga efektiivset mehhanismi, mille abil saab kombineerida erinevaid komponente aplikatsiooni ja teenusesse, mida on vaja tänapäeva arenduses ja mida on lisaks ka väga lihtne uuendada ning ka hooldada. Käesolev töö kasutab Docker'it, et viia rakendus kasutamiseks veebiserverisse. Sarnaselt eelnevale alampeatükile oli soov autoril õppida uut keskkonda, mis on populaarne ja võib professionaalses karjääris kasulikuks tulla.

4.3 Google Cloud Platform

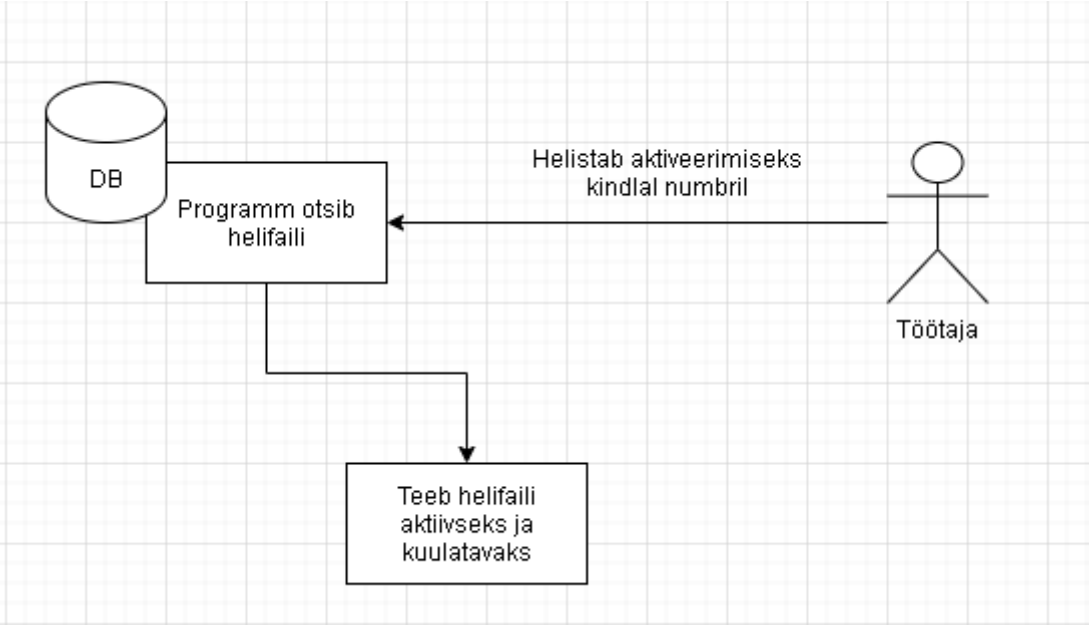
Tänasel päeval kasutatakse peamiselt pilvteenuseid. Pilvteenuse ehk pilvandmetöötluse all mõeldakse majutatud teenuste pakkumist, mis tähendab, et serverid ja andmed asuvad eemal lõppkasutajast. Nii on võimalik pakkuda kiirelt ja mugavalt suuremahulist aruvutusvõimsust. Kasutamiseks peab kasutaja sisenema pilvepõhisesse teenusesse läbi rakenduse või veebibrauseri ja sealt pääsetakse oma andmetele ligi. SADA Systems'i läbi viidud uuringust selgub, et 84% IT juhtidest kasutavad oma lahendustes juba mingit pilvandmetöötlust [21]. Nendest omakorda pooled eelistavad Google'i pilvteenust nimega *Cloud Platform*. Google'i teenustel on mitmeid eeliseid, allika sõnul on nad esiteks odavamad kui konkurendid. Neile ei maksta igakuiseid tasusid, vaid arve esitatakse selle eest, kui palju vastavat teenust kasutati. See on kliendisõbralik lähenemine testfaasis olevatele rakendustele, nagu ka käesoleva lõputöö rakendus töö kirjutamise hetkel on. Allika järgi on Google'i eeliseks ka nende suurepärane internetivõrk. Nad on viinud oma enda kaabelinterneti igale poole maailmas, mistõttu suudavad pakkuda erakordset kiirust. Google'i pilvteenused on lisaks veel turvalised: Google'i serverite ja kliendi vahelises suhtluses krüpteeritakse iga fail ja et üldse andmeid näha, peavad olema kasutajal vastavad õigused. Jackson [21] väidab, et Google suudab varundada andmeid 99.9999% tõenäosusega, mistõttu on tegu ka väga kindla ja vastupidava platvormiga. Odavuse, kiiruse ja töökindluse tõttu otsustas autor koos ettevõttega kasutada Google'i pilvteenuseid helifailide salvestamiseks. Autor sai ka uurida teisi lahendusi, nt Amazoni veebiteenuseid. Amazoni veebiteenuste peamiseks nõrkuseks oli tema hind, mis oli ca 3-4 korda kallima kuuhinnaga kui Google'i teenused. Muus osas on tegu võrdsete suurfirmadega ja kahtlemata on ka Amazoni pakutud teenus väga kvaliteetne.

5. Valminud lahendus

Käesolevas peatükis tutvustatakse käesoleva lõputöö rakendusele eelnevat protsessi, sellega seotud probleeme ja uue protsessi loogikat. Tutvustatakse valminud kasutajaliidest ja selle kasutamist ning selgitatakse rakendusega seotud andmebaasi haldust. Lõpetuseks pakutakse ideid ka edasiarendusteks. Valminud lahendust on võimalik lokaalseks kasutamiseks leida GitHubi lingilt https://github.com/maaksuhtemaak/konesyntees-angular.

5.1 Vana protsess

Varasemalt pidi töötaja *prompti* ehk eelsalvestatud helifaili aktiveerimiseks helistama kindla kombinatsiooniga numbril, üldist protsessi kirjeldab joonis 1. Igal numbril oli kindel teenus, millega helifail oli seotud, ja teatud kindel ettesalvestatud ning näitleja poolt sisseloetud tekstiga helifail. Peale töötaja poolt tulnud kõnet otsis programm üles helifaili vastavast andmebaasist (joonisel 1 “DB”) ja tegi selle aktiivseks. See tähendab, et kui ettevõttele helistas klient, kes oli helifailis märgitud teenuse klient, esitati talle leitud helifail automaatselt. Aktiivse helifaili esitamise lõpetamine käis samamoodi: tuli helistada teatud koodiga kindlal numbril ja programm tühistas faili esitamise.



Joonis 1. Varasem ettevõttes kasutatav protsess

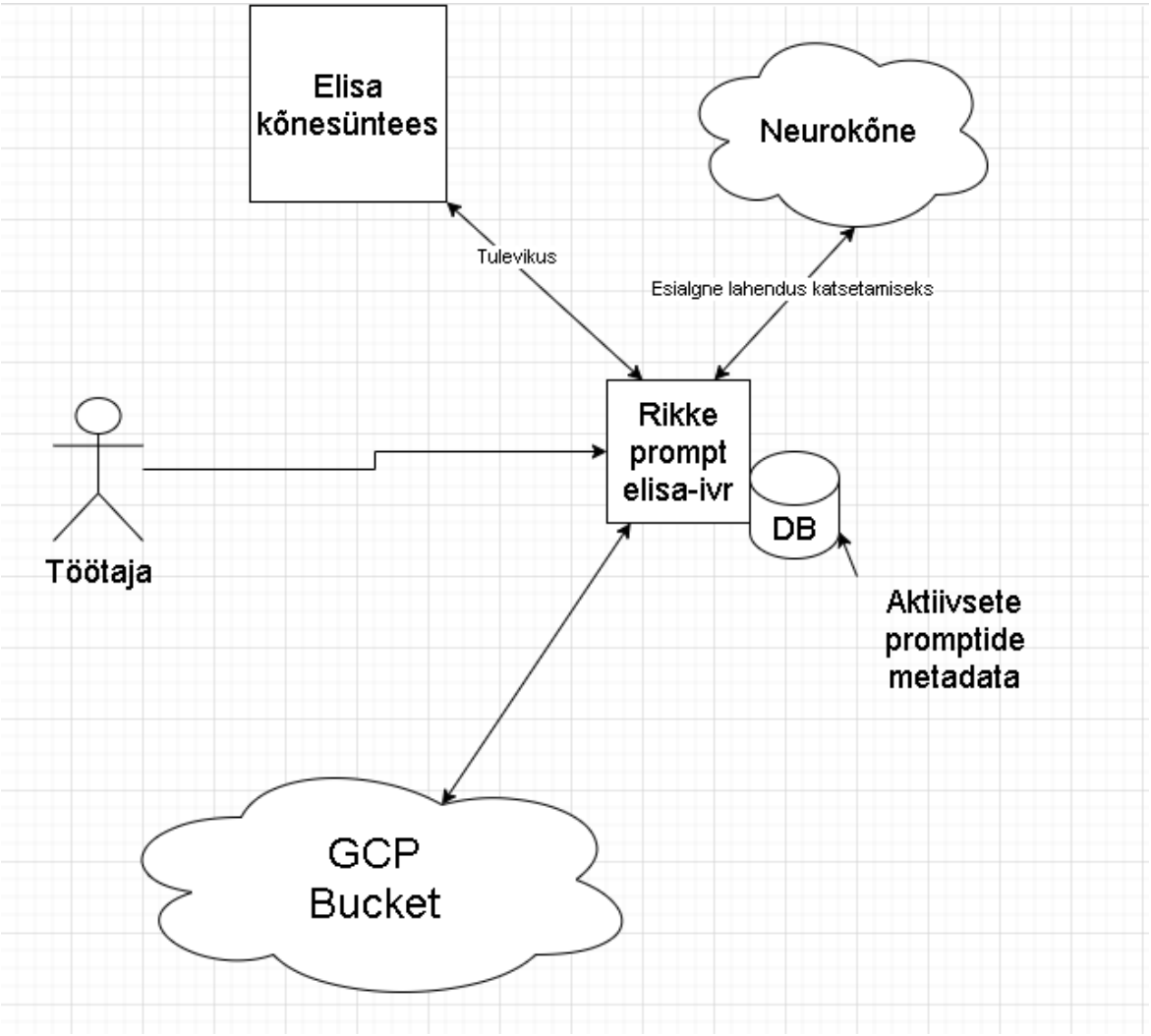
Protsess on töötajale ebamugav, sest koodi valesti sisestamisel peab tegema vähemalt kaks lisakõnet: ühe tühistamiseks ja teise õigele teenusele faili lisamiseks, mis omakorda võtab aega. Lisaks polnud mugavat kohta, kust kontrollida, millised helifailid on aktiivsed, vaid seda pidi eraldi pärima andmebaasist. Ettevõtte jaoks oli puuduseks uute helifailide lisamise aeglus ja hind, sest igaks uueks helifailiks on vaja tellida pealelugemiseks inimene ja peale salvestust on vaja teha ka järeltöötlust. Kogu protsess võis võtta aega nädalaid. Kuna selline lahendus on ettevõttele ebamugav, ei olnud võimalik ega finantsiliselt mõttekas salvestada palju eridetailsusega faile, vaid salvestati kõige üldisemad ja vajalikumad helifailid, mida oleks võimalik vastavas olukorras ette mängida.

Nende puuduste kõrvaldamiseks oli rakenduse nõueteks teha kasutajaliides, kust on mugav valida teenus, millele helifaili esitada, ja mugav soovitud teksti helifailiks sünteesida. Lisaks pidi kasutajaliidesest näha olema, millised häälviibad on hetkel aktiivsed, ja võimalus neid

sealt eemaldada. Seetõttu oli nõudeks ka tekitada vastav andmebaas, kus hoida helifailide metaandmeid.

5.2 Uus protsess

Seetõttu oli vaja mõelda uus mugavam ja kiirem protsess, mis sobiks rakendusega kokku ja oleks parem kui eelmine (vt. joonist 2). Üldjoontes jäi idee samaks: töötaja kasutab kasutajaliidest helistamise asemel. Sealt on võimalik valida teenus, millele häälviipa mängida, ja lisada soovitud tekst, mida soovitakse esitada.



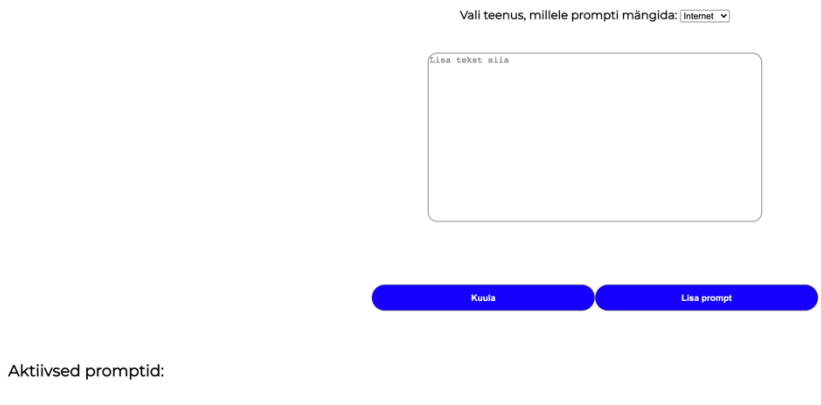
Joonis 2. Rakenduse uue protsessi kujutis

Protsessi alustab töötaja, kes tahab lisada uut helifaili ja seetõttu kasutab ta rakendust (joonisel 2 “rikke prompt-elisa-ivr”). Seal valib ta teenuse ja lisab teksti, mida sünteesida. Edasi sünteesitakse tekst helifailiks Tartu Ülikooli kõnesünteesiliidesega (joonisel 2 “Neurokõne”), mille metaandmed salvestatakse andmebaasi (joonisel 2 “DB”) ja salvestatakse helifail pilve (joonisel 2 “GCP Bucket”). Edasi päritakse pilvelt helifaile – kui on lisandunud uued, siis laetakse need koheselt alla, et oleks võimalik koheselt neid ka esitada. Kuna süntees ise ja allalaadimine võtavad pisut aega (ca 10 sekundit), muudetakse helifail aktiivseks alles siis, kui ta on alla laetud. Kui klient helistab ettevõttele ja omab mingit teenust, millele on aktiivne häälviip koos allalaetud helifailiga olemas, esitatakse talle kõigepealt helifail. Kui klient selle peale kõnet ei lõpeta, liigub kõne edasi ettevõtte töötajale, kes kliendiga edasi tegeleb. Hetkel katsetatakse lahendust Tartu Ülikooli

kõnesünteesiliidesega, kuid tulevikus soovib ettevõte teha oma kõnesünteesi (joonisel 2 “Elisa kõnesüntees”).

5.3 Kasutajaliides

Kasutajaliides koosneb teenuse valikust, soovitud teksti lisamise võimalusest, kahest nupust (kuulamiseks ja edastamiseks) ja aktiivsetete *promptide* loetelust, et oleks võimalik näha, millised on juba aktiivsed. Kasutajaliidest on võimalik näha joonisel 3.



Joonis 3. Valminud kasutajaliides

Kasutajaliidese päises on tekst “Vali teenus, millele prompti mängida:”, millele järgneb rippmenüü, kus on valikuvõimalus teenustest, millele on võimalik sünteesitud kõnet ette mängida. Sellele järgneb suurem tekstikast, kuhu on võimalik soovitud sisend kirjutada, ja kaks nuppu tekstiga “Kuula” ning “Lisa prompt”. Kõige põhjas on tekst “Aktiivsed promptid:”, millele järgneb ülevaade hetkel aktiivsetest helifailidest (joonisel 3 aktiivsena ühtegi pole).

5.4 Rakenduse kasutamine

Rakenduse kasutamiseks on peamised osad tekstikast, mis on mõeldud soovitud teksti lisamiseks, ja nupud. Nupuga „Kuula“ on võimalik soovitud teksti sünteesitud tulemust kuulata ja veenduda, et see on sobilik ning piisavalt hea kvaliteediga. Nupuga „Lisa prompt“ on võimalik kohe teha *prompt* aktiivseks ja saata info andmebaasi, mille järel tekib ta alla loetellu (vt. joonis 4).

Joonis 4. Kasutajaliidese vaade peale andmete sisestamist

Peale sisendi lisamist tekib ka uus nupp „eemalda“, mis eemaldab vastava *prompti* aktiivsete hulgast ja teda enam ei esitata. Andmebaasi jääb ta info alles, et oleks võimalik hilisemalt tuvastada, miks midagi lisati ja kas üldse lisati. Andmebaasis on info teenuse, teksti ja lisamis- ning eemaldamisaja kohta. Lisaja kohta pole hetkel infot vaja hoida, sest õiguste hulk on töö kirjutamise hetkel väga väiksel inimgrupil.

5.5 Andmebaasi haldus

Helifailide metaandmete hoidmiseks pidi looma uue andmebaasi nimega “Prompts” ja lisama sinna uue tabeli nimega “Prompts” (tabeli näidisisu demonsteerib joonis 5). Tabeli ülesehitus on järgmine: id-nimeline veerg, mis sisaldab automaatselt genereeritud täisarve ja on ka tabeli võtmeks. Peale seda on sõne-tüüpi veerg “randomId”, mis tähistab helifaili nime pilves ja on tabelis unikaalne. Tabelis on veel sõne-tüüpi veerg “service” ehk tähistab teenuse nime, millega sisend seotakse ja ka prompt-nimeline veerg, mis sisaldab endas teksti, mida kliendile ette mängitakse. Lõpetuseks on veel kaks veergu: “created” ehk kuupäev, mis on vaikeväärtusega hetk, millal häälviip lisatakse ja veerg “removed”, mis on vaikimisi tühi, aga saab omale väärtuse, kui ta aktiivsete *promptide* hulgast eemaldatakse.

id	randomid	service	prompt	created	removed
1	_y7k9j2n	Internet	See on test-prompt internetiteenuse rikke jaoks	2021-04-24 15:10:01	NULL
2	_9veirge	Mobiil-ID	See on test-prompt Mobiil-ID teenuse rikke jaoks	2021-04-24 15:10:20	NULL

Joonis 5. Andmebaasi *select*-päringu näidistulemus tabelist “prompts”

5.6 Rakenduse tagasiside

Suures pildis toimis lahendus nii nagu plaanitud: oli mugav, kiire ja lihtne käsitleda. Tagasiside andsid neli inimest, kes olid töö kirjutamise hetkel sama projekti teiste otstega seotud. Tagasiside andmine toimus koosoleku vormis, kus kõik said küsida ja avaldada arvamust. Peamiselt oli küsija rollis töö autor, kes küsis iga funktsiooni kohta eraldi, et mis meeldis ja mis ei meeldinud ning ideid parandusteks.

Oldi rahul teenuste valiku lihtsusega ja võimalusega kuulata üle, kuidas kõnesüntees toimib. Hästi õnnestus ka disain, mille osas oli autoril kõige rohkem kahtlusi – tagasisideks mainiti, et on arusaadavalt ülesehitatud ja silmale ilus.

Töötajatele meeldis ka kasutajasõbralikkus, näiteks kinnituste küsimised. Selle all mõeldakse olukordi, kus töötaja tahtmatult vajutab eemaldamiseks mõeldud nuppu. Siis avaneb kinnitus, et kas ta soovis seda siiski teha. Sarnaselt tuleb ka teavitus, kui kogemata soovitakse lisada tühja sisuga helifaili.

Siiski leiti ka kohti, mida saaks parandada. Kasutajaliideses pole hetkel võimalust näha eemaldatud helifaile ja nende sisu. Sooviti lisada võimalus, kus saab eemaldatud helifaili uuesti aktiivseks panna. See lihtsustaks töötajate tööd ja annaks ka ülevaate, millised on olnud varem kasutatud helifailid. See on kasulik näiteks olukorras, kus uus töötaja peab lisama mingi uue faili. Siis on võimalus näha teisi näiteid ja võtta need eeskujuks. Teisest küljest saaks mugavalt panna varasemalt sünteesitud helifaili uuesti aktiivseks, mis hoiaks kokku aega ja ka pilvel ruumi.

Lisaks ei oldud mõnel juhul rahul sünteesitulemusega: töötajate väitel lõikas süntees mingi osa lausest ära või hääldas kohanime valesti. Töötajaid häiris ka taustale tekkiv müra ja monotoonsus.

See-eest peeti sünteesitulemust arusaadavaks ja mõisteti, et alati saab parema lause välja mõelda, mille sünteesitulemusega ollakse ise rohkem rahul. Muu osas kommentaare ei tehtud.

5.7 Edasiarenduse võimalused

Käesoleva bakalaureuse töö jaoks valminud rakendus pole kaugeltki täiuslik ja on võimalik edasi arendada erinevates suundades. Ühe võimalusena saab programmeerida ja täiendada olemasolevat lahendust või ehitada kõnesünteesiliides eraldi serverisse või lausa muule keelele. Teisest küljest on tegemata ka põhjalikum järelanalüüs.

Esimesena tuleks täiendada valmislahendust ja lisada juurde funktsioonid, mille puuduse tõid tagasisideks töötajad, kelle käest autor tagasisidet küsis.

Teisest küljest kasutatakse hetkelahenduse puhul Tartu Ülikooli kõnesünteesiliidest, mistõttu pole teenuse haldamine ettevõtte käsitleda. See on ebamugav, sest kui teenus peaks mingil põhjusel ära kaduma, pole võimalik enam helifaile esitada. Sellele on kaks võimalikku lahendust: esimesena võiks viia kõnesünteesiliidese oma serveritesse. See on võimalik, kuna liidese kood on avalik. Teine lahendus oleks ehitada oma sünteesimudel, mis on aga väga mahukas ja keeruline töö.

Ettevõtte jaoks on tähtsad kõik kliendid ja tänapäevases globaalses maailmas ei piisa ainult eesti keelele ehitatud lahendusest. Seetõttu on võimalik töö edasiarenduseks luua sarnane lahendus ka muudele keeltele. Seeläbi saaks lahendust kasutada suuremal osal klientidest.

Lisaks pole rakendusele süvitsi analüüsi tehtud: tuleks kaardistada, kuidas antud tööriist meeldib ettevõtte töötajatele ja mida saaks või lausa peaks parandama. Samamoodi tuleks uurida ka klientide arvamust, kellele sünteesitud vastust esitatakse. Kui kliendid pole sellise lahendusega rahul, s.t näiteks on vastus liialt arusaamatu, siis pole samuti mõtet rakendust kasutada, kui see klientides veel rohkem segadust tekitab.

Need on mõned näited, kuidas saaks valminud tööd täiendada ja rakenduse toimivust parendada.

6. Kokkuvõte

Käesoleva töö eesmärk oli teha kõnesünteesi kasutav rakendus ettevõttele Elisa Eesti AS, mille abil on võimalik sünteesida mugavalt ja kiirelt teksti kõneks. Nagu tööst selgus, oli varasem ettevõttes kasutusel olnud lahendus aeglane, jäik ja ettemängitavatel tekstidel jäi puudu detailsusest. Tänases maailmas, kus info liigub kiiresti ja seda on võimalik kätte saada pea igat kanalit pidi, peavad ka suurfirmad seda suutma pakkuda.

Töös sai antud ülevaade kõnesünteesi ajaloost nii Eestis kui mujal, erinevatest võimalikest mudelitest, nende tugevustest ja nõrkustest ning probleemidest, mis võivad sünteesimisega seotud olla. Põhjendati kasutajaliideseks vajaminevaid tehnoloogilisi vahendeid ja seletati lahti, miks oli vaja luua uus protsess. Näidati, milline on lõpptulemus ja kuidas seda kasutada.

Käesoleva bakalaureusetöö raames valmis kõnesünteesi kasutav rakendus ettevõttele Elisa Eesti AS, mille töötajal on võimalik väga lihtsalt ja kiirelt lisada klientidele ettemängimiseks detailne sünteesitud helifail. Enne rakenduse ehitamist oli tarvis välja mõelda uuem ja mugavam protsess ettevõtte töötajale. Kõnesünteesirakendus ehitati Angularis ja viidi veebiserverisse Dockeriga. Rakendus edastab helifaili pilve, mis paikneb Google'i pilvteenuses, ning salvestab edastatud faili metaandmed andmebaasi. Sellise lahendusega on võimalik hõlpsalt leida infot, millal mingi fail esitamiseks mõeldud oli. Valminud rakendusega olid nii kasutajad kui ka autor rahul, sest lahendus oli mugav, kiire ja kasutajasõbralik. Tagasisidet anti mõne funktsiooni lisamiseks ja kõnesünteesimudeli parandamiseks.

Edasiarendustena oleks võimalik täiendada olemasolevat lahendust, lisades juurde funktsioone. Teise variandina oleks võimalik viia kõnesünteesiliides ettevõtte serveritesse või lausa ehitada uus sünteesimudel. Kolmanda variandina tuleks kaaluda ka võimalust sünteesida erinevates keeltes, seega saaks juurde lisada keeli. Samuti oleks võimalus uurida töötajate ja klientide rahulolu antud rakenduse ning sünteesi tulemusega.

Lõputöö autorile andis tehtud töö väga arvestataval määral uusi teadmisi kõnesünteesi ajaloost ja kasutatavatest mudelitest, lisaks valdkonna seisust Eestis. Autor sai ka elus esimest korda täiesti algusest ehitada oma rakenduse, mistõttu kasvasid ka programmeerimisoskused ja teadlikkus erinevatest võimalikest lahendustest.

Viidatud kirjandus

[1] Schroeder M. A Brief History of Synthetic Speech. *Speech Communication vol. 13*, 1993, pp 231-237.

[2] Story B. History of Speech Synthesis. 2019, https://www.researchgate.net/publication/342693675_History_of_speech_synthesis (05.02.2021)

[3] Klatt D. Review of Text-to-Speech Conversion for English. *Journal of the Acoustical Society of America*, 1987, JASA vol. 82 (3), pp 737-793, https://doi.org/10.1121/1.395275, (08.01.2021)

[4] Euphony. The exciting future of voice synthesis technology. 10.01.2017, https://medium.com/@EuphonyInc/the-exciting-future-of-voice-synthesis-technology-6b8d05dea7fc (25.03.2021)

[5] Skinner O. Text to Speech technology: How voice computing is building a more accessible world. 09.06.2020, https://www.voices.com/blog/text-to-speech-technology/ (25.03.2021)

[6] Mihkla M. Eesti keel tehnoloogiate mõjutuses. 2009, *Õiguskeel*, *nr 4*, https://www.just.ee/sites/www.just.ee/files/meelis_mihkla._eesti_keel_tehnoloogiate_mojutuses.pdf (11.02.2021)

[7] Meister E., Alumäe T. Kuidas arvuti kuulab ja kõneleb. 2010, *Horisont*, *nr 5*, http://cs.ioc.ee/excs/popular/meister-alumae-horisont510.pdf (11.02.2021)

[8] Teadusprogrammid, Haridus- ja Teadusministeerium. 2021, https://www.hm.ee/et/tegevused/teadus/teadusprogrammid (11.02.2021)

[9] Vainu I. Current practical state of Estonian language Voice AI (speech-to-text and text-to-speech). 11.03.2020, https://alphablues.com/current-practical-state-of-estonian-language-voice-ai-speech-to-text-and-text-to-speech/ (24.03.2021)

[10] Mihkla M. Kõnesüntees? … See on imelihtne., *Oma keel*, 2008, nr 16 / kevad 2008, lk 5-15, https://www.emakeeleselts.ee/omakeel/2008_1/OK_2008-1_01.pdf (12.01.2021)

[11] Kröger B. Minimal Rules for Articulatory Speech Synthesis. *Proceedings of EUSIPCO92 (1)*, 1992, pp 331-334

[12] Mihkla M., Meister E. Eesti keele tekst-kõne-süntees., *Keel ja Kirjandus*, 2002, nr 2, pp 88–97

[13] Lemmety S. Review of Speech Synthesis Technology.

1999, http://research.spa.aalto.fi/publications/theses/lemmetty_mst/thesis.pdf (14.01.2021)

[14] Raitio T. Hidden Markov Model Based Finnish Text-to-Speech

System Utilizing Glottal Inverse Filtering. 2008, http://lib.tkk.fi/Dipl/2008/urn012274.pdf (10.02.2021)

[15] Sabaliauskas D. Hidden Markov Model (HMM) — simple explanation in high level. 06.10.2020, https://towardsdatascience.com/hidden-markov-model-hmm-simple-explanation-in-high-level-b8722fa1a0d5 (10.02.2021)

[16] Abbood N. jt. Speech synthesis using neural network. 2018, *Open Science Journal, 3 (1)*, https://doi.org/10.23954/osj.v3i1.1257 (10.02.2021)

[17] Näslund P. Artificial Neural Networks in Swedish Speech Synthesis. 2018, https://www.diva-portal.org/smash/get/diva2:1264794/FULLTEXT01.pdf (10.02.2021)

[18] Kuligowska K. jt. Speech synthesis systems: Disadvantages and limitations. 2018, *International Journal of Engineering & Technology, 7 (2), pp 234-239,* http://dx.doi.org/10.14419/ijet.v7i2.28.12933 (11.02.2021)

[19] Shubham S. Angular tutorial: Getting started with Angular 8. 25.11.2020, https://www.edureka.co/blog/angular-tutorial/ (25.03.2021)

[20] Yegulalp S. What is Docker? The spark for container revolution. 19.04.2019, https://www.infoworld.com/article/3204171/what-is-docker-the-spark-for-the-container-revolution.html (25.03.2021)

[21] Jackson B. Top 7 Advantages of choosing Google Cloud Hosting. 11.03.2021, https://kinsta.com/blog/google-cloud-hosting/ (27.04.2021)

Lisad

I. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Markus Mikko,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

   Kõnesünteesi kasutav rakendus ettevõttele Elisa Eesti AS,

   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 Commonsi litsentsiga CC BY NC ND 3.0, 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.

Markus Mikko

07.05.2021