TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Ilja Samoilov 192480IAPM

TELEVISIOONIPROGRAMMIDE AUTOMAATSETE TRANSKRIPTSIOONIDE TEISENDAMINE LOETAVATEKS SUBTIITRITEKS

Magistritöö

Juhendaja
Tanel Alumäe
PhD

Tallinn 2022

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Ilja Samoilov

Kuupäev: 10.05.2022

Annotatsioon

Magistritöö eesmärk on uurida võimalusi subtiitrite toimetamiseks kasutades loomuliku keele mudeleid. Varem on seda tehtud, kas ekstraheerivaid meetodeid kasutades või automaatset kõnetuvastust teksti toimetamisega ühendades. Kahjuks need meetodid ei arvesta tänapäevaste keelemudelite arenguga. Magistritöös kasutab autor mudelite treenimiseks transkribeeritud Eesti Rahvusringhäälingu telesaateid ja nende juurde kuuluvaid subtiitreid. Paralleelkorpuse loomiseks arendati algoritm andmete ettevalmistamiseks. Seda saab edasipidi kasutada mitte ainult eesti vaid ka teiste keelte jaoks või suurema korpuse tekitamiseks.

Kõik magistritöö raames valminud tehisnärvivõrgu mudelid suudavad toimetada subtiitreid, mis on kvaliteedi poolest ligilähedased käsitsi loodutele. Kõige parema tulemuse saavutasid keelemudelid. Magistritöö käigus loodud mudelid võivad olla suureks abiks toimetajatele. Automaatselt transkribeeritud subtiitrite toimetamisele kuluv aeg on väiksem, kuna käsitsi jäävad parandada ainult fakti- ja grammatikavead. Lisaks kui treeningandmestikus on transkribeerimisvead piisavalt sagedased, suudavad mudelid ka need automaatselt parandada.

Autor viis läbi uuringu, mille käigus selgitati välja, kas vaatajad eelistavad mudelite poolt loetavaks teisendatud subtiitreid või automaatse kõnetuvastaja poolt transkribeerituid. Uuringus osales 80 inimest. Selgus, et 63% juhtudest eelistasid vaatajad mudeli poolt teisendatud subtiitreid või ei omanud eelistust kahe valiku vahel.

Magistritöö on kirjutatud eesti keeles, sisaldab teksti 43 leheküljel, seitset peatükki, 18 joonist, 20 tabelit ja kahte lisa.

Abstract

Converting automatic transcriptions for television programs to readable subtitles

Using same language subtitles can greatly improve the viewing experience and help learn foreign languages [1]. Creating them is not easy and takes time even for an experienced editor. Videos can be transcribed straight to subtitles. It is good enough for movies and television series, but for television shows and interviews, for example, that have natural and unscripted language, automatic transcription can be problematic as natural language is fast-paced and unstructured. Repeated words and transcription errors can render subtitles unusable. Hence, compressing subtitles is important for end-users and they must be as similar to the original speech as possible.

Automatic compression of subtitles using extractive summarization and combining compression with automatic speech recognition (ASR) have already been researched before [2, 3]. Although it can show good results, extractive summarization misses an opportunity to abstract sentences. Also, combining compression with ASR is not using the recent developments in natural language processing. With the emergence of big transformer-based language models (like BERT [4] and BART [5]) subtitle compression can be achieved even for small languages. Additionally, it is possible to use the same models for fixing transcription errors, thus improving the quality of automatically created subtitles [6].

The objective of this master thesis is to research the possibility of compressing subtitles that are transcribed from videos in Estonian using pre-trained language models. In addition, the author examines if compressing subtitles and fixing transcription errors can be achieved together. In order to achieve this, data preparation methods were developed and several models were trained and their performance was measured. Transcriptions of Estonian Public Broadcasting television programs were used for creating a parallel corpus, which was utilized to train multiple models, including seq2seq, extractive summarization and multilingual models.

The results of models show that it is indeed possible to create more readable subtitles that are similar to edited ones, language models being the best-performing ones. The author also researched if adding timing information to subtitles can help the model decide how long the final subtitle must be. This usually resulted in shorter subtitles, but in general, it makes no difference when additional inputs are added to the usual language model because editors are not consistent enough to make use of that information. Also, some transcription errors are fixable by models, but it depends on how many errors are present in the training set.

Lastly, a survey was conducted to check if viewers prefer compressed subtitles over transcription. 80 participants needed to watch 6 videos with subtitle options underneath and had to choose which one they preferred. They did not know the origin of these subtitle options. In 63% of cases, participants either preferred subtitles created by the model or didn't have a preference.

The results of this paper can be used to help editors create subtitles and automate most of their work. Full automation is not possible because of grammatical and factual errors in model output sentences. Using the developed data preparation algorithms it is possible to create even better models or adapt them to other languages.

The master's thesis is in Estonian language and contains 43 pages of text, 7 chapters, 18 figures, 20 tables and 2 appendices.

Lühendite ja mõistete sõnastik

Subtiiter	Tekst, mis on tuletatud kas dialoogi või kommentaari transkriptsioonist filmides, telesaadetes, videomängudes jms. Enamasti asuvad need ekraani allservas, kuid võivad paikneda ka ekraani ülaosas, kui ekraani allservas on juba tekst olemas.
Mudel	Masinõppe algoritm, mida on treenitud sisendandmete peal. See on võimeline leidma mustreid või tegema otsused varem nägemata andmestiku peal.
N-gramm	N elemendist koosnev jada teksti- või kõneproovist. Vastavalt n-i väärtusele kasutakse nimetuses ladinakeelseid numbrite eesliiteid (unigramm, bigramm, trigramm jne).
Seq2seq	Masinõppe lähenemine, mis muudab ühe järjendi teiseks.
BERT	*Bidirectional Encoder Representations from Transformers*.
BART	*Bidirectional and Auto-Regressive Transformers*.
Kooder	Muudab lause vektoriks sõnastiku abil.
Dekooder	Muudab vektori väljundkujuks.
Transformer	Loomuliku keele mudel, mis kasutab isetähelepanu (ing k *self-attention*) mehhanismi kaaludes iga sisendi tookenit erinevalt.
GPT	*Generative Pre-trained Transformer*.
BPE	*Byte-Pair Encoding*.
Tooken	Märgi eksemplaresitlus (antud töö puhul sõna või BPE).
OCR	Optiline märgi- või tähetuvastus (ing k *Optical character recognition*).
MT	Masintõlge.

Sisukord

Jooniste loetelu

Tabelite loetelu

1. Sissejuhatus

Kõnekeelest arusaamine on otsustava tähtsusega audiovisuaalsete teoste tarbimisel. Subtiitrid on abiks, kui kõne on raskesti kuuldav, vaataja keeleline tase ei ole piisav või vaatajal on tervislik seisund, mille puhul on kuulmine raskendatud. Audioga samakeelsed subtiitrid aitavad paremini eristada kõnet ning arendada võõrkeele mõistmist [1]. On leitud, et subtiitreid kasutavad õpilased suudavad paremini vastata teose kohta käivatele küsimustele ning subtiitrid on abiks harjumatute aktsentide puhul [7]. Lisaks avaldab nende kasutamine lastesaadete vaatamisel positiivset mõju ka laste lugemisoskuse arendamisele [8].

Vaatamata subtiitrite plussidele, on nende tootmisel omad miinused aja- ja ressursikulu osas. Näiteks kulub ühe tunnipikkuse audiofaili teksti üleskirjutamise ehk transkribeerimise jaoks kogenud transkribeerijal keskmiselt 4–5 tundi [9]. Kuigi subtiitreid on võimalik luua otse kõnekeelest transkribeerides, võib selline tekst olla raskesti loetav, sest kõnekeeles esinevad parasiit- ja täitesõnad ning kõneleja lauseehitus võib olla kohmakas. Kui selline kõne on kiire või ekraani suurus on piiratud, nõuavad otse audiofailist genereeritud subtiitrid vaatajalt ka kiiremat lugemist.

Inimese lugemiskiirus on aeglasem rääkimise ja kuulamise kiirusest [10]. Seetõttu võib olla kasulik otsekõnest saadud tekst koondada kokku ainult kõige asjakohasemaks sisuks [11], sealjuures tuleb võtta arvesse, et tekst oleks piisavalt sarnane helisalvestises olevale kõnele ja sõnu tuleks muuta ainult vajadusel. Vastasel juhul võib teksti erinevus audiosisust publikut häirida [3]. Seega mugavama lugemise võimaldamiseks kulub transkribeerijatel aega mitte ainult otsekõne üleskirjutamisele, vaid ka subtiitrite toimetamisele ja sobitamisele. Tabelis 1 on toodud mõned näited, kuidas teksti kokkuvõtmine ja ümbersõnastamine aitab subtiitrit paremini lugeda.

Tabel 1. Näide otsekõnest transkribeeritud subtiitrite korrastamisest. „Transkribeeritud“ lause kujutab transkribeerimisalgoritmi tulemust ja „Subtiiter“ sellele vastavat Eesti Rahvusringhäälingu poolt korrastatud subtiitrit.

Transkribeeritud	**Subtiiter**
Et, et, et miks mitte olla siis tasakaalus, ma noh, hüpoteetiliselt viskan selle palli üles,	Miks mitte siis olla tasakaalus. Hüpoteetiliselt viskan selle palli ülesse.
te olete ka noh, noh, päris korralikult ka Rahvusringhäälingu teatud mõttes sellisesse keerulisse olukorda pannud,	te olete ka rahvusringhäälingu keerulisse olukorda pannud.
kus see tähendab seda, et iga inimene mitte ei pea lugema meedia pandud pealkirju,	Iga inimene ei pea lugema mitte meedia pandud pealkirju,

1.1 Probleem

Otsekõnest transkribeeritud laused tihti ei sobi subtiitriteks, kuna otsekõne on kohmakas ja selle tulemusena peab meedia tarbija subtiitrite lugemisel liigselt pingutama. Käsitsi toimetamine on kallis, kuna see on aeganõudev ja nõuab spetsiifilist koolitust. Seega lahendusena soovitakse kasutada automaatseid süsteeme kõnesisu toimetamiseks ja sobitamiseks. Loomuliku keele valdkonnas võib antud probleemi liigitada lausete tihendamise alla.

Subtiitrite loomise kasvõi osaline automatiseerimine muutub aina olulisemaks ka Eesti Vabariigi seadusmuudatuste tõttu. Meediateenuste seadus sätestab: „Audiovisuaalmeedia teenuse osutaja muudab järk-järgult proportsionaalsete meetmete abil oma teenuse ligipääsetavaks puudega inimestele, kasutades selleks subtiitreid, viipekeelset tõlget, kirjeldustõlget, eraldi audiokanaleid, teleteksti ja teisi lisateenuseid, mis võimaldavad puudega inimestel pakutavat teenust kasutada.“ [12]

Lisaks alates 01.01.2026 jõustavad uued nõudmised teenuste ligipääsetavuses: „Valdkonna eest vastutav minister kehtestab määrusega täpsemad nõuded audiovisuaalmeedia teenuste ligipääsetavusele ja nende täitmise tähtajad, sealhulgas:

1. subtiitritega varustatud eestikeelsete saadete ning viipekeelse tõlkega, kirjeldustõlkega ja audiosubtiitritega varustatud saadete miinimummahu programmis või programmikataloogis;
2. nõuded subtiitrite valitavusele ja helilistele subtiitritele.“ [12]

1.2 Eesmärk

Loomuliku keele töötluse valdkond on viimase kümnendi jooksul kiirelt arenenud. See võimeldab tekste kiiresti tõlkida, kokku võtta, klassifitseerida jne. Magistritöö eesmärk on uurida, kuidas tihendada subtiitriteks tekste, mis on transkribeeritud automaatse kõnetuvastaja ppolt. Magistritöös annab autor ka ülevaate varasematest töödest samas valdkonnas. Selle käigus selgus, et neis on pööratud rohkem tähelepanu ekstraheeritavatele meetoditele, mis küll eemaldavad soovimatud sõnad, kuid ei kasuta tekstide töötlemisel ümbersõnastamist. Magistritöös otsustas autor panna olulist rõhku vähem uuritud monolinguaalse masintõlke ja abstraheerivate lausete kokkuvõtmise meetoditele. Varasemates töödes on keskendutud pigem transkribeerimise ja lausete tihendamise ühendamise meetoditele. Kusjuures mõlemad meetodid kasutavad sarnaseid masinõppe mudelite arhitektuure.

Lisaeesmärgiks on uurida võimalusi transkribeerimisvigade parandamiseks kasutades tekstide tihendamist. Transkribeerimismudeli vead võivad ühes tekstis korduda. Näiteks organisatsiooni nime asemel kasutatakse sõna, mis on küll sarnane öeldule, kuid ei sobi konteksti. Selliste vigade parandamine on tõhus viis vaataja subtiitrite lugemiskogemuse parendamiseks. Antud probleemi lahendamiseks ei loo töö autor eraldi masinõppe mudeleid, vaid uurib viise vigade parandamiseks tihendamise käigus võttes arvesse lahendusi, kus subtiitreid sobitatakse juba transkribeerimise ajal. Kasutuses olevad siirdeõppe mudelid sobivad mõlema probleemi lahenduseks [3, 6].

Andmestikuks autor võttis Eesti Rahvusringhäälingu (ERR) saadete transkribeerimise väljundi ja vastavad eestikeelsed subtiitrid. Kuna masinõppe mudelite treenimiseks andmed sellisel kujul ei sobinud, vajasid need vastavat ettevalmistust. Mudelite treenimiseks oli vajalik luua paralleelkorpus, kus masina poolt kirjutatud tekst on pandud vastavusse toimetaja poolt koostatu tekstiga. Kuna transkribeeritud teksti ja subtiitri vastavusse seadmine ei ole triviaalne ülesanne, vajas andmete ettevalmistamine eraldi algoritmilist lahendust.

Eesti keel on loomuliku keele töötluses väike keel, st pole palju kergelt kättesaadavaid andmestikke ja mudeleid. Lahendusena on loodud vaba ligipääsuga mudelid, mida on treenitud mitmete keelte peal ning on saavutanud üleüldise loomuliku keele mõistmise. Üks selline masintõlke mudeli näide on MBART [13]. Vaatamata sellele, et eesti keelt oskavad mudelid on enamasti mitmekeelelised mudelid, mis on arendatud masintõlke otstarbeks, on siiski võimalik panna selliseid mudeleid ühest keelest teise tõlkimise asemel tekste tihendama ehk „tõlkima“ sisendiga samasse keelde. Abstraheeritud lause kokkuvõtmiseks on kasutatud tipptasemel seq2seq mudeleid, näiteks Fairseq [14]. Autor püstitab hüpoteesi, et eelmainitud meetodeid kasutades on võimalik luua inimese poolt toimetatud tekstiga ligilähedase kvaliteediga subtiitreid.

Antud töös valmivad subtiitrid ei sobi vaegkuuljatele, kes kasutavad öeldu mõistmiseks huultelt lugemist. Sõnade asendamine ja järjekorra muutmine võib olla segav ja vähendada tarbimise kvaliteeti. Lisaks ei ole magistritöös keskendutud transkribeeritud teksti subtiitriteks lõikamisele, kuna selleks on juba tehtud teadustöid ja loodud andmekorpuseid, mis võimaldavad treenida vajalikke mudeleid. Üks selline korpus on MuST-cinema, kuhu on lisatud eraldavad märgistused nii subtiitrite ridade kui ka subtiitrite endi jaoks. Artiklis „MuST-cinema: a speech-to- subtitles corpus“ näitavad autorid Karakanta, Negri ja Turchi, et korpust saab kasutada mudelite ehitamiseks, mis tõhusalt segmenteerivad lauseid subtiitriteks ja pakuvad välja meetodi olemasolevate subtiitrite korpuste annoteerimiseks subtiitritevaheliste pausidega, mis vastavad aja- ja märgipikkuse piirangutele [15].

1.3 Ülevaade tööst

Magistritöö teises peatükis autor annab ülevaate loomuliku keele töötlemisest ja selle meetoditest. Kolmandas peatükis kirjeldatakse kolme varasema töö seost käesolevaga ja tuuakse välja lahendused probleemidele, mis on sarnased magistritöös püstitatutele. Neljandas peatükis kirjeldatakse autori poolt kasutatud arendusmeetodeid, mudelite lahendusi ja tulemuste analüüsiks valitud mõõdikuid. Viiendas peatükis annab autor ülevaate tulemustest. Kuuendas peatükis analüüsitakse mudelite ja läbiviidud küsitluse tulemusi ning pakutakse välja edasiarendusi tulevikuks.

2. Loomuliku keele töötlemine

Tehisintellekt uurib inimeste ja arvutite vahelisi interaktsioone loomuliku keele kaudu. Selle üks oluline haru on loomuliku keele töötlemine (ing k *natural language processing*, NLP), mis omakorda uurib fundamentaalseid tehnoloogiaid sõnade, fraaside, lausete ja dokumentide tähenduste väljendamiseks ning süntaktiliseks ja semantiliseks töötlemiseks. NLP-d rakendatakse sellistel väljadel nagu masintõlge ja on kasutuses näiteks otsingumootorites, klienditoe süsteemides, äriintelligentsis ja suulistes assistentides [16].

2.1 Masintõlge

Masintõlge on arvutuslingvistika alamväli, mis uurib tarkvara kasutamist teksti või kõne tõlkimiseks ühest keelest teise. Masintõlkel on oluline roll näiteks sotsioloogide, keele- ja arvutiteadlaste jaoks. Peale seq2seq mudelite arendamist on masintõlke valdkond kõvasti muutnud. Tehisnärvivõrkudel põhinev masintõlge, mis mõjutas tervet NLP valdkonda, on muutunud aina populaarsemaks ja kättesaadavamaks. Masintõlke valdkonnas on kõige rohkem kvaliteetseid paralleelkorpuseid, mis võimaldavad eeltreenida mitmekeelseid mudeleid. Näiteks tekstide kokkuvõtmine ja genereerimine on palju edasi arenenud just tänu meetoditele, mis olid algselt loodud masintõlke jaoks [17, 18].

2.2 Tekstide kokkuvõtmine

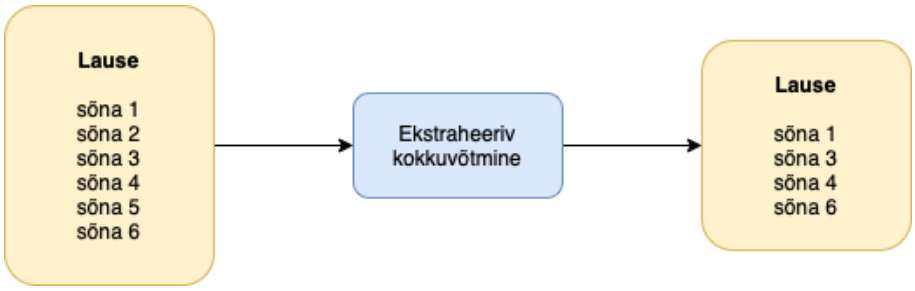
Tekstide kokkuvõtmist või summeerimist võib jagada kaheks: käsitsi tehtavaks ja automaatseks. Esimene on protsess, mis hõlmab käsitsi kokkuvõtete loomist inimekspertide poolt. See on äärmiselt aeganõudev, raske, kallis ja stressirohke töö. Siinkohal oleks kasulik automatiseerimine, et muuta töö kiiremaks, odavamaks ja kergesti korratavaks. Automaatne teksti summeerimine on teabeallika sisu automaatne lühendamine säilitades vaid kõige olulisem informatsioon. Kokkuvõtvate süsteemide eesmärk on aidata lugejal mõista pika teksti sisu ilma tervet teost lugemata. Hea kokkuvõte peab olema ladus ja järjepidev, haarama kõiki olulisi teemasid, kuid mitte kordama sama teavet. Tipptasemel tehnoloogiaks antud valdkonnas on jadalt jadale modelleerimine (ing k *Sequence-to-Sequence modelling*), mis muudab ühe jada teiseks, sõltuvalt mudeli eesmärgist [19, 20].

Tekstide summeerimine keskendub enamasti dokumentide ja teiste pikkade tekstide lü-

hendamisele. Üksikute lausete pikkuse vähendamine on pigem uurijate fookusest väljas [21].

2.2.1 Ekstraheeriv lausete lühendamine

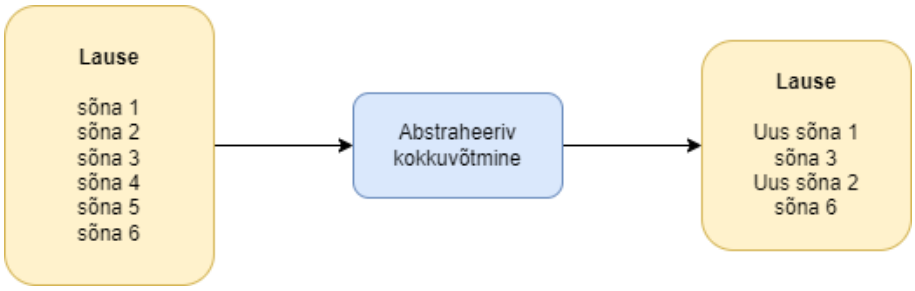
Ekstraheeriv lausete lühendamine leiab algsest sisendist sõnad, mida on vaja säilitada. Tavaliselt lahendatakse seda lekseemide klassifitseerimise meetoditega. Ekstraheeriv lähenemine on abstraheerivast kiirem ja lihtsam, sest väljund ekstraheeritakse ehk võetakse välja otse sisendtekstist. Nii on võimalik säilitada originaaltekstiga sama terminoloogia ja tagada suurem täpsus [22]. Tööprintsiip on kujutatud Joonisel 1.



Joonis 1. Ektraheeriv lausete lühendamine.

2.2.2 Abstraheeriv lausete lühendamine

Tekstide summeerimise uurimise keskmes on viimastel aastatel olnud järk-järguline üleminek ekstraheerivalt abstraheerivale tehnoloogiale. See on osaliselt tingitud olulistest edusammudest tehisnärvivõrkudel põhinevate meetodite väljatöötamisel. Algselt masintõlke jaoks välja töötatud sellised meetodid on vaieldamatult ümber kujundanud abstraheerivad summeerimisuuringud [23]. Tööprintsiip on kujutatud Joonisel 2.



Joonis 2. Abstraheeriv lausete lühendamine.

2.3 Seq2seq modelleerimine

Seq2seq (*Sequence-to-Sequence*) mudelid on NLP valdkonnas olulised, kuna traditsioonilised masinõppe mudelid ja närvivõrgud ei suuda tabada jada teavet tekstist, sh sõnade omavahelisi suhteid ja järjestusi. *Sequence-to-Sequence* modelleerimine püüab luua ühest jadast (nt lause vektorkujutisest) teist. Jadadena saab lahendada mitmeid NLP ülesandeid, näiteks masintõlkes lähtekeelest sihtkeele sõnajada genereerimiseks. Samuti on need kasutuses juturobotites küsimuste sõnajadast vastuse sõnajada genereerimiseks või dialoogis kasutaja sisendi sõnajadast roboti vastuse sõnajada genereerimiseks. Lisaks leiab seq2seq kasutust ka teksti summeerimisel, kus on võimalik sisendi sõnajadast genereerida kokkuvõte sõnajada kujul [16].

Selliste mudelite põhikomponentideks on kooder ja dekooder. Kooder võimaldab muuta sisendvektori peiduvektoriks, mis sisaldab konteksti ja sisendi teavet. Dekooder on võimeline tegema vastupidist ehk looma peiduvektorist väljundvektorit.

2.4 Siirdeõpe

Siirdeõpe on tehnika, mille puhul kasutatakse suurel andmestikul treenitud sügavõppe mudelit, et täita sarnaseid ülesandeid teist andmestikku kasutades. Sellist treenitud sügavõppe mudelit nimetakse eeltreenitud mudeliks. Enamasti ongi parem kasutada probleemi lahendamiseks eelnevalt treenitud mudelit, mitte ehitada uut nullist [24]. Suurte andmestike peal treenitud mudelitel tekkib nii-öelda üleüldine loomuliku keele mõistmine, mida saab edaspidi kasutada teatud valdkondade ülesannete lahendamiseks. Lisaks on selliste mudelite loomiseks kasutatud suurt arvutusvõimsust, mis ei ole kergelt kättesaadav. Seega sarnase võimekusega mudelite loomine iga ülesande täitmiseks on ebaotstarbekas [25].

2.4.1 BERT

BERT (*Bidirectional Encoder Representations from Transformers*) on eeltreenitud sügavõppe mudel, mille treenimiseks on kasutatud märgendamata ehk lisateabeta teksti. Mudel on võimeline õppima teksti nii vasakult kui ka paremalt lugedes. Selline mudeli kahesuunalisus on väga tähtis, et mudel oleks võimeline keelt mõistma. Näiteks lausetes „see on maitsev tee“ ja „näita tee koju“ võib sõna „tee“ kontekst mudeli jaoks sõltuda sellest, kummas suunas lauset lugeda. BERT on mõeldud peenhäälestamiseks spetsiifiliste ülesannete jaoks. Mudel on eeltreenitud kasutades peidetud sõna või järgmise lause ennustamise meetodeid [4].

BERT ei ole hea valik teksti genereerimiseks, sest see on treenitud peidetud sõnade ennustamise meetodil, mis arvestab sõna ees ja taga oleva kontekstiga. Seetõttu ei ole mudel võimeline ennustama järgmist sõna, vaid oskab ainult öelda, milline sõna on lauses peidetud. Sellisel viisil saab BERTiga genereerida teksti sõnahaaval, kuid leidub ka palju efektiivsemaid mudeleid, näiteks järgmises alampeatükis kirjeldatud BART [4, 26].

2.4.2 BART

BART kasutab tavalist seq2seq mudeli arhitektuuri. Selle kodeerija on kahesuunaline (nagu BERTil) ja kasutab vasakult-paremale dekooderit (nagu GPT). BARTi eeltreenitakse mürafunktsioonidega rikutud dokumentide peal ja seejärel nende rekonstruktsioonide peal. Sisendis on laused omavahel segatud ja tekstide vahed on juhuslikult asendatud spetsiaalse tookeniga, mille mudel peab ära täitma. BART on eriti tõhus siis, kui see on peenhäälestatud teksti genereerimiseks (näiteks masintõlke jaoks), kuid toimib hästi ka teksti sisu mõistmise puhul. BART on üks populaarseim teksti genereerimise mudel ning selle edasiarendused MBART ja MBART-50 on antud magistritöö kirjutamise ajal tipptehnoloogia [5].

3. Varasemad tööd

Magistritöös püstitatud probleemi võib algselt liigitada tavatekstide summeerimise alla, kuid lähedalt uurides erineb see suuresti. Subtiitrite puhul peab helikujul öeldud lause olema üsna sarnane sellele, mida kuvatakse ekraanile, aga ei tohi olla liiga pikk ega jätta lauseid vahele, sest see raskendab lugemist [3]. Seetõttu ei ole autor varasemate tööde hulka arvanud pikkade tekstide ja dokumentide kokkuvõtmist, kuna nendes ei üritata säilitada lausetes piisavat sarnasust, vaid pigem keskendutakse lausete pikkuse ja arvu hulgalisele vähendamisele. Küll aga kasutab autor selliste tööde võtteid ja mudeleid magistritöös esitatud probleemile lahenduse otsimiseks. Järgnevates alampeatükkides annab autor ülevaate varasematest seotud teadustöödest.

3.1 Subtiitriks tihendamine liigsete sõnade märgendamise abil

Angerbauer jt [2] uurisid subtiitrite automaatset tihendamist kasutades tehisnärvivõrke. Selgus, et subtiitrite tihendamise jaoks on võimalik kasutada ekstraheerivat lausete lühendamise meetodit. See annab igale sisendlause sõnale märgi, kas sõna JÄTTA(1) või KÕRVALDADA(0). Artiklis välja pakutud mudel loeb kõigepealt läbi terve lause ja ehitab sellest kujutise (kooderi). Traditsiooniline kooder-dekooder mudel tekitaks seejärel dekooderiga järgmise järjestuse, mis võib olla algsest lausest erineva pikkusega. Kuid kirjeldatud mudeli seadistust on muudetud nii, et väljund oleks sisendiga pikkuse poolest võrdne ja otsus oleks tehtud iga sõna kohta. Seetõttu on dekooder kohandatud ning kooderi varjatud olekud toidetakse uuesti dekooderisse, et oleks võimalik teha eraldi otsus iga üksiku tookeni kohta. Lisaks on dekooderile lisatud juurde informatsioon, mis näitab kui palju on vaja lauset vähendada.

Angerbrauer jt uurisid ka tihendatud subtiitrite mõju lõppkasutaja vaatamiskogemusele olukorras, kus subtiitri teksti on vähendatud 50–60% võrra. Eksperimendis osalenutel paluti hinnata subtiitrite lugemisraskust ning teabe piisavust. Uuringute tulemused näitasid, et sellised subtiitrid on arusaadavad, kuid põhjustavad vaatajale suuremat kognitiivset koormust. Vastajad ei täheldanud olulisi erinevusi käsitsi ja artikli autorite poolt pakutud meetodil loodud subtiitrite vahel.

3.2 Automaatse kõnetuvastuse rakendamine koos tihendamisega otse subtiitrite loomiseks

Liu jt [3] keskendusid oma töös automaatse kõnetuvastuse (ing k *Automatic Speech Recognition*, ASR) abil subtiitrite loomisele. ASR süsteeme hinnatakse tavaliselt eelkõige transkriptsiooni täpsuse alusel. Mõnel juhul, näiteks subtiitrite puhul, vähendaks otsekõnest transkribeeritud tekst väljundi loetavust, kui võtta arvesse ka ekraani piiratud suurust ja vaatajal subtiitrite lugemiseks kuluvat aega. Alguses proovisid teadustöö autorid paralleelkorpuse puudumise tõttu treenida mudeleid juhendamata viisil, kasutades järjestikus automaatset kõnetuvastajat ja lause tihendajat. Selline lahendus kasutas liiga palju ümbersõnastamist ega võtnud arvesse subtiitri pikkuse kitsendust. Seetõttu otsustasid autorid hiljem keskenduda kõnetuvastaja väljundi tihendamisele, mis võimaldab lausete summeerimist ilma vahepealset teksti genereerimata. Esimene vaadeldud lähenemine oli sõnade vektorikujutistele (ing k *word embeddings*) informatsioon lisamine selle kohta, kui palju tookeneid tohib veel kasutada. Teine lähenemine oli muuta *transformer*'i trigonomeetrilise asukoha kodeeringut nii, et see näitaks ülejäänud lause pikkust, mitte sõna praegust asendit. Mõlemad lähenemised töötasid ja võimaldasid genereerida tihendatud teksti, aga ei garanteerinud selle pikkust.

3.3 BARTi kasutamine OCR vigade parandamiseks

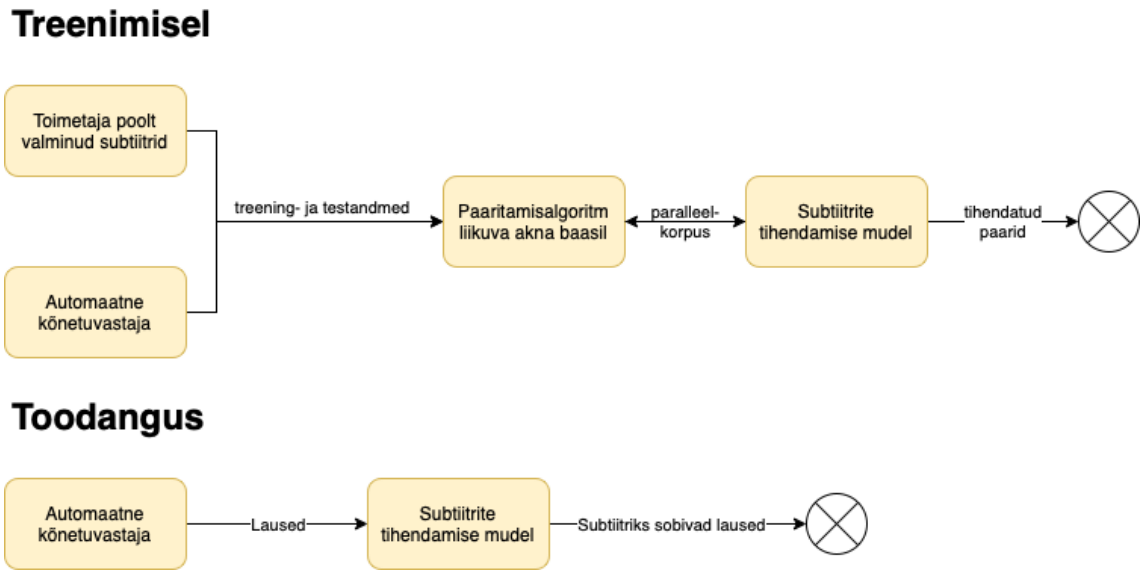
Soper jt [6] kasutavad oma töös BART mudelit, et parandada optilise märgituvastuse (ing k *optical character recognition*, OCR) vigu. Kuigi esmapilgul OCR ei tundu olevat seotud transkribeeritud teksti tihendamisega, on oluline märgata, et subtiitri kvaliteedi tõstmiseks on tähtis audiofailis öeldu ja ekraanile kuvatava teksti omavaheline sarnasus. Soper jt uurisid, kas oleks võimalik parandada vigu kasutades *transfomer*-põhist mudelit BARTi, mis tekkisid vanade ingliskeelsete ajakirjade piltidelt märkide tuvastamisel. Täpsemini uuriti puuduvate või liigsete märkide korrigeerimist ja olematu teksti juurdetekkimist. BART on selliste vigade parandamise jaoks ideaalne, kuna selle eeltreenimiseks on juba kasutatud erineval moel „rikutud" tekstide taastamist. Vaatamata sellele näitas peenhäälestamata mudel siiski sisendist halvemat tulemust.

Autoritel oli väike andmestik (40 000 lauset), kus optiliselt tuvastatud lausega oli vastavusse seotud inimese poolt käsitsi loodud viitetekst. Andmestikku kasutati, et mudel märgituvastuse vigade parandamiseks peenhäälestada. Tulemusena näitas mudel 29,4% tõusu täpsuses algse ja viiteteksti vahel. Autorid väidavad oma töös, et sellist lähenemist kasutades on võimalik parandada ka teisi tekstilisi vigu. Kasutades BARTi mitmekeelelist versiooni MBART on võimalik seda lähenemist kasutada ka teiste keelte jaoks.

4. Metoodika

4.1 Arhitektuur

Töös esitletud probleemi lahenduseks pakub autor välja ahelarhitektuuri, mida on kujutatud Joonisel 3. Toimetaja poolt töödeldud subtiitrid ja automaatse kõnetuvastaja väljund on omavahel vastavusse viidud ehk paaridesse seatud. Nende abil loob paaritamise algoritm paraleellkorpuse edasiste mudelite treenimiseks. Korpuses on olemas vajalik teave: millisest tekstist on vaja luua subtiiter, toimetaja poolt loodud subtiiter, paaridevaheline kaugus ning subtiitri ekraanil kuvamise ajavahemik. Sama korpusega on võimalik treenida kõiki töös kasutatud mudeleid.



Joonis 3. Ahelarhitektuuri diagramm.

Toodangus on subtiitrite ajavahemik ja sisendtekst mudelile ette antud. Seejärel annab mudel väljundina tihendatud tekstilõigu.

4.1.1 Automaatne kõnetuvastus

Magistritöö kasutab eestikeelsete televisioonisaadete kõne tekstiks muutmiseks Tallinna Tehnikaülikooli kõnetranskriptsioonisüsteemi. See on mõeldud poolspontaanse kõne käsitlemiseks, näiteks eetrivestlused, loengusalvestised ja intervjuud, mis on salvestatud erinevates akustilistes tingimustes. Süsteem põhineb Kaldi tarkvara tööriistakomplektil. Töökindluse parandamiseks kasutatakse ülekantavatest andmetest eraldatud taustamüra peal treenimist. Sõnastikuvälised sõnad taastatakse foneemil n-grammise dekodeerimise alamgraafi ja FST-põhise (ing k *finite-state transducer*) foneem-grafeem mudeli abil. Süsteem saavutab saatevestluste testimisel sõnavea määra 8,1% ja tegeleb ka kirjavahemärkide taastamise ja kõneleja identifitseerimisega. Kõneleja identifitseerimise mudeleid treeniti hiljuti väljapakutud nõrga juhendatud õpetamise meetodil [27].

4.1.2 Andmete ettevalmistamine

Magistritöö kirjutamise ajal ei olnud eestikeelne monolinguaalne paraleellkorpus kättesaadav, mille abil oleks võimalik õpetada mudelile transkribeeritud teksti tihendamist subtiitrite jaoks sobivale kujule. Selline korpus on vajalik, kuna masinõppe puhul on üheks efektiivseimaks õpetamisviisiks näidata mudelile väljundit, mis vastab sisendile. Sellist õppimisviisi nimetakse juhendatud õppeks. Juhendatud õppe juurutamine võimaldab paremini ennustada, milline teave on oluline ja peaks säilima [28].

Korpuse loomiseks kasutati ERR-i poolt toimetatud subtiitreid ja Tallinna Tehnikaülikooli kõnetranskriptsioonisüsteemi väljundit samadele toimetatud telesaadetele. Andmete ettevalmistamise algoritmi on kirjeldatud alampeatükis 4.2.

4.1.3 Teksti tihendamine subtiitriks

Subtiitrite loetavamaks muutmiseks kasutatakse rakendusliidest (ing k *Application Programming Interface*, API), mis võib sisaldada nii masinõppe mudelit kui ka sisendteksti tagastavat programmikoodi. Automaatsete mõõdikute arvutamiseks peab võrdluseks olema sisendisse lisatud ka alustekst.

4.2 Andmestiku loomine

Andmete algkuju ei luba lihtsasti vastavusse seada subtiitrit ja sellele vastavat transkribeeritud tekstilõiku. Toimetatud subtiitrites on kõnest transkribeeritud tekst tükeldatud ja sellele on lisatud ekraanil kuvamise ajavahemik. Väljalõige subtiitrite failist on näha Joonisel 4.

```
0
00:00:14.160 --> 00:00:15.760
Tere \u00f5htust, head vaatajad.

1
00:00:15.840 --> 00:00:18.320
Valitsus leppis kokku
4 aasta eelarvestrateegias

2
00:00:18.400 --> 00:00:22.240
ja nii m\u00f5nedki v\u00e4ga olulised valdkonnad
nagu tervishoid v\u00f5i julgeolek
```

Joonis 4. Toimetaja poolt loodud subtiitrite faili näide.

Automaatse kõnetuvastaja poolt transkribeeritud tekst on eraldatud kõnelejate järgi ning igal sõnal on olemas kõnelemise algus ja lõpp. Kuid lühikeste repliikide puhul ei pruugi mudel kõnelejate vahetust tuvastada. Subtiitritele vastavat automaatse kõnetuvastaja väljundit on näha Joonisel 5.

```
"sections": [
   {
      "start": 0,
      "type": "speech",
      "end": 3492.52,
      "turns": [{
            "speaker": "S1",
            "unnormalized_transcript": ...
            "start": 14.01,
            "transcript": "Tere \u00f5htust, head vaatajad,
               valitsus leppis kokku nelja aasta eelarve
               strateegias ja nii m\u00f5nedki v\u00e4ga
               olulised valdkonnad, nagu tervishoid v\u00f5i
               julgeolek saavad raha juurde. [...]",
            "end": 86.23,
            "words": [...]
            ...}
            }
```

Joonis 5. Automaatse kõnetuvastaja väljundi näide.

4.2.1 EstNLTK

EstNLTK (NLTK ehk *Natural Language ToolKit*) on tarkvaratööristade kogum eestikeelsete tekstide töötluseks. See võimaldab analüüsida näiteks tekstide morfoloogiat ja sõnade seoseid. Antud kogum on ainuke, mis pakub sellist võimekust eesti keele jaoks [29]. Andmete ettevalmistamiseks kasutati EstNLTK lemmatiseerijat ja sõnestajat, mis aitasid määrata, kas transkribeeritud sõna on oluline säilitamiseks. See on ainus tööriist eesti keele morfoloogilise analüüsi jaoks.

4.2.2 Levenšteini kaugus

Levenšteini kaugus on algoritm, mis kirjeldab kahe sõne vahelist kaugust, täpsemini minimaalset põhitoimingute arvu (lisamine, kustutamine või asendamine), mida on vaja, et üks sõne teiseks teisendada. Näiteks sõne „abcde" teisendamiseks „acrde"-ks on vaja kasutada kahte toimingut: kustutamist ja lisamist. Pikemate sõnekauguste ja -pikkuste suhet loetakse nende sõnede sarnasuseks [30]. Levenšteini kaugus sobib hästi kahe sama tähendusega lause võrdlemiseks, kuna algoritm arvestab ka võimalike asendamistega, mis võivad tekkida näiteks üleliigsete sõnade või ümbersõnastamise puhul.

4.2.3 Paaritamise algoritm

Subtiitrite juurde kuulub ka teave, millises ajavahemikus tuleb subtiitrit ekraanil kuvada. Automaatse kõnetuvastaja väljund sisaldab iga sõna kohta ajastamise informatsiooni. Paralleelkorpuse loomiseks oli vajalik võimalikult täpselt panna paari subtiitrile kuuluv ajavahemik kõnetuvastaja väljundiga. Loomuliku keele mudelite treenimise jaoks oli tähtis säilitada maksimaalselt palju toimetaja poolt kustutatud või muudetud sõnu subtiitrite kuvamise ajavahemikes. Seda silmas pidades loodi algoritm algsete paaride sobitamiseks, mida on näha Joonisel 6.

```
1: procedure PREPROCESS(captions, words)
2:     i ← 0
3:     pairs ← list
4:     for each caption ∈ captions do
5:         f ← list
6:         while i <= len(words) & words[i].end < caption.end do
7:             f.append(words[i])
8:             i ← i + 1
9:         end while
10:        pairs.append(f)
11:    end for
12:    return pairs
13: end procedure
```

Joonis 6. Algsete paaride sobitamise algoritm.

Sellisel viisil valmistatud paarid ei ole piisavalt täpsed. Mõned sõnad võivad sattuda järgmisesse paari või jääda eelmisesse sisse, kuna subtiitrite kuvamise ajavahemik ei kattu alati kõnelemise ajaga. Joonisel 7 on näha, et mõnesid paare on raske nimetada samaväärseteks, kuna üleliigseid sõnu on palju või just olulised puuduvad. Selliste paaridega ei ole otstarbekas mudeleid treenida, kuna siis ei ole neuraalvõrgul arusaama, miks toimetatud subtiitris on mõned sõnad juurde tekkinud või hoopis kadunud. Probleemi lahendamiseks kasutati lisaks Joonisel 6 toodud algoritmile ka liugakna meetodit, mida kirjeldatakse järgmises alampeatükis.

Kõnetuvastaja: sellel nädalal esitles öönstanud yang Riias värsket Rail Balticu
Subtiiter: sellel nädalal esitles Ernst & Young Riias värsket Rail Balticu tasuvusuuringut.

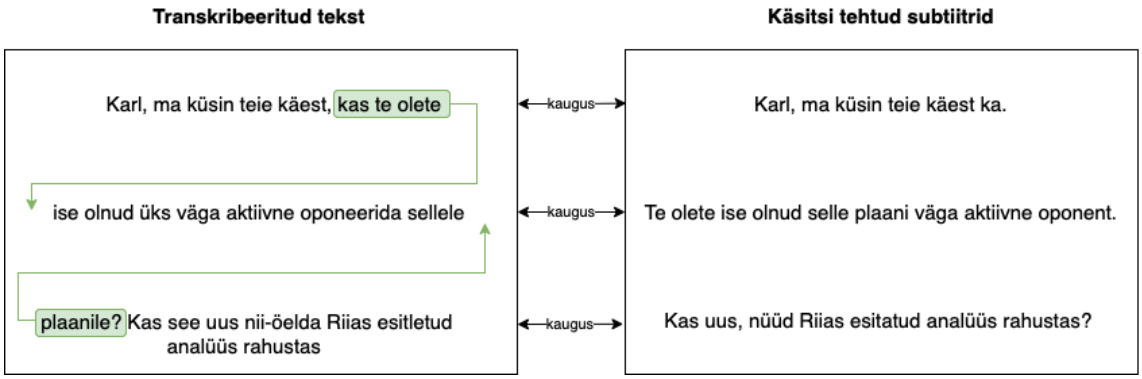
Kõnetuvastaja: tasuvusanalüüsi. Mis oli selle analüüsi peamine kokkuvõte teie
Subtiiter: Mis oleks selle analüüsi peamine kokkuvõte?

Kõnetuvastaja: jaoks? No üldises plaanis ta ütles sedasama, mida on öelnud ka varasemad uuringud, et
Subtiiter: Ta ütles sama, mida on öelnud ka varasemad uuringud:

Joonis 7. Algsete paaride algoritmi väljund.

4.2.4 Paaritamise algoritm liugakna meetodil

Eelmises alampeatükis mainitud probleemi lahenduseks valiti liugakna meetod. Akna suuruseks seati kolm lauset. Liugakna meetod valiti, sest sõnade liigutamine on võimalik ainult naaberpaaride vahel ehk aknas. Sellest tulenevalt saab järeldada, et saavutades akna sees lausete kauguste lokaalne miinimum, on võimalik jõuda terve telesaate transkriptsiooni globaalse kauguste miinimumini. See tähendab, et kauguse mõttes on transkribeeritud laused maksimaalselt lähedased toimetatud lausetele. Liugakna algoritmi skeemi on näha Joonisel 8.



Joonis 8. Liugakna algoritmi töötamise skeem.

Liugakna meetodil liigub kindlaksmääratud pikkusega aken ***Len*** üle andmete ning statistika arvutatakse aknas olevate andmete põhjal. Iga sisendproovi väljund on praeguse prooviakna ja ***Len - 1*** varasemate näidiste statistika. Kõigepealt esimese ***Len - 1*** väljundi arvutamiseks, kui aknal ei ole veel piisavalt andmeid, täidab algoritm akna nullidega. Järgmistes etappides kasutab algoritm akna täitmiseks eelmiseid näidiseid. Liikuvatel statistilistel algoritmidel on olek ja need mäletavad varasemaid andmeid [31].

Algoritmi eesmärk on liugakna sees minimeerida paaride omavahelist Levenšteini kauguse summat. Kõik paarides olevad sõnad on viidud algvormile, et erinevad käändelõpud või sõnavormid ei mõjutaks lausete kauguse arvutamist. Seejärel üritab algoritm liigutada sõnu paaride vahel akna sees hoides algset sõnade järjekorda. Iga sammu puhul arvutatakse terve akna sees kaugus ümber ning lõpuks valitakse parim seis, mis salvestakse. Niimoodi analüüsitakse terve salvestise lausepaarid ära. Joonisel 9 on näha liugakna algoritmi väljundit.

Kõnetuvastaja: sellel nädalal esitles öönstanud yang Riias värsket Rail Balticu tasuvusanalüüsi.
Subtiiter: sellel nädalal esitles Ernst & Young Riias värsket Rail Balticu tasuvusuuringut.
Kaugus: 15

Kõnetuvastaja: Mis oli selle analüüsi peamine kokkuvõte teie jaoks?
Subtiiter: Mis oleks selle analüüsi peamine kokkuvõte?
Kaugus: 11

Kõnetuvastaja: No üldises plaanis ta ütles sedasama, mida on öelnud ka varasemad uuringud, et
Subtiiter: Ta ütles sama, mida on öelnud ka varasemad uuringud:
Kaugus: 22

Joonis 9. Parandatud paaritamise algoritmi väljund.

4.3 Kontrollmudelid

Kontroll- või baasmudel on sisuliselt lihtne mudel, mis on masinõppeprojektide võrdlusaluseks. Selle peamine ülesanne on mõtestada treenitud mudelite tulemusi. Selleks, et mõista, kas töös väljatöödeldud mudelid on võimelised teksti tihendama sarnaselt toimetajaga, loodi mõned kontrollmudelid. Need põhinevad eelnevalt uuritud mudelitel või magistritöö kirjutamise ajal tavalahendustena kasutatud meetoditel.

- **Lühendamata paaride mudel**, mis tagastab sisendiga võrdset teksti ja näitab, kui sarnane on algne tekst toimetaja poolt loodud tekstiga.
- **Seq2seq mudelid** ehk jadamudelid, mis on kõige tihedamini kasutuses olevad mudelid masintõlke ja teksti kokkuvõtmise puhul. Nende treenimine on kerge ja kiire, kuid puudub keele arusaamine enne treenimist [32].
- **Liigseid sõnu märgendav mudel EstBERTi baasil** toimib samal põhimõttel, mida on kirjeldatud alampeatükis 3.1. Erinevuseks võrreldes Angerbauer jt [2] tehtud tööga on rekurrentse närvivõrgu asendamine siirdeõppe mudeliga. Asendust eelistati keelemudelite kasutuselevõtu tõttu. Üldjuhul need näitavaid paremaid tulemusi ning nii nagu rekurrentsete närvivõrkudega on ka keelemudelitega võimalik summeerida tekste ekstraheerivalt [33]. Siirdeõppe mudeliks valiti EstBERT, mida on juba varasemalt eeltreenitud eestikeelsete tekstide peal [34].

4.4 Eeltreenitud mudelid

Loomuliku keele töötlemise tipptehnoloogiaks on magistritöö kirjutamise ajal eeltreenitud mudelite kasutamine. Märkimisväärselt palju teadustöid on näidanud, et suure korpuse peal võivad eeltreenitud mudelid õppida selgeks universaalseid keele kujutisi. Neid mudeleid kasutades saab vältida mudeli nullist treenimist ja vähendada eesmärgi saavutamiseks vajaminevat andmemahtu. Lisaks aitab see vältida ülesobitamist ehk olukorda, kus mudel ei oska õpitut üldistada ja saab hakkama ainult nähtud andmetega [25].

Eeltreenitud mudelitena on antud töös kasutatud masintõlke jaoks valmistatud mudeleid, kuigi on olemas ka juba valmis eeltreenitud mudeleid tekstide kokkuvõtmiseks. Dokumentide ja tekstide summeerimise probleem on esmapilgul palju sarnasem töös lahendatavale probleemile, sest selle käigus samuti genereeritakse sisendist tihendatud versioon. Küll aga tekitab raskusi asjaolu, et selliseid mudeleid on treenitud eesmärgiga võtta pikad tekstid kokku paariks lauseks, seevastu subtiitrid võiksid olla üsna sarnased kõneldule. Vaatamata sellele, et magistritöös kasutatud masintõlke mudelid on loodud ühest keelest teise tõlkimise jaoks, on võimalik kasutada meetodeid, mis sunnivad mudelit andma väljundit sisendiga samas keeles. Eelmainitud põhjustel on autor otsustanud kasutada subtiitrite loomiseks masintõlke mudeleid.

4.5 Automaatmõõdikud

Suuremahulisi tekste kokkuvõtvate mudelite töösoorituse käsitsi ja ka poolautomaatne hindamine on kulukas ja tülikas. Seetõttu on nähtud palju vaeva, et töötada välja automaatsed mõõdikud, mis võimaldaksid mudelite tööd kiiresti ja odavalt hinnata [35]. Enamasti võrdlevad sellised mõõdikud mudeli väljundit toimetaja poolt tihendatud viitetekstiga. Abstraheeriva kokkuvõtmise puhul on kõige populaarsem mõõdik ROUGE, kuid kasutatakse ka BLEU ja METEOR automaatmõõdikuid. Ekstraheeriva kokkuvõtmise kvaliteedi mõõtmiseks kasutatakse veel lisaks täpsust, saagist ja f-skoori [36].

4.5.1 BLEU

BLEU (*Bilingual Evaluation Understudy*) on algoritm kvaliteedi hindamiseks tekstide puhul, mis on masintõlgitud ühest loomulikust keelest teise. Kvaliteediks peetakse vastavust automaatse ja inimtõlke vahel. BLEU'l on põhimõte: mida lähemal on masintõlge professionaalsele inimtõlkele, seda parem see on. BLEU oli üks esimesi mõõdikuid, mis näitas kõrget korrelatsiooni käsitsi läbi viidud kvaliteedihinnanguga ning on jätkuvalt üks populaarseim ja odavaim automatiseeritud mõõdik [37].

Skoorid arvutatakse üksikute tõlgitud segmentide (tavaliselt lausete) kohta võrreldes neid hea kvaliteediga viitetõlke kogumiga. Seejärel keskmistatakse need hinded kogu korpuse ulatuses, et jõuda tõlkekvaliteedi üldise hinnanguni. Intelligentsust või grammatilist korrektsust ei arvestata. BLEU väljund on alati 0 ja 1 vahel. See väärtus näitab, kui sarnane on kandidaattekst viitetekstidega, kusjuures väärtused, mis on lähemal kui 1, tähistavad sarnasemaid tekste. Ainult mõned inimtõlked saavutavad skoori 1, sest see näitab, et kandidaat on identne viitetõlkega, mistõttu ei ole ka vaja saavutada punktisummat 1. Kuna tekstide omavahelisse vastavusse seadmiseks on mitmeid võimalusi, suurendab täiendavate viitetõlgete lisamine BLEU punktisummat [37, 38].

4.5.2 ROUGE

ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) on kogum mõõdikuid ja tarkvarapakett, mida kasutatakse automaatse kokkuvõtte ja masintõlke tarkvara hindamiseks loomuliku keele töötlemisel. Mõõdikutes võrreldakse omavahel automaatselt kokkuvõetud teksti või tõlget inimese poolt loodud kokkuvõtte või tõlkega ehk viite või viitekogumiga. ROUGE on tõstutundmatu, mis tähendab, et suurtähti käsitletakse samamoodi nagu väiketähti [39]. ROUGE jaguneb omaette veel mitmeks erinevaks mõõdikuks:

- **ROUGE-n**, mis on saagisel põhinev mõõdik ja kasutab n-grammide võrdlust. Viitekokkuvõtetest ja kandidaadi kokkuvõttest (automaatselt genereeritud kokkuvõte) tuleneb rida n-gramme (enamasti üks või kaks). Mõõdik arvutatakse jagades mõlemas tekstis leiduvate n-grammide arv viitekokkuvõtte n-grammide arvuga.
- **ROUGE-L**, mille puhul kasutatakse kahe tekstijada pikima ühisosa meetodit. Mida pikem on ühisosa kahe kokkuvõtva lause vahel, seda sarnasemad nad on. Kuigi see mõõdik on paindlikum kui eelmine, võib puudusena välja tuua selle, et kõik n-grammid peavad olema järjestikused;
- **ROUGE-LSUM**, mis on sarnane eelmise mõõdikuga, kuid jagab teksti mitte laseteks vaid lõikudeks, mis on eraldatud reavahega [40].

4.5.3 METEOR

METEOR on masintõlke hindamise jaoks välja töödeldud automaatne mõõdik, mis põhineb masintõlke ja käsitsi loodud viitetõlke üldistatud unigrammi kontseptsioonil. Unigramme saab sobitada nende pinna- ja tüvevormide ning tähenduste põhjal. Lisaks saab METEORi hõlpsasti laiendada lisades täiustatud sobitamise strateegiaid. Kui kõik üldistatud unigrammi vasted on kahe sõne vahel leitud, arvutab METEOR selle sobitamise skoori kasutades unigrammi täpsuse, saagise ja fragmenteerumise kombinatsiooni. Eesmärk on teada saada,

kui ligilähedased on masintõlgitud sõnad võrdlusalustega [41].

4.6 Uuring

Mõõdikute abil ei ole võimalik kindlalt määrata, kas televaataja jaoks sobib töös välja töötatud masinõppe mudeli väljund subtiitritena rohkem kui otse kõnetuvastajaga transkribeeritud tekst. Magistritöö käigus läbi viidud uuringus osalejad nägid saatelõiku ning seejärel valisid, kas üks kahest valikust sobib paremini subtriitriteks või on need võrdväärsed. Vastajad ei teadnud tekstide päritolu ja nende järjekord valikute hulgas oli juhuslik. Uuringus kasutatavad subtiitrid ei olnud mudelile varasemalt tuttavad ning need said valituks enne mudelipoolset tihendamist.

5. Tulemused

Magistritöö raames valmis paralleelkorpus ja kuus teksti subtiitriteks tihendavat mudelit. Andmestik ja osa mudelitest on kättesaadavad HuggingFace'is[1]. Korpuse loomise lähtekood on saadaval Githubis[2].

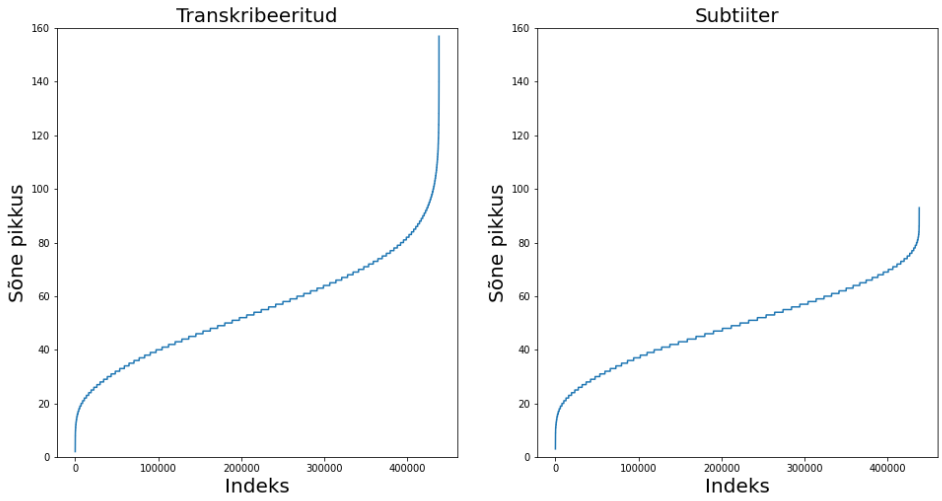
5.1 Andmestiku omadused

Andmestiku loomiseks kasutati 996 audiovisuaalset salvestist, mille puhul kõnetuvastaja väljund oli paari viidud toimetaja poolt loodud subtiitritega. Kokku loodi sellisel viisil 604 892 subtiitripaari. Nendest filtreeriti välja kõik paarid, mis olid liiga erinevad, tühjad, sisaldasid kirillitsat või olid muud moodi vigased. Näiteks kõnetuvastaja algoritmil ei õnnestunud laulude transkribeerimine, mistõttu selliseid lauseid ei saanud õppimiseks kasutada.

Lausete erinevust mõõdeti jagades Levenšteini kaugus paaris suurima sõne pikkusega. Sellist suhet nimetatakse Levenšteini suhteks. Andmeid uurides otsustas autor välja arvata kõik need laused, mille Levenšteini suhe oli alla 70, kuna tavaliselt see tähendas, et transkribeerides oli tehtud viga, mistõttu ei olnud otstarbekas seda mudelile õpetada. Õppimise jaoks sobivaid paare oli 438 229. Valimist ei arvatud välja juhtumeid, kus automaatne kõnetuvastaja tuvastas mõne sõna valesti, kuna ükski kõnetuvastaja ei paku 100% transkribeerimise täpsust ja loodeti, et mudel suudab mõne vea tuvastada ja parandada.

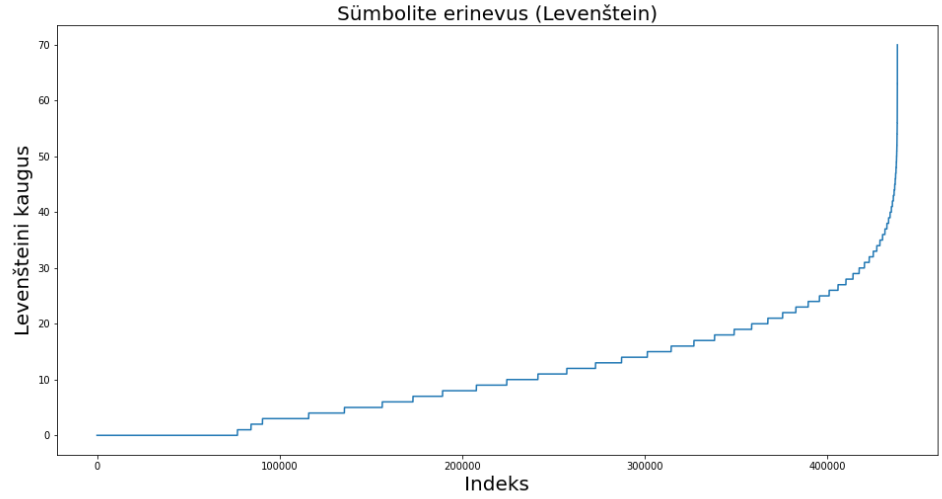
Joonisel 10 selgub, et keskmiselt on subtiitriks toimetatud tekst lühem, kui sellele vastav transkribeeritud tekst. Keskmine vahe on natuke üle kuue sümboli. Seega võib öelda, et keskmisel juhul on toimetatud tekst vähemalt ühe sõna võrra lühem.

[1] `https://huggingface.co/IljaSamoilov`
[2] `https://github.com/IljaSamoilov/subtitle_prettifier`



Joonis 10. Lausete pikkuse jaotuse diagramm.

Joonisel 11 on näha, et vaatamata keskmisele pikkuse vahele, mis on ainult kuus sümbolit, oli keskmiselt siiski vaja 11 toimingut, et omavahel võrdsustada tekstid, kus kõik sõnad on viidud algvormile. Andmestikku otsustati sisse jätta ka lausepaarid, kus Levenšteini kaugus oli võrdne nulliga, mis õpetas mudelile, et laused ei vajagi iga kord tihendamist.



Joonis 11. Levenšteini kauguste jaotuse diagramm.

Terve andmestik jagunes kolmeks osaks:

- treeningandmestik, mis koosnes 416 317 eksemplarist. Seda korpust kasutati mudelite treenimiseks, millel oli ligipääs sisendile ja soovitud väljundile;
- valideerimisandmestik, mis koosnes 10 956 eksemplarist. Korpust kasutati mudeli vahetulemuste arvutamiseks, mis aitas aru saada, kas mudeli õppimisiteratsioonide tulemus on piisavalt üldine, et mudel saab hakkama ka andmetega, mida ei kasutatud treenimiseks;
- testandmestik, mis koosnes 10 956 eksemplarist. Korpust kasutati mudelite võrdlemiseks andmete peal, mida mudelid ei ole varasemalt sisendiks saanud.

Kõikide mudelite treenimiseks, valideerimiseks ja testimiseks kasutati samasuguseid korpuseid.

5.1.1 Loomuliku keele mõtestamine mudelite jaoks

Matemaatilised algoritmid ei ole suutelised aru saama sõnadest ja lausetest töötlemata kujul. Selle probleemi ületamiseks luuakse sõnastikke, kus igale sõnale on määratud number. Neid numbreid kasutades võib algoritmile esitada loomuliku keelt. Selle lähenemise puhul tekib probleeme haruldaselt esinevate sõnadega ja morfoloogiliselt rikaste keeltega nagu eesti keel. Algoritm ei tea, mida teha mitte kunagi nähtud sõnadega. Näiteks sõnad „kirjutama“ ja „kirjutada“ on sellise lähenemise puhul algoritmi jaoks erinevad, aga inimese jaoks tulenevad need samast sõnast ning nende tähendus on sarnane.

Nende sõnade ühendamiseks kasutatakse alamsõnade algoritme (ing k *subword algorithms*), mis otsivad ühiseid sõnaosasid ja ühendavad need omavahel. Magistritöös kasutatakse täpsemalt lihtsat andmete pakkimise meetodit *byte pair encoding* (BPE), mis asendab iteratiivselt jadas kõige sagedasema baidipaari ühe kasutamata baidiga. Loomuliku keele jaoks on antud algoritm kohandatud sõnade segmenteerimiseks. Sagedaste baidipaaride ühendamise asemel ühendatakse tähemärke või nende järjestusi [42]. See lähenemine on laialt kasutatav ja kujunenud loomuliku keele töötlemise standardiks. See ei ole aga optimaalne lähenemine eeltreenimiseks ja on saanud selle osas ka omajagu kriitikat [43]. Vaatamata sellele on siirdeõppe jaoks kasutatavad mudelid juba antud lähenemist kasutades niikuinii eeltreenitud ja seq2seq mudelite jaoks on see lähenemine optimaalne.

5.2 Tihendamise mudelid

Kõikide mudelite tulemusi mõõdeti sama andmestiku ja samade mõõdiku funktsioonidega. Kuigi iga treenimiseks kasutatud tarkvarapakett pakub juba tulemuste mõõtmist, ei saa ikkagi neid tulemusi nimetada omavahel võrreldavateks. Näiteks mõõdiku BLEU implementatsioon võib erineda sõltuvalt tarkvarapaketist. BLEU'l on olemas ka parameetrid, mida tihti ei täpsustata teadustöödes. Andmete töötlemisel on samuti suur mõju mõõdikutele: erinevaid töötlusviise kasutavad tulemused ei ole omavahel võrreldavad [44]. Seetõttu ka magistritöös mõõdeti kõikide mudelite tulemusi sama funktsiooniga ja ka mõõdikute parameetrid olid samad.

Seq2seq mudelite treenimiseks kasutati neile vastavaid tarkvaratööristade kogumeid. Siirdeõppe mudeleid treeniti kasutades Huggingface teeki, mis osutus valituks, sest see võimaldab eeltreenitud mudeleid alla laadida, vahetada mudeli pead ehk väljundi tüüpi ning mudeleid häälestada kasutades treenimise liidest. Tulemuste mõõtmine ehitati samuti Huggingface teegi abil, kuna see pakub kerget liidest mõõdikute allalaadimiseks ja kasutamiseks [45].

Edaspidi esitleb autor tulemuste võrdlemiseks iga mudeli kohta automaatmõõdikute tulemusi ja valitud näiteid mudelite väljunditest. Näidetes tähendab „Sisend“ automaatse kõnetuvastaja poolt valminud transkribeeritud teksti, „Võrdlusalus“ toimetaja poolt valminud vastavat subtiitrite lõiku ja „Väljund“ mudeli tihendamise tulemus.

Kõik mudelid treeniti kasutades ühte NVIDIA Tesla P100 videokaarti, mis oli kättesaadav tänu Google Colab Pro'le. Treeningpakkide (ing k *batch*) suurused sõltusid antud ressursside kitsendusest ja videokaardi tootja poolt soovitatud paki suurusest. Videokaardi piirangute tõttu ei olnud treenimise kiirendamiseks võimalik kasutada poole täpsusega ujukomaesitust (fp16, ing k *half-precision floating-point*), mis on üks populaarseim kiirendamise viise [46]. Iga mudeli treenimisperiood kestis vahemikus 2–12 tundi.

5.2.1 Tihendamata paarid

Tihendamata paaride mudel tagastab sisendiga võrdse väljundi ehk kirjeldab *status-quo*'t. Lisaks näitab see, kui hästi on paarid omavahel seotud. Edaspidi võrreldakse iga väljatöödeldud mudelit kõigepealt *status-quo* mudeliga.

Olemasolevad paarid (Tabel 2) on juba väga sarnased ning seda näitab nii kõrge METEORi kui ka ROUGE-i väärtus. Neid andmeid on aga raske võrrelda varasemate töödega, kuna peatükis 3.1 kirjeldatud töös kasutatakse ainult tookenite täpsust. Peatükis 3.2 kirjelda-

tud töös on küll ROUGE arvutatud, kuid seal lisanduvad lisaks tihendamise vigadele teistsuguse automaatkõnetuvastaja vead. Seega ka neid väärtusi ei saa omavahel võrrelda.

Tabel 2. Status-quo mudeli tulemused.

Mudel	BLEU	METEOR	R-1	R-2	R-L	R-LSUM
status-quo	41,27	0,75	80,67	64,00	78,95	78,97

Kõikide mõõdikute puhul on tähtsad mitte absoluutsed väärtused, vaid erinevus *status-quo* mudeliga. Seega isegi väiksed väärtuste tõusud näitavad, et mudel on suutnud muuta oma väljundi sarnasemaks toimetaja tekstile.

5.2.2 Fairseq seq2seq mudel

Fairseq on spetsialiseeritud tarkvaratööristade kogum seq2seq mudelite loomiseks ja treenimiseks. See sisaldab kõike vajalikku, et luua masintõlke mudel *transformer*'i arhitektuuri baasil [14]. Selle mudeli treenimiseks kasutati masintõlke jaoks soovitatud seadistust, mis koosneb kihilisest *transformer* mudelist, kuuest kooderi ja kuuest dekooderi kihist. Lisaks oli selle õppimiskiiruseks seatud $5e^{-4}$ ning ülesobitamise vältimiseks kasutati *dropout* kihte. Andmed valmistati ette kasutades Moses [47] tarkvarapaketti, nagu soovitavad ka Fairseq-i autorid. Sõnastik koosnes 32 tuhandest BPE-st ning kao funktsioonina kasutati ristentroopiakahju funktsiooni koos märgendite silumisega (ing k *label smoothed cross entropy*).

Mudeli tulemustele (Tabel 3) tuginedes võib öelda, et isegi tavalist seq2seq *transformer*'it kasutades on juba võimalik saavutada esmast lausete tihendamist. Üldjuhul saab selline mudel hakkama liigsete sõnade eemaldamisega ja harva ka ümbersõnastamisega. See tuleneb mudeli ebapiisavast keele mõistmisest, mida põhjustab piiratud andmestik ja eeltreenimise puudumine.

Tabel 3. Fairseq mudeli tulemused.

Mudel	BLEU	METEOR	R-1	R-2	R-L	R-LSUM
status-quo	41,27	**0,75**	80,67	64,00	78,95	78,97
Fairseq	**42,41**	0,75	**81,43**	**64,95**	**79,71**	**79,74**

Mõned näited Fairseq mudeli väljundist on leitavad Tabelist 4. Põhiraskusteks antud mudeli puhul on numbrid ja nimed. Kuigi BPE aitab seda probleemi vähendada, tekivad siiski vead, kui leidub tekstitükke, mida mudel varasemalt näinud ei ole ja mõnel juhul on väljund täis teadmata sümboleid.

Tabel 4. Fairseq mudeli tihendamise näited.

Sisend	selles mõttes, et me olemegi, ma olen kindlasti nõus, et me oleme sellisel teataval määral stabiilsel platool
Võrdlusalus	Ma olen nõus, et me oleme sellisel teataval määral stabiilsel platool
Väljund	ma olen kindlasti nõus, et me oleme sellisel teataval määral stabiilsel platool.

(a) Eduka tihendamise näide

Sisend	see circa 760-le väikeaktsionärile, kes on, teeme ettepaneku aktsiad müüa
Võrdlusalus	ca 760 väikeaktsionärile ettepaneku aktsiad müüa
Väljund	see on umbes 60% fyleine väikeaktsionärile, kes on ettepaneku aktsiad müüa.

(b) Vigase tihendamise näide

Fairseq mudeli probleemiks on ka vahepeal tekkivad pikad väljundid, kus üks sõna kordub ja väljund on subtiitriks kasutamatu. Sellisele käitumisele ei ole autor seletust leidnud ning seda ei ole ka dokumentatsioonis. Üks selline näide on leitav Tabelist 5.

Tabel 5. Fairseq mudeli vigase väljundi näide.

Sisend	et me ei saa ju, et üks mees nii-öelda jookseb ja teised mehed sõidavad tankiga.
Võrdlusalus	Me ei saa nii, et üks mees jookseb ja teised sõidavad tankiga?
Väljund	me ei saa ju, et üks mees nii f-isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, isa, nagu giyyyyyyyyyyyyyyyyyyyyyyyy[...],

5.2.3 Joey NMT seq2seq mudel

Joey NMT on Fairseq-iga sarnane tarkvaratööristade kogum, mis on küll mõeldud algajatele masinõppes, kuid pakub palju paindlikkust mudelite ehitamiseks. Kogum võimaldab seadistada erinevaid parameetreid: kuidas seda treenida, testida, kui palju kihte või võrgupäid on kooderil või dekooderil jne. Samuti nagu eelmise mudeli puhul, koosneb sõnastik 32 tuhandest BPE-st. Kihtide arv on Fairseq-iga võrdne, kuid neuronite arv on kaks korda väiksem: 512 asemel 256. Joey NMT mudelite eksemplaridest parima leidmiseks kasutati ka *perplexity*'t (ppl), mis näitab, kui hästi mudel suudab ennustada väljundi valimi peal. Lisaks otsustas autor selle mudeli treenimiseks mitte viia andmestikku väiketähtede kujule, vaid säilitada algse kuju [48].

Mudel on eelmistest parem kõikide mõõdikute poolest peale METEORi (Tabel 6), kuid ka selle väärtus on juba kõrge ja raske on eeldada edasist suurt tõusu. Võrreldes eelmise

Fairseq mudeliga saab see mudel paremini hakkama numbritega, tavaliselt ilma tükke kaotamata. Isikute nimed tekitavad samuti vähem probleeme.

Tabel 6. Joey NMT mudeli tulemused.

Mudel	**BLEU**	**METEOR**	**R-1**	**R-2**	**R-L**	**R-LSUM**
status-quo	41,27	**0,75**	80,67	64,00	78,95	78,97
Fairseq	42,41	0,75	81,43	64,95	79,71	79,74
Joey NMT	**48,09**	0,74	**82,22**	**65,82**	**80,68**	**80,68**

Suurem osa tihendamisest on siiski saavutad sõnade kustutamise ja ümberpaigutamise abil, aga ümbersõnastamine on haruldane. Võimalik, et üheks edu põhjuseks võrreldes eelmise seq2seq mudeliga on väiksemad võrgukihid, kuna sisend ja väljund ei erine nii palju. Tabelis 7 on näha, et mõnel juhul on mudel võimeline märkama kõnelejate vahetust ilma eelmiseid lauseid nägemata.

Tabel 7. Joey NMT mudeli tihendamise näited.

Sisend	ma arvan, et dekrendiili saladus peitub nädalavahetuse Siim Kallas, Edgar Savisaar kohtumises
Võrdlusalus	EKRE diili saladus peitus Siim Kallase ja Edgar Savisaare kohtumises.
Väljund	Ma arvan, et EKRE saladus peitub nädalavahetuse Siim Kallas, Edgar Savisaar kohtumises.

(a) Eduka tihendamise näide, kus mudel on suutnud parandada kõnetuvastaja vea.

Sisend	arvan, et see ettekujutus, et ERR-i juhatuse esimees peab tema töö peamine sisu, on parkimiskorraldus
Võrdlusalus	Ettekujutus, et ERRi juhatuse esimehe peamine töö on parkimiskorraldus
Väljund	ERR-i juhatuse esimees peab tema töö peamine sisu parkimiskorraldus

(b) Vigase tihendamise näide, kus lause tähendus on kadunud.

Sisend	Aitäh. Praegu Heldur, ma räägin sulle midagi tobedat. No lase aga käia,
Võrdlusalus	Heldur, ma räägin sulle midagi tobedat... - Lase aga käia.
Väljund	Praegu Heldur, ma räägin sulle midagi tobedat. - Lase aga käia.

(c) Vahel oskab mudel leida ka kõnelejate vahetuse.

5.2.4 Liigseid sõnu märgendav mudel EstBERT-i baasil

Mudeli baasina kasutati eestikeelse korpuse peal eeltreenitud EstBERTi. Eeltreenimiseks kasutati peidetud sõnade ja järgmise lause ennustamist, nagu on ka kirjeldatud originaalses BERTi artiklis [34, 4]. Mudeli eesmärk on anda igale tookenile märgend, mis näitab, kas tooken peaks jääma lausesse või mitte. EstBERTil on juba olemas tookenite klassifitseerimise mudelid, mis näiteks määravad sõnaliikmeid, kuid need on juba treenitud kindla eesmärgiga ja nende ümberõpetamine oleks ebatõhus. Seetõttu valiti siirdeõppe baasmudeliks peidetud sõnu ennustav mudel. Väljundi tüübi muutmiseks vahetati mudeli pea, mis andis võimaluse säilitada keelt mõistvad peidetud kihid. EstBERT kasutab sõnastiku loomiseks samuti BPE lähenemist.

Mudeli treenimiseks pidi algselt viima võrdlusaluse ehk viiteteksti märgendite kujule. Igale sõnale ja kirjavahemärgile anti märgend 1 või 0 vastavalt sellele, kas need esinevad nii sisendis kui ka viitetekstis või ainult sisendis. Probleemseks osutusid olukorrad, kus toimetaja oli mõnda sõna muutnud või automaatne kõnetuvastaja oli vea teinud. Pythoni standardteegi klassi SequenceMatcher [49] muudeti nii, et tähthaaval võrdlemise asemel võrreldi sõnu hoopis Levenšteini suhte põhjal. Sõnade ja kirjavahemärkide asukoht võeti samuti arvesse, et ennetada nende kordumised siis, kui sisendis ja viitetekstis on nende arv erinev. Kuna mudel kasutab sõnastiku loomiseks BPE-d siis viiteteksti märgendeid oli vaja laiendada igale sõnaosale.

Kõikide teiste mudelite puhul taastati väljundi tekstiline kuju kasutades mudeli sõnastikku, kuid EstBERTi sõnastikus on suurel osal tookenitel täpitähed kaotatud. See mõjutab mudeli täpsust märkimisväärselt, kuna viitetekstis on need tähed olemas ja mõõdikud on sellise muutuse osas tundlikud. Samuti on raske nimetada eestikeelseid subtiitreid ilma täpitähtedeta kvaliteetseteks. Seepärast taastati tekstiline kuju sisendi põhjal, kasutades tokeniseerija algset infot selle kohta, milline tooken kulub millisele sõnale.

EstBERT mudeli (Tabel 8) ROUGE mõõdikud on ootustele vastavalt kõrged ekstraheeriva kokkuvõtmise puhul, kuna mudel ei muuda sõnu enda sees vaid annab väljundiks neid samu, mis leiduvad ka sisendis. Seevastu BLEU ja METEOR on oluliselt nõrgemad, kui seq2seq mudelitel, mis tuleneb sellest, et toimetajad muudavad tihti viitetekstides sõnade järjekorda, aga antud mudel ei ole selleks võimeline. Lisaks on kõnetuvastaja vead palju suurema kaaluga ekstraheeriva kokkuvõtmise puhul, kuna puudub võimalus sõnu parandada.

Tabel 8. EstBERTi baasil liigseid sõnu märgendava mudeli tulemused.

Mudel	BLEU	METEOR	R-1	R-2	R-L	R-LSUM
status-quo	41,27	**0,75**	80,67	64,00	78,95	78,97
Joey NMT	**48,09**	0,74	82,22	65,82	80,68	80,68
EstBERT *mudeli sõnastik*	27,06	0,58	63,02	40,21	61,84	61,82
EstBERT *taastatud sõnad*	46,85	0,73	**83,28**	**67,16**	**81,64**	**81,64**

Kasutades mudeli sõnastiku ei olnud võimalik saavutada vastuvõetavat tulemust. Peamiselt põhjustasid seda poolikud sõnad ja täpitähtede puudumine. Samuti ei saanud antud mudeliga parandada transkribeerimisvigu, kuna uut tekstilist väljundit ei genereeritud. Mõned EstBERT mudeli tihendamise näited on leitavad Tabelis 9.

Tabel 9. EstBERT mudeli tihendamise näited.

Sisend	kes hakkavad siis kiiresti toituma ja jõuavad siis jälle välja lõpuks selle nukufaasi
Võrdlusalus	kes hakkavad kiiresti toituma ja jõuavad jälle välja nukufaasi.
Väljund	kes hakkavad kiirest toituma ja jõuavad välja nukufaasi

(a) Eduka tihendamise näide.

Sisend	Eesti 200-l on plaanis, et hoopis
Võrdlusalus	Eesti 200-l on plaanis hoopis
Väljund	Eesti on plaanis

(b) Vigase tihendamise näide, kus lausejupi tähendus on kadunud.

5.2.5 MBART-50

MBART on mitmekeeleline versioon BARTist, mida on treenitud taastades rikutud tekste 25 keelest. Eelnevad uuringud on näidanud, et mitmekeelsed mudelid on võimelised paremini keelt modelleerima ja mõistma. Vaatamata sellele, et MBARTi eesmärk on masintõlge, on mudelit õpetatud juhendamata stiilis monolinguaalsetel andmestikel. Kuigi mudel oskab kõiki eeltreenimise käigus kasutatud keeli, siis kasutamisel on soovitatud mudelit eeltreenida just nende keelte peal, mida on plaanis kasutada. Eesti keel on samuti üks nendest 25 keelest, mida kasutati mudeli treenimiseks. MBARTist on loodud ka 50 keele mudel MBART-50, mille tulemused on veelgi paremad [13, 50].

Selliseid mudeleid, saab kasutada peale masintõlke ka muude loomuliku keele ülesannete jaoks. Magistritöö autor kasutab sisendteksti tihendamiseks MBART-50 mudelit. Selle jaoks on võimalik mudelit panna teksti muutma ehk tõlkima ainult ühes keeles ehk sisend- ja väljundkeel on samad.

Tõestamaks, et mudeli häälestamine on tähtis ja näitab paremaid tulemusi (Tabel 10), lisati võrdlusesse treenimata versioon MBART-50 mudelist. Häälestamata mudel näitas palju halvemaid tulemusi võrreldes teistega ja oli isegi halvem transkribeeritud tekstist.

Tabel 10. MBART-50 mudeli tulemused.

Mudel	**BLEU**	**METEOR**	**R-1**	**R-2**	**R-L**	**R-LSUM**
status-quo	41,27	0,75	80,67	64,00	78,95	78,97
Joey NMT	48,09	0,74	82,22	65,82	80,68	80,68
EstBERT	46,85	0,73	83,28	67,16	81,64	81,64
MBART-50 *Treenimata*	41,24	0,71	78,14	61,59	76,41	76,44
MBART-50	**52,40**	**0,77**	**83,64**	**68,19**	**82,04**	**82,04**

Treenitud mudeli tulemused, mille näited on leitavad Tabelis 11, olid eelmistest oluliselt paremad. Seda võib seostada parema keele mõistmisega. Miinuseks võib tuua mudeli kiiruse. MBART-50 oli kõvasti aeglasem EstBERTist ja natuke aeglasem seq2seq mudelitest.

Tabel 11. MBART-50 mudeli tihendamise näited.

Sisend	olin ja see Jaapani 20 aastat tagasi täpselt. Naganos olin võistkonna esindaja
Võrdlusalus	See oli täpselt 20 aastat tagasi. Naganos olin ma võistkonna esindaja.
Väljund	20 aastat tagasi olin Naganos võistkonna esindaja.

(a) Eduka tihendamise näide.

Sisend	siis praegu on olemas vahendeid üle 400000. Nii et meie Sävendame
Võrdlusalus	siis praegu on vahendeid üle 400 000, et seda teha.
Väljund	siis praegu on vahendeid üle 40 000.

(b) Vigase tihendamise näide, kus mudel kaotas numbri ja eemaldas arusaamatu osa.

5.2.6 MBART-50 koos subtiitri ajavahemikuga

Autori eeldas, et andes mudelile ette teave, kui kaua kuvatakse subtiitrit ekraanil, on võimalik aidata mudelil ennustada, kui pikk peaks olema väljundsubtiiter. Lausete kauguse ja subtiitri kestvuse numbriline korrelatsioon on 0.47, mis on küll nõrk, aga võib siiski mudelit aidata. Lisaks sisendi ning oodatava väljundi sõnede pikkuste korrelatsioon subtiitri kestvusega on 0.45, mis näitab, kas subtiitritest sõnade eemaldamine üldse mõjutab subtiitrite ekraanil kuvamise aega.

MBART mudelile ei saa uut sisendit lihtsalt lisada, kuna see on juba eeltreenitud, kuid on võimalik manipuleerida, kust mudel alustab väljundi genereerimist. MBART ootab sisendandmeid vektorkujul, kus genereerimise algus ja lõpp on märgistatud spetsiaalse tookeniga. Sisendi algusesse lisati subtiitri kestvus sekundites (sõnena) kodeeritud kujul ja tähelepanuvektorit suurendati selle tookeni võrra. Mudel võtab ikkagi arvesse sekundite informatsiooni, kuid ei alusta kunagi sellest genereerimist. Antud lähenemine sai innustust sellest, kuidas BERT mudeli juurde on võimalik kodeerida lisainfot.

Mudel oli võrdväärne tavalise MBART-50 mudeliga ja mõõdikute väärtuste erinevusi võib lugeda statistiliseks veaks (Tabel 12). Seda võib põhjustada mitu asjaolu. Esiteks on võimalik, et lause kuju ja pikkus juba ütlevad mudelile piisavalt täpselt, kui palju ja mida on vaja lauses muuta. Teiseks on võimalik, et antud teave ei ole mudelile piisav.

Tabel 12. MBART-50 koos subtiitri ajavahemikuga mudeli tulemused.

Mudel	**BLEU**	**METEOR**	**R-1**	**R-2**	**R-L**	**R-LSUM**
status-quo	41,27	0,75	80,67	64,00	78,95	78,97
Joey NMT	48,09	0,74	82,22	65,82	80,68	80,68
EstBERT	46,85	0,73	83,28	67,16	81,64	81,64
MBART-50	**52,40**	**0,77**	83,64	**68,19**	82,04	82,04
+ kestvuse teave	52,26	0,77	**83,64**	68,19	**82,06**	**82,06**

Siiski uurides mudelite väljundeid, mille näited on leitavad Tabelis 13, on erinevused suurimad siis, kui subtiitri kestvus ekraanil on pikem. Näiteks ajavahemiku informatsiooniga mudel ei muuda tihti rohke sümbolite arvuga lauseid nii palju, kui tavaline MBART mudel, aga ekraanil kuvamise kestvus on keskmisest madalam. Võib järeldada, et mudel ikkagi võtab ajavahemiku arvesse, aga see ei ole piisav parema tulemuse saavutamiseks.

Tabel 13. MBART-50 + subtiitri aeg ekraanil mudeli tihendamise näited.

Sisend	Nüüd, liikumise puhul on lugu selline, et et siin me pea jooksma nii nagu Erki Nool või hüppama nagu Erki Nool,
Võrdlusalus	Liikumisega on lugu nii, et me ei pea jooksma või hüppama nagu Erkki Nool.
Väljund	Liikumise puhul on lugu selline, et siin me ei pea jooksma nagu Erki Nool,

(a) Eduka tihendamise näide.

Sisend	ütleme nii, et see laia äärega kübar on üks, üks nooremate inimeste fas songaga,
Võrdlusalus	Ütleme, et laia äärega kübar on nooremate inimeste fassong.
Väljund	Ütleme nii, et see laia äärega kübar on üks nooremate inimeste faas.

(b) Vigase tihendamise näide, kus mudel on muutnud lause tähendust huvitaval kombel.

6. Tulemuste analüüs ja järeldused

Tuginedes automaatmõõdikute tulemustele (Tabel 14) võib öelda, et loomuliku keele mudeleid kasutades on võimalik luua transkribeeritud tekstist käsitsi toimetatud subtiitritega ligilähedase kvaliteediga tulemus. Kõige paremini teevad seda häälestatud siirdeõppe mudelid. Järelikult on subtiitrite tihendamise puhul väga tähtis ka hea andmestik ning selle laiendamine võib tulemusi veelgi parendada, ent juba praeguse magistritöö andmestiku suuruse juures on siirdeõppe mudelite treenimine aeganõudev protsess. Veelgi suurema andmestiku juures see võib osutada eriti probleemseks.

Tabel 14. Tihendamise mudelite koondtulemused.

Mudel	**BLEU**	**METEOR**	**R-1**	**R-2**	**R-L**	**R-LSUM**
status-quo	41,27	0,75	80,67	64,00	78,95	78,97
Fairseq	42,41	0,75	81,43	64,95	79,71	79,74
Joey NMT	48,09	0,74	82,22	65,82	80,68	80,68
EstBERT	46,85	0,73	83,28	67,16	81,64	81,64
MBART-50	41,24	0,71	78,14	61,59	76,41	76,44
Treenimata						
MBART-50	**52,40**	**0,77**	83,64	**68,19**	82,04	82,04
+ kestvuse teave	52,26	0,77	**83,64**	68,19	**82,06**	**82,06**

Probleemid, mis mudelite juures veel lahendamist vajavad:

- Mudelid kaotavad tihti arvude puhul numbrid ära. Esimest tüüpi vead tekkisid peamiselt rohkem kui neljakohaliste arvude puhul, kus näiteks kaotakse saja tuhande puhul üks null ära. Teiseks veatüübiks oli numbrite täielik kadumine tekstist. Kahjuks ei ole võimalik loomuliku keele mudelite arhitektuuri tõttu kontrollida, miks mudel sellise otsust tegi.
- Kõnelejate vahetumisele vastava märgi lisamine sinna, kus seda tegelikkuses ei juhtu. Mudelil puudub kontekst, kus kõneleja on vahetunud, ja üritab seda ennustada puhtalt lause struktuuri pealt. Sellist viga võib seostada andmestiku ettevalmistusega, mille käigus selline harva andmestikus esinev informatsioon välja lõigatakse.
- Andmestik ei ole ühekülgne. Subtiitrite toimetajad küll tihendavad tekste, kuid teevad seda erinevalt, ja toimetamisviisid erinevad saatest saatesse. Siinkohal oleks võimalikuks lahenduseks näiteks saadete liigitamine teema järgi, millest on rohkem kirjutatud alampeatükis 6.2.

Magistritöö suurimaks saavutuseks võib nimetada selle käigus valminud mudeleid, mida saab kasutada toimetajate abistamiseks. Täielik tihendamise automatiseerimine ei ole veel võimalik. Seda nii eelnimetatud probleemide kui ka grammatikavigade tõttu. Telesaate transkribeerimine võtab natuke vähem aega, kui saade ise kestab, ning keskmiselt läheb MBARTi mudelil ühe saate toimetamiseks kuni 2 minutit, mis on oluliselt kiirem käsitsi toimetamisest. Lisaks on ka võimalik antud protsess muuta paralleelseks ja toimetada mitut saadet korraga.

Massiline subtiitride tootmine võib ka kergendada eesti keele õppimist [7]. Eesti keele õppimise tõhustamiseks oleks võimalik ka kõigile eelsalvestatud saadetele lisada juba valmis tehtud samakeelsed subtiitrid. Otse-eetris näidatavate saadetega oleks see natuke raskem, kuid siiski võimalik. Praegune Tallinna Tehnikaülikooli lahenduse [51] viivitus on umbes 3 sekundit, millele lisanduks veel töös valminud mudeli viivitus kuni 3 lisasekundit. Tavavaataja jaoks oleks selline asi segav. See oleks aga kergelt lahendatav, kui saadet näidatakse väikese viivisega, et jõuaks subtiitreid automaatselt genereerida.

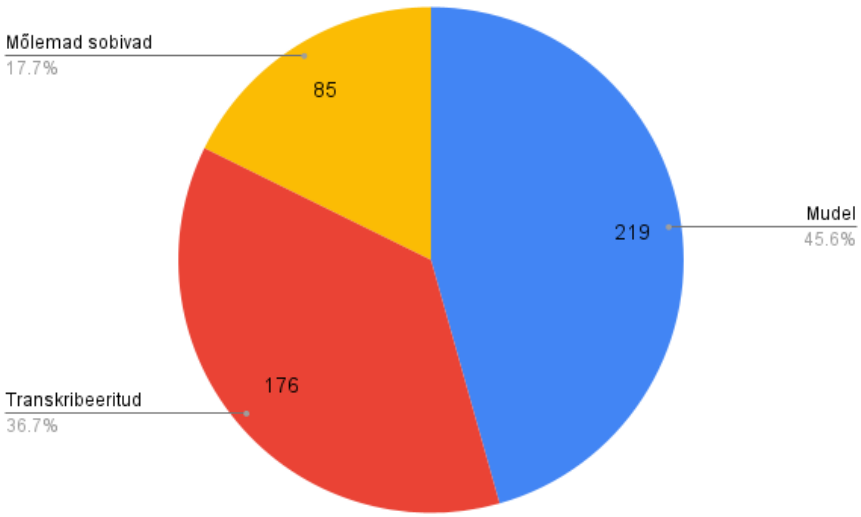
Töö teiseks saavutuseks võib nimetada andmestiku loomist. Kasutades töös olevaid algoritme, võib luua aina suuremat ja paremat andmestikku ja lisades väikseid muudatusi, saab seda kasutada mitte ainult eesti, vaid ka teiste keelte jaoks.

Lisaks võib tõdeda, et töös läbiviidud küsitluse käigus ei kogutud piisavalt informatsiooni, et täheldada, millised on vaatajate eelistused subtiitrite stiili osas. Vastanuid ei liigitatud ka emakeele põhjal, kuid autor usub, et kvaliteetsed subtiitrid on oluliselt suurema tähendusega neile inimestele, kelle emakeel ei ole eesti keel.

6.1 Küsitluse tulemuste analüüs

Uuringu jaoks valiti kasutamata andmete seast kahe telesaate episoodid: „Kahekõne: Ain Seppik“ ja „Prillitoos: 275“. Uuringu eesmärk oli aru saada, kas magistritöös väljatöötatud masinõppe mudeli väljund sobib subtiitriks rohkem kui otse kõnetuvastajaga transkribeeritud tekst. Uuringus osalenud mudel ei olnud kunagi varem uuringus kasutatud tekstilõike näinud. Mudeliks valiti MBART-50, kuna see näitas kõige paremaid automaatmõõdikute tulemusi. Lõigud valiti enne mudelipoolset töötlemist ega olnud töö autori poolt kuidagi muudetud. Ebaausa võrdluse vältimiseks ei valitud lõike, kus automaatne kõnetuvastaja tegi vea. Uuringus osalejad pidid vaatama kuut videolõiku ja valima, kumba kahest pakutud subtiitrist nad eelistasid. Sealjuures vastajad ei teadnud, millisel viisil valikus olevad subtiitrid loodi.

Uuringus osales 80 vastajat. See viidi läbi Google Forms keskkonnas. Uuringus kasutatud küsitlusega saab lähemalt tutvuda Lisas 2. Joonisel 12 on kujutatud vastajate arvamusi üle kõigi küsimuste. Mudeli poolt loodud subtiitreid eelistati 45.6% juhtudel, mis oli ka kõige populaarsem variant. Kuna mõlemat viisi loodud subtiitrid sobisid vastajatele 17.7% juhtudest, siis kokku 63.3% juhtudel eelistati mudeli poolt loodud subtiitreid vähemalt sama palju, kui automaatselt transkribeeritud teksti. Siiski üldtulemuste peal ei ole võimalik ühtselt öelda, et mudel on oma eesmärgi saavutanud. Seda põhjustavad ka uuringu jaoks valitud subtiitrite näited. Neid valides proovis autor kajastada erinevaid juhtumeid, k.a erandlikke, mida tuleks pigem vaadelda teistest lahus.



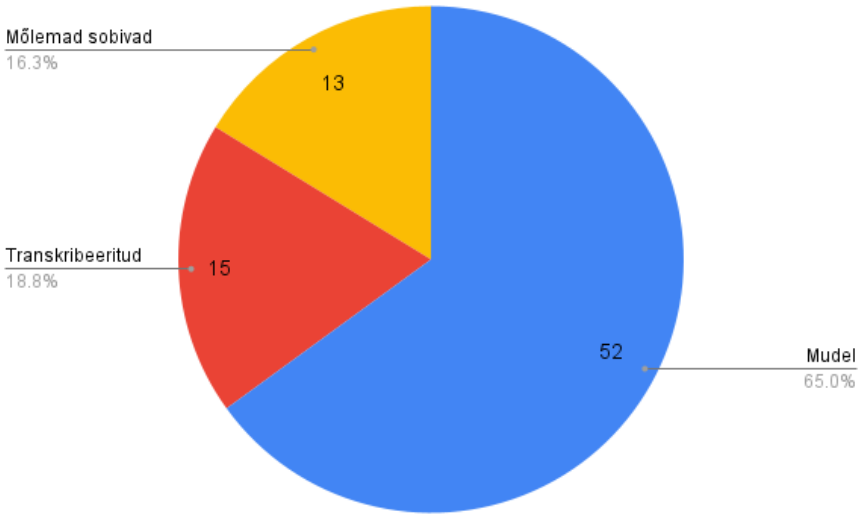
Joonis 12. Uuringus vastajate eelistused kokku.

Video 1 subtiitrid (Tabel 15) said valituks seetõttu, et toimetaja poolt loodud subtiiter on palju erinevam võrreldes transkribeeritud tekstiga. Võrdlusalune tekst erineb nii sõnade järjekorra kui ka pikkuse poolest.

Tabel 15. Video 1 subtiitrid.

Sisend	Need salmid, salmid on, on mõned 100 aastat vanad, mis siin kirjutatakse üksteisele.
Võrdlusalus	Mõned salmid, mida siin kirjutatakse, on sada aastat vanad.
Väljund	Need salmid on mõned sajad aastad vanad, mis siin üksteisele kirjutatakse.

Joonisel 13 on näha, et suurem osa uuringus osalejatest eelistas mudeli poolt loodud subtiitrit, kuigi see erines oluliselt toimetaja omast. Numbrite muutmine tekstiks oli alguses üllatuseks, kuna seda tavaliselt ei juhtunud. Andmeid lähemalt analüüsides selgus, et subtiitrite toimetajad asendavad tihti osa numbreid sõnadega, mis vastab ka eesti keele õigekirjareeglitele [52]. Mudel on suutnud antud käitumist korrata ja muuta sõnade järjekorra lugemiseks sobivamaks.



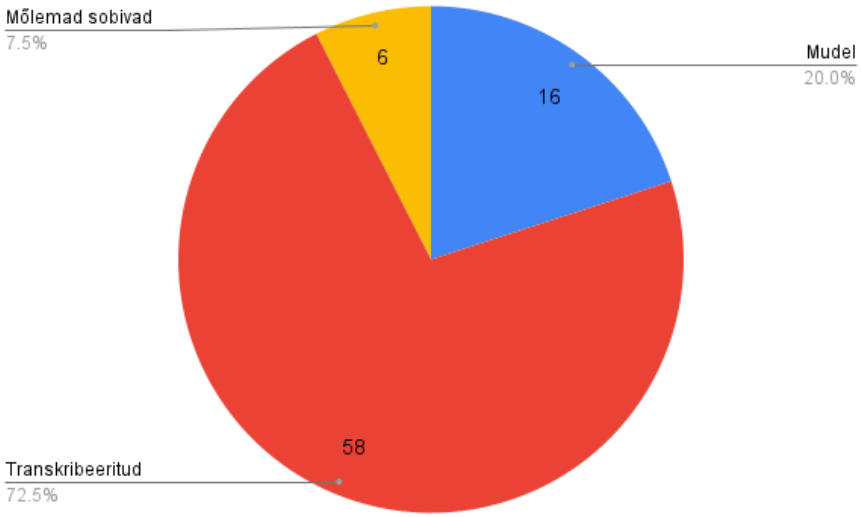
Joonis 13. Video 1 vastajate eelistused.

Video 2 subtiitrid (Tabel 16) valiti, sest autor täheldas, et toimetajad eristavad kahte kõnelejat sidekriipsuga ning mudelid on suutnud seda käitumist korrata. Lisaks oli huvitav näha, millised olid selle osas vastajate eelistused ning kas sidekriipsu eesmärk oli koheselt mõistetav.

Tabel 16. Video 2 subtiitrid.

Sisend	Kes selle arutelu võiks läbi viia? Noh, tegelikult püüti panna punti kokku
Võrdlusalus	kes peaks selle arutelu läbi viima? - Tegelikult püüti panna punti kokku
Väljund	Kes võiks selle arutelu läbi viia? - Püüti panna punti kokku,

Vaatamata sellele, et mudel suutis tuvastada kaks kõnelejat ja vähendas sõnade arvu ilma tähendust kaotamata, oli transkribeeritud variant osalejatele meelepärasem. Täpsemaid tulemusi on näha Joonisel 14. Selle üheks põhjuseks võib tuua emotsiooni kadumise sõnade arvu vähenemise tõttu. Mudel on eemaldanud tähtsa sõna „tegelikult“, mis on tihti küll parasiitsõna, kuid antud subtiitris on sel rõhutamise eesmärk. Lisaks ei saa kindlalt väita, et vastajad panid sidekriipsu olemasolu üldse tähele või pidasid seda sõna „tegelikult“ puudumisest ebaolulisemaks.



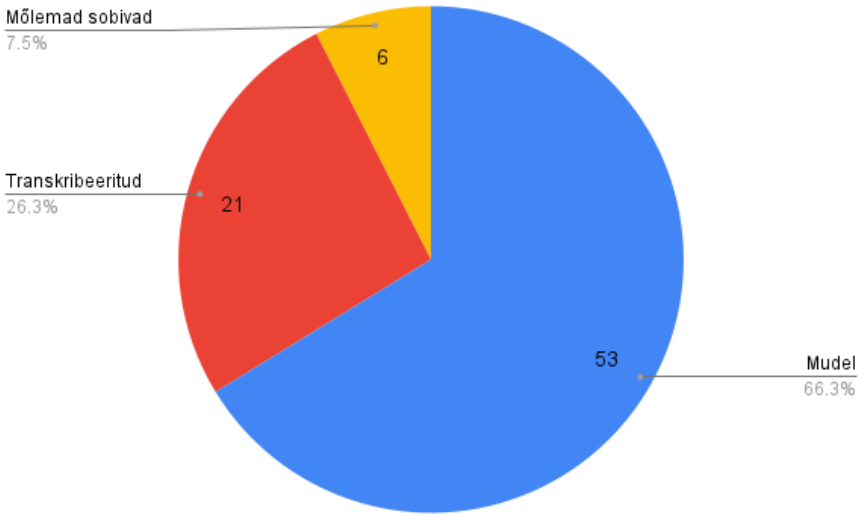
Joonis 14. Video 2 vastajate eelistused.

Video 3 oli teistest erinev, kuna koosnes kahest subtiitrist korraga (Tabel 17). Antud küsimuse eesmärk oli välja selgitada, kas mudel on võimeline järjestikuste lausetega hakkama saama ja tähendust edasi kandma. Lisaks olid suured erinevused transkribeeritud sisendi ja võrdlusaluse teksti vahel nii sõnade järjekorra kui ka parasiit- ja kordussõnade poolest.

Tabel 17. Video 3 subtiitrid.

Sisend	No juba aimab, et pidasin ennast ikkagi, et ma olen juba noh, jalg on seal ukse vahel selle, selle, selle, selle oskuste ja võimete ja, ja tahtmise ja, ja. Teadmiste poolest.
Võrdlusalus	Ma uskusin, et mul on jalg ikkagi uste vahel, oskuste ja võimete ja teadmiste ja tahtmiste poolest.
Väljund	Juba aimab ennast, et jalg on seal ukse vahel. oskuste, võimete, tahtmise ja teadmiste poolest

Video 3 subtiitrite eelistuste tulemused on leitavad Joonisel 15. Enamus uuringus osalenutest eelistas mudeli poolt tihendatud subtiitrit, mis oli kergemini loetavam ning ei sisaldanud liigseid sõnu. Vaatamata osalenute eelistusele, ei jätnud mudel peale tihendamist väljundteksti piisavalt arusaadavaks ning sidus omavahel sõnu, mis alglauses seotud ei olnud.



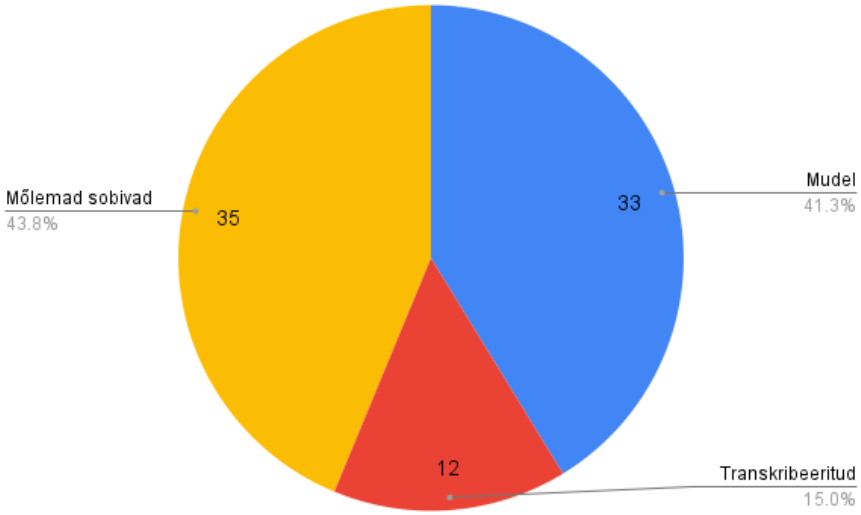
Joonis 15. Video 3 vastajate eelistused.

Video 4 kasutamise eesmärk oli aru saada, kas mudel on võimeline tuvastama, kui palju teksti on üldse muuta vaja, ja leidma sõnu, mida tavaliselt ei kustutata (Tabel 18). Eesnimede kustutamine ei ole toimetamisel just kõige tavalisem asi ehk see subtiiter on haruldane juhtum.

Tabel 18. Video 4 subtiitrid.

Sisend	Et Edgar Savisaar selle sajandi alguses ja Edgar Savisaar täna on kaks erinevat inimest
Võrdlusalus	et Savisaar selle sajandi alguses ja Savisaar täna on kaks eri inimest.
Väljund	Edgar Savisaar selle sajandi alguses ja Edgar Savisaar täna on kaks erinevat inimest

Autori jaoks jääb arusaamatuks, miks ainult ühe sõna kaotamine oli piisav, et 41.3% vastanutest eelistas algset, transkribeeritud subtiitrit rohkem (Joonis 16). Suurem osa oli ikkagi arvamusel, et mõlemad variandid sobivad sama hästi. Vaatamata sellele, et lause on subtiitri jaoks tõesti päris pikk, oli mudel siiski suutnud jätta lause peaaegu algsele kujule, mis ei ole andmestikus nii tihe juhtum.



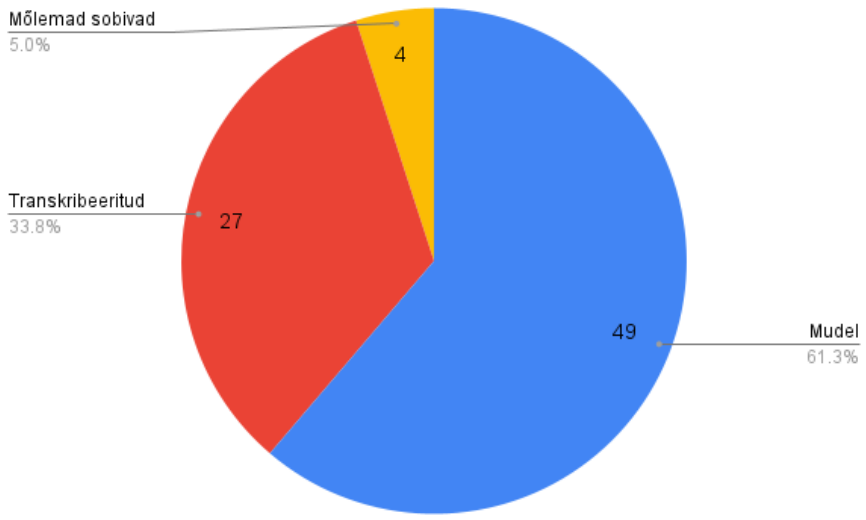
Joonis 16. Video 4 vastajate eelistused.

Video 5 valimise eesmärk oli selgitada välja, kas mudel on võimeline tegema üsna arusaamatu lause subtiiriks, mis siiski annaks mõtet edasi. Lisaks erines võrdlusalune tekst transkribeeritust suurel määral (Tabel 19).

Tabel 19. Video 5 subtiitrid.

Sisend	ja selles mõttes jah, et nüüd ma nagu värvisin veel seda ju see natuke seda, seda absurdsust, mis seal loos on.
Võrdlusalus	Värvisin veel natuke juurde seda absurdsust, mis seal loos on.
Väljund	Nüüd ma värvisin veel natuke absurdsust, mis seal loos on.

Mudel suutis antud lõigu puhul peaaegu korrata võrdlusalust teksti ning jättis kõik vajaliku alles. Suurem osa vastanutest eelistas mudeli poolt loodud subtiitrit, kuid kolmandikule meeldis muutmata kuju ikkagi kõige rohkem. Üheks põhjuseks võib tuua selle, et eesti keelt emakeelena kõnelevad isikud ei ole harjunud nägema eestikeelsetel saadetel samakeelseid subtiitreid, mistõttu nad eelistasid totaalset täpsust lause mõtte edasikandmise asemel. Samuti on võimalus, et osa osalenutest eelistavadki, kui subtiiter on maksimaalselt lähedane saates öeldule. Tulemusi on näha Joonisel 17.



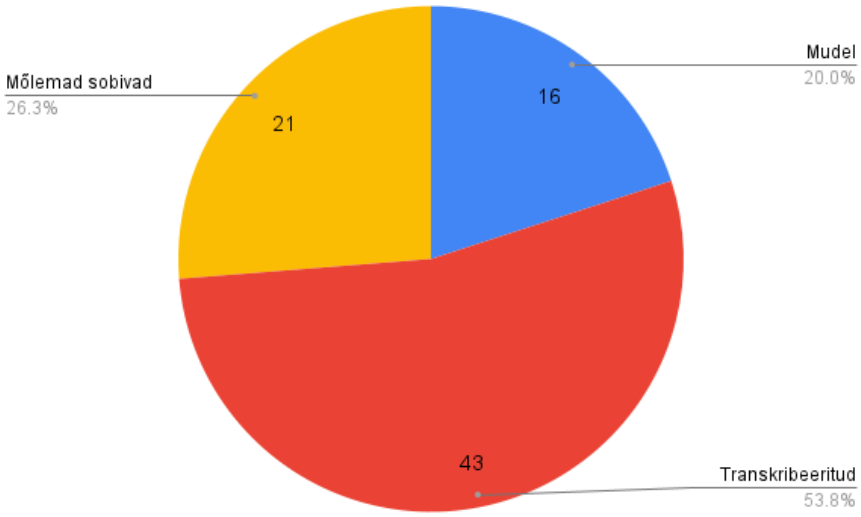
Joonis 17. Video 5 vastajate eelistused.

Video 6 lõik valiti kontrollimaks, kas mudel on peale tihendamist võimeline edasi andma lause emotsionaalset värvikust. Kuigi käsitsi toimetatud tekst on algsest oluliselt lühem, annab see edasi nii öeldu mõtte kui ka emotsiooni. Video 6 subtiitrid on leitavad Tabelis 20.

Tabel 20. Video 6 subtiitrid.

Sisend	Vot ei teagi, millise mudeli juures me oleme. Sest tegelikult ei ole seda mitte kunagi juhtunud,
Võrdlusalus	Ei teagi, mis mudeli juures me oleme. Kunagi varem pole juhtunud
Väljund	Ei teagi, millise mudeli juures me oleme, sest seda pole kunagi juhtunud.

Tuginedes Joonisel 18 leitavatele tulemustele, võib öelda, et enamusele vastanutest Video 6 puhul ei meeldinud mudeli väljund või lugesid seda transkribeeritud tekstiga samaväärseks. Põhjusteks võib tuua lausete kokkuliitmise ning emotsiooni edasikandvate sõnade kaotamise.



Joonis 18. Video 6 vastajate eelistused.

Uuringu tulemuste kokkuvõttena saab öelda, et mudel on võimeline looma automaatselt transkribeeritud tekstist paremaid subtiitreid. Enamus vastanutest ikkagi eelistas neid subtiitreid, kust liigsed sõnad olid eemaldatud ja sõnade järjekord korrastatud. Mudeli probleemina võib välja tuua emotsionaalse või kõneleja isikupärasuse kaotamise, mille parimaks näiteks on video 6 tulemused.

6.2 Võimalikud edasiarendused

ERR-i arhiivis on olemas mitmete tuhandete saadete salvestused koos subtiitritega. ERR-i loa olemasolul on võimalik nende allalaadimine automatiseerida kasutades ERR-i arhiivi API-t. Neid saateid transkribeerides ja subtiitreid kasutades saab luua väga suure ja laia andmestiku, mille abil saaks tõenäoliselt treenida veelgi kvaliteetsemaid mudeleid. Suure andmestiku puhul võib alustada ka saate kategooria arvestamisega, kuna arhiivis on iga saade mõne kategooriaga märgistatud. Nii oleks võimalik luua iga kategooria jaoks oma mudel, mis oleks hea näiteks loodussaadete subtiitrite genereerimiseks. MBARTi ja Huggingface'i loojad soovitavad näiteks kasutada enamasti teemapõhiseid andmestikke [13, 45]. Lisaks on olemas tõestusi, et kasutades teemapõhist eeltreenimist, saab saavutada veelgi paremaid tulemusi [53].

Andmestikust on praegu puudu ka kõnelejate vahetumise informatsioon, mis on tegelikult automaatse kõnetuvastaja väljundis olemas. Selle abil saaks mudelile paremini ette näidata, millal oleks õige rääkija vahetuse korral panna sidekriips.

MBARTile ja sellega sarnastele mudelitele on võimalik anda sisendiks rohkem kui ühte lauset. Õppimiseks ja kasutamiseks saaks kasutada tekste tervikuna. Subtiitrite vahele

saaks lisada mingi spetsiaalse tookeni, nagu näiteks kasutatakse lause alguse määramiseks või MUST-Cinema teadustöös [15]. Niimoodi oleks võimalik õpetada mudelile korraga nii teksti tükkideks jagamist kui ka tihendamist. Lisaks võivad eelmised laused anda mudelile rohkem konteksti, miks toimetaja on mingi muudatuse teinud. Antud lähenemise miinusteks võib nimetada suurema arvutusvõimekuse vajadust ja mudeli täpsuse vähendamist sisendite erinevate suuruste tõttu. Lisaks on raskendatud sisendandmestiku ettevalmistamine, sest suurte tekstide puhastamine ja ka puuduvate tekstivahemike tuvastamine on raskendatud.

Magistritöö eesmärk ei olnud leida optimaalset mudelit loetavate subtiitrite loomiseks (näiteks siirdeõpe jaoks kasutati ainult EstBERTi ja MBART-50 mudelit). Järgmiseks sammuks olekski mudeli arhitektuuri edasine optimeerimine. Töös kasutatud MBARTile on olemas mitu teist alternatiivi, näiteks MT100. Mudeli arhitektuuri optimeerimise käigus võiks arvestada ka mudeli suuruse vähendamisega, mis sõltub kihtide ja neuronite arvust. Väiksema mudeli puhul tõuseb töötamise kiirus ja ka häälestamine oleks odavam ning kiirem. Mudeli suurus on eriti tähtis siis, kui seadmel on piiratud arvutusvõimekus. Näiteks juhul, kui soovitakse luua automaatseid subtiitreid televiisori või telefoni arvutusvõimekust kasutades. Erinevate lähenemiste abil on võimalik keelemudeli suurust vähendada ilma tunnetatava jõudluse kaotuseta [54]. Samuti üheks optimeerimise suunaks võiks mudelis olla kasutavate keelte arvu piiramine. Näiteks võiks piirduda ainult soome-ugri keelte peal treenitud mudelitega. Eeltreenitud keelte lähedus võib anda paremaid tulemusi magistritöö eesmärkide saavutamiseks.

7. Kokkuvõte

Magistritöö eesmärk oli uurida võimalusi automaatse transkribeerija väljundit tihendades subtiitreid luua, mis oleksid sarnased keeletoimetaja poolt loodavatele. Subtiitrite tihendamist ei ole varasemalt palju uuritud. Kuigi eesmärk on sarnane masintõlke ja teksti kokkuvõtmise ülesannetega, oli magistritöö fookuses just lahenduse loomine eesti keele jaoks, mille jaoks ei ole olemas piisavalt andmeid ega mudeleid.

Autor lõi andmestiku ettevalmistamiseks algoritmi ja mitmeid loomuliku keele mudeleid subtiitrite tihendamiseks. Andmestikku oli võimalik koostada tänu Tallinna Tehnikaülikoolis arendatud automaatsele kõnetuvastajale ja Eesti Rahvusringhäälingu toimetajate poolt eestikeelsetele telesaadetele loodud subtiitritele.

Töös valminud loomuliku keele mudelite tulemused näitasid, et tõepoolest sellised mudelid suudavad luua transkribeeritud tekstist subtiitreid, mis on kvaliteedi poolest ligilähedased toimetajate poolt loodutele. Nii nagu varasemates töödes, selgus ka magistritöös, et abstraheerivat lähenemist kasutavad mudelid näitavad oluliselt paremaid tulemusi kui ekstraheerivad. Siirdeõppe mudelid, millel on eelnev arusaam eesti keelest, saavad antud ülesandega oluliselt paremini hakkama, kuid vajavad selleks peenhäälestamist. Autor on võrrelnud erineva lähenemisega mudeleid ning kasutanud tulemuste mõõtmiseks ühtset meetodit.

Magistritöös läbi viidud uuring näitas samuti, et vaatajad eelistavad mudeli poolt tihendatud subtiitreid automaatselt transkribeeritutele. Mudel siiski tihti eksib kõneleja emotsiooni säilitamisega ja mõnel juhul kustutab sõnu, mida ei peaks. Mudeli väljundeid saab kasutada subtiitrite toimetajate töö kergendamiseks ning isegi otse subtiitrite loomiseks juhul, kui väike arv vigu on vastuvõetav.

Kasutatud kirjandus

[1] Mehdi Latifi, Ali Mobalegh ja Elham Mohammadi. “Movie subtitles and the improvement of listening comprehension ability: Does it help?” *The Journal of Language Learning and Teaching* 1.2 (2011), lk. 18–29.

[2] Katrin Angerbauer, Heike Adel ja Ngoc Thang Vu. “Automatic Compression of Subtitles with Neural Networks and its Effect on User Experience”. Teoses: *Proc. Interspeech 2019*. 2019, lk. 594–598. DOI: 10.21437/Interspeech.2019-1750.

[3] Danni Liu, Jan Niehues ja Gerasimos Spanakis. “Adapting end-to-end speech recognition for readable subtitles”. *arXiv preprint arXiv:2005.12143* (2020).

[4] Jacob Devlin *et al.* “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. *CoRR* abs/1810.04805 (2018). arXiv: 1810.04805. URL: http://arxiv.org/abs/1810.04805.

[5] Mike Lewis *et al.* “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension”. *arXiv preprint arXiv:1910.13461* (2019).

[6] Elizabeth Soper, Stanley Fujimoto ja Yen-Yun Yu. “BART for Post-Correction of OCR Newspaper Text”. Teoses: *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*. 2021, lk. 284–290.

[7] Yasser Ebrahimi ja Parisa Bazaee. “The effect of watching English movies with standard subtitles on EFL learners’ content and vocabulary comprehension”. *Journal of Applied Linguistics and Language Research* 3.5 (2016), lk. 284–295.

[8] Monica Brady-Myerov. “Closed captioning gives literacy a boost”. *Education Week* (2015).

[9] Danielle Chazen. *How long does it take to transcribe audio? Turnaround Time explained*. Märts 2022. URL: https://verbit.ai/how-long-it-really-takes-to-transcribe-accurate-audio.

[10] Helen Williams ja David Thorne. “The value of teletext subtitling as a medium for language learning”. *System* 28.2 (2000), lk. 217–228.

[11] Pablo Romero-Fresco. “More haste less speed: Edited versus verbatim respoken subtitles”. *Vigo International Journal of Applied Linguistics* 6 (2009), lk. 109–133.

[12] Riigikogu. *Meediateenuste seadus*. https://www.riigiteataja.ee/akt/112122018055?leiaKehtiv. 2022.

[13] Yinhan Liu *et al.* "Multilingual Denoising Pre-training for Neural Machine Translation" (2020). arXiv: 2001.08210 [cs.CL].

[14] Myle Ott *et al.* "fairseq: A Fast, Extensible Toolkit for Sequence Modeling". Teoses: *Proceedings of NAACL-HLT 2019: Demonstrations*. 2019.

[15] Alina Karakanta, Matteo Negri ja Marco Turchi. "MuST-cinema: a speech-to-subtitles corpus". *arXiv preprint arXiv:2002.10829* (2020).

[16] Ming Zhou *et al.* "Progress in Neural NLP: Modeling, Learning, and Reasoning". *Engineering* 6.3 (2020), lk. 275–290. ISSN: 2095-8099. DOI: https://doi.org/10.1016/j.eng.2019.12.014. URL: https://www.sciencedirect.com/science/article/pii/S2095809919304928.

[17] Mia Xu Chen *et al.* "The best of both worlds: Combining recent advances in neural machine translation". *arXiv preprint arXiv:1804.09849* (2018).

[18] Felix Stahlberg. "Neural machine translation: A review". *Journal of Artificial Intelligence Research* 69 (2020), lk. 343–418.

[19] Oleksandra Klymenko, Daniel Braun ja Florian Matthes. "Automatic Text Summarization: A State-of-the-Art Review." *ICEIS (1)* (2020), lk. 648–655.

[20] Som Gupta ja S. K Gupta. "Abstractive summarization: An overview of the state of the art". *Expert Systems with Applications* 121 (2019), lk. 49–65. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2018.12.011. URL: https://www.sciencedirect.com/science/article/pii/S0957417418307735.

[21] Alexander M Rush, Sumit Chopra ja Jason Weston. "A neural attention model for abstractive sentence summarization". *arXiv preprint arXiv:1509.00685* (2015).

[22] Wafaa S El-Kassas *et al.* "Automatic text summarization: A comprehensive survey". *Expert Systems with Applications* 165 (2021), lk. 113679.

[23] Hui Lin ja Vincent Ng. "Abstractive Summarization: A Survey of the State of the Art". *Proceedings of the AAAI Conference on Artificial Intelligence* 33.01 (juuli 2019), lk. 9815–9822. DOI: 10.1609/aaai.v33i01.33019815. URL: https://ojs.aaai.org/index.php/AAAI/article/view/5056.

[24] Fuzhen Zhuang *et al.* "A comprehensive survey on transfer learning". *Proceedings of the IEEE* 109.1 (2020), lk. 43–76.

[25] Xipeng Qiu *et al.* "Pre-trained models for natural language processing: A survey". *Science China Technological Sciences* 63.10 (2020), lk. 1872–1897.

[26] Alex Wang ja Kyunghyun Cho. “BERT has a Mouth, and It Must Speak: BERT as a Markov Random Field Language Model”. *CoRR* abs/1902.04094 (2019). arXiv: 1902.04094. URL: http://arxiv.org/abs/1902.04094.

[27] Tanel Alumäe, Ottokar Tilk *et al.* “Advanced rich transcription system for Estonian speech”. *arXiv preprint arXiv:1901.03601* (2019).

[28] Jin-ge Yao, Xiaojun Wan ja Jianguo Xiao. “Recent advances in document summarization”. *Knowledge and Information Systems* 53.2 (2017), lk. 297–336.

[29] Sven Laur *et al.* “EstNLTK 1.6: Remastered Estonian NLP Pipeline”. Teoses: *Proceedings of The 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, mai 2020, lk. 7154–7162. URL: https://www.aclweb.org/anthology/2020.lrec-1.884.

[30] Mamdouh Farouk. “Measuring sentences similarity: a survey”. *arXiv preprint arXiv:1910.03940* (2019).

[31] Vladimir Braverman. “Sliding Window Algorithms”. Teoses: *Encyclopedia of Algorithms*. Toim. Ming-Yang Kao. New York, NY: Springer New York, 2016, lk. 2006–2011. ISBN: 978-1-4939-2864-4. DOI: 10.1007/978-1-4939-2864-4_797. URL: https://doi.org/10.1007/978-1-4939-2864-4_797.

[32] Daniel W Otter, Julian R Medina ja Jugal K Kalita. “A survey of the usages of deep learning for natural language processing”. *IEEE transactions on neural networks and learning systems* 32.2 (2020), lk. 604–624.

[33] Yang Liu. “Fine-tune BERT for extractive summarization”. *arXiv preprint arXiv:1903.10318* (2019).

[34] Hasan Tanvir *et al.* “Estbert: A pretrained language-specific bert for estonian”. *arXiv preprint arXiv:2011.04784* (2020).

[35] Wojciech Kryściński *et al.* “Neural text summarization: A critical evaluation”. *arXiv preprint arXiv:1908.08960* (2019).

[36] Adhika Pramita Widyassari *et al.* “Review of automatic text summarization techniques & methods”. *Journal of King Saud University - Computer and Information Sciences* 34.4 (2022), lk. 1029–1046. ISSN: 1319-1578. DOI: https://doi.org/10.1016/j.jksuci.2020.05.006. URL: https://www.sciencedirect.com/science/article/pii/S1319157820303712.

[37] Kishore Papineni *et al.* “BLEU: a Method for Automatic Evaluation of Machine Translation”. Teoses: 2002, lk. 311–318.

[38] Chin-Yew Lin ja Franz Josef Och. “ORANGE: a Method for Evaluating Automatic Evaluation Metrics for Machine Translation”. Teoses: *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*. Geneva, Switzerland: COLING, august 2004, lk. 501–507. URL: https://www.aclweb.org/anthology/C04-1072.

[39] Chin-Yew Lin. “ROUGE: A Package for Automatic Evaluation of Summaries”. Teoses: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, juuli 2004, lk. 74–81. URL: https://www.aclweb.org/anthology/W04-1013.

[40] Mehdi Allahyari *et al.* “Text summarization techniques: a brief survey”. *arXiv preprint arXiv:1707.02268* (2017).

[41] Satanjeev Banerjee ja Alon Lavie. “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”. Teoses: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ann Arbor, Michigan: Association for Computational Linguistics, juuni 2005, lk. 65–72. URL: https://www.aclweb.org/anthology/W05-0909.

[42] Rico Sennrich, Barry Haddow ja Alexandra Birch. “Neural machine translation of rare words with subword units”. *arXiv preprint arXiv:1508.07909* (2015).

[43] Kaj Bostrom ja Greg Durrett. “Byte pair encoding is suboptimal for language model pretraining”. *arXiv preprint arXiv:2004.03720* (2020).

[44] Matt Post. “A call for clarity in reporting BLEU scores”. *arXiv preprint arXiv:1804.08771* (2018).

[45] Thomas Wolf *et al.* “Huggingface’s transformers: State-of-the-art natural language processing”. *arXiv preprint arXiv:1910.03771* (2019).

[46] Azzam Haidar *et al.* “Harnessing GPU tensor cores for fast FP16 arithmetic to speed up mixed-precision iterative refinement solvers”. Teoses: *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE. 2018, lk. 603–613.

[47] Philipp Koehn *et al.* “Moses: Open source toolkit for statistical machine translation”. Teoses: *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*. 2007, lk. 177–180.

[48] Julia Kreutzer, Jasmijn Bastings ja Stefan Riezler. “Joey NMT: A Minimalist NMT Toolkit for Novices”. Teoses: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*. Hong Kong, China: Association for Computational Linguistics, november 2019, lk. 109–114. DOI: 10.18653/v1/D19-3019. URL: https://www.aclweb.org/anthology/D19-3019.

[49] *Difflib - helpers for computing deltas¶*. URL: https://docs.python.org/3/library/difflib.html.

[50] Yuqing Tang *et al.* “Multilingual translation with extensible multilingual pretraining and finetuning”. *arXiv preprint arXiv:2008.00401* (2020).

[51] Tanel Alumäe. *Kiirkirjutaja - realtime speech-to-text tool designed for real-time subtitling of TV broadcasts and streaming media.* https://github.com/alumae/kiirkirjutaja. 2021.

[52] Kristiina Ross Mati Erelt Tiiu Erelt. *EESTI KEELE KÄSIRAAMAT 2007*. https://www.eki.ee/books/ekk09/index.php?p=2&p1=10. 2007.

[53] Yu Gu *et al.* “Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing”. *ACM Trans. Comput. Healthcare* 3.1 (oktoober 2021). ISSN: 2691-1957. DOI: 10.1145/3458754. URL: https://doi.org/10.1145/3458754.

[54] Sam Shleifer ja Alexander M. Rush. “Pre-trained Summarization Distillation”. *CoRR* abs/2010.13002 (2020). arXiv: 2010.13002. URL: https://arxiv.org/abs/2010.13002.

Lisa 1 - Lihtlitsents

Mina, Ilja Samoilov

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Televisiooniprogrammide automaatsete transkriptsioonide teisendamine loetavateks subtiitriteks“ , mille juhendaja on Tanel Alumäe.

1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.

3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

Lisa 2 - Küsitlus

Sissejuhatav tekst

Tere! Olen TalTech-i Informaatika eriala magistrant Ilja Samoilov. Uurin oma magistritöös "Toimetatud subtiitrite loomine transkribeeritud audiovisuaalsetele salvestistele kasutades loomuliku keele mudeleid"võimalusi, kuidas muuta automaatse kõnetuvastaja tekst subtiitriteks kasutades masinõpet.

Uuringu eesmärk on aru saada, kas töös välja töötatud masinõpe mudeli väljund sobib subtiitrina rohkem kui otse kõnetuvastajaga transkribeeritud tekst.

Antud mudel ei ole kunagi varem uuringus kasutatud tekstilõike näinud. Lõigud said valitud enne mudeli poolt töötlemist ning ei ole töö autori poolt kuidagi muudetud.

Kõik uuringus olevad videod on ERRi omad ja autor ei oma nende õigusi.

Ülesanne

Palun vaadake läbi video ja vastake, kumb subtiiter sobiks Teie meelest paremini saatelõigu juurde. Tähtis ei ole mitte subtiitri ülim täpsus, vaid lugemise kergus ja mõtte edastamine.

Videoid kirjeldav tekst

Teie ees on kuus videot. Nende all on juhuslikus järjekorras automaatse kõnetuvastaja poolt loodud ja masinõppe mudeli pool tihendatud subtiitrid. Tähtis ei ole mitte subtiitri ülim täpsus, vaid lugemise kergus ja mõtte edastamine.

Video 1	`https://youtu.be/EHI2_mwPmjY`
Video 2	`https://youtu.be/rBdowN4hPIg`
Video 3	`https://youtu.be/6qWumIwMMbA`
Video 4	`https://youtu.be/qWELdLtukYo`
Video 5	`https://youtu.be/7R44yee6lKI`
Video 6	`https://youtu.be/n19v4DaLFx0`