TARTU ÜLIKOOL
Loodus- ja täppisteaduste valdkond
Arvutiteaduse instituut
Informaatika õppekava

Karen Saksakulm

Tõlkebüroode Grata ja Interlex masintõlke mudelite vigade analüüs ja lahendused

Bakalaureusetöö (9 EAP)

Juhendajad: Andre Tättar, MSc
Liisa Rätsep, MSc

Tartu 2021

Tõlkebüroode Grata ja Interlex masintõlke mudelite vigade analüüs ja lahendused

Lühikokkuvõte:
Masintõlkimine on aina enam populaarsust koguv keele masinõppe haru, mis lihtsustab käsitsi tehtavate tõlgete hulka ning aitab säästa aega. Tõlkimiseks on tutvustatud arvukalt nii erinevaid mudeliarhitektuure kui ka eeltöötluse protsesse, kuid kahjuks puudub üks parim lähenemisviis tõlkimiseks. Antud lõputöö käigus viidi kahe tõlkebüroo - Grata ning Interlex - tõlkeandmete peal läbi eel- ja järeltöötlused ning nii mudeli treenimise kui ka tõlkimise protsess. Eesmärgiks oli uurida erinevate masintõlke mudelite tõlketäpsusi, analüüsida tekkinud probleeme ning pakkuda välja lahendusi nende parandamiseks. Töö tulemustena ilmnes, et üks treenitud mudelitest ei oska vaatluse alla võetud andmeid tõlkida.

Võtmesõnad:
Masintõlge, tehisnärvivõrgud, maskimine, struktureeritud tekst, märgistuskeel

CERCS: P176 tehisintellekt

Error analysis and solutions of translation agencies Grata and Interlex machine translation models

Abstract:
Machine translation is a machine learning sector that is gaining popularity. It simplifies handmade translations and helps to save time. Several model architectures alongside pre- and post-processing methods have been introduced, but a single most effective translating solution has never been found. During the research of this thesis, the data of two translation companies, Grata and Interlex, was used for pre- and post-processing, model training and translation processes. The goal was to compare the translation accuracy of different machine translation models, analyze encountered problems and propose solutions for fixing these issues. The results showed that one of the models is incapable of translating the input data.

Keywords:
Machine translation, artifical neural networks, masking, structured text, markup language

CERCS:
P176 artificial intelligence

Sisukord

Sissejuhatus

Masintõlkimine on aina enam populaarsust koguv keele masinõppe haru, mis lihtsustab käsitsi tehtavate tõlgete hulka ning aitab oluliselt säästa aega. Masintõlkimiseks on tutvustatud arvukalt nii erinevaid mudeliarhitektuure, eel- ja järeltöötluse protsesse kui ka teisi tõlketäpsust parandavaid meetodeid. Antud lõputöös kasutatavate tekstide tõlkimiseks võetakse kasutusele Transformer mudel, millest tehnilises peatükis ka lähemalt räägitakse.

Tihtipeale ei salvestada tekste lihtkujul, vaid dokumendi struktuuri ning andmetega seotud informatsiooni salvestamiseks on kasutusele võetud märgistuskeel [2]. Struktureeritud tekstide käsitlemine on võimalus uue masintõlkekvaliteedi saavutamiseks päris rakenduste peal [2]. Käesoleva töö käigus võetakse tähelepanu alla kahe tõlkebüroo - Grata ning Interlex - struktureeritud tõlketekstide eeltöötlusprotsessid ning mudelite treenimine. Treenitud mudeleid rakendatakse testandmete tõlkimiseks. Töö väljundiks on uurida masintõlke mudelite tõlketäpsusi, analüüsida tekkivaid probleeme ning pakkuda välja lahendusi nende parandamiseks.

Uurimisküsimused

Autor soovib töö käigus leida vastuseid järgnevatele küsimustele:

1. Kui palju õpib masintõlke mudel panema liiga palju silte väljundisse?
2. Kui palju õpib masintõlke mudel panema liiga vähe silte väljundisse?
3. Kui palju tekib probleeme tühikutega?
4. Kui palju tekib probleeme suurtähtedega?
5. Kas terminite maskimine teeb rohkem halba kui head?

Töö ülesehitus

Käesoleva lõputöö põhiosa on jagatud viieks. Esimeses peatükis annab autor ülevaate lõputööga seotud mõistetest.

Teises peatükis kirjeldatakse masintõlkimise ning selle eeltöötluste tehnilisi protsesse. Vaatluse alla võetakse Transformer mudel, mida töö käigus tõlkimise jaoks kasutati, eeltöötluse jaoks rakendatud teksti tokeniseerimine ja suurtähtede normaliseerimine ning viimaks tõlketäpsuse hindamiseks kasutatav BLEU skoor.

Käesoleva lõputööga sarnaseid teadustöid tutvustatakse kolmandas peatükis.

Neljandas osas antakse ülevaade töö jaoks kasutatud tehnoloogiatest, kirjeldatakse lähemalt tõlkemudeli treenimisele eelnenud ning järgnenud andmete töötlusprotsesse. Autor kirjeldab ka tõlkemudelite treenimist ning andmete tõlketööd.

Viiendas peatükis viib autor tõlgitud andmete peal läbi kvalitatiivse ning kvantitatiivse analüüsi, mille käigus uuritakse mudelite tõlketäpsuseid, tekstisiseseid silte ning siltidega seotud probleeme.

1 Mõisted

GRU (ingl *gated recurrent unit*) - rekurrentsete närvivõrkude tüüp, mis on sarnane LSTM arhitektuuriga, kuid kasutab lisaks veel ka lähtestamise ning uuendamise väravat [1]

Järjestuste modelleerimine (ingl *sequence modelling*) - protsess, kus sisend väärtuste seeriate analüüsimise käigus genereeritakse väärtuste jada [2]

Maskimine (ingl *masking* või *placeholding*) - protsess, mille käigus asendatakse kindlat tüüpi andmed, näiteks märkesildid või urlid, uue kujuga

Metaandmed - andmeid kirjeldavad andmed, mis võivad olla administratiivsed (näiteks väljaandja, kuupäev), kirjeldavad (pealkiri, autor) või tehnilised (tarkvara, versioon)[3]

Märgistuskeel - arvutikeel, mis kasutab dokumendisiseste elementide defineerimiseks silte[4]

Pikk lühiajaline mälu (LSTM) - rekurrentsete närvivõrkude tüüp, mis on võimeline õppima järjestuste (ingl *sequence*) ennustuste probleemide puhul järjestuste sõltuvusi [5]

Rekurrentsed närvivõrgud (RNN) - tehisnärvivõrkude klass järjestikuste andmete töötlemiseks [6]

Silt (ingl *tag*) - tekstis esinev mittetõlgitav element, mida kasutatakse teksti struktuuri ja vorminduse väljendamiseks

Tehisnärvivõrgud - inimese ajutegevuse järgi modelleeritud algoritmide hulk, mis on loodud mustrite ära tundmiseks [7]

Tokeniseerimine - meetod, mille käigus jagatakse tekst väiksemateks tükkideks

Tõst - märgistiku vahetus klaviatuuril, sh suur- ja väiketähtede vahetus[8]

---

[1] https://paperswithcode.com/method/gru

[2] https://www.allerin.com/blog/sequence-modeling-for-beginners

[3] https://sisu.ut.ee/teadusandmed/metaandmed

[4] https://techterms.com/definition/markup_language

[5] https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/

[6] https://www.tensorflow.org/guide/keras/rnn

[7] https://wiki.pathmind.com/neural-network

[8] https://www.eki.ee/dict/qs2018/index.cgi?C03=1Q=tõst

2 Tehniline taust

Masintõlkimine eeldab mitmete tehniliselt keerukate protsesside mõistmist ning rakendamist. Antud peatüki eesmärgiks on tutvustada käesolevas töös kasutatud tõlkemudeli arhitektuuri, anda ülevaade eeltöötluste tehnilistest lahendustest ning hindamismeetodist.

2.1 Transformer mudel

Töös kasutatavate andmete masintõlkimiseks on kasutatud Transformerit[9]. Transformer on Vaswani jt poolt 2017. aastal tutvustatud närvivõrgu mudeli arhitektuur [11].

Vaswani jt [11] tõdevad, et pikalt olid järjestuste modelleerimisel ja transduktsiooni probleemide lahendamisel kasutusel tipptaseme meetodid nagu rekurrentsed närvivõrgud, pikk lühiajaline mälu (LSTM) ning GRU. Nende meetodite kvaliteedi parandamise peamiseks probleemiks on olnud mälulimiit pikemate sisendite puhul, kuna nende jadana toimiv olemus välistab juba iseenesest treeningandmete paralleelsuse [11]. Transformer mudel ei kasuta korduvust ning pakub seetõttu ka oluliselt rohkem paralleelsust andmete treenimisel võrreldes teiste rekurrentsete meetoditega [11]. Mudeli treenimisaeg on oluliselt kiirenenud tänu lihtsamalt paralleeliseeritavale (ingl *parallellizable*) arhitektuurile. Lisaks paralleelsuse lubamisele kasutab Transformer mudel ka tähelepanu mehhanisme.

Alljärgnev lõik tugineb Vaswani jt teadusartiklil [11]. Tähelepanu mehhanismid lubavad sõltuvuste modelleerimist sõltumata nende kaugustest sisend- või väljundjärjestustes. Tähelepanu funktsioonid koosnevad päringutest, võtmetest, väärtustest ning väljundist. Väljund arvutatakse kaalutud keskmisena väärtustest, kus iga väärtuse kaal arvutatakse sobivusfunktsiooni päringut ning sellele vastavat võtit kasutades. Transformer mudel tugineb sisend ning väljund sõltuvuste märkimisel täielikult tähelepanu mehhanismidele. Lisaks tähelepanu mehhanismidele ning paralleelsuse võimaldamisele on Transformer esimene transduktsiooni mudel, mis tugineb sisend ning väljund esitusviiside arvutamisel täielikult enese-tähelpanule (ingl *self-attention*) ilma RNN või konvolutsiooni kasutamata.

2.2 Teksti tokeniseerimine

Kuigi tänapäeva neuronmasintõlke süsteemidel on potentsiaal tõlkida lähtekeele tekst ilma tokeniseerimata otse sihtkeelde, toetuvad paljud masintõlke süsteemid siiski siiani keelest sõltuvatele eel- ja järeltöötlustele [6]. Paljud tokeniseerimise lahendused on mõeldud Euroopa keeltele, mille sõnad on tühikutega segmenteeritud [6]. Mittesegmenteeritud keelte jaoks, nagu Hiina, Korea või Jaapani keel, on vaja erinevaid sõnade eraldajaid [6].

[9] https://github.com/jadore801120/attention-is-all-you-need-pytorch

Selliste keelte puhul on ka raske treenida mitmekeelseid neuronmasintõlke mudelid, kuna eel- ja järeltöötlus toimub keele kaupa, kuid mudeli arhitektuur on keelest sõltumatu [6].

Selleks, et tõlkemudelid oskaksid tõlkida tundmatuid sõnu, on vaja tekst eelnevalt segmenteerida. Tokeniseerimine on meetod, millega jagatakse tekst väiksemateks osadeks. Transformer arhitektuurile toetuvate mudelite kvaliteet saavutatakse tänu eelnevale teksti tokeniseerimisele. Tokeniseeritud tekstist on masinal lihtsam leida nii mustreid kui ka korduvsõnu. Näiteks kui mudeli treenimisel on masin õppinud ära sõna *laps* ning sõnade mitmust tähistava lõpu *ed*, kuid mitte sõna kujul *lapsed*, ei oska ta viimast tõlkida. Selle probleemi suudab teksti tokeniseerimine lihtsalt parandada, lahutades sõna *lapsed* kaheks - *laps* ning *ed*. Tokenisaatori eesmärk ongi teksti segmenteerimine väiksemateks osadeks, mille kaudu masin õpib paremini teksti tõlkima.

2.3 Suurtähtede normaliseerija

Suurtähtede normaliseerijat (ingl *truecaser*) kasutatakse tõstutundlike (ingl *case sensitive*) keelte puhul, kus eristatakse suur- ning väiketähti. Lita jt. kirjeldavad suurtähtede normaliseerimist kui protsessi, mille käigus taastatakse tõstu informatsioon tekstidel, mis on kas puudulikult suur- ning väiketähtestatud või täielikult tõstu eiranud [7]. Kana seevastu kirjeldab seda kui NLP (loomuliku keele töötlus) probleemi, mille kaudu otsitakse sõnade õiget tõstu tekstides, kus see informatsioon ei ole kättesaadav [5]. Igal juhul on suurtähtede normaliseerimine oluline eeltöötluse samm kas tõlkemudeli loomisel või tõlgete järeltöötlusel.

Kana toob välja neli erinevat suurtähtede normaliseerimise lähenemisviisi [5]:

1. lausete segmenteerimine - sisendtekst jagatakse lauseteks ning iga lause esitäht suurtähestatakse;
2. sõnaliigi määramine (ingl. *part-of-speech*) - uuritakse iga sõna definitsiooni ning konteksti lauses, määratakse kõige sobivam silt ning suurtähestatakse spetsiifiliste siltidega sõnad;
3. nime olemi tuvastamine (ingl *name-entity-recognition*) - klassifitseeritakse lausetes leiduvad sõnad ning suurtähestatakse neist teatud kategooriad, näiteks inimeste nimed;
4. statistiline modelleerimine - treenitakse statistiline sõnade mudel ning sõnade grupp, mis tavaliselt esinevad suurtähtedes.

Lisaks erinevatele lähenemisviisidele leiduvad ka erinevad klassid, mida suurtähtede normaliseerimisel rakendatakse. Lita jt tutvustavad oma artiklis [7] nelja järgnevat kategooriat:

1. läbiv väiketähestus;
2. esitähe suurtähestus;
3. läbiv suurtähestus;
4. vahelduv suur- ja väiketähestus.

Tutvustatud nelja meetodit saab kasutada nii sõna- kui ka lausetasandil suurtähtede normaliseerimisel.

2.4 BLEU skoor

Mida lähemal on masintõlge professionaalsele inimtõlkijale, seda parem on selle kvaliteet [8]. Papineni jt toovad välja, et selle hindamiseks on vaja kahte osa: arvulist mõõdikut, millega hinnata tõlke ligilähedust ning inimeste poolt tõlgitud hea kvaliteedilist andmekorpust [8]. Nende poolt loodud meetodi BLEU põhiidee on võrrelda kandidaattõlgete n-gramme vastavate tõlkelausete n-grammidega ning loendada üle mõlemas lauses leiduvad vasted. Parima tõlkekandidaadi valimisel tuleb arvesse võtta ka lausete pikkust - tõlkekandidaat ei tohi võrreldes originaaltõlkega liiga pikk ega liiga lühike olla [8].

BLEU skoor toetub oma töös järgnevatele meetoditele [8]:

- N-grammi täpsus keelab ära sõnad, mida ei leidu üheski kandidaattõlkele vastavates originaallausetes.
- Kõrvale jäetakse kandidaatlaused, kus sõna esinemissagedus on suurem kui originaallausetes leiduva sõna sagedus.
- Kõrge vastega tõlkelause pikkus peab jääma originaallausete pikkuste vahemikku ning olema sama sõnavaliku ning sõnade järjekorraga.

BLEU skoori mõõdetakse vahemikus 0-st 1-ni, kus vaid originaallausega identsed laused võivad saada BLEU skooriks 1.0 [8].

3 Seotud kirjandus

Antud peatükis kirjeldatakse käesoleva tööga seotud teadustöid.

3.1 Andmete maskimine

Käesoleva tööga seotud uurimus on Matt Posti jt 2019. aastal ilmunud teadusartikkel *An Exploration of Placeholding in Neural Machine Translation*. Töös tutvustatakse andmete maskimise protsessi masintõlkimisel.

Andmete maskimine on protsess, mille käigus sisendsümbolid asendatakse korduvama ning lihtsama kujuga. Peale masintõlkimist viiakse maskitud tõlkelaused tagasi originaalteksti kujule. Matt Posti jt 2019. aastal ilmunud teadustöös on vaatluse alla võetud prantsuse-inglise ning jaapani-inglise keelsed tõlked Transformer mudelil [9]. Töö autorid nendivad, et kuigi üldiselt on tehisnärvivõrkudel põhineva masintõlke kvaliteet parem kui fraasipõhine masintõlge, on selle edusammu tõttu kaotatud teatud kontroll tõlkimise üle. Enam ei ole otsest lüli lähtesõnade, nende tõlke seadistuste ning järjestatud dekoodri väljundi vahel, mis on tekitanud olukorra, kus tihtipeale ohverdatakse tõlkekvaliteet tõlke garantiile, et kindlad sisendsümbolid tõlgitakse peaaegu täiusliku saagisega (ingl *recall*) [9]. Tõlkekvaliteedi hoidmiseks ning garantii saamiseks on kasutusele võetud standardsemal ning lihtsamal kujul olevad maskid. Sellega on saavutatud olukord, kus soovitud sisesendtermin on õigel kujul ka väljundis [9]. Maskide eelis terminite algkuju ees ongi nende lühem ning korduvam kuju, mis lihtustab ning parandab nende tõlget.

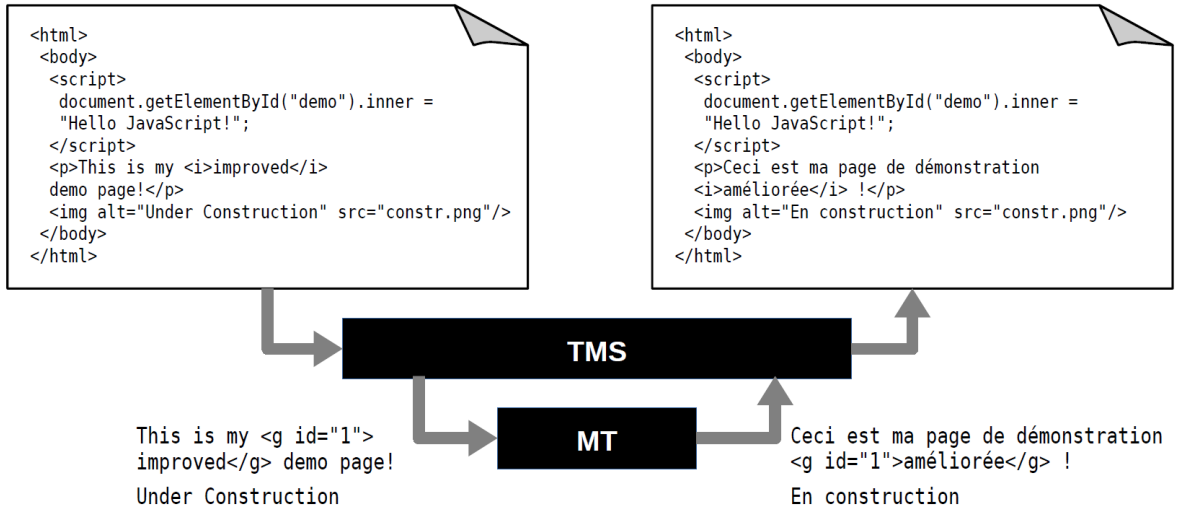
Post jt keskenduvad oma teadustöös kopeeritavatele terminitele, mida ei tõlgita ning mis kopeeritakse maskimise teel otse väljundlausesse. Nendeks terminiteks on regulaaravaldistega leitavad väljendid nagu numbrid, URL'id, e-maili aadressid [9].

Tulemustena toovad nad välja, et maskitud tõlkesüsteemid ei tõlgi maske usaldusväärselt, mistõttu tuleks kasutatada piiranguid maskide väljundisse kirjutamise kontrollimiseks [9]. Post jt toovad ka välja, et maskitud terminite puhul tõlgib alussüsteem (ingl *baseline system*) neid juba piisavalt hästi. Saagised maskitud süsteemi ning alussüsteemi puhul olid mitmete testandmete puhul üle 90-ne [9].

3.2 Siltide tuvastamine ning tõlkimine

Tekstisiseste siltide tuvastamist ning tõlkimist tutvustavad Hanneman ja Dinu oma 2019. aastal ilmunud teadustöös *How Should Markup Tags Be Translated?* [1]. Töös võrreldakse omavahel kahte tihti kasutusel olevat märgistussilti ning testitakse masintõlke mudelite võimet õppida kunstlikult silte lisama läbi treening andmestiku suurendamise.

Suur osa tõlketekstidest ei leidu puhta tekstina, vaid pärineb struktureeritud dokumentidest, kasutades sellele vastavaid spetsifikatsioone, näiteks HTML'i, Miscrosoft Word'i või PDF'i [1]. Struktureeritud dokumentide tõlketöö jagatakse ära tõlkejuhtimissüsteemi (TMS) ning selle alla kuuluva masintõlke süsteemi (MT) vahel [1]. Joonis 1 illustreerib struktureeritud teksti tõlkeprotsessi.



Joonis 1. Struktureeritud teksti tõlkimine[1].

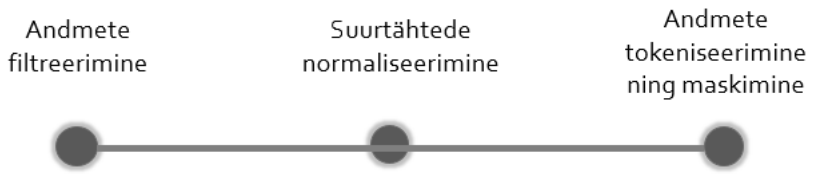
Tekstide tõlkimiseks eemaldavad Hinneman ja Dinu märgistussildid sisendist täielikult ning sisestavad need taas väljundisse järeltöötluse käigus. Võrdluseks võetakse kaks alternatiivset lähenemist - maskitud andmed ning toorandmed (ingl *raw data*), kuhu on alles jäetud sildid[1].

Tõlketäpsuste hindamiseks kasutavad nad BLEU skoori SacreBLEU lähenemisviisi. Maskitud ning toorandmete täpsust hinnati ka erineva arvu siltide lisamisel lausetesse [1]. Lisaks hinnati täpsusi ka inimvaatluse käigus, mille puhul jagati tõlkelausete siltide täpsused kolmeks: hea, halb ning võimatu hinnata [1].

Tulemustena toovad nad välja, et siltide asendamisel maskidega koos andmestiku suurendamisega annab võrdväärse või parema tulemuse võrreldes meetodiga, mille käigus sildid treenimise ajaks täielikult eemaldatakse ning pärast tagasi lisatakse [1]. Algsiltid (ingl *raw tags*) ebaõnnestuvad aga nende läbi viidud testandmetel - siltide asukoha tuvastamine on väga hea, kuid mudel ei osanud harvaesinevaid silte väljundisse kopeerida, ilma et ei oleks läbi tehtud korduvaid siltide moonutamise, kustutamise ning duubeldamise samme [1].

4 Metoodika

Antud lõputöö käigus treeniti kaks erinevat masintõlkemudelit. Tõlkemudelite treenimiseks oli vajalik andmed viia kõigepealt treenimiseks sobivale kujule. Käesolevas töös võeti kasutusele kolm eeltöötluse protsessi, mille käigus viidi algandmed mudelite treenimiseks ning tõlkimise hindamiseks sobivasse vormingusse. Selles peatükis antakse ülevaade lõputöö koostamiseks kasutatud tehnoloogiatest ning kirjeldatakse tõlkeandmete peal läbi viidud protsesse. Joonis 2 illustreerib tõlkemudelite treenimiseelseid samme. Antud peatükis tutvustatavad meetodid viidi läbi kahel mudelil. Baasmudel treeniti algkujul olevatel andmetel ning teine maskitud andmetel.



Joonis 2. Tõlkemudelite andmete treenimiseks läbi viidud eeltöötluse sammud.

4.1 Kasutatud tehnoloogiad

Käesolevas töös kasutatud mudeleid treeniti Tartu Ülikooli alla kuuluva Teadusarvutuste keskuse, lühidalt HPC[10] (ingl. *High Performance Computing Center*) Rocketi klastris. Rocketi klaster koosneb 135 serverist, kuuest GPU serverist, 20 AMD kõrgtihedusega protsessori sõlmest, neljast kõrgmälu seadmest, 12 CPU sõlmest ja peasõlmest, milleks on rocket.hpc.ut.ee [10].

Rocketi mäluruumi kasutamiseks tuli kasutusele võtta Slurm[11] (ingl. *Simple Linux Utility for Resource Management*). Järgnev tekst on refereeritud Jette jt poolt 2002. aastal loodud Slurm'i tuvustava dokumendist [3]. Slurm on avatud lähtekoodiga, veakindel, kõrgelt skaleeritav klastri juhtimis- ning tööd ajastav süsteem. Slurm'il on kolm põhifunktsiooni. Esiteks tagab Slurm kasutajatele ligipääsu ressurssidele. Teiseks pakub Slurm raamistikku, kus käivitada, jooksutada ning jälgida oma tööd. Kolmandaks lahendab ta ressursside kasutamise probleemi, hallates ootel olevate tööde järjekorda.

[10] https://hpc.ut.ee/en/home/
[11] https://www.schedmd.com/

Mudelite töötlemiseks ning analüüsimiseks kasutatavaid koode jooksutati Rocketi klustril asuvas Jupyter Notebook[12] keskkonnas. Jupyteri kodulehel leiduva informatsioonikohaselt [4] on Notebook avatud lähtekoodiga veebirakendus, mis annab kasutajatele võimaluse jagada dokumente, mis sisaldav koodi, valemeid, joonised ja teksti. Jupyteris on võimalus valida üle 40 programmeerimiskeele vahel [4]. Käesoleva lõputöö mudelid on kirjutatud kasutades Pythoni programmeerimiskeelt.

4.2 Andmete filtreerimine

Esimese eeltöötlusena viidi läbi tõlkeandmete maskimise ning filtreerimise protsess, mille käigus eemaldati edasise vaatluse jaoks ebavajalikud andmed. Sisendina kasutati .tmx[13] faili formaadis XML-sildistatud andmestikke, mis sisaldasid lähte- ning sihtkeele tõlkelauseid. Grata lähteandmeid oli inglise-eesti keelepaari puhul 1 662 690 ning saksa-eesti keelepaaril 847 073. Interlexi tõlkeid oli inglise-läti keelepaaril 78347, inglise-leedu keelepaaril 32681, inglise-eesti keelepaaril 151 085, saksa-leedu keelepaaril 88608 ning saksa-eesti keelepaaril 147 297. Kokku oli Grata tõlkebüroo andmeid 2 509 763 ning Interlexi andmeid 578 203.

Esimese sammuna puhastati algandmetest teksti struktuuri sildid ning metaandmed (joonis 3) ning salvestati <seg> siltide vahel leiduvad lähte- ning sihttõlgete informatsioon (joonis 4).

```
<tu creationdate="20150917T030341Z" creationid="LGE\taeho.bae" changedate="20150917T030341Z" changeid=
"LGE\taeho.bae" lastusagedate="20150917T030341Z">
  <prop type="x-LastUsedBy">LGE\taeho.bae</prop>
  <prop type="x-Origin">TM</prop>
  <prop type="x-ConfirmationLevel">Translated</prop>
  <tuv xml:lang="en-US">
    <seg>Please read the safety information carefully before using the product.</seg>
  </tuv>
  <tuv xml:lang="et-EE">
    <seg>Palun lugege ohutusteave enne toote kasutamist hoolikalt läbi.</seg>
  </tuv>
</tu>
```

Joonis 3. Tõlkeandmete algkuju.

Järgmise sammuna kontrolliti, et struktuurisiltidest ning metaandmetest puhastatud lausete puhul oleksid olemas nii lähte- kui ka sihttõlked. Kolmandaks rakendati lausetele andmete maskimist. Selle sammu ajal asendati lausetesisesed sildid maskidega kujul «MASK_ARV», kus mask märgib maski tüüpi ning arv asukohanumbrit. Lausetes leiduvad järjestikused ühte tüüpi sildid grupeeriti ühe maski alla. Maski tüüpidena olid kasutusel sildi tüüpidele vastavad maskid nagu *ept* (ingl *end paired tag*), *bpt* (ingl *begin*

[12] https://jupyter.org/
[13] https://www.reviversoft.com/file-extensions/tmx

paired tag), *ph* (ingl *place holder*) ning internetiaadresse ja termineid asendavad *url* ning *ter* maskid. *Bpt* ning *ept* puhul on tegemist paarisjärjestuste märkimiseks kasutatavate siltidega, mille puhul esimene neist tähistab vaadeldava järjendi algust ning teine lõppu. *Ph* tüüpi silte kasutatakse eraldiseisvate järjestuste tähistamiseks.

Please read the safety information carefully before using the product.
Palun lugege ohutusteave enne toote kasutamist hoolikalt läbi.

Joonis 4. Joonisel 3 toodud puhastatud tõlkeandmed.

Lausete maskimise järel rakendati maskitud lausetele esimest filtrit, millega eemaldati pikemad kui kolmekohalised sümbolite järjendid. Teise filtri käigus kontrolliti, et lausetel oleks säilinud korrektne .tmx formaat ning selle puudumisel parandati seda. Andmed, mis ka pärast paranduste tegemist õigel kujul ei olnud, eemaldati edasisest protsessist. Teise filtri läbinud andmetele rakendati kolmandat filtrit, mille käigus ühendati lähte- ning sihttõlgete sildid. Selleks kontrolliti mõlema poole lausete siltide vastavust. Juhul kui sildid erinesid, eemaldati need laused edasisest tööst. Kui lähte- ning sihtlausete sildid võrdusid omavahel, jäeti edasisteks protsessideks alles üks koopia siltidest. Töö käigus kontrolliti ka ainult siltidest koosnevate lausete olemasolu ning nende leidumisel eemaldati need vaatluse alt. Joonisel 5 on toodud näitelause filtritest läbinud tõlkelausest. Tabelites 1 ja 2 on välja toodud Grata ning Interlexi tõlkeandmete hulgad enne ning pärast filtrite rakendamist.

Tabel 1. Grata tõlkebüroo andmehulk enne ning peale filtrite rakendamist.

Keelepaar\ Samm	Algandmed	Esimene filter	Teine filter	Kolmas filter
EN-ET	1 662 690	1 611 004	1 610 468	1 607 067
DE-ET	847 073	815 042	814 695	811 436
Kokku	**2 509 763**	**2 426 046**	**2 425 163**	**2 418 503**

["Visit website «url_1»", "Külastage veebilehte «url_1»", {"«url_1»":
"`https://courses.cs.ut.ee/`*"}, "en-et", "pc", "Visit website,*
`https://courses.cs.ut.ee/`*", "Külastage veebilehte*
`https://courses.cs.ut.ee/`*"]*

Joonis 5. Filtritest läbinud ning väljundfaili kirjutatud lause, kus on kirjas töödeldav lause ning selle tõlge, sisemine sõnastik maski ning sellele vastava tekstiga, keelepaar, teksti domeen ning algne lause ja selle tõlge.

Tabel 2. Interlex tõlkebüroo andmehulk enne ning peale filtrite rakendamist.

Keelepaar\ Samm	Algandmed	Esimene filter	Teine filter	Kolmas filter
EN-LV	78347	78234	78228	78156
EN-LT	32681	32506	32503	32467
EN-ET	151 085	150 246	150 199	149 208
DE-LV	80185	80064	80057	79440
DE-LT	88608	88438	88430	87806
DE-ET	147 297	146 973	146 943	146 328
Kokku	**578 203**	**576 461**	**576 360**	**573 405**

Müraseid andmeid oli Grata tõlgete puhul 4,21% ning Interlexi tõlgete puhul kõigest 0,83% algandmetest (tabel 3).

Tabel 3. Filtrite rakendamisel tööst eemaldatud tõlkelaused.

Keelepaar\ Põhjus	Esimene filter	Teine filter	Kolmas filter	**Kokku**
Grata	83985	615	6660	**91260**
Interlex	1755	88	2955	**4798**

Enim Grata andmeid eemaldati esimese filtri puhul, mille käigus jäeti kõrvale 3,80% algandmetest. Interlexi andmete puhul oli suurim kadu peale kolmanda filtri rakendamist, mil eemaldati 0,42% algandmetest. Järgnevate eeltöötlusprotsesside käigus andmetehulgad enam ei muutunud ning masintõlke mudeli treenimiseks jäi alles 2 418 503 Grata tõlkelauset ning 573 405 Interlexi tõlkelauset, mida on vastavalt 96,36% ning 99,17% algandmetest.

4.3 Suurtähtede normaliseerimine

Teise eeltöötluse sammuna viidi läbi suurtähtede normaliseerimine. Esiteks kontrolliti, kas leidub ainult suurtähtedes kirjutatud lauseid. Suurtähtedes kirjutatud lausete leidumisel salvestati see info koos tõlkelausetega lõppväljundisse. Lausete puhul, mis ei olnud kirjutatud ainult suurtähtedes, kontrolliti suurtähtedes kirjutatud fraaside olemasolu. Kui leiti selliseid fraase, salvestati see informatsiooni <UP> </UP> siltidena koos tõlkelausetega väljundisse.

Andmed, mis ei olnud kirjutatud läbivalt suurtähtedes ega sisaldanud ka ühtegi suurtähtedes kirjutatud sõna, kontrolliti üle mudeli abil. Suurtähtede normaliseerimise (ingl *truecase*) mudel koosnes 7,8 miljonist sõnast viies erikeeles - eesti, läti, leedu, inglise ning saksa. Esimesena kontrolliti, kas lausetes kasutatavad väiketähestatud sõnad leiduvad ka mudelis. Kui ei leidunud, jäeti sõna muutmata. Juhul kui väiketähestatud kujul sõna oli mudelis olemas, kontrolliti järgnevat kolme juhtu:

1. Kas tegu on lause esimese sõnaga?
2. Kas sõna on kirjutatud suurtähtedes?
3. Kas sõnale eelneb punkt, koolon, semikoolon, küsimärk või hüüumärk?

Kui sõna vastas vähemalt ühele ülaltoodud tingimustest, kirjutati sõna läbivalt väiketähtedes. Järgnevalt uuendati kõiki lauseid vastavalt eelmise sammu käigus läbi viidud kirjapildi muudatustele.

4.4 Tokeniseerimine

Enne kolmandat eeltöötlust liideti Grata ning Interlexi andmestikud üheks. Kasutusele võetud ühendatud andmestik sisaldas 2 991 908 tõlkelauset.

Esimese sammuna rakendati andmetele SentencePiece'i tokeniseerijat. Alljärgnev lõik tugineb Kudo ja Richardsoni 2018. aastal ilmunud teadusartiklile [6]. SentencePiece on keelest sõltumatu alamsõnede sõnesti ja pöördsõnesti, mis on mõeldud masintõlke tekstide töötlemiseks. Kui senini kasutusel olnud segementatsiooni vahendid on vajanud lausete eelsõnestamist, siis SentencePiece võimaldab lähteteksti treenida otse alamsõnede mudeliteks. SentencePiece koosneb neljast põhikomponendist: normaliseerijast, treenijast, sõnestist ja pöördsõnestist. Normaliseerija on moodul, mis viib semantiliselt samaväärsed Unicode'i märgid standard kujule. Treenija eesmärgiks on treenida alamsõnede segmentatsiooni mudel normaliseeritud korpusest. Tokeniseerija ning detokeniseerija töö on vastavalt eeltöötlus (sõnestamine) ning järeltöötlus (pöördsõnestamine).

['see on näidislause']
['_', 'see', '_on', '_', 'näidis', 'lause']

Joonis 6. Näitelause illustreerib SentencePiece poolt kodeeritud lauset.

Kui teised sõnestamisega tegelevad meetodid eeldavad sisendlausete eelsõnestust järjenditesse, siis SentencePiece treenib alamsõnede mudelid otse algtekstist [6]. See lihtsustab

ning kiirendab masintõlkimisega seotud tööd.

Viimase sammuna jagati töödeldud andmed kolme andmestikku - treeninghulk, valideerimisandmestik ning testandmed. 2 991 908 maskitud tõlkelausest 99,4% jäeti tõlkemudeli treeningandmeteks. 6000 tõlkelauset jäeti mudeli valideerimiseks ning 12000 mudeli testimiseks.

4.5 Baasmudeli tokeniseerimine

Baasmudeli viimase eeltöötluse jaoks võeti kasutusele juba eelmise eeltöötluse käigus kasutatud Grata ning Interlexi ühendatud andmestik, mis koosnes 2 991 908 tõlkelausest. Antud mudeli jaoks treenitavaid andmeid ei olnud vaja maskida, vaid lausetes leiduvad sildid jäeti originaalkujule.

Viimase eeltöötluse esimese sammuna vaadati läbi, kas vaadeldav lause on eelnevalt suurtähestatud ning märgistatud </UP> <UP> siltidega. Juhul kui oli, eemaldati need märgised. Lausetes, kus vastavad märgistused puudusid, jäeti tõstud samaks.
Sarnaselt eelmisele eeltöötlusele, rakendati ka siin kolmanda sammuna SentencePiece tokeniseerijat. Töödeldud tõlkeandmed jagati treening-, test- ning valideerimisandmeteks. 2 973 908 lauset eraldati mudeli treenimiseks, 6000 mudeli valideerimiseks ning 12000 testimiseks.

4.6 Mudelite treenimine ning tõlkimine

Tõlkemudelite treenimiseks võeti kasutusele viimase eeltöötluse läbinud treengingandmed. Antud töö käigus ei treenitud uusi mudeleid, vaid häälestati eeltreenitud mudelid uutel andmestikel. Pärast treenimist valiti andmete tõlkimiseks viimane mudel. Mõlemat mudelit treeniti Rocketi klastris 26 tundi. Treenimiseks kasutati kahte protsessorit (CPU) ning ühte graafikaprotsessorit (GPU). Muutmälu oli kasutusel 33 gigabaiti. Loodud mudeleid kasutati testandmete tõlkimiseks.

Andmete tõlkimiseks kasutati mõlema mudeli puhul 12000 tõlkelauset, 2000 lauset iga keelepaari kohta. Tõlkimiseks kasutati samuti kahte protsessorit ning ühte graafikaprotsessorit. Muutmälu oli kasutusel 17 gigabaiti. Tõlkimine võttis mõlema mudeli puhul ligikaudu pool tundi.

4.7 Tõlkelausete järeltöötlus

Tõlgitud lausete analüüsimiseks ning hindamiseks viidi läbi järeltöötlus, mille käigus maskitud ning tokeniseeritud kujul andmed asendati nende algkujuga. Kõigepealt rakendati tõlkelausetele SentencePiece'i detokeniseerijat. Detokeniseeritud lausete peal

viidi läbi suurtähtede normaliseerimine, mille käigus muudeti lausete esitähed suureks. Lisaks kontrolliti ainult suurtähtedes kirjutatud sõnade olemasolu ning nende leidumisel muudeti nende kuju vastavalt ka tõlkelauses. Eelnevalt baasmudelist, tehti maskitud mudeli puhul tagasiasendus maskidest siltidele. Peale järeltöötlust sisaldasid mõlema mudeli tõlkelaused silte (joonis 7).

Mit Taste<bpt type="1"x="1"/> AUF<ept x="1"/> <bpt type="2"x="2"/>AII 1.03 <ept x="2"/>wählen.

Joonis 7. Järeltöötluse läbinud lause nii baasmudeli kui ka maskitud mudeli näitel.

5 Andmete analüüs

Käesolevas peatükis võetakse vaatluse alla järeltöötluse läbinud tõlkeandmed. Põhirõhk on pööratud sildistatud lausete tõlketäpsustele ning tekkinud probleemide analüüsile.

5.1 Kvantitatiivne analüüs

Antud peatükis võrreldakse kahe treenitud mudeli SacreBLEU skoore silte sisaldavate lausete puhul. Antakse ka ülevaade silte sisaldavate lausete osakaalust ning tüüpidest testandmetel.

5.1.1 Tõlketäpsused

Järeltöötluse läbi teinud andmetele rakendati SacreBLEU[14] skoori. Selle puhul on tegemist BLEU lihtsustatud variandiga, mille arvutamine käib kiiremalt ja mugavamalt. Erinevusena on arvud toodud nullist sajani, mitte nullist üheni.

SacreBLEU skooridest ilmnes, et maskitud mudeli skoorid olid baasmudeli skooride võrdluses paremad iga vastava keelepaari puhul. Küll aga oli maskitud mudeli eesti-leedu keelepaari skoor madalam inglise-eesti, saksa-eesti ning inglise-läti baasmudeli tõlkeandmete skooridest. Inglise-eesti ning saksa-eesti SacreBLEU skoorid olid ootuspäraselt kõrgemad, kuna treeningandmeid oli rohkem (tabel 4). Antud keelepaaride info moodustas vastavalt 58,70% ning 32,01% treeningandmetest.

Tabel 4. Maskitud mudeli ning baasmudeli tõlgete SacreBLEU skoorid.

Keelepaar\ Tõlgitud andmed	Maskitud mudel	Baasmudel
EN-LV	43,01	36,11
EN-LT	35,02	28,61
EN-ET	62,81	46,30
DE-LV	46,55	30,09
DE-LT	42,19	24,99
DE-ET	51,11	37,06
Keskmine	**46,78**	**33,86**

Mudelite võrdluses ilmneb, et maskitud mudeli SacreBLEU keskmine skoor oli märkimisväärselt kõrgem baasmudeli skoorist - 46,78 punkti võrreldes baasmudeli 33,86-punktilise

[14] https://github.com/mjpost/sacreBLEU

keskmisega. Kuigi SacreBLEU skoorid olid võrdlemisi kõrged, siis edasiste analüüside käigus osutus, et baasmudel õppis silte sisaldavaid sisendeid väljundisse kopeerima, mitte tõlkima. Seega on BLEU üksi nõrk meetrika tõlkekvaliteetide hindamiseks.

5.1.2 Siltide analüüs

Mõlema mudeli järeltöötluse läbinud tõlkelausete seast ilmnes, et 12000 lausest sisaldasid silte vaid 1730 lauset, mida on vastavalt 14,41% testandmetest. Huvitava tähelepanekuna ilmnes, et lähtelausete ning tõlgitud lausete silte sisaldavate lausete arv erines sihttõlke lausete arvust. Viimase puhul osutus, et silte sisaldasid 1751 lauset, mida on 21 võrra enam tõlgitud lauseid arvesse võttes. Ilmnes, et nende 21 lisalause puhul sisaldasid oodatavad tõlkelaused <ph> tüüpi silte, mida teistes lausetes ei olnud. Tegu on veaga, mis oleks pidanud andmete filtreerimisel eemalduma, kuid mille filtrid siiski läbi lasksid. Joonis 8 illustreerib tekkinud olukorda.

lähe: Menü (Navigationshinweise)
siht: Meniu <ph type="2"/> (navigacijos nurodymai)
tõlge: Meniu (navigacijos nurodymai)

Joonis 8. Näitelause, kus sihtlause (oodatav tõlge) sisaldab silti, kuid lähtelause ning tõlgitud lause seda ei sisalda.

Edasiste vaatluste käiguks jäeti antud 21 lauset kõrvale, kuna analüüsiks on oluline tõlgitud lausete siltide sisalduvus.

Kui eelnevate töötluste ning analüüside käigus oli vaatluse all 6 keelepaari, siis edasistes siltidega seotud küsimustes on kõrvale jäetud inglise-läti ning inglise-leedu keelepaarid, kuna nende testandmetes ei leidunud silte sisaldavaid lauseid (tabel 5).

Tabel 5. Mudelite testandmete silte sisaldavate lausete hulk keelepaaride kaupa.

Keelepaar	Silte sisaldavad laused	% testandmetest
EN-LV	0	0
EN-LT	0	0
EN-ET	274	13,70%
DE-LV	515	25,75%
DE-LT	545	27,25%
DE-ET	396	19,8%
Kokku	**1730**	**14,41%**

Maskitud mudeli siltidega laused sisaldasid kokku 4379 silti kolmes eritüübis. Tabel 6 kirjeldab testandmetel leitud silditüüpe esinemissageduste kaupa.

Tabel 6. Maskitud mudeli testandmetel leiduvad silditüübid esinemissageduste kaupa.

Silditüüp	Siltide arv
bpt	1170
ept	1170
ph	2039
Kokku	**4379**

Baasmudeli tõlkeandmed sisaldasid 1730 lause kohta 4381 silti, mida on 2 võrra enam võrreldes maskitud mudeliga (tabel 7). Baasmudeli testandmetel ilmnes ka huvitav viga, mille käigus on kolm ph tüüpi silti kaotanud oma tüübinime protsessi käigus (joonis 9).

Tabel 7. Baasmudeli testandmetel leiduvad silditüübid esinemissageduste kaupa.

Silditüüp	Siltide arv
bpt	1171
ept	1171
ph	2036
tühi	3
Kokku	**4381**

lähe: <ph x="6"type="6"/>HomeLink
tõlge: < x="6"type="6"/>HomeLink

Joonis 9. Näide kaduma läinud tüübinimest baasmudeli tõlkelause sildis.

5.2 Kvalitatiivne analüüs

Tõlkeandmete peal viidi läbi vaatlus, mille käigus uuriti iga keelepaari kohta juhuslikult valitud 100 tõlgitud lauset ning nende oodatavat tõlkelauset. Kokku vaadeldi 800 tõlkelauset, 400 ühe mudeli kohta. Vaatluse käigus võrreldi siltide korrektsust tõlkelausetes, tühikuid ning suurtähe probleeme. Andmed jagati omakorda kahte hulka: laused, kus oli 1 või 2 silti ning laused, kus oli üle 3 sildi. Tabelites 8 ning 9 on välja toodud maskitud mudeli vaatluse andmed.

Vaatluse käigus ilmnes maskitud mudelite väga oluline eelis baasmudeli ees - baasmudeli järeltöödeldud laused olid kõik tõlkimata. See tähendab, et mudel ei õppinud tõlkima silte sisaldavat teksti, vaid kopeeris lähtelaused tõlkelauseteks. Antud informatsiooni teades ilmneb, et meie baasmudel on kasutu struktureeritud tekstide tõlkimisel, kuna ei õpi silte tõlkima. Selle teadmisega saab juba tõdeda andmete maskimise vajalikkust silte sisaldavatel tekstidel.

Tabel 8. Maskitud mudeli 1-2 silti sisaldavate vaatluslausete siltide tõlketäpsused.

Keelepaar \ Otsus	Õigesti	Puudulikult
EN-ET	90	0
DE-LV	60	1
DE-LT	70	1
DE-ET	69	1
Kokku	**289**	**3**

Tabel 9. Maskitud mudeli 3 või rohkemat silti sisaldavate vaatluslausete siltide tõlketäpsused.

Keelepaar \ Otsus	Õigesti	Puudulikult
EN-ET	10	0
DE-LV	37	2
DE-LT	29	0
DE-ET	30	0
Kokku	**106**	**2**

1-2 silti sisaldavate lausetest olid puudulikult sildistatud 3 lauset 292-st, mis on 1,04% vaatlusandmetest. 3 ning enam silti sisaldavat lauset seas leidus 2 puudulikult sildistatud lauset, mida on 1,85% vaatlusandmetest. Puudulikult sildistatud lausete puhul oli lausete

ning siltide struktuur õige, kuid leidusid kas lisasildid või ilmnesid erinevused nimedes. Joonistel 10 ning 11 ning on toodud näited leitud siltide probleemidest.

lähe: Drehschalter<Bpt type="15050"x="1"/>
siht: Keerake pöördlüliti<bpt type="15050"x="1"/>
tõlge: Keerata pöördlüliti<Bpt type="15050"x="1"/>

Joonis 10. Näide maskitud mudeli oodatava tõlke ning tõlgitud lause sildinime erinevusest.

lähe: <ept x="1"/> neu
siht: <ept x="1"/> uus<ph type="4"/>
tõlge: <ept x="1"/> uus

Joonis 11. Näide maskitud mudeli oodatava tõlke ning tõlgitud lause siltide erinevusest.

Siltide probleemide sügavamal uurimisel leidus, et vead ei tekkinud mitte mudeli poolt, vaid erinevustest lähtelause ning sihtlausete vahel. Mudel kasutab õppimiseks lähtekeeles olevaid lauseid ning kujundab ka tõlked antud kujule vastavalt.

Lisaks siltide täpsusele uuriti ka tühikute ning suurtähtedega tekkinud probleeme. Tabelites 10 ning 11 on välja toodud vaatluse käigus leitud mudelitepoolsed tühikute ning suurtähtede vead.

Tabel 10. Maskitud mudeli silte sisaldavate lausete vead 1-2 silti sisaldavates lausetes.

Keelepaar \ Probleem	Tühikute vead	Suurtähtede vead
EN-ET	9	0
DE-LV	11	3
DE-LT	7	3
DE-ET	6	0
Kokku	**33**	**6**

Lisaks mudeli poolt tekitatud vigadele oli tõlke ning oodatud tõlkelause vahel ka mudelist mittesõltuvaid tühikute ning suurtähtede erinevusi. Vaadeldud 400 lausest 25,25%-il esines tühikute erinevusi. 1-2 silti sisaldavate lausete seas oli tühikute probleemid 19,52% andmetest ning 3 või enam silti sisaldavate lausete puhul oli tühikute probleeme 40,74%

lausetest. Kui 1 või 2 silti ning kolm ja rohkem silti sisalduvate lausete tühimike probleemidest vastavalt 57,89% ning 56,81% olid seotud mudeli poolt puudulikult või üleliigselt märgitud tühikutega, siis ülejäänud andmete puhul tekkisid probleemid taaskord sihttõlke ning oodatava tõlke lausete erinevustega.

Tabel 11. Maskitud mudeli silte sisaldavate lausete vead 3 või enam silti sisaldavates lausetes.

Probleem\ Keelepaar	Tühikute vead	Suurtähtede vead
EN-ET	1	0
DE-LV	9	0
DE-LT	6	0
DE-ET	9	1
Kokku	**25**	**1**

Enamus tekkinud lisatühikuid olid seotud keeltevaheliste lausestruktuuride erinevustega. Kui lähtelauses oli sildi ning lauselõpu punkti vahel sõna, kuid tõlgitud lause struktuuri puhul oli sildi ning punktivaheline sõna liikunud ettepoole, tekkis sildi ning lauselõpumärgi vahele üleliigne tühik. Joonis 12 illustreerib tekkinud olukorda.

lähe: Zurückschalten: Am Lenkradschaltpaddle <ph x="1"type="2"/> ziehen.

siht: Perjungti žemesnę pavarą: Patraukite ant vairo esančią pavarų perjungimo svirtelę <ph x="1"type="2"/>.

tõlge: Žemesnės pavaros perjungimas: patraukite ant vairo esančią pavarų perjungimo svirtelę <ph x="1"type="2"/> .

Joonis 12. Näide üleliigsest tühikust maskitud mudeli tõlgitud lause lõpus, mida sihttõlkes ning lähtelauses ei ole.

Antud maskitud mudel õppis valesti panema tühikuid ka komade ette, mida originaallauses ei olnud, kuid tõlkes oli. Joonisel 13 on näide antud olukorrast. Ülejäänud tühikutega seotud probleemid tekkisid alglause ning lähte tõlkelause tühikute erinevustest. Läbiv erinevus oli siltide vahel asetsevate üksiksõnade ümber leiduvate tühikute lahknevus.

Suurtähtede probleeme esines 1-2 silti sisaldavate lausete seas 9,59% ning 3 või enamat silti sisaldavate lausete seas 7,41% andmetel. 1-2 silti sisaldavate andmete puhul oli 28-st

probleemist 6 tõstu mudeli poolt vigaselt määratud ning 3 või enam sildiga lausete puhul oli vigaselt märgitud 1 tõst 8-st. Vigadena mitte arvesse võetud andmete puhul ilmnesid taaskord erinevused alglause ning oodatava tõlkelause vahel.

lähe: Bei einem <bpt type="b960.257_960.1"x="1"/>Wert = 0<ept x="1"/> wird dieser Parameter ignoriert.

siht: Kui <bpt type="b960.257_960.1"x="1"/>väärtus = 0<ept x="1"/>, eiratakse seda parameetrit.

tõlge: Kui <bpt type="b960.257_960.1"x="1"/>väärtus = 0<ept x="1"/> , ignoreeritakse seda parameetrit.

Joonis 13. Näide tõlkelausesse koma ette lisatavast tühikust, mida lähte- ning sihttõlkelauses ei ole.

Vaadeldud andmete puhul saab tõdeda, et paljud lahknevused tõlke ning oodatud tõlke vahel on seotud algandmete erinevustega.

Kokkuvõte

Käesoleva lõputöö käigus võeti vaatluse alla struktureeritud tekstide sildistatud tõlkeandmete tõlkimine. Algandmete peal läbi viidud eeltöötluste käigus tekitati kaks andmestikku. Esimene neist sisaldas maskitud kujul andmeid ning teine säilitas silte sisaldavate tõlkelausete algkuju. Mõlema andmestiku peal treeniti masintõlke mudel, mida kasutades tõlgiti 12000 testlauset. Tõlgitud andmetele rakendati omakorda järeltöötlust, mille käigus viidi eeltöötluse käigus kuju muutnud laused tagasi algsele silte sisaldavale kujule. Tõlgitud lausetele rakendati SacreBLEU skoori, millega hinnati tõlketäpsuseid. Tulemustest ilmnes, et maskitud mudel saavutas parema tulemuse. Keskmine SacreBLEU skoor antud mudelil oli 46,78 punkti, kui baasmudeli keskmine tulemus oli vaid 33,86 punkti. Vaatluse käigus ilmnes, et baasmudel ei õppinud tõlkima silte sisaldavaid lauseid, vaid kooperis sisendandmed väljundisse. Seda teades saab esiteks tõdeda, et BLEU skoor üksi on nõrk meetrika tõlkekvaliteetide hindamiseks. Teiseks saab kinnitada maskimise vajalikkust struktureeritud tekstide masintõlkimisel.

12000-st testlausest 14,41% sisaldasid silte. Kuuest keelepaarist nelja testandmed sisaldasid silte, mida järgnevate lähemalt vaadeldi. Siltide uurimisel ilmnes ka, et eeltöötluse käigus rakendatud andmete filtrid on teatud tüüpi andmete puhul katki ning lasevad läbi laused, mis tulnuks eeltöötluste käigus andmete hulgast eemaldada.

Siltidega tõlkelaused sisaldasid kolme erinevat silditüüpi. Baasmudeli puhul ilmnes ka viga, mille käigus mudel unustas silditüübi kirjutada. 1730-st sildistatud tõlkelausest 800 puhul viidi läbi ka vaatlus, mille käigus hinnati siltide täpsust tõlkelausetes ning vaadeldi tühikute ning suurtähtede probleeme. Töö käigus saadi teada, et masintõlke mudel ei õpi panema liiga palju ega vähe silte väljundisse. Läht- ning sihtkeelte lausete struktuuri erinevustest tulenevalt tekib aga omajagu probleeme tühikutega lauselõpu märkide ning kirjavahemärkide ees. 1 või 2 silti ning 3 või enam silti sisaldavate maskitud mudeli tõlkelausete tühikute probleemidest vastavalt 57,89% ning 56,81% puuduse puhul oli tegemist tõlkemudeli poolse veaga. Ülejäänud probleemid tulenesid algandmete erinevustest. Maskitud mudeli poolt vigaselt märgitud tõstude arv oli 1 või 2 silti sisaldavate lausete puhul 2,05% ning 3 või enamat silti sisaldavate lausete puhul 0,93%.

Antud tööga kinnitati, et andmete maskimine struktureeritud silte sisaldavate tekstide puhul on vajalik. Käesoleva töö edasiarendusena võiks maskitud andmete tõlketäpsuste tõstmiseks sisse viia parandused tühikute probleemide parandamiseks. Tõlkeandmete lähtetõlgete ning oodatavate tõlgete omavahelist sobivust peaks samuti parandama, kuna töö käigus ilmnes mitmeid probleeme nende andmete omavahelistest erinevustest.

Viidatud kirjandus

[1] Greg Hanneman and Georgiana Dinu. How should markup tags be translated? In *Proceedings of the Fifth Conference on Machine Translation*, pages 1160–1173, Online, November 2020. Association for Computational Linguistics.

[2] Kazuma Hashimoto, Raffaella Buschiazzo, James Bradbury, Teresa Marshall, Richard Socher, and Caiming Xiong. A high-quality multilingual dataset for structured documentation translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 116–127, Florence, Italy, August 2019. Association for Computational Linguistics.

[3] Morris A. Jette, Andy B. Yoo, and Mark Grondona. Slurm: Simple linux utility for resource management. In *In Lecture Notes in Computer Science: Proceedings of Job Scheduling Strategies for Parallel Processing (JSSPP) 2003*, pages 44–60. Springer-Verlag, 2002.

[4] Project Jupyter. The jupyter notebook, 2021. (20.03.2021).

[5] Michel Kana. Truecasing in natural language processing. https://towardsdatascience.com/truecasing-in-natural-language-processing-12c4df086c21, 2019. (24.04.2021).

[6] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[7] Lucian Vlad Lita, Abe Ittycheriah, Salim Roukos, and Nanda Kambhatla. tRuEcasIng. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 152–159, Sapporo, Japan, July 2003. Association for Computational Linguistics.

[8] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation, 2001.

[9] Matt Post, Shuoyang Ding, Marianna Martindale, and Winston Wu. An exploration of placeholding in neural machine translation. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, pages 182–192, Dublin, Ireland, August 2019. European Association for Machine Translation.

[10] University of Tartu. Ut rocket, 2018. (20.03.2021).

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

Lisad

Lõputöö käigus loodud analüüsimise koodid on kättesaadavad autori GitHubis: https://github.com/skaren99/bachelor_thesis

I. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Karen Saksakulm**,

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

   Tõlkebüroode Grata ja Interlex masintõlke mudelite vigade analüüs ja lahendused,

   mille juhendajad on Andre Tättar ja Liisa Rätsep,

   reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commonsi litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Karen Saksakulm
07.05.2021