TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Martin Kivisikk

Nimeüksuste tuvastaja loomine puudepanga korpuse põhjal

Bakalaureusetöö (9 EAP)

Juhendaja:
Siim Orasmaa, PhD

Tartu 2025

Nimeüksuste tuvastaja loomine puudepanga korpuse põhjal

Lühikokkuvõte:

Keeletehnoloogias on nimeüksuste tuvastamise eesmärk märgendada tekstis infoüksused, näiteks isiku-, organisatsiooni- ja kohanimed. Eesti kirja- ja veebikeele puudepankadele on lisatud nimeüksuste märgendused, aga nendel korpustel ei ole veel nimeüksuste tuvastamise mudeleid loodud. Töös peenhäälestati BERTil põhinevad mudelid nii eraldi kui ka ühisel treeningandmestikul. Parimaks mudeliks osutus ühisel treeningandmestikul peenhäälestatud Est-RoBERTa, mis saavutas testandmestikul F-skoori 0,828. Töös selgus, et välistel andmestikel on mudelitel keerulisem nimeüksuseid tuvastada, sest ei ole tagatud, et nimeüksused on eri andmestikes sarnaselt defineeritud ja tekstis märgendatud.

Võtmesõnad: nimeüksuste tuvastamine, BERT

CERCS: P176 Tehisintellekt; P175 Informaatika, süsteemiteooria

Developing a Named Entity Recognition Model Based on Treebank Corpora

Abstract:

In natural language processing, named entity recognition aims to tag information units in text, such as names of people, organizations and locations. Named entity tags have recently been added to the Estonian UD treebanks, but no named entity recognition models using the datasets have been made. In this thesis, models based on BERT were fine-tuned on both individual and combined training sets. The best model turned out to be Est-RoBERTa fine-tuned on the combined training set, which achieved an F-score of 0.828 on the test set. The study revealed that models perform worse on external datasets, as named entities are not necessarily defined and annotated consistently across different corpora.

Keywords: named entity recognition, BERT

CERCS: P176 Artificial intelligence; P175 Informatics, systems theory

Sisukord

Sissejuhatus

Nimeüksuste tuvastamise eesmärk on automaatselt leida ja märgendada tekstis olulised infoüksused, näiteks isiku-, organisatsiooni- ja kohanimed. Nimeüksuste tuvastamine on muuhulgas kasulik struktureerimata andmetest informatsiooni eraldamisel, tekstide liigitamisel ja tundlike andmete anonüümseks tegemisel. Eesti keele jaoks on välja töötatud mitmeid nimeüksuste tuvastamise mudeleid, nt EstBERTil põhinevad neuromudelid (Sirts, 2023) ning Est-RoBERTal põhinev vana kirjakeele nimeüksuste tuvastamise mudel (Orasmaa jt, 2022). Ka eesti keele puudepangale on lisatud käsitsi nimeüksuste märgendused (Muischnek jt, 2023), aga teadaolevalt ei ole sellele korpusele veel automaatseid nimeüksuste tuvastamise katseid tehtud ega mudeleid loodud. Töö eesmärk on täita see lünk ning eksperimenteerida nimeüksuste tuvastamisega nii kirjakeele[1] kui ka veebikeele[2] puudepankadel, leida parim nimeüksuste tuvastamise mudel ning võrrelda selle märgenduskvaliteeti ka olemasolevate neuromudelite kvaliteediga.

Töö koosneb kolmest peatükist. Esimeses peatükis on antud ülevaade nimeüksuste tuvastamise meetoditest, BERT mudeli taustast ning eesti keele jaoks loodud mudelitest ja nimeüksustega märgendatud andmestikest. Teises peatükis on antud tehnilisem ülevaade töös kasutatud andmestikest ja andmestike eeltöötlusest, lisaks on kirjeldatud alusmudelite peenhäälestamiseks kasutatud keskkonda ning nimeüksuste tuvastamise mudelite hindamist. Kolmandas peatükis on välja toodud töö tulemused.

Töös on mudelite ennustuste hinnangud testandmestikel esitatud tabelitena ning ühe kokkuvõtva joonisena. Lisades on tabelitele vastavad joonised, mis võivad tulemustest anda parema ülevaate.

[1] https://github.com/UniversalDependencies/UD_Estonian-EDT
[2] https://github.com/UniversalDependencies/UD_Estonian-EWT

Mõisted ja terminid

Neurovõrk[3] (ingl *neural network*) on tehisintellektis kasutatav andmetöötlusmudel, mis jäljendab bioloogilise neuronivõrgu omadusi. Nende abil saab muuhulgas luua mudeleid, mis märgendavad sisendiks antud teksti.

Puudepank[4] (ingl *treebank*) on süntaktiliselt märgendatud keelekorpus, milles teksti iga lause jaoks on leitud selle lause puukujuline struktuur.

Siirdeõpe[5] (ingl *transfer learning*) on masinõppe tehnika, mille puhul ühe ülesande täitmiseks omandatud infot kasutatakse teiste ülesannete täitmiseks.

Peenhäälestamine ehk **täppistreenimine** (ingl *finetuning*) on siirdeõppe alamliik, mille käigus kohandatakse eeltreenitud mudelit kasutades väiksemat märgendatud andmestikku (Sügis jt, 2024, lk 156).

[3] https://akit.cyber.ee/
[4] Müürisep, K. Eesti keele puudepank. 2009. https://kodu.ut.ee/~kaili/Korpus/puud/
[5] Bogdanov, D., Etti, P., Kamm, L., Ostrak, A., Stomakhin, F., Toomsalu, M., Valdma, S.-M., Veldre, A. Tehisintellekti ja masinõppe tehnoloogia riskide ja nende leevendamise võimaluste uuring. 2024. https://www.ria.ee/sites/default/files/documents/2024-03/Tehisintellekti-masinoppe-tehnoloogia-riskide-uuring-2024.pdf

1. Taust

Selles peatükis antakse ülevaade nimeüksuste tuvastamise meetoditest, BERT keelemudelist, eesti keele jaoks nimeüksustega märgendatud andmestikest ja analüüsitakse teemaga seotud uurimistöid.

1.1 Nimeüksuste tuvastamise meetodid

Järgnev alapeatükk on kirjutatud Maud Ehrmanni jt (2021) töö põhjal, kui ei ole väidetud teisiti.

Maud Ehrmanni jt (2021) sõnul käsitletakse nimeüksuste tuvastamist kui tekstijada märgendamise ülesannet. Autorid selgitavad, et mudeli treenimise eesmärk on õppida olemasolevate sõna-märgend paaride põhjal märgendama seni nägemata tekste. Maud Ehrmanni jt (2021) töös kirjeldatakse ka põhilisi nimeüksuste tuvastamise meetodeid.

Esiteks on välja toodud reeglitepõhine lähenemine: selle meetodi puhul tuvastatakse nimeüksused käsitsi loodud reeglite abil. Kuigi treeningandmeid ei ole siis vaja, on puudus see, et andmestikule vastavate reeglite loomine on aeganõudev.

Teiseks on antud ülevaade masinõppel põhinevatest lähenemistest: need meetodid vajavad suuri märgendatud tekstikorpuseid ning õppimine toimub andmetest käsitsi valitud tunnuste põhjal. Nimeüksuste tuvastamisel sai tähtsamaks just CRF (ingl *linear-chained conditional random field*) (Lafferty jt, 2001; viidatud Ehrmann jt, 2021 kaudu)[6], sest suudab arvestada nii eelnevate- kui ka järgnevate sõnedega.

Viimasena on kirjeldatud süvaõpet kasutavaid meetodeid, kus on keskne roll kahel tehnoloogial. Esiteks kahesuunalistel pika lühimäluga neurovõrkudel (ingl *bidirectional long short-term memory network*), mida rakendasid esimesena Huang jt (2015; viidatud Ehrmann jt, 2021 kaudu)[7]. Kahesuunalisus tähendab seda, et sisendit töödeldakse nii vasakult paremale kui ka paremalt vasakule. Samuti on märkimisväärne enesetähelepanuga võrkude ehk transformerite (Vaswani jt, 2017; viidatud Ehrmann jt, 2021 kaudu)[8] areng, kusjuures

[6] Lafferty, J., McCallum, A., Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.
[7] Huang, Z., Xu, W., Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015. https://arxiv.org/abs/1508.01991
[8] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł., Polosukhin, I. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. https://user.phil.hhu.de/~cwurm/wp-content/uploads/2020/01/7181-attention-is-all-you-need.pdf

nimeüksuste tuvastamisel on eriti silmapaistev BERT-arhitektuur (Devlin jt, 2019; viidatud Ehrmann jt, 2021 kaudu).

Raamatus "Praktiline andmeteadus" (Sügis jt, 2024) on kirjutatud, et transformer-arhitektuur võimaldab võrreldes rekurrentsete närvivõrkudega paremat paralleeltöötlust, mis teeb treenimise kiiremaks ja tõhusamaks. Transformerite põhiline eelis on enesetähelepanu mehhanism, mis annab mudelile võime mõista konteksti ja seoseid andmetes (Sügis jt, 2024, lk 155). BERT mudeli tausta on täpsemalt kirjeldatud järgmises alapeatükis.

1.2 BERT

BERT (ingl *Bidirectional Encoder Representations from Transformers*) mudeli töötasid välja tehnoloogiaettevõtte Google teadlased 2018. aastal (Devlin jt, 2019). BERT on arhitektuuri poolest mitmekihiline kahesuunaline transformer kodeerija, millest loodi esialgu kaks varianti. Tabelis 1 on välja toodud BERT alusmudelite parameetrite väärtused, kus L tähistab transformeri plokkide ehk kihtide arvu, H tähistab peidetud suurust (ingl *hidden size*) ehk mudeli dimensionaalsust ning väärtus A näitab, mitu enesetähelepanu pead igal transformeri plokil on (Devlin jt, 2019).

Tabel 1. BERT alusmudelite parameetrid Devlini jt (2019) töö põhjal

Mudel	**Transformer plokkide arv (L)**	**Peidetud suurus (H)**	**Enesetähelepanu peade arv (A)**
$\text{BERT}_{\text{BASE}}$	12	768	12
$\text{BERT}_{\text{LARGE}}$	24	1024	16

Mudelite loomisel katsetati ka väiksemate kihtide ja enesetähelepanu peade arvudega, aga Devlini jt (2019) töös tuli välja, et suuremad mudelid saavad paremini hakkama nii suuremate, näiteks masintõlke ja keele modelleerimise kui ka väiksemate ülesannetega. $\text{BERT}_{\text{BASE}}$ mudeli parameetrid valiti samad, mis OpenAI GPT mudelil, et mudeleid võrrelda. BERT kasutab kahesuunalist enesetähelepanu ning kahesuunalise transformeri asemel kasutatakse ka mõistet "transformer kodeerija" (ingl *transformer encoder*), samas kui GPT (ingl *generative pre-trained transformer*) mudelid saavad kasutada vaid sõnest (ingl *token*) vasakule jäävat konteksti ning selle jaoks kasutatakse mõistet "transformer dekodeerija" (ingl *transformer decoder*) (Devlin jt, 2019).

Järgnevad lõigud eeltreenimisest ja peenhäälestamisest on kirjutatud Jurafsky jt (2025) raamatu "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models" 11. peatüki põhjal, kui ei ole väidetud teisiti.

BERT-arhitektuuril põhineva mudeli eeltreenimisel on kasutusel kaks ülesannet: lünkteksti täitmine ehk MLM (ingl *masked language modeling*) ning järgmise lause ennustamine ehk NSP (ingl *next sentence prediction*). MLMi puhul valitakse mingi osa (BERTi jaoks 15%) mudelile sisendiks antud järjendi sõnadest, mida maskeerida. Maskeerimisel on iga valitud sõna jaoks kolm varianti: 80% juhtudel asendatakse valitud sõna erilise sõnaga "[MASK]", 10% juhtudel asendatakse valitud sõna mõne teise sõnastikust suvaliselt valitud sõnaga ning ülejäänud 10% ajast jääb valitud sõna samaks. Nii treenitakse mudelit ennustama algset sisendit iga maskeeritud sõna kohta. MLM ülesanne keskendub ümbritseva konteksti abil maskeeritud sõnade ennustamisele ja treenib mudelit aru saama sõnade tähendustest.

Mudelit õpetatakse ka järgmise lause ennustamise ülesandel, kus eesmärk on ennustada, kas antud lausepaar esineb treeningandmestikus järjestikku või koosneb omavahel mitteseotud lausetest. Nii õpib mudel määrama lausetevahelisi suhteid, näiteks kas kahel lausel on sarnane tähendus või mitte. BERTi treenimisel NSP jaoks kasutatakse lausepaare, kus pooled lausepaarid on omavahel seotud ning ülejäänud paaridel valitakse teine lause andmestikust suvaliselt. Devlini jt (2019) sõnul kasutati mudeli eeltreenimisel 90% sammudest sisendi pikkust 128 ja viimased 10% sammudest treeniti pikkusel 512, sest tähelepanu on sisendi pikkuse suhtes ruutkeerukusega ning pikemad sisendid muudavad arvutused liiga kalliks.

Eeltreenitud keelemudelid on võimsad just seepärast, et suudavad märgendamata tekstil treenides tuletada ja meelde jätta üldiseid teadmisi, mida saab kasutada teisteks ülesanneteks. Nende ülesannete hulka kuuluvad näiteks teksti ja lausepaaride klassifitseerimine ning sõnajärjendite märgendamine. Keelemudelite peenhäälestamisel lisatakse alusmudelile *softmax* aktiveerimisfunktsiooniga klassifitseerimiskiht, mille sisend on keelemudeli väljund. Peenhäälestamise protsessis kasutatakse märgendatud andmestikku, et treenida lisatud klassifitseerimiskihti. Alusmudeli kaale selle protsessi käigus ei muudeta või kasutatakse madalat õppimiskiirust. Sõnajärjendite märgendamise ülesandes, mille üks levinumaid rakendusi on nimeüksuste tuvastamine, peab mudel igale järjendi elemendile määrama ühe märgendi kindlast märgendite hulgast. Selle ülesande puhul suunatakse iga sõna väljundvektor klassifitseerimiskihti, mis omakorda tagastab tõenäosusjaotuse üle võimalike märgendite hulga.

Liu jt (2019) töös tuli välja, et algseid BERT mudeleid (Devlin jt, 2019) treeniti liiga vähe ning uuriti, kuidas vastavate mudelite treenimist optimeerida. Liu jt (2019) töö tulemusena valmisid RoBERTa alusmudelid, mille eeltreenimisel tehti võrreldes Devlini jt (2019) tööga neli olulist

muudatust. Esiteks kasutati ainult lünkteksti täitmise ülesannet, sest leiti, et NSP ülesande eemaldamisel on mudeli tulemused kas samad või paremad. MLM ülesande jaoks kasutati staatilise maskeerimise asemel dünaamilist maskeerimist. Algselt tekitati lüngad treeningandmetesse andmestiku eeltöötluse ajal, aga dünaamilise maskeerimisega genereeritakse lüngad vahetult enne sisendi andmist mudelile. Terve eeltreenimise protsessi käigus kasutati maksimaalset sisendi pikkust 512. Lisaks toodi välja, et mahukam treeningandmestik, pikem treenimine ja suurem ploki suurus (ingl *batch size*) toovad endaga kaasa paremad tulemused väiksematel andmestikel peenhäälestamisel (Liu jt, 2019).

Eesti keele jaoks on loodud erinevaid alusmudeleid, kasutades suuri märgendamata eestikeelseid tekstikorpuseid ja eelnevalt määratud BERTil põhinevat mudeli arhitektuuri. EstBERTi treenimisel kasutati originaalset $\text{BERT}_{\text{BASE}}$ arhitektuuri ning mudelit treeniti 2017. aasta eesti keele ühendkorpusel, mis pärast puhastamist koosnes 1,154 miljardist sõnast (Tanvir jt, 2021). Eesti keele ühendkorpus 2017 (Kallas jt, 2018), mille suurus on 1,3 miljardit sõna, koosneb peamiselt veebi- ja ajakirjandustekstidest ning eestikeelsetest vikipeedia artiklitest.

Ulčar jt töös (2021) treeniti RoBERTa alusmudelil põhinevad Est-RoBERTa ja LitLat BERT mudelid, kusjuures LitLat BERT treeniti läti, leedu ja inglisekeelsetest tekstidest koosnevast andmestikul. Est-RoBERTa mudel treeniti eestikeelsel korpusel, mille suurus oli 2,51 miljardit sõna ning koosnes peamiselt Ekspress Meedia artiklitest ja CoNLL 2017 korpuse eestikeelsest osast. Est-RoBERTa treenimiseks kasutatud andmestikus oli märgatavalt rohkem eestikeelseid tekste kui teiste eesti keele jaoks mõeldud mudelite treeningandmestikes (Ulčar jt, 2021).

1.3 Seotud tööd

Muischneki jt (2023) töö raames on käsitsi lisatud nimeüksuste märgendid eesti kirja-[9] (ingl *Estonian Universal Dependencies Treebank*, ka EDT) ja veebikeele[10] (ingl *Estonian Universal Dependencies Web Treebank*, ka EWT) puudepankadele. Esimene neist koosneb umbes 440 000 sõnast ja sisaldab nii ilukirjandus-, ajakirjandus- kui ka teadustekste. Ajakirjandustekstid moodustavad üle poole andmestikust, ilukirjandusel ja teadustekstidel on väiksem osakaal. Teiseks on kirjeldatud väiksemat veebikeele puudepanka, mis koosneb umbes 90 000 sõnast ja on loodud blogipostitustest, kommentaaridest ja foorumitest pärinevate tekstide põhjal

[9] https://github.com/UniversalDependencies/UD_Estonian-EDT
[10] https://github.com/UniversalDependencies/UD_Estonian-EWT

(Muischnek jt, 2023). See võimaldab uurida nimeüksuste tuvastamist tekstides, kus kirjakeele normid ei pruugi alati kehtida.

Muischnek jt (2023) otsustasid korpustes märgendada kaheksat tüüpi nimeüksusi: isikud *PER* (sh loomade ja väljamõeldud tegelaste nimed), asukohad *LOC* (maastikuobjektid ja asulad), geopoliitilised üksused *GEP* (riigid ja mõnel juhul linnad), organisatsioonid *ORG*, tooted *PROD* (inimese loodud objektid, ideed ja teooriad), sündmused *EVE*, muud *OTHER* ja tundmatud nimeüksused *UNK*. Nimeüksusele määrati kategooria *OTHER*, kui see ei sobinud ühtegi teise kategooriasse, näiteks "U3 projekt". Kategooriat *UNK* kasutati nende nimeüksuste jaoks, mille tähendus ei olnud selge. Autorite sõnul piirduti vaid nende kategooriatega, sest liigsete klasside kasutamine raskendab nii märgendamist kui ka lõpuks mudeli treenimisel parema täpsuse saavutamist.

Sirtsi (2023) töö käigus märgendati nimeüksustega samuti kaks andmestikku: varasem 220 000-sõnaline ajakirjandustekstide korpus EstNER[11] ning uus 130 000-sõnaline korpus EstNER_new[12], mis sisaldab nii ajakirjandus- kui ka veebitekste. Tanviri jt (2021) töö käigus loodi esialgne BERTil põhinev nimeüksuste tuvastamise mudel, EstBERT_NER[13], mille peenhäälestamiseks kasutati originaalset EstNER andmestikku (Tkachenko jt, 2013). EstNER andmestik märgendati Sirtsi töös uuesti, sest Tanvir jt (2021) leidsid sellest vigu. Nii Muischneki jt (2023) kui ka Sirtsi (2023) töös märgendati vastavates andmestikes nimeüksustena isikunimed *PER*, organisatsioonid *ORG*, asukohad *LOC*, geopoliitilised üksused *GPE* (Muischneki jt töös *GEP*), tootenimed *PROD* ja sündmused *EVENT* (Muischneki jt töös *EVE*). Lisaks märgendati Sirtsi töös nimeüksustena tiitlid *TITLE*, kuupäevad *DATE*, ajaväljendid *TIME*, rahalised väärtused *MONEY* ja protsente väljendavad üksused *PERCENT*. Kasutusel ei olnud aga kategooriaid *OTHER* ja *UNK*, mida kasutati Muischneki jt (2023) töös teistesse klassidesse mittesobivate või ebaselge tähendusega nimeüksuste märgendamiseks.

Pärast andmestike märgendamist jätkati Sirtsi (2023) töös mudelite treenimisega kahel eesmärgil. Esiteks sooviti kindlaks teha, kui hea tulemuse mudelid uutel andmestikel saavutavad ning teiseks uuriti, kas parem on treenida üks kombineeritud andmetega mudel või kaks mudelit eraldi andmestikel. EstBERTil[14] (Tanvir jt, 2021) põhinevad mudelid treeniti kõigepealt eraldi andmestikel ning lõpuks kasutades mõlema korpuse andmeid. Eraldi treenitud

[11] https://github.com/TartuNLP/EstNER
[12] https://github.com/TartuNLP/EstNER_new
[13] https://huggingface.co/tartuNLP/EstBERT_NER
[14] https://huggingface.co/tartuNLP/EstBERT

mudelid saavutasid vastavatel valideerimisandmestikel F-skoorid 0,747 ja 0,735. F-skoori kasutatakse mudelite ennustustulemuste hindamiseks. Nimeüksuste tuvastajate hindamismeetodeid on täpsemalt kirjeldatud siinses töös peatükis 2.4. Sirts leidis, et andmestike ühendamine andis üldiselt parema tulemuse: ühisel valideerimisandmestikul saavutati F-skoor 0,761 ning parim mudel, EstBERT_NER_v2[15] saavutas kombineeritud testandmestikul F-skoori 0,774 (Sirts, 2023). Järelikult tasub ka siinse töö raames peenhäälestada mudelid kirja- ja veebikeele puudepankadel eraldi ning treenida mudelid kasutades ka mõlemat korpust, sest suurema hulga andmetega on mudelil võimalik saavutada kõrgem täpsus.

Lisaks kaasaegsetele tekstidele on märgendatud ka vanemaid eestikeelseid tekste ning nendel andmetel mudeleid loodud. Orasmaa jt (2022) töö raames märgendati nimeüksustega korpus, mis sisaldab 19. sajandist pärit vallakohtuprotokolle. Muuhulgas näidati, et peenhäälestades BERTil (Devlin jt, 2019) põhinevaid mudeleid nimeüksuste märgendamiseks ajaloolistel andmetel on võimalik saavutada samasugune täpsus nagu parimatel mudelitel, mille treenimisel kasutati tänapäevaseid eestikeelseid tekste. Eksperimentide käigus osutus siirdeõppel parimaks mudeliks Est-RoBERTa[16], mis saavutas testandmestikul F-skoori 0,936. Kuigi Orasmaa jt (2022) töös mainitud keelemudelid eeltreeniti kaasaegsel eesti keelel, saavutasid need ka ajaloolistel tekstidel peenhäälestades häid tulemusi. Autorite arvates tulenesid kõrged skoorid andmete kvaliteedist ning ühtlasest struktuurist.

[15] https://huggingface.co/tartuNLP/EstBERT_NER_v2
[16] https://huggingface.co/tartuNLP/est-roberta-hist-ner

2. Metoodika

Selles peatükis on kirjeldatud andmestikku, selle sisselugemist ja töötlemist, mudelite peenhäälestamist ja hindamist.

2.1 Andmestik

Eesti kirja- ja veebikeele puudepankadele on lisatud Muischneki jt (2023) töö raames nimeüksuste märgendid. Andmestike märgendamisel on kasutatud BIO formaati (Ramshaw jt, 1995), kus B tähistab üksuse algust, I tähistab üksuse jätkamist ning O-märgendit kasutatakse nende sõnade jaoks, mis ei kuulu ühtegi nimeüksuse klassi. BIO formaadis märgendid määravad nii nimeüksuse tüübi kui ka piirid, seega nimeüksuste tuvastamist saab käsitleda kui järjendi märgendamise ülesannet, kus igale sõnale on määratud üks märgend (Jurafsky jt, 2025).

Mõlemad korpused on jagatud treening-, arendus- ja testandmestikeks. Andmefailid ise on CoNLL-U formaadis[17], kus iga sõna on kogu kaasneva infoga ühel real ja lauseid eraldab tühi rida. Sõna lisainfo hulka kuuluvad järgmised väärtused:

1. ID: sõna indeks, mis igas lauses algab arvust 1;
2. FORM: sõna või kirjavahemärk;
3. LEMMA: sõna algvorm või tüvi;
4. UPOS: universaalne sõnaliik;
5. XPOS: keelespetsiifiline sõnaliik (puudumisel alakriips);
6. FEATS: morfoloogiliste tunnuste loetelu (puudumisel alakriips);
7. HEAD: antud sõna süntaktiline ülem, väärtus on ID või 0;
8. DEPREL: süntaktiline sõltuvusseos ülemaga;
9. DEPS: täiustatud sõltuvusgraaf HEAD-DEPREL paaride listina;
10. MISC: muud väärtused, näiteks nimeüksuse märgend.

Sõnale vastavad infoväljad ei tohi olla tühjad ning ükski väli peale väljade FORM, LEMMA ja MISC ei tohi sisaldada tühikuid. Alakriipsu kasutatakse määramata väljade tähistamiseks, välja arvatud ID välja puhul (Universal Dependencies, 2024).

CoNLL-U-formaadis faile on mugav sisse lugeda EstNLTK (Laur jt, 2020) teegi funktsiooniga conll_importer.conll_to_text. See funktsioon teisendab sisendiks antud faili *Text* objektiks,

[17] https://universaldependencies.org/format.html

millest saab otse lugeda laused, sõnad ja sõnadega seotud nimeüksuse märgendid. EstNLTK on programmeerimiskeele Python teek, milles on erinevaid eesti keele jaoks mõeldud loomuliku keele töötluse tööriistu. Joonisel 1 on näide CoNLL-U formaadis lausest "Meediaõpe toob Lähte ühisgümnaasiumisse õpilasi üle Eesti", kus sõnade "Lähte" ja "Eesti" sõnaliik on PROPN ehk pärisnimi. Sõna "Lähte" tähistab nimeüksuse algust ning selle märgend on "B-Org". Nimeüksuse jätkumist tähistab sõna "ühisgümnaasiumisse", mille märgend on "I-Org". Sõna "Eesti" puhul on kasutatud märgendit "B-Loc", mis viitab kohanime algusele.

```
# sent_id = aja_ml200247_2056
# text = Meediaõpe toob Lähte ühisgümnaasiumisse õpilasi üle Eesti
1 Meediaõpe     meedia_õpe      NOUN  S  Case=Nom|Number=Sing                                          2 nsubj  2:nsubj  Arg=tooma_Arg_0
2 toob          tooma           VERB  V  Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|Voice=Act  0 root   0:root   Verb=tooma_1
3 Lähte         Lähte           PROPN S  Case=Gen|Number=Sing                                          4 nmod   4:nmod   NE=B-Org
4 ühisgümnaasiumisse ühis_gümnaasium NOUN S  Case=Ill|Number=Sing                                      2 obl    2:obl    Arg=tooma_Arg_3|NE=I-Org
5 õpilasi       õpilane         NOUN  S  Case=Par|Number=Plur                                         2 obj    2:obj    Arg=tooma_Arg_1
6 üle           üle             ADP   K  AdpType=Prep                                                 7 case   7:case   _
7 Eesti         Eesti           PROPN S  Case=Gen|Number=Sing                                          5 nmod   5:nmod   NE=B-Loc
```

Joonis 1. Näidis CoNLL-U formaadis lausest

Lisas I välja toodud tabelis on kirjas mõlema andmestiku statistika treening-, arendus- ja testandmestike lõikes. Saadud arvud on enamasti kooskõlas Muischneki jt (2023) töös välja toodud andmetega. Andmetöötluse käigus tulid välja üksikud märgendamisvead, näiteks olid puudu mõned "B-" või "I-" eesliited või oli kogemata kirjutatud "B-" asemel "B.". Erinevused tulenevad sellest, et nimeüksuste loendamisel on vaja arvestada ainult täielikke üksusi, mistõttu on loendatud vaid "B-" eesliitega märgendeid.

Nimeüksuseid on võrreldes sõnade koguarvuga andmestikes vähe. Veebikeele puudepangas on domineerivad isiku- ja tootenimed, ülejäänud klasside esindajaid on vähem. Kirjakeele puudepangas on ülekaalus just isikunimed, mõõdukalt on esindatud ka koha-, organisatsiooni- ja tootenimed ning geopoliitilised üksused. Mõlemas andmestikus on haruldasemad nimeüksused *EVE*, *MUU* ja *UNK*.

2.2 Andmetöötlus

Pärast andmestiku sisselugemist on tarvis andmed eeltöödelda, et viia need BERT-tüüpi mudeli jaoks sobivale kujule. Säästmaks andmete sisselugemiseks kuluvat aega, salvestasin nii treening-, arendus- kui ka testandmestikud JSON-formaadis[18] failidesse. Iga lause on üks objekt, mis sisaldab järjekorranumbrit *id*, märgendite listi *tags* ning sõnade listi *tokens*. JSON

[18] https://www.json.org/json-en.html

ehk JavaScript Object Notation on tekstipõhine failiformaat, mida kasutatakse andmete salvestamiseks ja edastamiseks.

Viimasena on vaja iga lause sõnestada kasutades mudeli sisemist sõnestajat (ingl *tokenizer*). Originaalne BERT mudel kasutas sõnestamisalgoritmina WordPiece algoritmi (Devlin jt, 2019), mida ei ole avalikult kättesaadavaks tehtud. Selle asemel kasutavad EstBERT (Tanvir jt, 2021) ja Est-RoBERTa (Ulčar jt, 2021) sõnestamisalgoritmina BPE algoritmi. Sõnestamise käigus tükeldatakse mõned sõnad vastavalt mudeli sõnastikule alamsõnedeks. Jurafsky jt (2025) kirjutasid, et kuna algselt on iga sõna vastavuses ühe märgendiga, siis tuleb treenimisel ka alamsõnedele märgendid lisada ning ennustamisel valida terve sõna jaoks üks märgend. Lahenduseks pakkusid Jurafsky jt (2025), et treenimisel saab igale alamsõnele määrata sama märgendi, mis oli originaalsel sõnal. Ennustamisel on lihtsaim valida esimese alamsõne tõenäoliseim märgend (vt joonis 2).

Lõplik ennustus	O		O		B-GEP	B-ORG			I-ORG
Kohandatud märgendid	O	O	O	O	B-GEP	B-ORG	B-ORG	B-ORG	I-ORG
BERT sõnestus	Panga	##juht	tude	##erib	Rootsis	Wall	##en	##bergi	instituudis
Algne sõnestus	Pangajuht		tudeerib		Rootsis	Wallenbergi			instituudis
Algsed märgendid	O		O		B-GEP	B-ORG			I-ORG

Joonis 2. BERT sõnestuse ja kohandatud märgendite võrdlus esialgse lausega

Selles näites on näha, kuidas BERT mudeli sõnestajaga sõnestatud lause sõned ei kattu esialgse lause BIO-märgenditega. Näiteks sõna “Wallenbergi” on tükeldatud alamsõnedeks “Wall”, “##en”, “##bergi” ning igale alamsõnele on määratud esimese alamsõne “Wall” märgend “B-ORG”. Ennustamisel valitakse terve sõna jaoks esimese alamsõne märgend ning ülejäänusid ignoreeritakse: selles näites võetakse alamsõne “Wall” ennustatud märgend, mis määratakse sõnale “Wallenbergi”.

2.3 Peenhäälestamine

Eeltreenitud mudelite peenhäälestamiseks kasutasin Google Colaboratory[19] keskkonda, kus saab veebibrauseris kirjutada ja käivitada Python koodi ning kasutada lisatasuta graafikaprotsessoreid. Tavarežiimis on kasutusel kahetuumaline protsessor ja 12,7 gigabaiti muutmälu. Lisaks saab kasutada NVIDIA T4 graafikaprotsessorit, millel on 15 gigabaiti

[19] https://colab.google/

kasutatavat mälu. Täpseid kasutuslimiite ei ole Google avaldanud, aga töö käigus sai T4 graafikaprotsessorit järjest kasutada kuni neli tundi. Alternatiivselt saab kasutada sarnaste võimalustega Kaggle veebikeskkonda[20], Tartu Ülikooli kõrge arvutusjõudlusega Rocket klastrit (University of Tartu, 2018) või lokaalset masinat.

Mudelite allalaadimist ja treenimist võimaldab Hugging Face Transformers[21] teek. Hugging Face[22] on platvorm, kus on vabalt saadaval üle ühe miljoni eeltreenitud või peenhäälestatud mudeli ning üle 300 tuhande andmestiku. Mudelite peenhäälestamisel kasutasin hüperparameetritena ploki suurust (ingl *batch size*) 16, õppimiskiirust (ingl *learning rate*) 5e-5 ning epohhide arvu 3. Ploki suurus näitab, mitu ühikut andmeid korraga mälus hoitakse ning töödeldakse. Õppimiskiirus näitab, kui palju mudeli kaale treenimisel muudetakse.

2.4 Hindamine

Töös on üksusepõhiseks hindamiseks testandmestikel kasutatud programmeerimiskeele Python teeki nervaluate[23]. Järgnevalt on kirjeldatud, kuidas nimeüksuste tuvastajaid hinnatakse.

Mudelite hindamisel ei pruugi sõnapõhine hindamine olla kõige parem variant, sest nimeüksused võivad koosneda mitmest sõnast. Vaja oleks hinnata terve üksuse ennustamise täpsust. Chinchor jt (1993) kirjutasid oma töös kuuest kategooriast, mille abil mudeli ennustusi ja tegelikke väärtuseid võrrelda:

1. Õige (ingl *correct*) ehk mudeli ennustus on sama, mis tegelik väärtus.
2. Vale (ingl *incorrect*) ehk mudeli ennustus ei ole sama, mis tegelik väärtus.
3. Osaline (ingl *partial*) ehk mudeli ennustus on "sarnane" tegeliku väärtusega, aga ei ole samad.
4. Puudu (ingl *missing*) ehk mudel ei ennustanud midagi, kuigi oleks pidanud.
5. Võlts (ingl *spurious*) ehk mudel tegi ennustuse, aga tegelik väärtus puudus.
6. Tühi (ingl *noncommittal*) ehk mudel ei ennustanud midagi ja tegelik väärtus puudus.

Chinchori jt (1993) tööd kasutasid Segura-Bedmar jt (2013), kes kirjeldasid oma töös nelja hindamisskeemi:

[20] https://www.kaggle.com/code
[21] https://huggingface.co/docs/transformers/en/index
[22] https://huggingface.co/
[23] https://pypi.org/project/nervaluate/

1. Range (ingl *strict*), mis arvestab nii üksuse tüübi kui ka täpsete piiridega.
2. Täpne (ingl *exact*), mis arvestab ainult üksuse täpsete piiridega.
3. Osaline (ingl *partial*), mis arvestab vaid sellega, et üksuse piirid oleksid osaliselt õiged.
4. Tüüp (ingl *type*), mis arvestab ainult tüübiga, aga ennustus ja tegelik üksus peavad mingil määral kattuma.

Need hindamisskeemid võimaldavad erinevatel juhtudel paindlikumat analüüsi, sõltuvalt sellest, kui rangelt soovitakse mudeli täpsust mõõta. Range skeem võib mudeli hindamisel olla kõige kasulikum, sest annab hea ülevaate mudeli võimest ennustada nii õiget nimeüksuse klassi kui ka üksuse piire. Tabelis 2 on ülevaade sellest, kuidas erinevad hindamisskeemid ennustusi ja tegelikke märgendeid võrdlevad.

Tabel 2. Hindamisskeemide võrdlus ühel lausel

Sõna	**Tegelik märgend**	**Ennustatud märgend**	**Range**	**Täpne**	**Osaline**	**Tüüp**
Mari	B-Per	B-Per	Õige	Õige	Õige	Õige
ja	O	O	Tühi	Tühi	Tühi	Tühi
Arno	B-Per	O	Puudu	Puudu	Puudu	Puudu
õpivad	O	O	Tühi	Tühi	Tühi	Tühi
Tartu	B-Org	B-Org	Vale	Vale	Osaline	Õige
Ülikoolis	I-Org	O	Tühi	Tühi	Tühi	Tühi
.	O	B-Loc	Võlts	Võlts	Võlts	Võlts

Jurafsky jt (2025) sõnul kasutatakse nimeüksuste tuvastamiseks mõeldud mudelite hindamiseks saagist (ingl *recall*), täpsust (ingl *precision*) ja F-skoori (ingl *F-score* või *F-measure*), mis on saagise ja täpsuse harmooniline keskmine. Saagis näitab, kui suur osa kõigist tuvastamist vajavatest üksustest on õigesti tuvastatud, samas täpsus näitab, kui suur osa tuvastatud üksustest on õiged. Tabelis 2 toodud andmete põhjal täpsuse ja saagise arvutamiseks on enne vaja teada kahte väärtust: mudeli ennustatud märgendite arvu ehk tõsi- ja valepositiivsete summat (1) ning tegelike märgendite arvu ehk tõsipositiivsete ja valenegatiivsete summat (2). Kõik järgnevad valemid on võetud programmeerimiskeele Python teegi nervaluate dokumentatsioonist (Batista jt, 2024). Valemites kasutatud väärtused *Õige*, *Vale*, *Osaline*, *Võlts*, *Puudu* tähistavad vastavate väärtuste koguarvu.

$$Tegelik = Õige + Vale + Osaline + Võlts = TP + FP \quad (1)$$

$$Võimalik = Õige + Vale + Osaline + Puudu = TP + FN \quad (2)$$

Täpse ja range hindamisskeemi jaoks arvutatakse täpsus (3) ja saagis (4) järgnevalt.

$$Täpsus = \frac{Õige}{Tegelik} = \frac{TP}{TP+FP} \quad (3)$$

$$Saagis = \frac{Õige}{Võimalik} = \frac{TP}{TP+FN} \quad (4)$$

Osalist kattuvust kasutavate hindamisskeemide (osaline ja tüüp) jaoks arvutatakse täpsus (5) ja saagis (6) nii, et tõsipositiivsetena arvestatakse ka osaliselt õiged ennustused.

$$Täpsus = \frac{Õige+0{,}5*Osaline}{Tegelik} = \frac{TP}{TP+FP} \quad (5)$$

$$Saagis = \frac{Õige+0{,}5*Osaline}{Võimalik} = \frac{TP}{TP+FN} \quad (6)$$

F-skoor arvutatakse üksusepõhisel hindamisel samal viisil nagu sõnapõhisel hindamisel, täpsuse ja saagise harmoonilise keskmisena (7).

$$F - skoor = 2 * \frac{Täpsus*Saagis}{Täpsus+Saagis} \quad (7)$$

Tabelis 3 on kirjas näite põhjal (vt tabel 2) saadud tulemused iga hindamisskeemi jaoks.

Tabel 3. Üksusepõhised hinnangud näidislausel kasutades erinevaid hindamisskeeme

	Range	**Täpne**	**Osaline**	**Tüüp**
Õige	1	1	1	2
Vale	1	1	0	0
Osaline	0	0	1	0
Puudu	1	1	1	1
Võlts	1	1	1	1
Täpsus	0,33	0,33	0,5	0,66
Saagis	0,33	0,33	0,5	0,66
F-skoor	0,33	0,33	0,5	0,66

Töös on välja toodud ainult range hindamisskeemi põhjal saadud mudelite ennustustulemused, sest oluline on hinnata nii õige kategooria kui ka üksuse alguse ja lõpu ennustamise võimekust.

3. Tulemused

Töö eesmärgi täitmiseks peenhäälestasin EstBERT ja Est-RoBERTa alusmudelid nii kirja- ja veebikeele puudepankadel eraldi kui ka kasutades ühist treeningandmestikku. Ühisel andmestikul treenitud mudeleid testisin ka eraldi testandmestikel, et võrrelda tulemusi väiksematel treeningandmestikel treenitud mudelitega. Viimaks testisin ühisel treeningandmestikul peenhäälestatud mudeleid teistel eesti keele andmestikel, millele on lisatud nimeüksuste märgendid ning teisi eestikeelsete tekstide jaoks mõeldud nimeüksuste märgendamise mudeleid eesti keele puudepankadel.

Töö tulemuste ja koodiga saab tutvuda GitHubi lehel[24]. Ühisel eesti keele puudepanga treeningandmestikul peenhäälestatud Est-RoBERTa mudel on saadaval Hugging Face lehel[25].

3.1 Eraldi andmestikel peenhäälestatud mudelid

Selles alapeatükis on välja toodud eraldi andmestikel peenhäälestatud EstBERT ja Est-RoBERTa mudelite tulemused vastavatel testandmestikel. Joonised tabelite 4 ja 5 kohta on lisades II ja III.

Tabelis 4 on kirjas kirjakeele puudepangal peenhäälestatud keelemudelite ennustustulemused vastaval testandmestikul. Est-RoBERTa baasil peenhäälestatud mudel saavutas igas kategoorias kõrgemad F-skoorid kui EstBERT. Suurimad erinevused on sündmuste ja toodete kategooriates, kus Est-RoBERTa F-skoorid on vastavalt 18,7 ja 7,5 protsendipunkti võrra kõrgemad. Andmete vähesuse tõttu ei suuda mudelid nimeüksuseid, mille märgend on *MUU* või *UNK* õigesti tuvastada. Põhjalikum näide kategooriate *MUU* ja *UNK* kohta on kirjas selle alapeatüki lõpus.

Tabel 4. EDT treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud EDT testandmestikul.

	EstBERT			**Est-RoBERTa**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,419	0,342	0,377	0,606	0,526	**0,563**
GEP	0,777	0,777	0,777	0,791	0,791	**0,791**
LOC	0,679	0,666	0,672	0,703	0,686	**0,694**
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,774	0,789	0,782	0,813	0,811	**0,812**
PER	0,942	0,938	0,940	0,965	0,964	**0,965**
PROD	0,514	0,518	0,516	0,590	0,592	**0,591**
UNK	0,000	0,000	0,000	0,000	0,000	0,000

[24] https://github.com/martinkivisikk/ner_thesis
[25] https://huggingface.co/vbius01/est-roberta-ud-ner

Üldine	0,801	0,799	0,800	0,835	0,831	**0,833**

Veebikeele puudepankadel peenhäälestatud mudelid saavutasid testandmestikul üldiselt madalamad tulemused (vt tabel 5), kui kirjakeele puudepangal peenhäälestatud ja testitud mudelid. Veebikeele puudepank on mahult väiksem ning sisaldab vabama keelekasutusega lauseid, seega mudelitel võib olla keerulisem nende andmete põhjal õppida. EstBERT mudel tuvastas veebikeele puudepanga testandmestikul kohanimesid paremini, kui Est-RoBERTa, aga Est-RoBERTa üldine F-skoor oli siiski 5,3 protsendipunkti võrra kõrgem.

Tabel 5. EWT treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud EWT testandmestikul.

	EstBERT			**Est-RoBERTa**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,136	0,136	0,136	0,150	0,136	**0,143**
GEP	0,617	0,707	0,659	0,705	0,756	**0,729**
LOC	0,593	0,571	**0,582**	0,524	0,393	0,449
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,438	0,424	0,431	0,493	0,402	**0,443**
PER	0,830	0,805	0,817	0,869	0,879	**0,874**
PROD	0,406	0,387	0,396	0,467	0,528	**0,496**
UNK	0,000	0,000	0,000	0,000	0,000	0,000
Üldine	0,675	0,660	0,667	0,722	0,718	**0,720**

Nimeüksuste klasside *MUU* ja *UNK* puhul on hindamistulemused nullid, sest andmestikes on nende kohta näiteid niivõrd vähe. Veebikeele puudepanga testandmestikus ei ole ühtegi nimeüksust, mille kategooria oleks *MUU* või *UNK*. Kirjakeele puudepanga testandmestikus on kaks nimeüksust, mille kategooria on *MUU* ning üks nimeüksus, mille kategooria on *UNK*, aga kumbki mudel ei suutnud neid õigesti tuvastada. Mõlemad mudelid ennustasid sõna "Goldbergi-haiguse" märgendiks "B-PER", mis tähistab isikunime algust, aga tegelik märgend oli "B-MUU". Selles näites võis viga tekkida sellest, et sõna algus "Goldberg" viitabki tegelikult isikunimele ning ennustamisel valitakse esimese alamsõne tõenäoliseim märgend.

3.2 Ühisel andmestikul peenhäälestatud mudelid

Selles alapeatükis on välja toodud ühisel treeningandmestikul peenhäälestatud mudelite tulemused nii ühisel testandmestikul kui ka kirja- ja veebikeele puudepankade testandmestikel eraldi. Joonised tabelite 6, 7 ja 8 kohta on vastavalt lisades IV, V ja VI.

Kasutades nii ühist treening- kui ka testandmestikku, on Est-RoBERTa jätkuvalt parem, üldine F-skoor testandmestikul on võrreldes EstBERT mudeliga 3,9 protsendipunkti võrra kõrgem (vt tabel 6).

Tabel 6. Ühisel treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud ühisel testandmestikul.

	EstBERT			**Est-RoBERTa**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,571	0,533	0,552	0,731	0,633	**0,679**
GEP	0,788	0,804	0,796	0,838	0,848	**0,843**
LOC	0,679	0,667	0,673	0,696	0,676	**0,686**
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,706	0,713	0,709	0,756	0,733	**0,744**
PER	0,911	0,913	0,912	0,949	0,958	**0,953**
PROD	0,551	0,508	0,529	0,581	0,542	**0,561**
UNK	0,000	0,000	0,000	0,000	0,000	0,000
Üldine	0,792	0,785	0,789	0,833	0,823	**0,828**

Kasutades ühist treeningandmestikku jäid üldised F-skoorid kirjakeele puudepanga testandmestikul samaks või paranesid. EstBERT mudeli puhul jäi F-skoor samaks, Est-RoBERTa F-skoor oli 0,4 protsendipunkti võrra kõrgem (vt tabel 7). Võrreldes tulemusi ainult EDT andmestikul peenhäälestatud ja testitud mudeliga (vt tabel 4), tuvastas ühisel treeningandmestikul peenhäälestatud EstBERT mudel paremini *EVE*, *GEP*, *LOC*, *PROD* ja Est-RoBERTa *GEP*, *PER* klassidesse kuuluvaid nimeüksuseid. Seega veebikeele puudepanga treeningandmed kirjakeele tekstides nimeüksuste tuvastamisel suurt eelist ei anna.

Tabel 7. Ühisel treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud EDT testandmestikul.

	EstBERT			**Est-RoBERTa**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,486	0,447	0,466	0,606	0,526	**0,563**
GEP	0,777	0,797	0,787	0,831	0,850	**0,841**
LOC	0,688	0,676	0,682	0,696	0,686	**0,691**
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,741	0,759	0,750	0,805	0,805	**0,805**
PER	0,935	0,938	0,937	0,967	0,971	**0,969**
PROD	0,535	0,540	0,538	0,567	0,576	**0,571**
UNK	0,000	0,000	0,000	0,000	0,000	0,000
Üldine	0,797	0,803	0,800	0,836	0,838	**0,837**

Ühise treeningandmestiku kasutamine aitas märgatavalt parandada ennustustulemusi veebikeele puudepanga testandmestikul (vt tabel 8). Võrreldes ainult veebikeele puudepanga treeningandmestikul peenhäälestatud mudelitega (vt tabel 5), tõusis üldine F-skoor EstBERT puhul 8,3 ja Est-RoBERTa puhul 7,6 protsendipunkti võrra.

Tabel 8. Ühisel treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud EWT testandmestikul.

	EstBERT			**Est-RoBERTa**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,714	0,682	0,698	0,947	0,818	**0,878**
GEP	0,875	0,854	**0,864**	0,895	0,829	0,861
LOC	0,593	0,571	0,582	0,696	0,571	**0,627**
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,557	0,533	**0,544**	0,526	0,446	0,482
PER	0,849	0,847	0,848	0,902	0,923	**0,913**
PROD	0,620	0,415	0,497	0,644	0,443	**0,525**
UNK	0,000	0,000	0,000	0,000	0,000	0,000
Üldine	0,774	0,728	0,750	0,824	0,769	**0,796**

Eeltoodud tulemused kinnitavad, et Est-RoBERTa alusmudel on nimeüksuste tuvastamise ülesandel puudepanga korpusel peenhäälestamiseks parem kui EstBERT. Lisaks on ühise treeningandmestiku kasutamine õigustatud, sest tulemused veebikeele puudepanga testandmestikul tõusid märgatavalt ning kirjakeele puudepanga testandmestikul jäid tulemused samaks või tõusid vähesel määral. Tulemusi võib veelgi parandada täiendav katsetamine hüperparameetrite valikul ning andmestike parandamine[26].

3.3 Mudelite testimine teistel eesti andmestikel

Selles alapeatükis on välja toodud ühisel puudepankade treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa mudelite ennustustulemused EstNER_new ja EstNER testandmestikel. Joonised tabelite 9 ja 10 kohta on lisades VII ja VIII.

Võrreldes ühisel puudepankade testandmestikul saadud tulemustega (vt tabel 6), langes EstNER_new testandmestikul üldine F-skoor EstBERT puhul 8,6 ja Est-RoBERTa puhul 6,8 protsendipunkti (vt tabel 9). Sarnast langust on märgata ka EstNER testandmestikul (vt tabel 10). Tulemused on madalamad, sest tegemist on väliste andmestikega, mille puhul märgendusjuhised ja andmete jaotus on erinevad. Puudepankade andmestik sisaldab nii ajakirjandus-, ilukirjandus-, teadus- kui ka sotsiaalmeediatekste. EstNER andmestik sisaldab ainult ajakirjandustekste ning EstNER_new andmestik koosneb ajakirjandus- ja sotsiaalmeediatekstidest. Seega puudepankade andmestik on tekstiliikide mõttes mitmekesisem.

[26] Töö algusfaasis jäid silma ja said parandatud mõned märgendusvead, aga neid võib veel esineda.

Tabel 9. Ühisel treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud EstNER_new testandmestikul

	EstBERT			Est-RoBERTa		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,500	0,231	**0,316**	0,385	0,192	0,256
GEP	0,606	0,593	0,600	0,648	0,645	**0,646**
LOC	0,676	0,657	**0,667**	0,633	0,543	0,585
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,679	0,593	0,633	0,773	0,660	**0,712**
PER	0,896	0,896	0,896	0,958	0,980	**0,969**
PROD	0,589	0,534	0,560	0,667	0,593	**0,628**
UNK	0,000	0,000	0,000	0,000	0,000	0,000
Üldine	0,724	0,682	0,703	0,782	0,739	**0,760**

EstNER_new testandmestikul oli mudelitel enim raskusi sündmuste tuvastamisel. Seevastu EstNER testandmestikul on sündmuseid tuvastatud paremini, aga tootenimede *PROD* tuvastamine on mudelite jaoks keerulisem (vt tabel 10).

Tabel 10. Ühisel treeningandmestikul peenhäälestatud EstBERT ja Est-RoBERTa ennustuste hinnangud EstNER testandmestikul

	EstBERT			Est-RoBERTa		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,562	0,529	0,545	0,600	0,706	**0,649**
GEP	0,601	0,664	0,631	0,636	0,699	**0,666**
LOC	0,627	0,770	0,691	0,645	0,803	**0,715**
MUU	0,000	0,000	0,000	0,000	0,000	0,000
ORG	0,644	0,512	0,571	0,675	0,542	**0,601**
PER	0,888	0,956	0,921	0,916	0,976	**0,945**
PROD	0,243	0,273	0,257	0,358	0,364	**0,361**
UNK	0,000	0,000	0,000	0,000	0,000	0,000
Üldine	0,715	0,721	0,718	0,749	0,752	**0,750**

Mõlemal välisel testandmestikul saavutas ühisel puudepankade treeningandmestikul peenhäälestatud Est-RoBERTa mudel üldiselt kõrgema F-skoori kui EstBERT.

Siinses töös saavutas Est-RoBERTa igal hindamisel kõrgema üldise F-skoori kui EstBERT. Est-RoBERTa mudeli eeltreenimisel (Ulčar jt, 2021) kasutati efektiivsemat treenimisprotsessi ning suuremat andmestikku kui EstBERTi puhul, mistõttu on Est-RoBERTa nimeüksuste tuvastamise ülesandel võimekam.

3.4 Teiste mudelite testimine puudepankadel

Selles alapeatükis on välja toodud EstBERT_NER ja EstBERT_NER_V2 mudelite ennustustulemused nii kirja- kui ka veebikeele puudepankade testandmestikel. Joonised tabelite 11 ja 12 kohta on lisades IX ja X.

Tanviri jt (2021) töö käigus loodud nimeüksuste tuvastamise mudel EstBERT_NER tuvastab ainult nimeüksuseid klassidest *LOC*, *ORG* ja *PER*, seega selle mudeli testimisel on testandmestikes kõik muud märgendid asendatud O-märgendiga.

EstBERT_NER saavutas kirjakeele puudepanga testandmestikul F-skoori 0,756 ning veebikeele puudepanga testandmestikul F-skoori 0,592 (vt tabel 11). Mudeli täpsus asukohanimede *LOC* tuvastamisel puudepankade testandmestikel on madal, sest algselt on riigid ja mõnel juhul linnad märgendatud andmestikes geopoliitiliste üksustena *GEP*, mis on asendatud O-märgendiga. Teine variant selle mudeli testimisel puudepankade testandmestikel oleks asendada kõik *GEP* märgendid *LOC* märgenditega.

Tabel 11. EstBERT_NER ennustuste hinnangud mõlemal testandmestikul

	EDT			**EWT**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
LOC	0,384	0,771	0,512	0,230	0,500	0,315
ORG	0,611	0,745	0,672	0,510	0,543	0,526
PER	0,875	0,889	0,882	0,852	0,520	0,646
Üldine	0,687	0,840	0,756	0,682	0,523	0,592

Samuti võib tähele panna, et EstBERT_NER mudeli saagis isikunimede tuvastamisel veebikeele puudepanga testandmestikul on madal. Veebikeele puudepanga tekstides esineb kasutajanimesid (tihti väikese algustähega), mis on märgendatud isikunimedena. EstBERT_NER mudeli treeningandmestikus isikunimesid sellisel kujul ei esinenud, seega mudelil on keeruline kasutajanimesid veebikeele puudepanga testandmestikus tuvastada.

EstBERT_NER_V2 mudeli puhul on näha, et kõige raskem on mudelil tuvastada sündmuseid, asukoha- ja tootenimesid (vt tabel 12). Mudelil on raske eristada sõnasid, mis võivad olla kas geopoliitiline üksus *GEP* või asukoht *LOC* ning on kaldu *GEP* klassi poole. Kirjakeele puudepanga testandmestikul ennustas mudel 165 korda nimeüksuse märgendiks *GEP*, kui tegelik klass oli *LOC*. Veebikeele puudepanga testandmestikul tegi mudel seda viga 12 korda. Kuigi Muischneki jt (2023, lk 181) töös kirjutati, et riiginimed märgendatakse puudepankades alati märgendiga *GEP*, leidub kirjakeele puudepangas siiski näiteid, kus riiginimedel on märgend *LOC*.

Tabel 12. EstBERT_NER_V2 ennustuste hinnangud mõlemal testandmestikul

	EDT			**EWT**		
	Täpsus	**Saagis**	**F-skoor**	**Täpsus**	**Saagis**	**F-skoor**
EVE	0,250	0,184	0,212	0,296	0,364	0,327
GEP	0,757	0,807	0,781	0,818	0,878	0,847
LOC	0,245	0,222	0,233	0,200	0,143	0,167
ORG	0,495	0,564	0,528	0,463	0,413	0,437

PER	0,866	0,843	0,854	0,864	0,705	0,777
PROD	0,232	0,197	0,213	0,308	0,226	0,261
Üldine	0,640	0,629	0,634	0,687	0,575	0,626

Sündmuste ja toodete klassi kuuluvate üksuste tuvastamist võivad samuti raskendada klassidevahelised kattuvused ning mudeli peenhäälestamiseks kasutatud andmestike ja puudepankade märgendamiserinevused. Näiteks võib viga tekkida sellest, kui ürituse pealkiri on märgendatud tootena *PROD*, aga mudel ennustab üksuse märgendiks sündmuse *EVE*.

Teiste nimeüksuste tuvastajate (EstBERT_NER ja EstBERT_NER_V2) testimisel kirja- ja veebikeele puudepankade testandmestikel selgus, et mudelite tulemused sõltuvad nii andmetest kui ka klasside arvust. Mõlemad mudelid saavutasid EWT testandmestikul madalamad F-skoorid kui EDT testandmestikul, seega veebikeele tekstidest on mudelitel keerulisem nimeüksuseid tuvastada. EstBERT_NER mudeli F-skoor EDT testandmestikul oli 0,756, aga EstBERT_NER_V2 saavutas samal andmestikul F-skoori 0,634. Võrreldes EstBERT_NER mudeliga, tuvastab EstBERT_NER_V2 mudel ka nimeüksuseid *EVE*, *GEP* ja *PROD*[27]. Sündmuseid *EVE* ja tooteid *PROD* oli EstBERT_NER_V2 mudelil puudepankade testandmestikel keeruline tuvastada, sest mudeli peenhäälestamiseks kasutatud treeningandmestiku ja puudepankade märgendamisjuhised on erinevad. Lisaks oli EstBERT_NER_V2 mudelil raskusi geopoliitiliste üksuste *GEP* ja asukohtade *LOC* ennustamisega. Mudel oli kaldu *GEP* klassi poole, mistõttu asukohtade klassi F-skoor mõlemal testandmestikul oli madal (EDT 0,233 ja EWT 0,167).

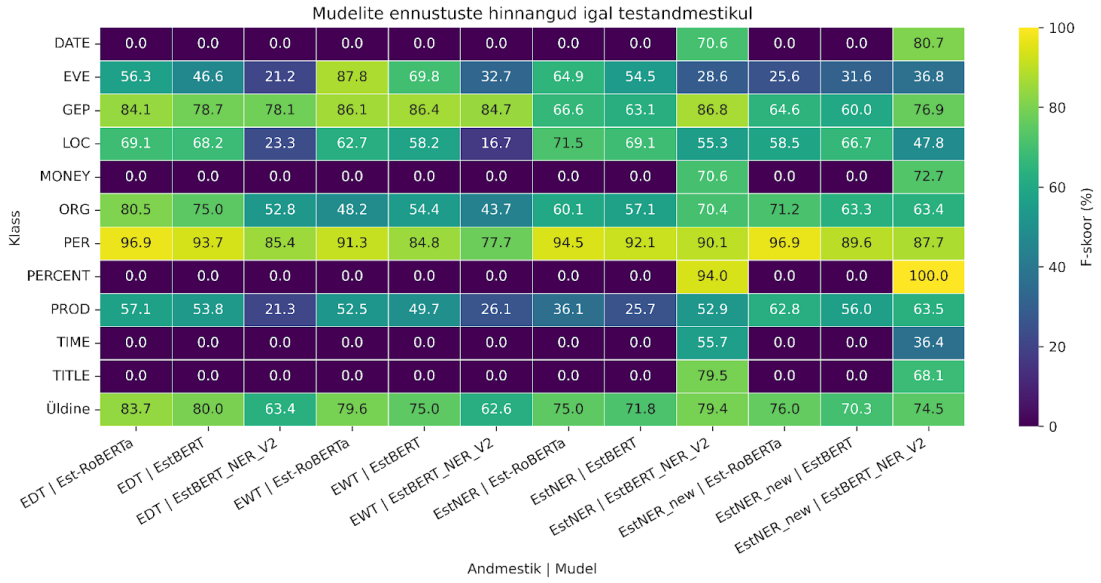
3.5 Koondtulemus

Selles alapeatükis on välja toodud ühisel puudepanga treeningandmestikul peenhäälestatud Est-RoBERTa ja EstBERT mudelite ning Sirtsi (2023) töös avalikustatud mudeli EstBERT_NER_V2 ennustuste hinnangud igal testandmestikul eraldi. Tanviri jt (2021) töös loodud mudel, EstBERT_NER, on jooniselt välja jäetud, sest see mudel peenhäälestati vanal EstNER andmestikul, milles olid nimeüksustena märgendatud ainult kohanimed *LOC*, organisatsioonid *ORG* ja isikunimed *PER*. Sirtsi (2023) töös toodi välja parima mudeli tulemused ainult ühisel testandmestikul (EstNER ja EstNER_new) ning eraldiseisvatel valideerimisandmestikel. Lisaks kasutati Sirtsi (2023) töös mudelite hindamisel seqeval[28]

[27] EstBERT_NER_V2 mudel suudab tuvastada ka üksuseid klassidest *TITLE*, *DATE*, *TIME*, *MONEY* ja *PERCENT*, aga eesti keele puudepankades neid märgendatud ei ole, seega siinkohal on neid klasse eiratud.
[28] https://github.com/chakki-works/seqeval

teeki. Ühtlasema ülevaate saamiseks testisin EstBERT_NER_V2 mudelit ka EstNER ja EstNER_new testandmestikel eraldi ja hindamisel kasutasin nervaluate teeki.

Joonisel 3 välja toodud tulemuste põhjal osutus paremaks mudeliks ühisel puudepankade treeningandmestikul peenhäälestatud Est-RoBERTa mudel. Eelnimetatud mudel saavutas parima üldise F-skoori nii kirja- ja veebikeele puudepankade kui ka EstNER_new testandmestikul. EstBERT_NER_V2 mudeli puhul langes üldine F-skoor puudepankade testandmestikel alla 0,64[29], aga siinses töös peenhäälestatud Est-RoBERTa mudeli üldine F-skoor EstNER ja EstNER_new testandmestikel jäi EstBERT_NER_V2 mudeliga võrreldes viie protsendipunkti piiresse[30].



Joonis 3. Mudelite ennustuste hinnangud igal testandmestikul

Kirjakeele puudepanga testandmestikul tuvastas kõiki nimeüksuseid paremini Est-RoBERTa mudel. Veebikeele puudepanga testandmestikul tuvastas EstBERT mudel paremini geopoliitilisi üksusi *GEP* ja organisatsioone *ORG*, aga Est-RoBERTa oli parem igas ülejäänud kategoorias.

EstNER testandmestikul tuvastas Est-RoBERTa mudel paremini sündmuseid *EVE*, asukohti *LOC* ning isikunimesid *PER*, EstBERT_NER_V2 oli parem igas ülejäänud kategoorias. EstNER_new testandmestikul tuvastas EstBERT paremini asukohti *LOC*, Est-RoBERTa

[29] Kuni 20,3 protsendipunkti võrra madalam tulemus võrreldes Est-RoBERTa mudeliga.
[30] Est-RoBERTa üldine F-skoor oli EstNER testandmestikul 4,4 protsendipunkti võrra madalam ja EstNER_new testandmestikul 1,5 protsendipunkti võrra kõrgem kui EstBERT_NER_V2 mudelil.

tuvastas paremini organisatsioone *ORG* ning isikunimesid *PER*, EstBERT_NER_V2 oli parem igas ülejäänud kategoorias.

EstBERT_NER_V2 mudeli eelis siinses töös peenhäälestatud mudelite ees on see, et suudab tuvastada rohkematest klassidest nimeüksuseid. Ühisel puudepanga treeningandmestikul peenhäälestatud Est-RoBERTa mudel on aga täpsem ning saab paremini hakkama võõraste tekstidega.

Kokkuvõte

Töös peenhäälestati EstBERTil ja Est-RoBERTal põhinevad mudelid eesti kirja- ja veebikeele puudepankadel nii eraldi kui ka ühisel treeningandmestikul. Parimaks mudeliks osutus ühisel treeningandmestikul peenhäälestatud Est-RoBERTa mudel[31], mis saavutas testandmestikul F-skoori 0,828. Ühise treeningandmestiku kasutamine aitas mudelitel paremini nimeüksuseid tuvastada veebikeele puudepanga testandmestikus.

Töö käigus testiti ühendatud puudepankadel peenhäälestatud mudeleid kahel teisel eesti keele nimeüksustega märgendatud andmestikul. Samuti testiti ka kahte varasemalt loodud eestikeelsete tekstide jaoks mõeldud nimeüksuste tuvastajat eesti keele puudepankadel. Selgus, et välistel andmestikel on mudelitel keerulisem nimeüksuseid tuvastada, sest erinevate andmestike puhul ei ole tagatud, et nimeüksused on sarnaselt defineeritud ning tekstis märgendatud.

Edasiste arendustena võib katsetada, kuidas mõjutavad erinevad hüperparameetrid, näiteks õppimiskiirus, ning täiendav andmestiku puhastamine mudelite tulemusi.

[31] https://huggingface.co/vbius01/est-roberta-ud-ner

Viidatud kirjandus

Batista, D. S., Upson, M. Nervaluate. 2024. https://pypi.org/project/nervaluate/ (25.04.2025).

Chinchor, N., Sundheim, B. M. MUC-5 evaluation metrics. *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*. 1993, pp. 69-78. https://aclanthology.org/M93-1007.pdf

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. 2019, pp. 4171-4186. https://aclanthology.org/N19-1423.pdf

Ehrmann, M., Hamdi, A., Pontes, E. L., Romanello, M., Doucet, A. Named Entity Recognition and Classification on Historical Documents: A Survey. 2021. https://doi.org/10.48550/arXiv.2109.11406

Jurafsky, D., Martin, J. H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models. 3rd edition. 2025, pp. 223-241. https://web.stanford.edu/~jurafsky/slp3/ (03.04.2025).

Kallas, J., Koppel, K. Eesti keele ühendkorpus 2017. 2018. https://doi.org/10.15155/3-00-0000-0000-0000-071E7L (03.04.2025).

Laur, S., Orasmaa, S., Särg, D., Tammo, P. EstNLTK 1.6: Remastered Estonian NLP Pipeline. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*. 2020, pp. 7152-7160. https://aclanthology.org/2020.lrec-1.884/

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. 2019. https://doi.org/10.48550/arXiv.1907.11692

Muischnek, K., Müürisep, K. Named Entity layer in Estonian UD treebanks. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 179-184. https://aclanthology.org/2023.nodalida-1.19

Orasmaa, S., Muischnek, K., Poska, K., Edela, A. Named Entity Recognition in Estonian 19th Century Parish Court Records. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*. 2022, pp. 5304-5313. https://aclanthology.org/2022.lrec-1.568

Ramshaw, L., Marcus, M. Text Chunking using Transformation-Based Learning. In *Third Workshop on Very Large Corpora*. 1995. https://aclanthology.org/W95-0107/

Segura-Bedmar, I., Martínez, P., Herrero-Zazo, M. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. 2013, pp. 341–350. https://aclanthology.org/S13-2056.pdf

Sirts, K. Estonian Named Entity Recognition: New Datasets and Models. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2023, pp. 752-761. https://aclanthology.org/2023.nodalida-1.76

Sügis, E., Tampuu, A., Aljanaki, A., Fišel, M., Kull, M. Praktiline andmeteadus. Tartu: Tartu Ülikooli arvutiteaduse instituut. 2024. https://courses.cs.ut.ee/t/andmeteadus/Main/HomePage

Tanvir, H., Kittask, C., Eiche, S., Sirts, K. EstBERT: A Pretrained Language-Specific BERT for Estonian. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*. 2021, pp. 11-19. https://aclanthology.org/2021.nodalida-main.2

Tkachenko, A., Petmanson, T., Laur, S. Named entity recognition in estonian. In *Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing*, 2013, pp. 78-83. https://aclanthology.org/W13-2412.pdf

Ulčar, M., Robnik-Šikonja, M. Training dataset and dictionary sizes matter in BERT models: The case of Baltic languages. 2021. https://doi.org/10.48550/arXiv.2112.10553

Universal Dependencies. CoNLL-U Format. 2024. https://universaldependencies.org/format.html (03.04.2025).

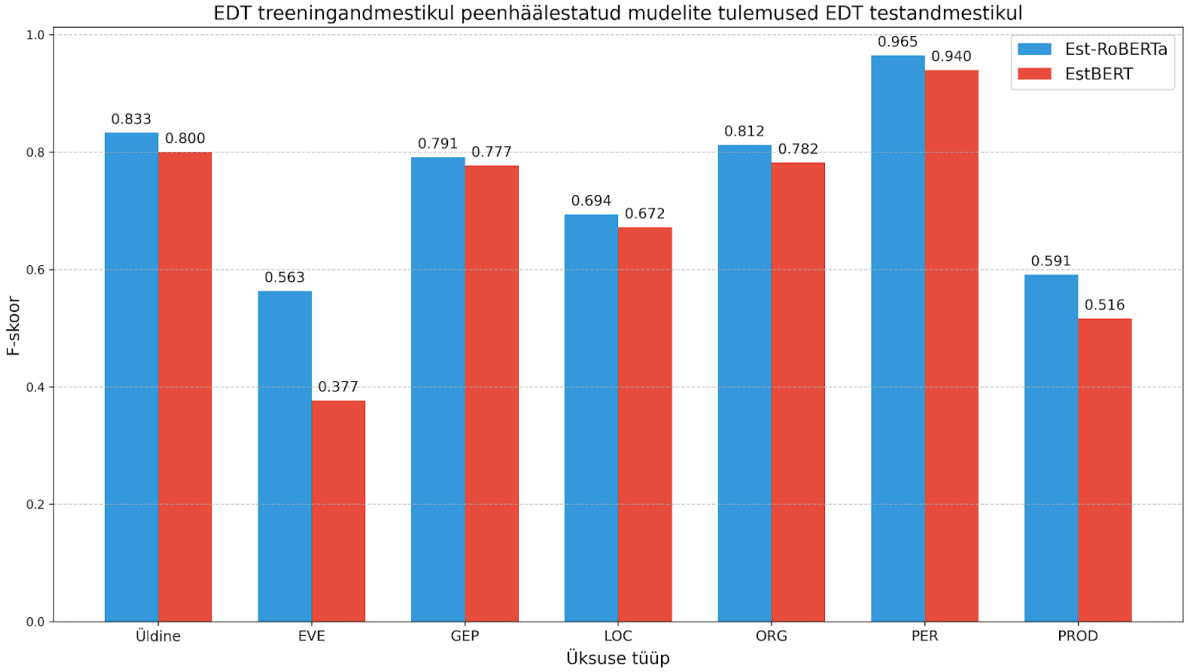
University of Tartu. UT Rocket. share.neic.no. 2018. https://doi.org/10.23673/PH6N-0144

Lisad

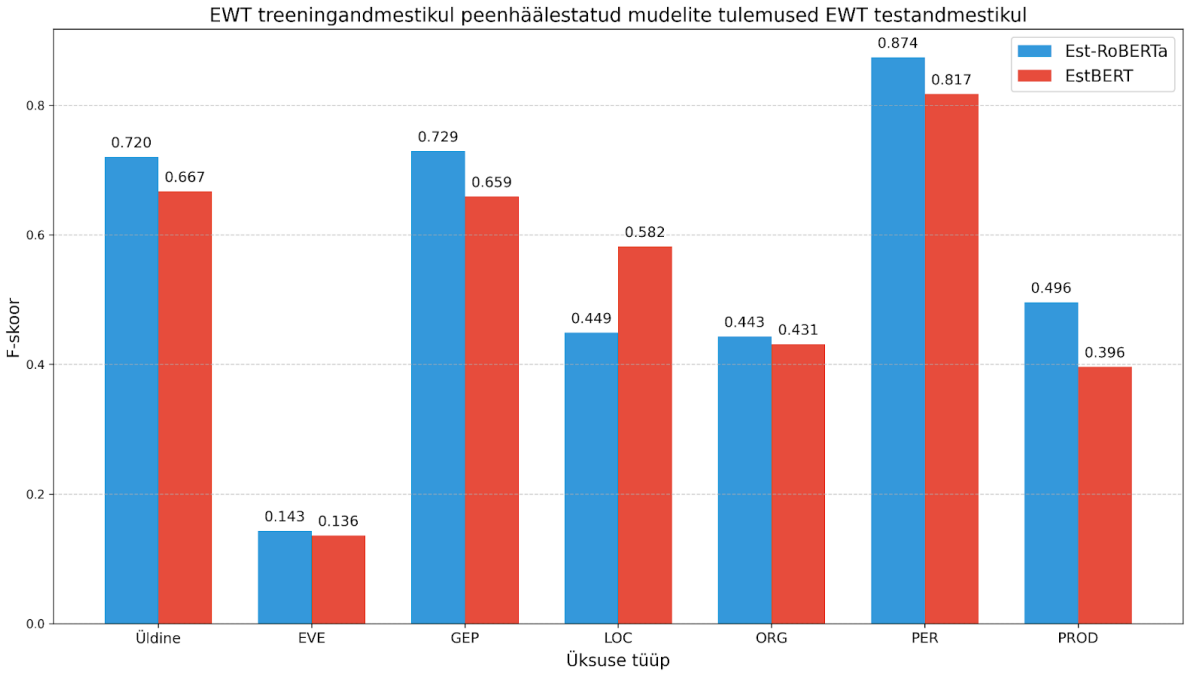
I. Kirja- ja veebikeele puudepankade (EDT ja EWT) statistika

	EDT				EWT			
	Treening	**Arendus**	**Test**	**Kokku**	**Treening**	**Arendus**	**Test**	**Kokku**
Lauseid	24601	3122	3207	30930	5444	833	913	7190
Sõnu	344581	44742	48465	437788	67431	10001	13152	90584
PER	6493	837	1116	8446	1226	235	431	1892
LOC	2639	377	290	3306	235	5	28	268
GEP	3447	295	300	4042	264	13	41	318
ORG	2433	215	365	3013	206	28	92	326
PROD	1534	245	308	2087	558	157	106	821
EVE	310	38	38	386	29	0	22	51
MUU	17	6	2	25	9	0	0	9
UNK	21	15	1	37	4	1	0	5

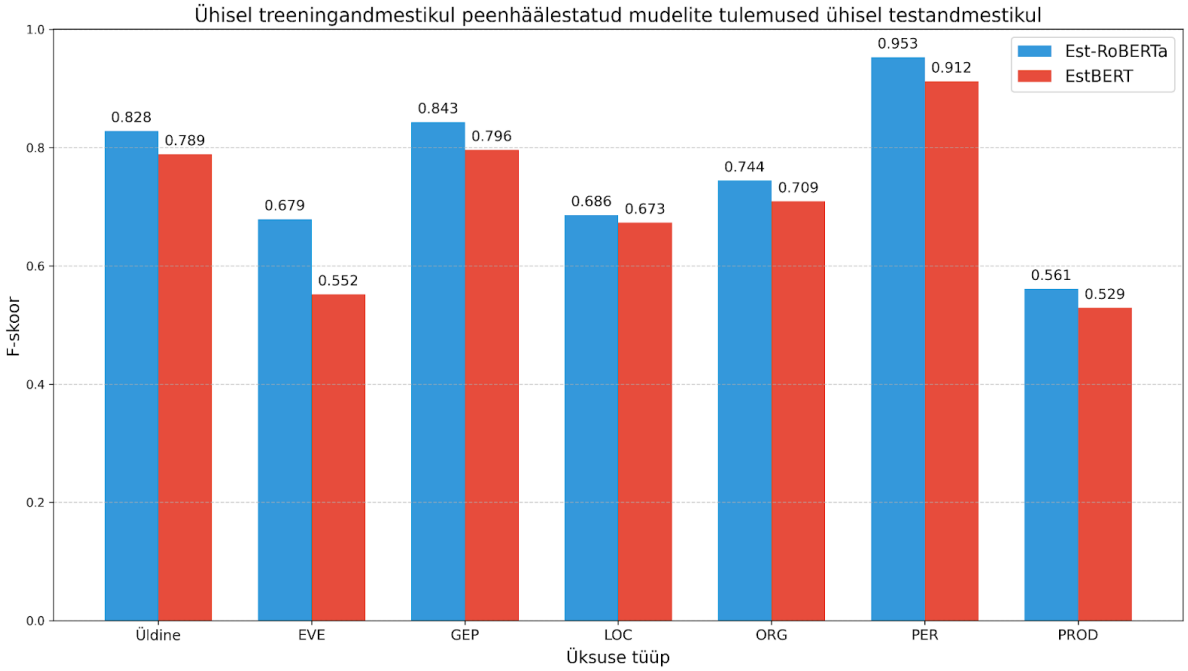
II. EDT mudelite tulemused EDT testandmestikul



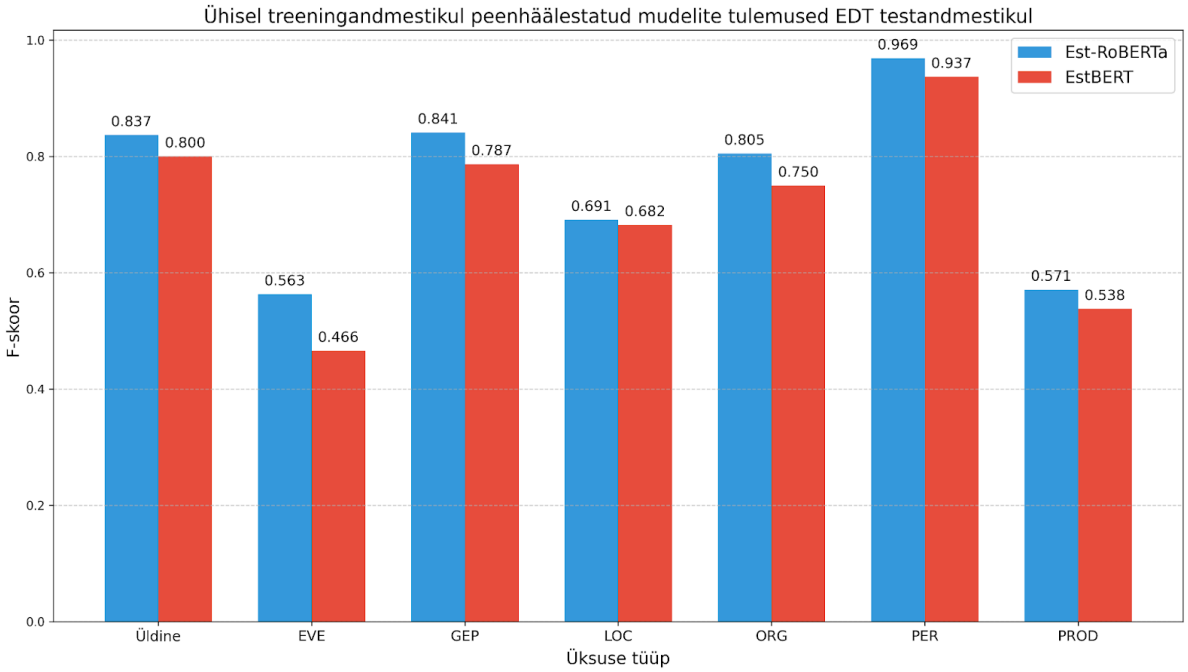
III. EWT mudelite tulemused EWT testandmestikul



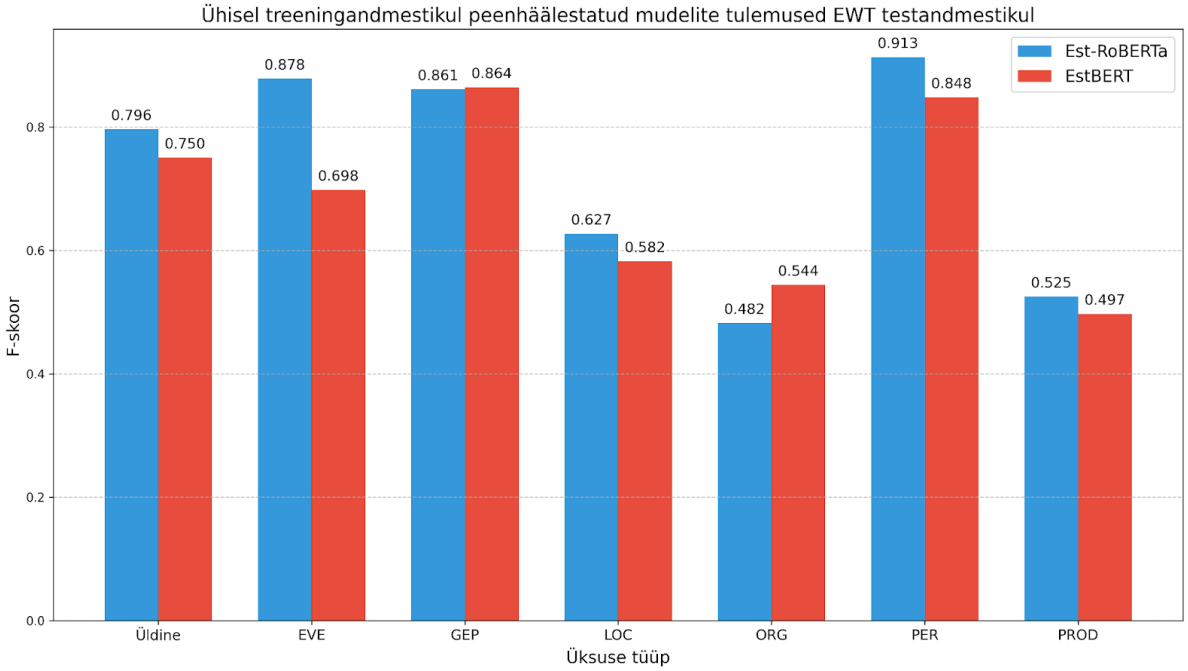
IV. Tulemused ühisel testandmestikul



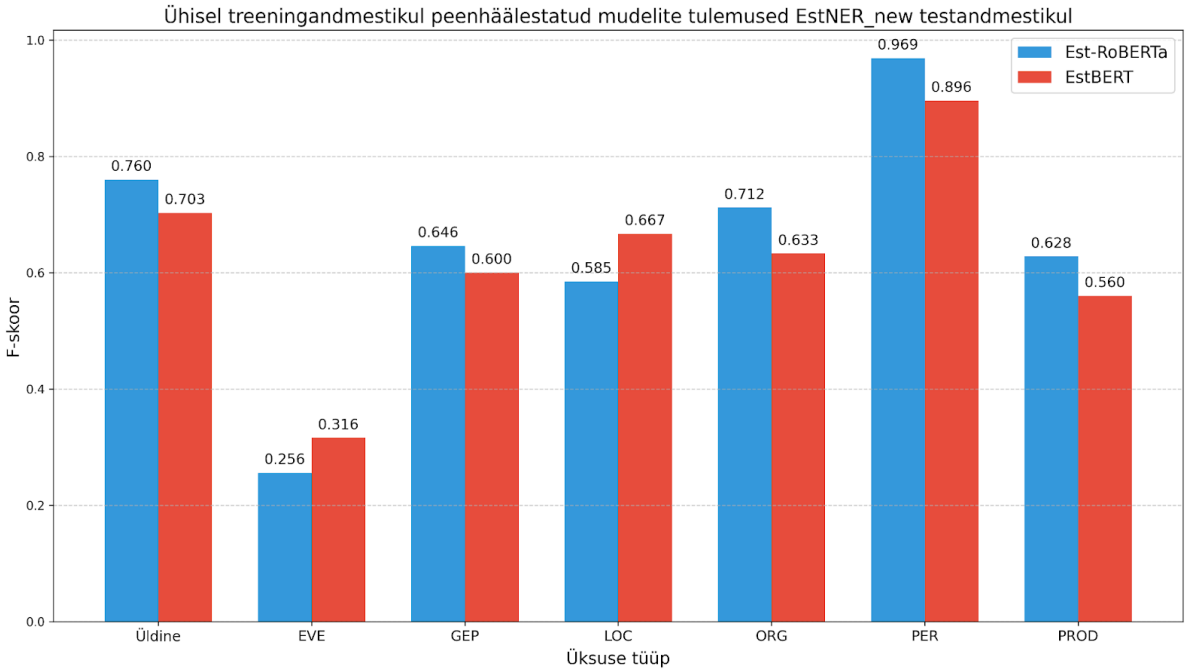
V. Tulemused EDT testandmestikul



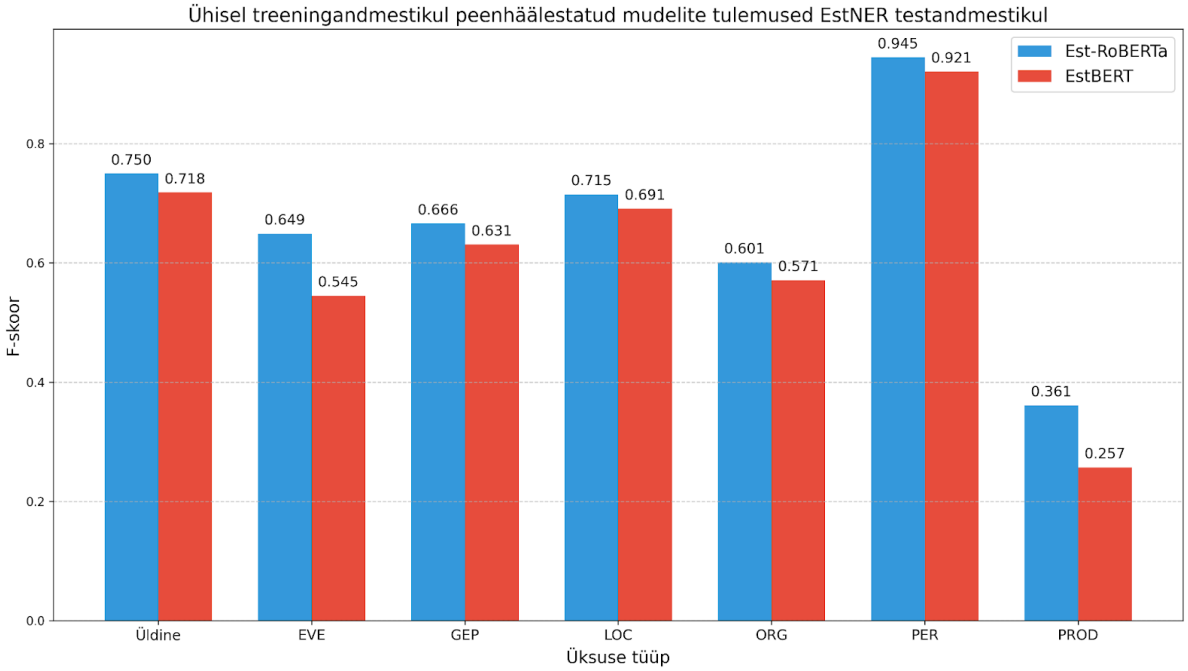
VI. Tulemused EWT testandmestikul



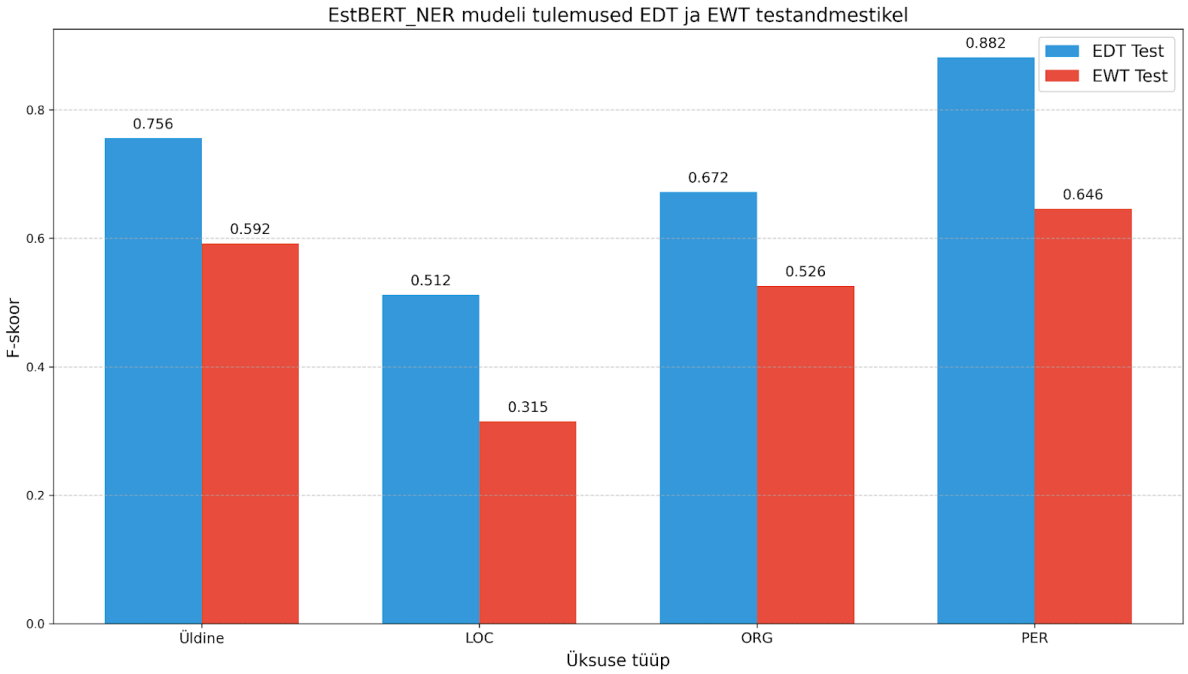
VII. Tulemused EstNER_new testandmestikul



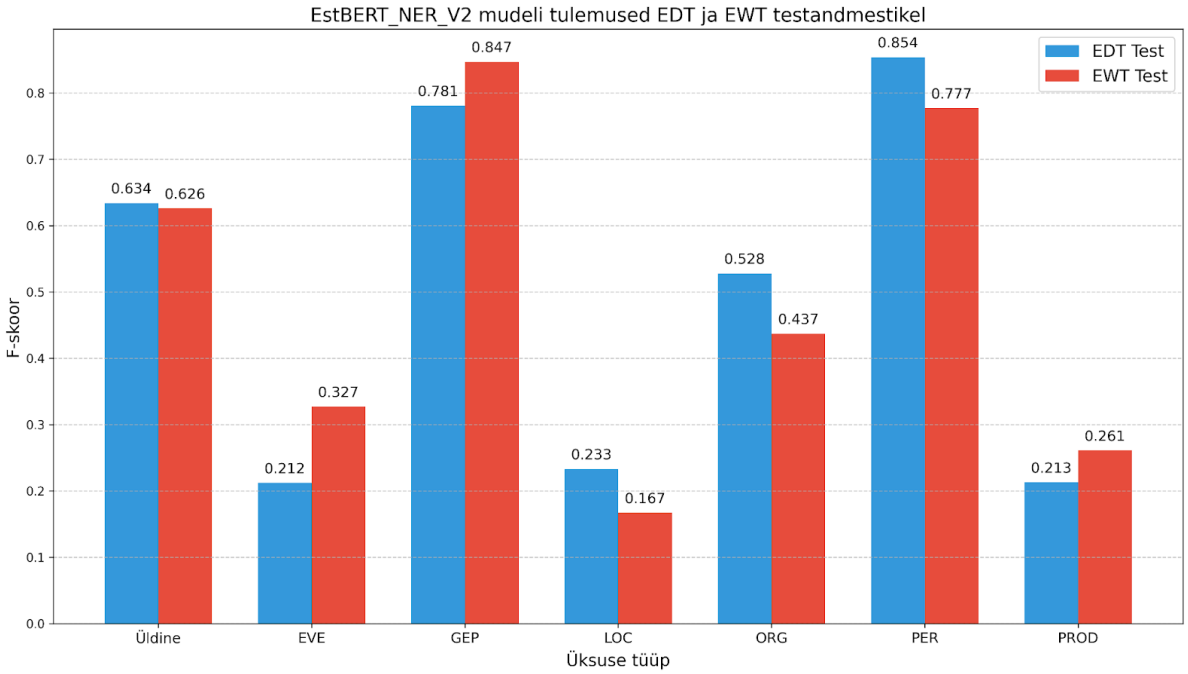
VIII. Tulemused EstNER testandmestikul



IX. EstBERT_NER mudeli tulemused EDT ja EWT testandmestikel



X. EstBERT_NER_V2 mudeli tulemused EDT ja EWT testandmestikel



Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks
Mina, Martin Kivisikk,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose "Nimeüksuste tuvastaja loomine puudepanga korpuse põhjal", mille juhendaja on Siim Orasmaa, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada Tartu Ülikooli digitaalarhiivi kuni autoriõiguse kehtivuse lõppemiseni;

2. annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi kaudu Creative Commonsi litsentsiga CC BY NC ND 4.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni;

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

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

Martin Kivisikk

15.05.2025