

TARTU ÜLIKOOL  
Arvutiteaduse instituut  
Informaatika õppekava

**Brait Õispuu**

**Eesti Vabariigi seaduste ja kohtulahendite  
andmebaas ning ristviitamine**

**Bakalaureusetöö (9 EAP)**

Juhendajad: Mart Randala  
Mare Koit

Tartu 2016

# **Eesti Vabariigi seaduste ja kohtulahendite andmebaas ning ristviitamine**

## **Lühikokkuvõte:**

Riigi Teataja on Eesti Vabariigi ametlik võrguväljaanne, kust on võimalik leida seaduseid ja kohtulahendeid. Selles bakalaureusetöös otsitakse Riigi Teataja veebilehelt üles hetkel kehtivad seadused ja viimase kümne aasta jooksul avaldatud kohtulahendid. Informatsioon laaditakse Riigi Teataja lehelt alla automaatselt, kasutades selleks Scrapy tööriista abil kirjutatud programmi. Kuna vajalikud andmed on leitavad vaid PDF-failidena, kasutatakse neist teksti eraldamiseks tööriista PDFMiner. Täistekstidest metaandmete eraldamiseks on kirjutatud vastav tekstianalüsaator. Seejärel seotakse seadused ja kohtulahendid omavahel ning saadud andmestikku kasutatakse andmeanalüüsiks, uurides eelkõige seaduste kasutatavust kohtulahendites. Andmetöötlust tehakse programmeerimiskeeles Python, päringuid koostatakse programmeerimiskeeles SQL. Töö tulemustest on näha, et põhiline töö tehakse kohtutes ära vaid mõne seadusega ning paljudele hetkel kehtivatele seadustele ei ole viimase kümne aasta jooksul kordagi viidatud.

## **Võtmesõnad:**

Andmebaas, seadused, kohtulahendid, ristviitamine, päringud

## **CERCS:**

**P175**

**Informaatika, süsteemiteooria**

## **Abstract:**

Riigi Teataja is an official publication of the Republic of Estonia, where laws and adjudications can be found. In this Bachelor's thesis, adjudications from the last 10 years and currently effective laws are downloaded using a tool named Scrapy. Because the data is only available as PDF files, an additional tool named PDFMiner is used, to parse text from the files. In order to extract metadata from the full texts, a corresponding text-analysing program is written. The data is then cross-referenced and analysed, especially with focus on the usage of laws in adjudications. Data processing is done using the Python language, queries are constructed in SQL. The results show that most of the work in courts is done using only a few laws and that many of the currently effective laws have not been referenced once in the last ten years.

## **Keywords:**

Database, laws, adjudications, cross-referencing, queries

## **CERCS:**

**P175**

**Informatics, systems theory**

## Sisukord

1.	Sissejuhatus .....	4
2.	Taustainfo.....	6
2.1	Python.....	6
2.2	Scrapy .....	6
2.3	PDFMiner .....	6
2.4	Peewee.....	6
2.5	MySQL.....	6
3.	Andmebaasi koostamine .....	7
3.1	Ülevaade olemasolevast projektist .....	7
3.2	Muudatused olemasolevale programmile.....	7
3.2.1	Scrapy võimalused .....	8
3.2.2	Süntaksianalüsaator.....	9
3.3	Muudatuste edukus .....	10
3.3.1	Scrapy programm .....	10
3.3.2	Süntaksianalüsaator.....	10
3.4	Seaduste ja kohtulahendite vastastikune indekseerimine .....	11
4.	Päringud .....	13
5.	Kokkuvõte .....	20
6.	Kasutatud materjalid .....	21
Lisad.....		23
I.	Loodud andmebaasi mudel .....	23
II.	Kasutamata seaduste loetelu.....	24
III.	Valminud programmi kood .....	26
IV.	Litsents .....	27

## 1. Sissejuhatus

Registrite ja Infosüsteemide Keskuse veebilehel kirjutatakse, et „Riigi Teataja on Eesti Vabariigi ametlik infokanal, mida annab alates 1. jaanuarist 2011 välja Justiitsministeerium. See ilmub vaid Internetis ning selle tehnilise poole toimimist kontrollib Registreerimis- ja Infosüsteemide Keskus (RIK)“ [1]. Riigi Teataja võimalusi on seal kirjeldatud järgmiselt:

- Riigi Teataja esilehel saab tutvuda värskelt ilmunuga ja leiab teavet peagi ilmuvate aktide kohta.
- Soovitud dokumendi leidmiseks saab kasutada otsingut. Otsing jaguneb: "otsing terviktekstidest" ja "otsing algtekstidest".
- Kronoloogia võimaldab leida kindlal ajal avaldatud õigusakte.
- Eelnõude otsing võimaldab otsida välja soovitud eelnõu ja vaadata selle menetluskäiku.
- Kohtuistungi aja ja koha otsingu kaudu leiab kohtuistungi ja kohtulahendi avalikult teatavaks tegemise aja ja koha.
- Kohtulahendite otsinguga leiab jõustunud kohtulahendeid.
- Õigusuudiste osas saab lugeda avaldatud kohtu-uudiseid, seadusuudiseid ja muid õigusuudiseid.
- E-postile teadete tellimiseks ja endale lingimärgmiku loomiseks tuleb registreeruda Riigi Teataja kasutajaks.

Riigi Teataja kodulehelt võib lugeda, et „2014. aasta 1. juunist on avaldatud peaaegu kõikide kohalike omavalitsuste kõik ajakohases seisus määrused. [...] Kokku oli 2014. aasta 1. juuni seisuga avaldatud 16 254 kehtivat kohaliku omavalitsuse määruse tervikteksti“ [2].

Minu bakalaureusetöö jaoks pakuvad enim huvi Riigi Teataja kodulehel olev otsinguvõimalus ning kehtivate ning varasemate lühenditega seaduste kättesaadavus.

Otsinguvõimaluse tutvustus on välja toodud selliselt: „Riigi Teataja lehele koondati kohtulahendite otsingud erinevatelt veebilehekülgedelt; on võimalik otsida jõustunud maa- ja ringkonnakohtu lahendeid ning eraldi Riigikohtu veebilehel olevaid lahendeid. Kuna Riigi Teatajas kohtuotsuseid ei avaldata ja otsinguvormid pöörduvad e-toimikusse sisestatud otsuste poole, siis otsitulemuste headust ja tulemuste leidmist Riigi Teataja väljaandja ülesannete täitja tulemusi mõjutada ei saa“ [2].

Praegu on võimalik teha otsinguid kõikide kohtumenetluste seas korraga järgmiste parameetrite järgi: kohtuasja number, lahendi kuupäev, menetluse alguse kuupäev, kohus, riigikohtu koosseisu tunnus, kohtunik/kohtu koosseis, annotatsiooni tekst, menetluse liik, lahendi liik ning lahendi tekst. Lahendi teksti järgi on võimalik leida vaid üks-üheselt pärin-gule vastavaid tulemusi.

Peale otsingu sooritamist kuvatakse kasutajale tabel, mis sisaldab endas kohtuasja numbrit, kohtu nime, lahendi liiki, kohtuniku/kohtu koosseisu andmeid ning jõustumise ja lahendi kuupäeva. Igal tabelireal on ka PDF-kujul fail, mis on vabalt allalaetav.

Sellisel kujul ei ole aga info hõlpsasti kättesaadav ning ilma täpseid asjaolusid teadmata on kasutajal võrdlemisi keeruline teda huvitavaid lahendeid leida.

Minu bakalaureusetöö eesmärgiks on Riigi Teataja lehel olevate kohtulahendite ja seaduste allalaadimine, nendest andmebaasi koostamine ning saadud andmehulga töötlemine.

Eesmärgi saavutamiseks kasutan ma põhiliselt programmeerimiskeeli Python ning MySQL. Suure andmehulga allalaadimiseks ning andmebaasi kandmiseks kasutan ma Pythoni vabavaralisi teeke Scrapy, PDFMiner ning peewee. Nende põhjalikuma tutvustuse esitan ma töö 2. peatükis.

Kohtulahenditel on olemas oma struktuur. Reeglina on teksti seest leitavad kohtu ning kohtunike nimed, lahendi liik, number ja kuupäev. Nende tuvastamise jaoks olen ma kirjutanud eraldi programmi, mis üritab vajalikku infot teksti seest üles leida. Samuti salvestan ma tervikteksti.

Töö järgmine osa (3. ptk) keskendub seaduste ning kohtulahendite omavahelisele võrdlemisele. Eesmärk on leida kohtulahendite seest üles selles lahendis kasutatavad seaduste lühendid ja paragrahvid ning võrrelda neid seadustega. Selle info põhjal on siis võimalik teha päringuid näiteks seaduste kasutatavuse kohta.

Viimase osana (4. ptk) teostan ma valminud andmebaasi peal päringuid, mis kirjeldavad seaduste kasutust kohtute töös.

Lisas nr. 1 on esitatud loodud andmebaasi mudel.

Lisas nr. 2 on esitatud töö käigus leitud viideteta seaduste nimekiri.

Lisas nr. 3 on töö käigus valminud programm.

## 2. Taustainfo

Selles peatükis annan ma lühiülevaate töös kasutatavatest tööriistadest ja tehnoloogiatest.

### 2.1 Python

Python on programmeerimiskeel, mis loodi 1980. aastate lõpus. Selle erilisteks tunnusteks võib pidada taanete tähtsust koodis ning keele lihtsust. Ka Tartu Ülikoolis õpetatakse algajatele programmeerimist just Pythonis.

### 2.2 Scrapy

Scrapy on programmeerimiskeeles Python kirjutatud raamistik, mis on mõeldud veebilehtedel liikumiseks ning struktureeritud andmete eraldamiseks. Saadud andmestikku saab kasutada paljudel erinevatel eesmärkidel, näiteks andmekaeveks, infotöötluks või arhiveerimiseks [3]. Scrapy võimaldab defineerida erinevaid andmeklasse ja ehitada „ämblikke“ (*spider*), mis liiguvad veebilehtedel ringi ning salvestavad programmeerijat huvitavat infot. Minu töö jaoks on eriti oluline mitte puhtalt HTML kujul leiduva teabe salvestamine, vaid PDF failide allalaadimine. Just sellel eesmärgil on Scrapy raames arendatud välja nn. failide torujuhe (*Files Pipeline*), mille eesmärgiks ongi erisuguste failide salvestamine [4].

### 2.3 PDFMiner

PDFMiner on samuti programmeerimiskeeles Python kirjutatud tööriist, millega eraldatakse informatsiooni PDF dokumentidest. Selle põhifookuseks on tekstitöötlus [5]. Seda saab kasutada väga mugavalt käsurea tööriistana, kuid minu töö jaoks on olulisem pääseda ligi koodile endale ning seda siis vastavalt vajadusele muuta. Tegemist ei ole kõikvõimsa vahendiga ja kahjuks jääb osade PDF kujul avatavate ning vaadatavate failide sisu kättesaamatuks.

### 2.4 Peewee

Viimane suurem Pythoni tööriist, mida ma oma töös kasutan, on peewee. See võimaldab kasutajal teha andmebaasi operatsioone otse Pythoni koodi sees. Peewee võimaldab kasutajal salvestada andmebaasi terveid objekte lihtsate sõnede ja numbrite asemel [6].

### 2.5 MySQL

MySQL on andmebaasi haldamise süsteem, mida arendab ja toetab Oracle Corporation. Andmed on salvestatud relatsioonidena, mitte ühe suure kogumina. Andmebaasi toiminguid tehakse keeles SQL (Structured Query Language) [7].

### 3. Andmebaasi koostamine

Varasemalt on minu tööga sarnasel viisil töötatud Riigikohtu lahenditega. Mul on ligipääs projektile, kus otsitakse samade tööriistade abil Riigikohtu veebilehelt [8] üles kõik seal olemasolevad lahendid ning võrreldakse neid Riigi Teataja lehelt saadud seadustega. Varasemalt tehtud ning minu töö vahe seisneb selles, et kui Riigikohtu lahendid on kättesaadavad HTML kujul lihttekstina, siis kõik ülejäänud lahendid, s.t. maa-, ringkonna- ja halduskohute lahendid, vaid PDF (*Portable Document Format*) kujul. Samuti on Riigikohtu lahendite struktuur, olgugi et ajas suuresti muutuv, ühtlasem ning sealt on lihtsam spetsiifilist infot välja lugeda.

#### 3.1 Ülevaade olemasolevast projektist

Eeskujuks olev projekt koosneb kolmest suuremast osast: seaduste ja Riigikohtu lahendite allalaadija, seaduste süntaksianalüsaator ja lahendite süntaksianalüsaator.

Seaduste allalaadimiseks on loodud programm, mis otsib Riigi Teataja leheküljelt „Lühendid ja nende vasted“ [9] üles kõik Eesti Vabariigis kehtivad seadused, laeb need XML kujul alla ning salvestab andmebaasi nende terviktekstid ja metaandmed. Need on tähtsad, kuna kohtulahendites viidatakse enim just lühenditega seadustele. Allalaadimine toimub Scrapy raamistiku abil.

Riigikohtu lahendite allalaadimiseks kasutatakse samuti vajadustele vastavalt kirjutatud Scrapy raamistiku programmi, mis otsib Riigikohtu otsingulehelt [10] aastate kaupa üles kõik avaldatud kohtulahendid ja salvestab need sõne kujul andmebaasi.

Süntaksianalüsaatorid teevad teksti järeltöötlust. Näiteks liigitab seaduste analüsaator seadused andmebaasi paragrahvi lõikude punktide täpsusega. Lahendite analüsaator otsib lahendite tekstide seest üles kõik viited paragrahvidele ning salvestab leitud viited uude tabelisse, et need kiiresti ligipäasetavad oleksid.

#### 3.2 Muudatused olemasolevale programmile

Minu töö esimene etapp on kohtulahendite allalaadimine. Selle jaoks tuleb mul kirjutada programm, mis laeb alla kõik Riigi Teataja lehel olevad kohtulahendid.

Riigi Teataja otsingulehel on kirjutatud, et „Riigi Teatajas ei avaldata kohtulahendeid, vaid on loodud võimalus otsida kohtute poolt avaldatud lahendeid. Kui mõni kohtulahend, mis peaks olema avaldatud, ei tule otsinguga välja, või soovite taotleda isikuandmete asendamist, siis palun pöörduge asjaomase kohtu poole.“ [11] Samuti ei ole kohtulahendid digitaalseeritud kujul kättesaadavad kogu taasiseseisvunud aja raames, vaid „Käesoleval otsingulehel on võimalik otsida kõiki I ja II kohtuastme lahendeid, mis on avalikustatud pärast 2006. aastat ning kõiki Riigikohtu lahendeid. Varasemaid I ja II kohtuastme lahendeid on võimalik otsida [...]“, [11] teisest Riigi Teataja veebilehel leiduvast otsingumootorist, kus on leitavad kohtulahendid, „mis on avalikustatud kohtustatistika ja kohtulahendite andmekogus (KOLA) enne 2006. aastat. Otsinguga on leitavad lahendid alates 2001. aastast, mis on kohtu poolt registreeritud ja avalikustatud. Tsiviilasjades ja kriminaalasjades on osaliselt kättesaadavad ka enne 2001. aastat tehtud lahendid“ [12]. On otsustatud, et kui otsida ja alla laadida kõik alates 2006. aastast avaldatud kohtulahendid, on see piisav andmehulk minu bakalaureusetöö jaoks.

Mõistlikke otsinguid, mis võimaldaksid näha kõiki kohtulahendeid, on minu hinnangul kaks: lahendi kuupäeva järgi, valides esimese võimaliku kuupäeva ning käesoleva kuupäeva, või kohtuliigi järgi, valides vastavalt Riigi-, ringkonna-, haldus- ning maakohtud. Oma töös otsustasin ma otsingut sooritada kohtuliigi järgi.

### 3.2.1 Scrapy võimalused

Scrapy põhilisteks tööriistadeks on klassid ja „ämblikud“. Klasse defineeritakse nii nagu on Pythonile omane ning klassiobjektidele saab lisada atribuute. „Ämblikud“ on klassid, mis luuakse eesmärgiga koguda veebilehtedelt infot.

„Ämbliku“ kasutamiseks tuleb sellele ette anda URL, kust soovitakse informatsiooni eraldada, ning defineerida *parse* funktsioon. *Parse* funktsioonile antakse „ämbliku“ käivitamisel argumendiks veebilehelt vastuseks saadud objekt, mis sisaldab endas kogu veebilehel olevat informatsiooni. Minu programmis on vaja selles funktsioonis defineerida ka reeglid, kuidas liikuda järgmise URLini, s.t. järgmise otsinguleheni.

Scrapy võimaldab kasutada XPathi ning CSSi selektoreid [13]. XPath on keel, mis on loodud XML tüüpi dokumentide sees päringute tegemiseks, kuid seda saab kasutada ka HTML tüüpi dokumentide korral. CSS on keel, millega saab kujundada HTML lehti. Kujundamise käigus lisatakse HTML elementidele klasse ning IDsid, mille põhjal saabki hiljem päringuid teha. Kui CSS võimaldab päringuid teha ainult veebilehe struktuuri peal, siis XPath suudab ka elementide sisu vaadelda [13].

Minu programm peab sisaldama „ämblikku“, mis saab URL parameetrina sisendiks päringu, mis on sooritatud kohtuliigi järgi. Järgmisele lehele minekuks tuleb CSS ja/või XPathi abil liikuda listi elemendini, millel on CSSi klass *next*. Juhul kui selline element leidub, on selle alamelemendi sisse salvestatud järgmise lehe URL parameeter, mille saabki Scrapy sisen-diks anda. Niimoodi on võimalik läbi käia kõik vastetega lehed.

Kuigi tulemuste tabel sisaldab ka informatsiooni kohtulahendite sisust, s.t. kohtuasja numbri, kohtu, lahendi liigi, kohtunike, jõustumise ja lahendi kuupäeva kohta, ei leidu kusagil informatsiooni, mis seletaks, kuidas see info sinna sisestatud on. Seetõttu, lähtudes lausest „Kuna Riigi Teatajas kohtuotsuseid ei avaldata ja otsinguvormid pöörduvad e-toimikusse sisestatud otsuste poole, siis otsitulemuste headust ja tulemuste leidmist Riigi Teataja väljaandja ülesannete täitja tulemusi mõjutada ei saa.“ [2] otsustasin ma minule vajamineva teabe ise faili seest välja lugeda.

Selleks, et saada kätte kõikvõimalikud kohtulahendid, on mul vaja lokaliseerida otsingutulemuste tabelis olevad PDF failide URL parameetrid. Need asuvad tabeli CSSi klassiga „Fail“ a elementide „href“ atribuudis. Salvestades need atribuudid kasutades XPath ja CSS päringuid oma objekti „file\_urls“ atribuuti, saan ma failid nn. faili torujuhtme kaudu hõlpsasti alla laadida.

Faili torujuhe (*FILES PIPELINE*) on Scrapy klass, mis on mõeldud veebilehtedelt failide alla laadimiseks. Kasutades seda eeldefineeritud klassi, saab hõlpsasti defineerida failide allalaadimise sihtkoha ning veenduda, et sama faili ei laeta alla mitu korda. Faili torujuhet rakendatakse juhul, kui Scrapy objektil on olemas „file\_urls“ atribuut. Selle väärtus, milleks on URL aadress, pannakse automaatselt allalaaditavate failide järjekorda, kusjuures allalaaditavad failid on kõrgema prioriteediga kui lehtedel liikumine, s.t. enne ei liiguta uuele leheküljele, kui failid on edukalt alla laaditud, või protsess on nurjunud. Kui allalaadimine on edukas, salvestatakse fail objekti „file“ atribuuti [4].

Tavaliselt salvestatakse failinimeks SHA1 paiskeväärtus faili originaalsest nimest, kuid minu töö jaoks oleks ülevaatlikum, kui fail salvestatakse sama nimega, mis tal Riigi Teataja andmebaasis on. Peale selle tuleb failid kindlasti salvestada PDF kujul. Seetõttu tuleb ümber kirjutada failidele salvestamisel nime andmise funktsioon.



### 3.2.2 Süntaksianalüsaator

Süntaksianalüsaatori puhul tuleb sisse viia päris palju fundamentaalseid muudatusi. Esiteks tuleb implementeerida PDFMineri kasutamine, et üldse oleks võimalik tekstile ligi pääseda, teiseks tuleb arvestada palju suurema hulga erinevate vormistusviisidega kui senises lahenduses.

Kuna on teada, et mingi hulk faile ei ole üldse PDFMineriga ligipääsetavad, tuleb kuidagi need failid välja filtreerida. Selleks teen ma kaks tabelit: esimesse salvestan need tekstid, milles on mingisugune sisu olemas, teise need failid, mille sisu ei ole loetav. On võimalik, et PDFMiner väljastab kohe faili avamisel vea, mis võib seisneda kas selles, et fail ei ole avatav, s.t. fail ei ole ka nt Adobe Acrobat tarkvaraga loetav, või et fail on parooliga kaitstud ning programm ei pääse sellele ligi. Viimasel juhul on võimalik, et failile on peale pandud n.-ö. nullparool, mis ei takista faili vaatamist selleks ette nähtud programmiga, kuid PDFMineri privaatsusloogika ei luba selle sisule läheneda. Ka kõik ülejäänud vead leiab programm üles ning salvestab need teise andmebaasi failinime ja vea kirjeldusega.

Peale selle on võimalik variant, kus PDFMiner tagastab teksti, ent see ei ole loetav. Seetõttu on mõistlik kontrollida, kas tekstifaili alguses on olemas sõnad „kohtuotsus“, „resolütiivotsus“, „kohtumäärus“, „määrus“ või „teatis“. Kohtulahendi tüüp on välja toodud kindlasti dokumendi alguses, kuid on võimalik, et tüüp on välja toodud kujul, kus iga täht on kirjutatud eraldi, nt. „R E S O L U T I I V O T S U S“. Sellisel juhul loeb programm kõik tähed eraldi sõnadeks ning ma pean suurendama otsitava ala suurust. Juhul kui esineb taoline tähtede lahkukirjutamise viis, piisab tüüpi määramiseks umbes 25 sõna seas otsingu tegemisest. Kuna ette võib tulla ka erandeid, kus tüüp on välja toodud kaugemal, võtsin piiriks 35 sõna. Juhul kui kohtulahendi tüüp on olemas, saab juba kindlalt öelda, mis tüüpi failiga on tegu ning analüüsi jätkata, kui need aga puuduvad, on see piisav teadmine oletamiseks, et fail ei ole tegelikult loetav ning leitud objekti võib salvestada teise tabelisse failinime ja seekord ka täistekstiga (et saaks hiljem tuvastada, kas ikka päriselt on loetamatu fail).

PDFMiner suudab lugeda tekstidest välja ASCII ning Unicode formaadis sõnesid. See konkreetne tarkvara ei ole mõeldud pildikujul teksti tuvastamiseks ega suuda optiliselt tähti tuvastada. Tekstitötluse poolelt on see tarkvara siiski väga tõhus ning suudab tuvastada nii sõnede asukohta tekstis, kirjastiilide nimesid, suuruseid kui ka kirjutamise suunda [5].

PDFMinerile on võimalik kaasa anda palju erineva funktsionaalsusega parameetreid. Käsurealt saab parameetritega määrata väljundifaili nime, töödeldavate lehekülgede numbreid, väljundi kodeeringut, väljundi formaati (nt. tekst, HTML, XML), JPEG formaadis piltide salvestamist, tähtede, ridade ja sõnade vahe informatsiooni, teksti horisontaalsuse ja vertikaalsuse tähtsust, maksimaalset töödeldavate lehtede arvu ning faili parooli [5]. Mina oma töös selle programmi käitamiseks käsurida ei kasuta, kuid annan siiski parameetrid programmile ette. Ma tahan, et tagastataks 'iso-8859-4' kodeeringus tekst. Töödeldud peavad saama kõik leheküljed ning katse-eksituse meetodil tegin kindlaks, et kõige parema vaste saan ma juhul, kui keelan igasuguse teksti paigutuse tuvastamise.

Pärast tüüpi määramist on mul olemas objektid, millesse on salvestatud failinimi, täistekst ning lahendi tüüp. Iga faili puhul peaks olema ka tuvastatav, kas otsus või määrus avaldati kriminaal-, väärteto-, tsiviil- või haldusasja kohta. Peale nende on kohtulahendi objektile veel järgmised atribuudid: kohus, kohtunik, sekretär, süüdistaja, kaitsja, süüalune, lahendi kuupäev, kohtulahendi number, kohtulahendi kirjeldus, resolutsioon, apellatsiooni tingimused ja sisu. Kõikides failides ei ole infot kõigi nende omaduste kohta, kuid kindlasti peaks olema tuvastatavad kohtunik, kohus, lahendi kuupäev ja kohtulahendi number. Andmebaasi mudel on esitatud lisas nr. 1. Töö tulemusel valminud andmebaasi ärisaladuse tõttu ei avaldata.

Kuna kohtulahenditel on olemas teatud struktuur ja erilisi andmeid, mida ma üritan lahendite teksti seest kätte saada ei ole palju, valin ma atribuutide kätte saamiseks viisi, kus mingi väiksema testgrupi peal üritan saajaprotsendiliselt andmed kätte saada ning siis loodan, et reeglid, mis kehtivad kontrollgrupi peal, kehtivad ka tervikul. Oma testgrupi suuruseks valisin ma 1000 faili, kuna nende järgi peaks olema võimalik üldisest kohtulahendite ülesehitusest aru saada. Atribuutide leidmiseks kasutan ma märksõnu, regulaaravaldisi ning atribuutide omadusi. Näiteks võib eeldada, et kui on leitud märksõna „kohtunik“, peavad vähemalt järgmised 2 sõna olema kas läbiva esisuurtähe või trükitähtedega kirjutatud, selleks et kohtuniku nime tekstist välja lugeda. Kohtuasja numbri tuvastamiseks kasutatud regulaaravaldis on toodud joonisel 1.

```
\d{1}\s?[\s-]\s?\d{2}\s?[-\s]\s?\d{1,5}\s?([\(/][\d{11}\d{1}][\(/])?
```

Joonis 1: Kohtuasja numbri tuvastamiseks mõeldud regulaaravaldis.

Kohtuinfo leidmiseks otsin ma kõigepealt tekstist märksõna „Kohus“: kui see ei tagasta tulemusi, siis otsin eraldi kõigi kohtute nimesid. Kui testgrupi peal on tulemused piisavalt täpsed, käitan ma programmi kogu andmestiku peal.

### 3.3 Muudatuste edukus

#### 3.3.1 Scrapy programm

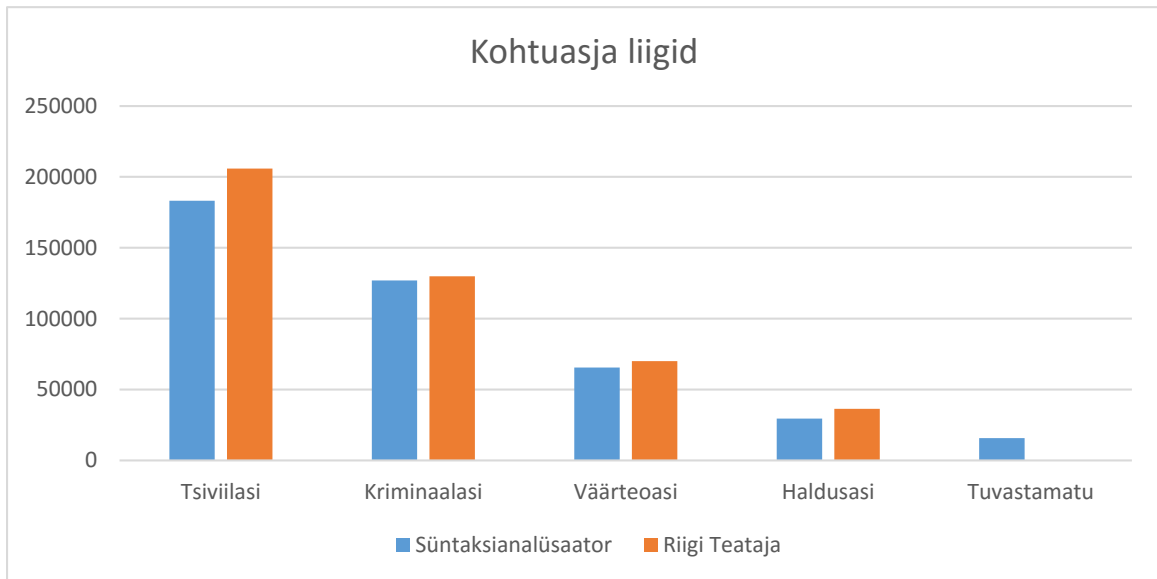
Sisseviidud muudatustega Scrapy „ämbliku“ käivitasin ma 9.04.2016 õhtul kell 21:26. Kõik töös kasutatavad failid on allalaaditud just selle hetkeseisuga. Programm lõpetas oma töö 15.04.2016 kell 11:50. Andmete reaajas uuendamist ei ole selle bakalaureusetöö raames ette nähtud. Programmi käitamise tulemusena laaditi alla 443 304 PDF faili, mille kogusuurus on 28,1 gigabaiti. Neist 420 981 puhul oli kohtulahendi tüüp leitav, s.t. päring oli edukas, ning 22 323 korral esines kas faili avamisel mingi viga või ei olnud faili sisu loetav. Samas sattus ka edukate päringute hulka faile, kus tekst on vaid osaliselt loetav, selliseid faile on 8215. Kokku teeb see 5,04% nurjunud päringuid, 1,85% osaliselt nurjunud päringuid ning 93,11% edukaid päringuid. See on piisavalt hea protsent, et nende vastete põhjal päringuid teha.

#### 3.3.2 Süntaksianalüsaator

Andmebaasi loetud kirjetel peaks kõigil olemas olema kohtulahendi ja kohtuasja tüüpide, kohtu, kohtuniku ja lahendi kuupäeva info. Kahjuks ilmnas, et tuhandene testgrupp oli liiga väike ning suured erinevused teatud andmete vormistamises ei tulnud testfaasis ilmsiks.

Kõigil lahenditel on olemas kohtulahendi tüüp, sest see oli eelduseks, et päring saaks õnnestuda. Tuli välja, et tüüp „Teatis“, mille töötlemise vajadus ilmnas testgrupis, esines kõigi lahendite seas vaid üks kord. Peale selle jagunesid lahendid 252 470 kohtuotsuseks ning 168 510 kohtumääruseks. Otsides lahenditüüpide järgi Riigi Teataja portaalis, on tulemuseks vastavalt 265 573 ja 176 076 lahendit. Teatiseid ei ole võimalik Riigi Teataja otsingumootoriga otsida. Riigi Teatajas sooritatud kontrollotsingud on tehtud 8.05.16.

Kõigi lahendite seast 15 757 ehk 3,74% puhul ei suutnud programm tuvastada, mis tüüpi asja arutati. Ülejäänud lahenditest 183 219 puudutasid tsiviil-, 126 998 kriminaal-, 65 507



Joonis 2: Kohtuasjade liigid.

väärteo- ning 29 500 haldusasu. Riigi Teatajast otsides on tulemuste arv vastavalt 205 882, 129 824, 69 967 ja 36 417 (vt. ka joonis 2).

Kohus jäi leidmata 5 481 ehk 1,30% lahendite puhul. Ülejäänud lahenditest 355 884 olid maa-, 26 205 haldus-, 32 293 ringkonna- ning 918 Riigikohtu poolt väljastatud. Riigi Teataja vasted on vastavalt 377 871, 26 040, 31 472 ja 10 396. On ilmne, et Riigikohtu lahendite kohtuinfo tuvastamisel ei olnud programm usaldusväärne. See tuleneb sellest, et märksõna „Kohus“ puudumisel otsiti tingimuslausete abil täistekstist kohtute nimesid, kus Riigikohtu otsiti kõige viimasena. Ilmselt on enamikes Riigikohtu lahendites mainitud, kus tehti varasemad otsused, ning programm liigitas lahendid nende kohtute alla. Riigikohtu raames ma seetõttu päringuid läbi ei vii.

Kohtunikku ei leidnud programm 27 046 faili korral, mis teeb 6,42% failide koguhulgast. Minu andmetel on viimase kümne aasta jooksul enim lahendeid välja andnud kohtunikud Tatjana Bogdanova, Galina Jasnjuk ja Anne Palmiste, vastavalt 5455, 4683 ja 4099. Otsides neid Riigi Teatajast, saan vasteteks 6003, 4836 ja 4221.

Lahendi kuupäev on puudu 8206 failil, mis teeb koguhulgast 1,95%. Kui otsida lahendeid 2015. aasta kohta, saab andmebaasist 29 197 vastet, Riigi Teatajas on tulemusi 30 075.

Üldiselt on näha, et iga atribuudi puhul on ilmnunud mingisugused kaod, kuid eelnevalt väljatoodud omaduste põhjal on siiski mõistlik koostada päringuid seaduste kasutatavuse kohta, kuna ei ole põhjust arvata, et kaotsiläänud informatsioon kuidagi seaduste kasutatavusega kooskõlas oleks.

### 3.4 Seaduste ja kohtulahendite vastastikune indekseerimine

Seaduste andmebaasi lugemiseks kasutan ma punktis 3.1 kirjeldatud seaduste allalaadimise „ämblikut“. See programm salvestab liigendatud kujul kõikide kehtivate seaduste terviktekstid. Olenevalt struktuurist salvestab see seaduste paragrahvid kas paragrahvi, lõigu või punkti täpsusega.

Laadides alla kõik uusimad lühenditega seadused, saan ma andmebaasi 361 kirjega. Selleks et kahte andmebaasi omavahel võrrelda, saan ma kasutada sama programmi, millega ühendati Riigikohtu lahendid ja seadused, tehes selle enda andmebaasi nõuetele vastavaks.

Programm, mida ma kasutan, võtab seaduste tabelist kõik kirjed, millel on olemas lühend, ning koondab need ühte listi. Seejärel võrreldakse seda listi regulaaravaldistega, mis otsivad kohtulahendi täistekstis vasteid. Juhul kui leidub vaste, salvestatakse uude tabelisse seaduse ja kohtulahendi IDd, leitud viide, ja vormistatud viide. Kuna ma käitan seda programmi niivõrd suure andmebaasi peal, kirjutasin ma juurde tingimuslause, mis rohkem kui 10 000 vaste puhul töötleb kirjeid korraga maksimaalselt 10 000 vaste kaupa.

Pärast programmi töö lõpetamist sain ma uue tabeli, milles on 2 916 652 kirjet, nende seas erinevaid 33 172. Kuna tabelis, kuhu on salvestatud seaduste sisu, on 172 817 kirjet, siis tähendab see, et 139 645 paragrahvile/lõigule/punktile ei ole viimase kümne aasta jooksul üheski kohtulahendis viidatud.

Tabelisse laetud seaduste seas on kaks sellist, millele ei ole veel võimalik viidata, kuna nad hakkavad kehtima vastavalt 1.07.2016 ja 1.01.2018, ja nii langeb seadusepunktide koguhulk 172 180 peale. Seega on viiteta punkte 139 008. Neid kaht seadust edasistes päringutes ei kasutata.

## 4. Päringud

Päringuid teen ma programmeerimiskeeles SQL. MySQL on populaarseim vabavarana kättesaadav relatsioonilise andmebaasi haldamise süsteem [14]. Mais 2016 jäi see populaarsuselt alla vaid Oracle andmebaasi haldamise süsteemile, mis on tasuta kättesaadav ainult piiratud kujul.

On tavaks, et SQL võtmesõnad kirjutatakse trükitähtedega, sest nii on neid lihtsam muust tekstist eristada. Reaalsuses ei ole tänapäeval enam vahet, kas kirjutada suurte või väikeste tähtedega, kuid kirjutan siin siiski nii nagu on tavaks. Oma töös kasutan ma järgmiseid SQL funktsioone: SELECT, FROM, JOIN - ON, WHERE, IN, GROUP BY, ORDER BY, LIKE, COUNT ja LIMIT.

SELECT on käsk, mis võimaldab teha tabelipäringuid. Sellega saab siduda ka alampäringuid. FROM määrab, millisest tabelist päring tehakse. Kui soovitakse päringut teha mitmest tabelist, tuleb kasutada võtmesõna JOIN [15]. MySQL tarkvara raames on JOIN, CROSS JOIN ja INNER JOIN samaväärsed, tavapärases SQLis mitte. MySQLis tuleb neid kõiki kasutada koos võtmesõnaga ON. JOIN abil saab päringusse tabeleid lisada, ON abil kirjeldatakse, milliste parameetrite kaudu tabelid ühendada [16]. Tabelid peavad alati mingi parameetri kaudu ühendatavad olema, muidu ei ole võimalik ühe kaudu teisest midagi leida. Võtmesõna WHERE abil on võimalik määrata, mis tingimustel read tabelist välja valitakse. WHERE klauslisse kirjutatud avaldised peavad omama tõest väärtust, et rida valituks osutuks. Juhul kui WHERE puudub, valitakse kõik read [15]. IN võimaldab tabeli väärtusi võrrelda mingi päringu tulemustega ning tagastab need read, mis kattuvad [17]. GROUP BY ja ORDER BY võimaldavad tulemusi sorteerida. GROUP BY abil saab kõik sama kattuva parameetri väärtused üheks koondada, ORDER BY võimaldab tulemusi järjestada, nt. kasvavas ja kahanevas järjestuses [15]. COUNT abil on võimalik tulemuste arvu loendada. Näiteks, valides COUNT(\*) mingist tabelist ilma lisaargumentideta, saab teada mitu rida selles tabelis on. COUNT(\*) funktsiooni saab kasutada ka koos GROUP BY võtmesõnaga, sel juhul väljastatakse ridade arv, kus GROUP BY parameeter kattub [18]. LIKE võtmesõna võimaldab kasutada erilisi võrdlusoperaatoreid „\_“ ja „%“. Esimene neist sobitub iga ühe sümboliga ning teine ükskõik kui suure hulga suvaliste sümbolitega. Nende abil saab sõnesid võrrelda [19]. LIMIT määrab maksimaalse tagastatavate ridade arvu [15].

Justiitsministeerium avaldas 17.08.2015 uudise, kus tuuakse välja, et „Eesti probleemiks on seaduste ja määruste rohkus ning see, kui kiiresti ja tihti neid muudetakse. See omakorda toob kaasa olukorra, kus muudatuste jälgimine on keeruline nii inimestele, ettevõtetele kui asutustele, [...] Näiteks 2014. aasta 1. jaanuaril kehtinud 384 seadusest on tänaseks muutunud 320 ja mõned neist mitmeid kordi“ [20]. Uudises tuuakse välja, et seaduste pidev muutmine toob kaasa ka lepingute, süsteemide ja praktikate muutusi ning tähendab lisanduvat halduskoormust ettevõtetele. Samuti on uudises kirjutatud, et näiteks riigi poolt makstavad sotsiaaltoetused on reguleeritud kümnes erinevas seaduses.

Seda uudist illustreerib hästi päring, mis uurib, kas leidub terveid seadusi, millele ei viidata kordagi üheski minu kasutatavas kohtulahendis.

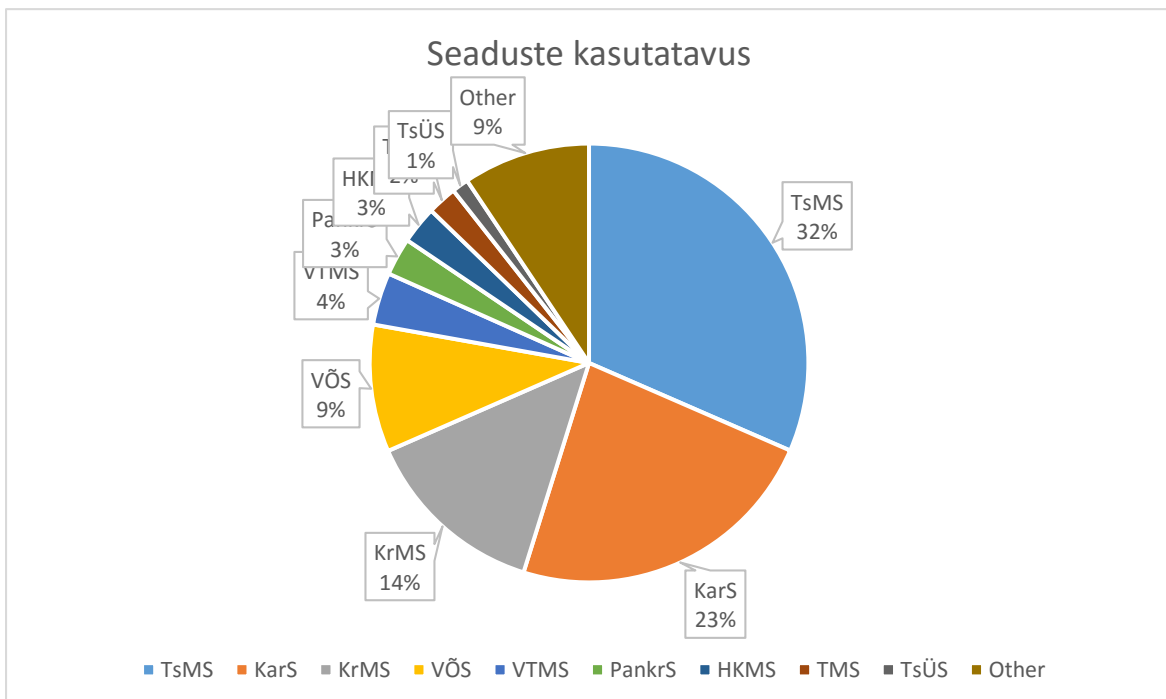
```
SELECT lyhend FROM legislationcontent WHERE legislationcontent.lyhend NOT IN (SELECT legislationcontent.lyhend FROM legislationcontent_cascontent JOIN legislationcontent ON legislationContent_id = legislationcontent.id GROUP BY legislationcontent.lyhend) GROUP BY lyhend;
```

Joonis 3: Päring, mis leiab seadused, millele ei viidata kohtulahendites kordagi.

Joonisel 3 toodud päring tagastab kõik seaduste lühendid, mis ei esine kordagi tabelis, kus on välja toodud kohtulahendite viited seaduste punktidele. Sulgudes olev päring grupeerib kõik viidatud punktid vastava seaduse järgi, mille all nad esinevad, üheks. Seejärel võrdleb väline päring kõikide lühenditega tabelit leitud tulemustega ning tagastab need lühendid, mida esialgsete tulemuste seas ei esine.

Päringutulemustest selgub, et 359 andmebaasis olevast seadusest ei viidata kordagi 52le, mis moodustab koguhulgast 14,49%. Otsustasin oma tulemusi ka Riigi Teataja lehel kontrollida. Väärteomenetluste hulgas on võimalik sooritada otsingut neis kasutatud seaduste lühendite järgi [21] ning kui kõrvutada minu saadud tulemusi sealsete valikus olevate lühenditega, on näha, et kõigist leitud seadustest on seal esindatud vaid üks – KultVS (Kultuuriväärtuste väljaveo, ekspordi ja sisseveo seadus) – ja kui selle järgi otsida, saab tulemuksiks 0 vastet. Leitud viideteta seaduste nimekiri on välja toodud lisas nr. 2.

Kõik kasutatavad seadused saan ma leida päringuga, mis on kujutatud joonisel nr. 3 sulgudes. Ülejäänud 85,51% seadusest, millele on viidatud, jagunevad viidete arvu järgi üsna

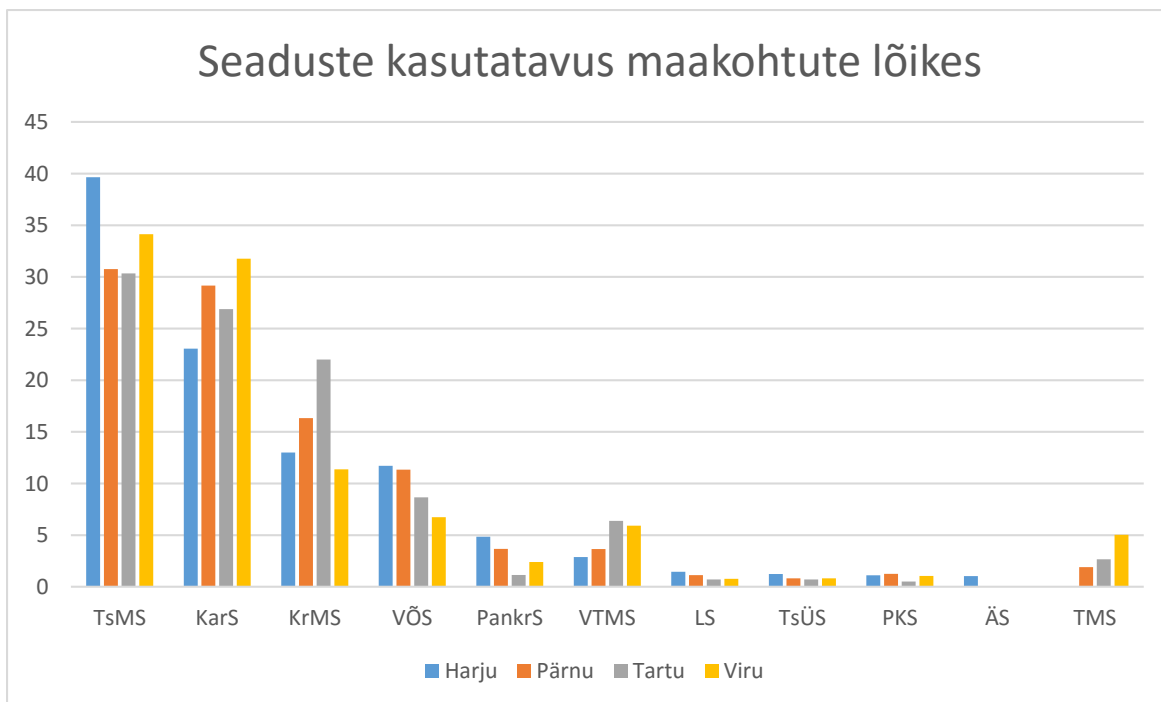


Joonis 4: Seaduste kasutatavus.

ebavõrdselt: Tsiviilkohtumenetluse seadustiku (TsMS) ja Karistusseadustiku (KarS) viited moodustavad kõigist viidetest 55%, ülejäänud esikümnesse jäävad seadused 36% ning ülejäänud 297 seadust kokku vaid 9% viidetest (vt. joonis 4).

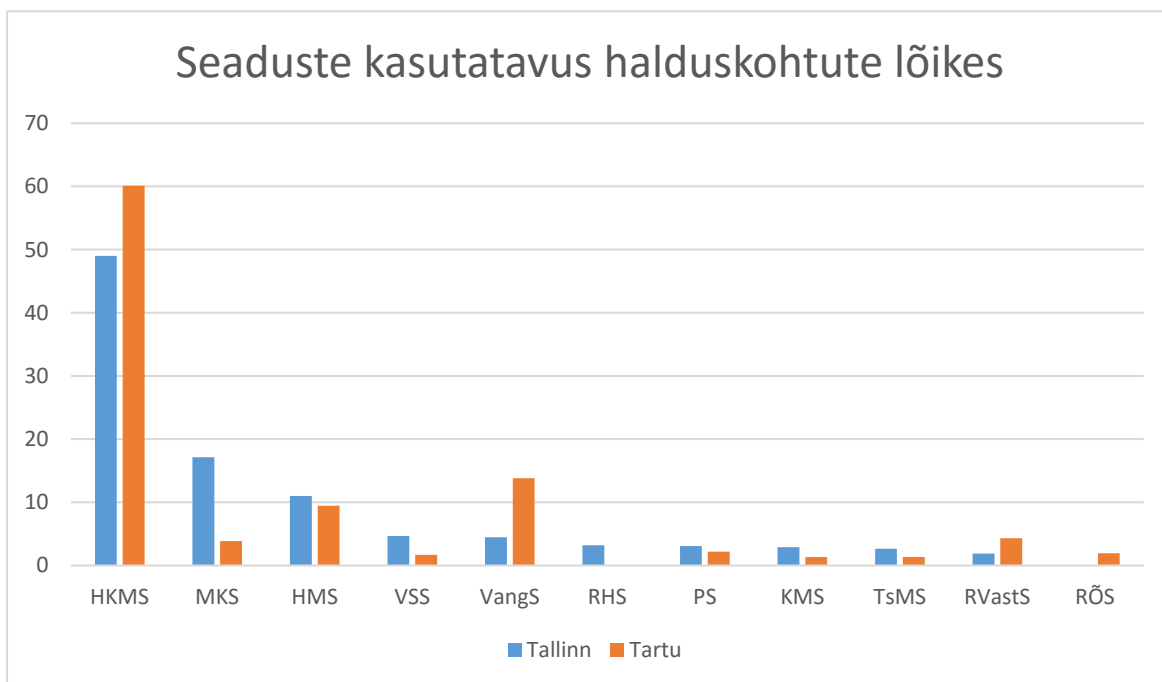
Järgmised päringud üritavad leida, kas kohtute töös kasutatakse mingisuguseid seaduseid tavapärasest rohkem või et kas kohtute töös on märgatavaid erinevusi. Selleks saab kasutada sama päringut, millega genereeriti eelmine seaduste üldise kasutatavuse tabel, kui lisada sellele päringule WHERE käsk, mis filtreerib võtmesõna LIKE abil välja need viited, mis viitavad otsitavale kohtule. Tulemused piiratakse esimese kümne peale. Otsinguid sooritatakse järgnevate kohtute seas: Harju, Tartu, Pärnu ja Viru maakohud, Tallinna ja Tartu halduskohtud, Tallinna ja Tartu ringkonnakohtud.

Joonisel 5 on kujutatud kümne enim kasutatava seaduse protsentuaalne viidete arv esikümnes olevate seaduste koguviidete hulgast maakohutute põhjal. On näha, et suures plaanis on proportsioonid samad, kuid näiteks Tallinna ja Viru maakohututes kasutatakse kohtu töös märgatavalt rohkem Tsiviilkohtumenetluse seadustikku (TsMS) kui Pärnu ja Tartu maakohututes. Samas kasutatakse Harju maakohutus võrreldes teistega palju vähem Karistusseadustikku (KarS). Kõige suurem erinevus kohtute töös on Kriminaalmenetluse seadustiku (KrMS) kasutamise osas, kus Tartu Maakohus kasutab seda oma töös üle 10% rohkem kui Viru Maakohus. Samuti on erinevusi näha tabeli allosas, kus Äriseadustik (ÄS) mahub 10 enim kasutatava hulka vaid Harju Maakohutus, teistes kohtutes on see asendunud Täitemenetluse seadustikuga (TMS). Ülejäänud enim kasutatavad seadused: Võlaõigusseadus (VÕS), Pankrotiseadus (PankrS), Väärteomenetluse seadustik (VTMS), Liiklusseadus (LS), Tsiviilseadustiku üldosa seadus (TsÜS) ja Perekonnaseadus (PKS) on kõigil maakohututel samad, ent nende kasutatavus varieerub kohtute lõikes üpris palju.



Joonis 5: Seaduste kasutatavus maakohutute lõikes.

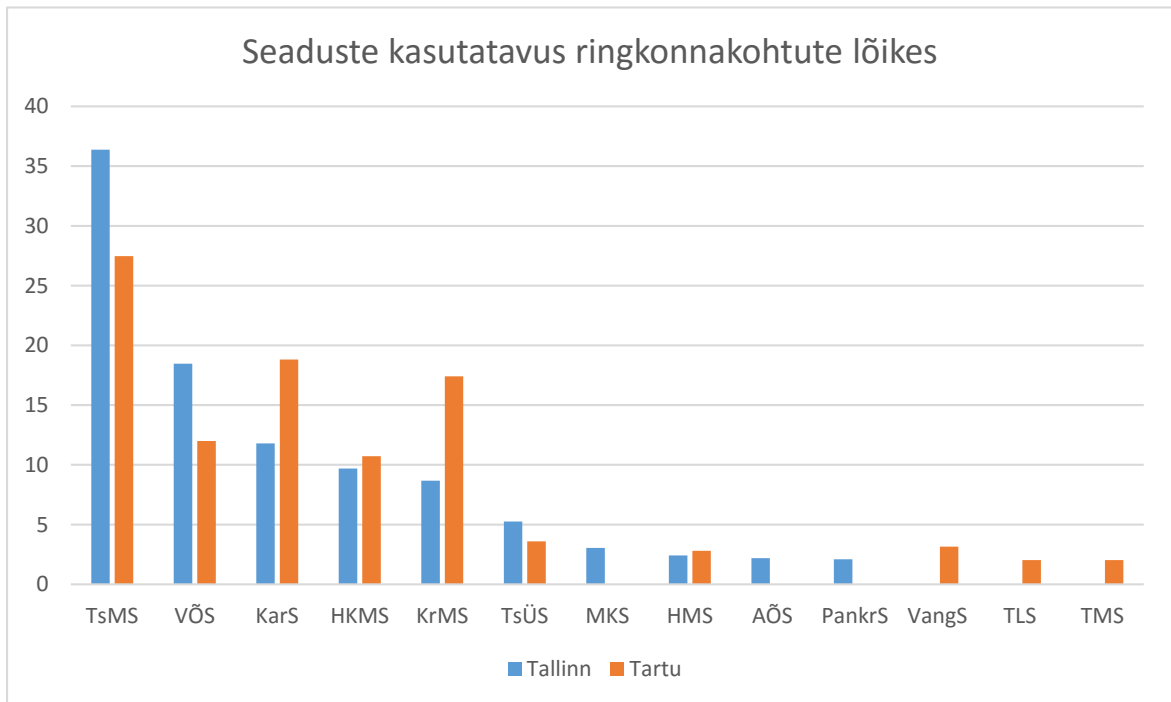
Joonisel 6 on kujutatud sama informatsioon halduskohtute kohta. Ka siin on näha üsna suuri erinevusi kohtute töös kasutatavate seaduste osakaaludes. Rohkem kui kümne protsendilised erinevused ilmnevad Halduskohtumenetluse seadustiku (HKMS) ning Maksukorralduse seaduse (MKS) kasutamises. Esimest kasutatakse märgatavalt rohkem Tartus ning teist Tallinnas. Samuti kasutab Tartu Halduskohus Vangistusseadust (VangS) oma töös umbes kolm korda rohkem kui Tallinna Halduskohus. Ka nende kahe kohtu võrdluses langevad enamik esikümne seadustest kokku, ainsad seadused, mis kokku ei lange, on Tallinnas Riigihangete seadus (RHS) ja Tartus Riigi õigusabi seadus (RÕS).



Joonis 6: Seaduste kasutatavus halduskohtute lõikes.

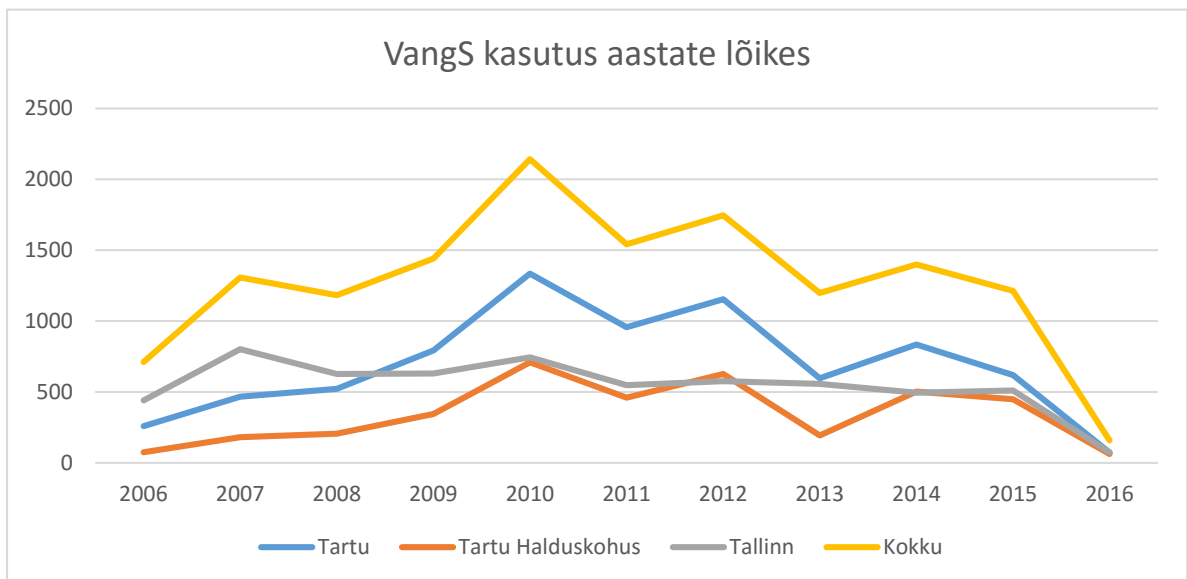
Eesti kohtute veebilehel on kirjeldatud teise astme kohtuid nii: „Ringkonnakohtud vaatavad teise astme kohtuna apellatsiooni või määruskaebuse korras läbi maa- ning halduskohtute lahendeid nende peale esitatud apellatsioon- või määruskaebuste alusel“ [22]. Selle järgi võib eeldada, et nendes kohtutes kasutatakse kõige rohkem ka eelmistel diagrammidel palju kasutatud seaduseid. Joonise 7 järgi on näha, et kolm enimkasutatavat seadust on pärit maakohtute edetabelist, s.t. ringkonnakohtud tegelevad rohkem maakohtutest kui halduskohtutest edasikaevatud juhtumitega. See on ka loogiline, kuna maakohtu lahendeid on minu käsitletavates lahendites umbes kolmteist ja pool korda rohkem kui halduskohtu lahendeid. Erinevused kasutatavate seaduste seas on suured: Tallinnas mõistetakse kohut enam Tsiviilkohtumenetluse seadustiku (TsMS), Võlaõigusseaduse (VÕS), Maksukorralduse seaduse (MKS), Asjaõigusseaduse (AÕS) ja Pankrotiseaduse (PankrS) põhjal, Tartus aga Karistusseadustiku (KarS), Kriminaalmenetluse seadustiku (KrMS), Vangistusseaduse (VangS), Töölepingu seaduse (TLS) ja Täitemenetluse seadustiku (TMS) põhjal.





Joonis 7: Seaduste kasutatavus ringkonnakohtute lõikes.

Vangistusseaduse suur kasutus Tartu kohtutes tuleneb Tartu vangla kinnipeetavate suurest kaebuste hulgast. On kaebusi, mis on esitatud põhjusega ja saavad rahuldatud [23], kuid need on vaid väike protsent kaebuste koguhulgast [24, 25]. Kuigi joonisel 8 esitatud statistika ei lange numbrite osas kokku Nils Niitra artiklis [25] välja toodud statistikaga, on 2014. aasta Vangistusseaduse kasutuse kasvu tõus siiski märgatav. Samas on näha, et probleem oli palju suurem 2010. aastal. 2016. aasta madal kasutus tuleneb sellest, et andmed on võetud aprilli seisuga. Tallinna Haldus- ja Ringkonnakohtus on kaebuste arv aastate lõikes olnud üsna stabiilne, pigem isegi langevas trendis.

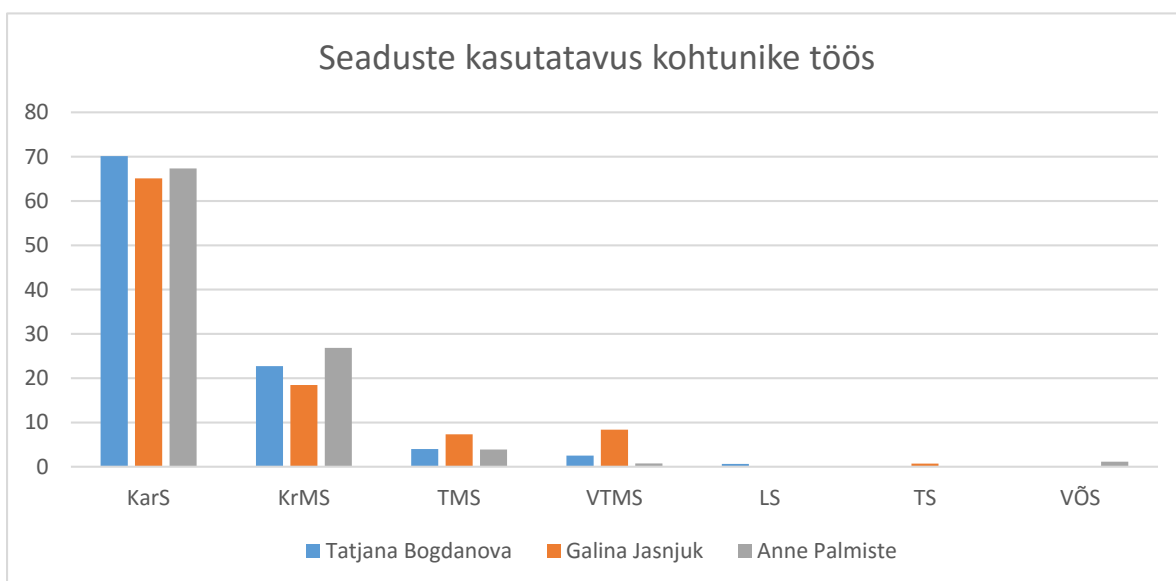


Joonis 8: Vangistusseaduse kasutus aastate lõikes.

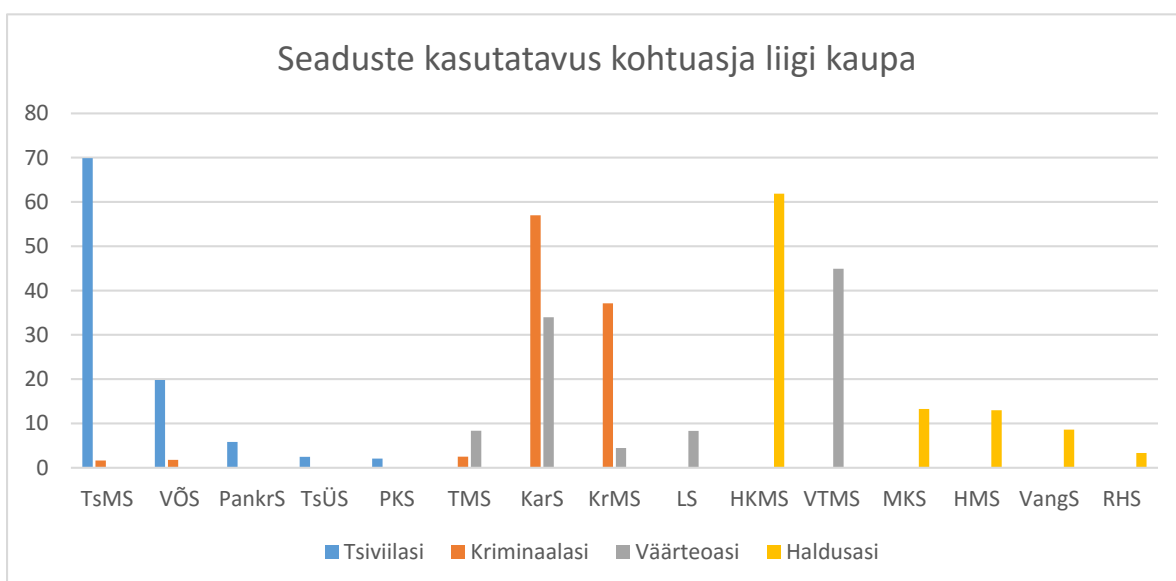
Riigikohtu lahendite kohta sarnast statistikat teha ei saa, kuna süntaksianalüsaator ei suutnud Riigikohtu lahenditest kohtu infot piisava usaldusväärsusega välja lugeda.

Joonis 9 kujutab kolme kõige produktiivsema kohtuniku seaduste kasutamist oma töös. Kõik nad on maakohtu kohtunikud, üks Harju, kaks Viru Maakohtus.

Andmed on esitatud samal viisil nagu ülal kohtute kohta. Võeti viis enim kasutatud seadust ning esitati protsentuaalselt nendele seadustele viitamise arv kogu viie seaduse viidete hulgast. On näha, et kõigi kolme puhul on tegu kriminaalasjadega tegelevate kohtunikega, kuna maakohtutes kõige populaarsemat Tsiviilkohtumenetluse seadustikku isegi ei esine tabelis. See-eest Karistusseadustiku (KarS) ja Kriminaalmenetluse seadustikku (KrMS) kasutavad nad oma töös üsna võrdselt.

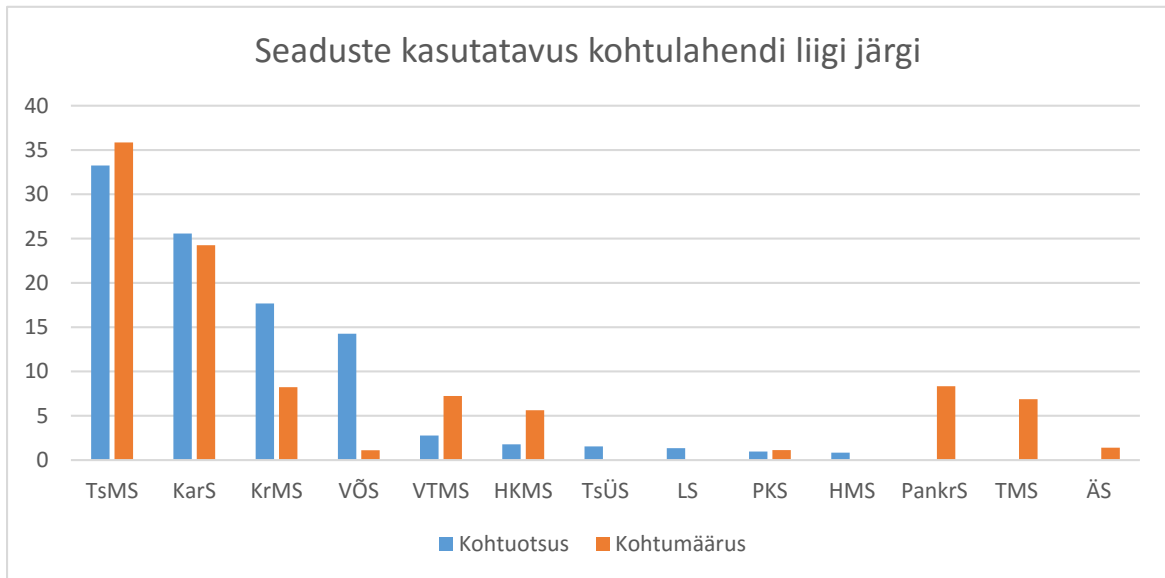


Joonis 9: Seaduste kasutatavus kohtunike töös.



Joonis 10: Seaduste kasutatavus kohtuasja liigi kaupa.

Lisatud on ka diagrammid seaduste kasutatavuse kohta kohtuasja ning kohtulahendi tüübi kaupa (vt. joonised 10 ja 11). Skaalad on esitatud protsentides vastavalt esiviisiku ja esikümne kasutatavuse koguhulgast. Kasutatavuses kohtuasja liigi kaupa midagi erilist ei esine, seadused üldiselt ei kattu ja iga liigi põhiseadused on selgelt välja eristunud. Kasutatavuses kohtulahendi liigi kaupa on aga näha, et näiteks Võlaõigusseaduse (VÕS) järgi väljastatakse kordades rohkem kohtuotsuseid kui –määruseid, kuid Pankrotiseaduse (PankrS), Täitemenetluse seadustiku (TMS) ja Äriseadustiku (ÄS) järgi põhiliselt kohtumääruseid.



Joonis 11: Seaduste kasutatavus kohtulahendi liigi järgi.

## 5. Kokkuvõte

Bakalaureusetöö raames valmistatud programm ning andmebaas osutusid suhteliselt edukateks, nagu näitavad peatükis 4 sooritatud päringud. Töö annab hea aluse parema seaduste ja kohtulahendite otsingumootori loomiseks, olgugi et vajab veel suurt lisapanust.

Valminud „ämblik“ töötas vigadeta ning kõik vajalikud failid said alla laaditud. Kahjuks võtab nii suure andmehulga allalaadimine kaua aega ning edaspidi tuleks seda kindlasti teha mõnes suurema jõudlusega statsionaarses masinas.

Seda sama „ämblikut“ tuleb edaspidi laiendada ka andmebaasile, kuhu on salvestatud kohtulahendid aastatest 2001-2006 ning varasemalt eksisteerinud Riigikohtu lahendite baas tuleb selle töö käigus valminud andmebaasiga ühendada. Nii saaks üle ka probleemist, kus Riigikohtu lahendite põhjal ei saanud statistikat teha, kuna süntaksianalüüs ei olnud piisavalt hea.

Süntaksianalüsaator oli edukas, arvestades kui suured erinevused kohtulahendite vormistamises tegelikult on. See, et teksti seest leitud informatsioon on puudujääke, oli oodatav ning see ei pärssinud päringute tegemise võimalikkust. Küll aga võib töö põhjal järeldada, et Riigi Teataja otsingutulemustel leitav teave on tõene ning see on mõistlik „ämbliku“ käitamisel koos failiga ära salvestada. Üks edasistest võimalustest oleks käitada „ämblik“ uuesti Riigi Teataja otsingulehe peal nii, et lahendite kohta leiduv info salvestatakse paralleelselt tekstist leitunga. Seejärel saaks tekkinud vastuolusid võrrelda ning kasvõi käsitsi parandada, eeldusel, et neid ei ole liiga palju.

Süntaksianalüsaatorit tuleb täiendada ka PDFMineri seisukohast. Praegu jäi analüüsimatele üle 20 tuhande faili, kuna neile ei olnud võimalik lihtsasti ligi pääseda. Üks võimalikest variantidest, millega saaks selliseid PDFe lugeda, oleks kasutada optilist tähetuvastust (*Optical Character Recognition*), s.t. pildituvastussüsteemi, mis suudaks eristada tekstis leiduvaid tähti üksteisest ning need tekstiks vormindada.

Seaduste osas tuleb töö edasises faasis laadida alla ka kõikide saadavalolevate seaduste varasemad redaktsioonid, et siis andmebaasis olevaid kohtulahendeid võrrelda ka ajakohaste seadustega. See on aeganõudev protsess, ent nii saaks teha statistikat ka selle kohta, kuidas on seaduste ületootmise hulk ajas muutunud.

Tulevikus on plaanis ühendada töös loodud andmebaas juba olemasoleva rakendusega [26], mis praegu näitab lingitud kujul seaduseid ja Riigikohtu lahendeid.

Andmebaasil tehtud päringud näitasid selgelt, et Eestis on väga palju seaduseid, mille kohta arutelu puudub, s.t. kohtulahendites ei ole neid kordagi mainitud või siis on seda tehtud üksikutel kordadel. Kas ja kuidas see puutub seaduste ületootmisesse, ei ole minu öelda, kuid annab ehk õigusega tegelevatele inimesele mõtteainet.

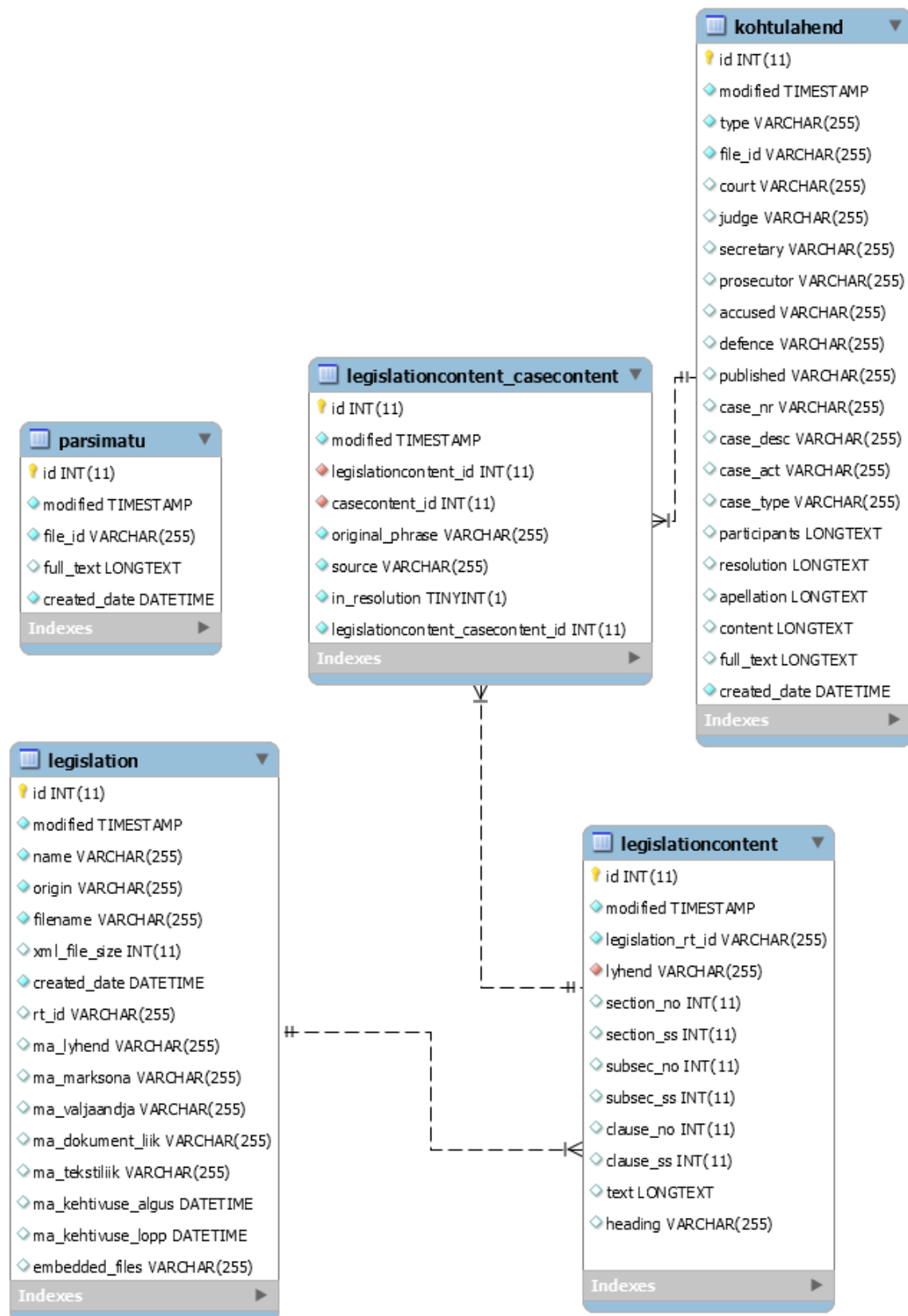
## 6. Kasutatud materjalid

- [1] Registrate ja Infosüsteemide Keskuse koduleht. <http://www.rik.ee/et/elektrooniline-riigi-teataja> (viimane külastus 11.05.2016)
- [2] Riigi Teatajat tutvustav veebileht. <https://www.riigiteataja.ee/abiLeht.html?id=1> (viimane külastus 11.05.2016)
- [3] Scrapy raamistikku tutvustav veebileht. <http://doc.scrapy.org/en/1.0/intro/overview.html> (viimane külastus 11.05.2016)
- [4] Scrapy raamistiku failide allalaadimist tutvustav veebileht. <http://doc.scrapy.org/en/latest/topics/media-pipeline.html> (viimane külastus 11.05.2016)
- [5] Tööriista PDFMiner koduleht. <http://euske.github.io/pdfminer/index.html> (viimane külastus 11.05.2016)
- [6] Tööriista peewee koduleht. <http://docs.peewee-orm.com/en/latest/> (viimane külastus 11.05.2016)
- [7] MySQL tarkvara tutvustav veebileht. <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html> (viimane külastus 11.05.2016)
- [8] Riigikohtu koduleht. <http://www.riigikohus.ee/> (viimane külastus 11.05.2016)
- [9] Riigi Teataja lühenditega seaduste <https://www.riigiteataja.ee/lyhendid.html> (viimane külastus 11.05.2016)
- [10] Riigikohtu lahendite otsingumootor. <http://www.nc.ee/?id=11> (viimane külastus 11.05.2016)
- [11] Riigi Teataja kohtulahendite otsingumootor. [https://www.riigiteataja.ee/kohtulahendid/koik\\_menetlused.html](https://www.riigiteataja.ee/kohtulahendid/koik_menetlused.html) (viimane külastus 11.05.2016)
- [12] Riigi Teataja enne 2006. aastat ilmunud kohtulahendite otsingumootor. [https://www.riigiteataja.ee/kohtuteave/kohtulahendite\\_otsing/haldusajad.html](https://www.riigiteataja.ee/kohtuteave/kohtulahendite_otsing/haldusajad.html) (viimane külastus 11.05.2016)
- [13] Scrapy raamistiku kasutamist tutvustav veebileht. <http://doc.scrapy.org/en/latest/intro/tutorial.html> (viimane külastus 11.05.2016)
- [14] Relatsiooniliste andmebaaside tarkvara kasutuse edetabelit vahendav veebileht. <http://db-engines.com/en/ranking/relational+dbms> (viimane külastus 11.05.2016)
- [15] MySQL tarkvara SELECT võtmesõna toimimist tutvustav veebileht. <http://dev.mysql.com/doc/refman/5.7/en/select.html> (viimane külastus 11.05.2016)
- [16] MySQL tarkvara JOIN võtmesõna toimimist tutvustav veebileht. <http://dev.mysql.com/doc/refman/5.7/en/join.html> (viimane külastus 11.05.2016)
- [17] MySQL tarkvara IN võtmesõna toimimist tutvustav veebileht. <http://dev.mysql.com/doc/refman/5.7/en/any-in-some-subqueries.html> (viimane külastus 11.05.2016)
- [18] MySQL tarkvara COUNT funktsiooni toimimist tutvustav veebileht. <http://dev.mysql.com/doc/refman/5.7/en/counting-rows.html> (viimane külastus 11.05.2016)
- [19] MySQL tarkvara LIKE võtmesõna toimimist tutvustav veebileht. <http://dev.mysql.com/doc/refman/5.7/en/pattern-matching.html> (viimane külastus 11.05.2016)

- [20] Tuulik, M.-E. (2015). *Seaduste ületootmine tuleb peatada*. Justiitsministeeriumi kodulehekül, <http://www.just.ee/et/uudised/seaduste-uletootmine-tuleb-peatada> (viimane külastus 11.05.2016)
- [21] Riigi Teataja vääртеomenetluste otsingumootor. <https://www.riigiteataja.ee/kohtulahendid/vaarteomenetlus.html> (viimane külastus 11.05.2016)
- [22] Eesti kohtute veebilehe ringkonnakohtute tutvustus. <http://www.kohus.ee/et/eesti-kohtususteem/ringkonnakohtud> (viimane külastus 11.05.2016)
- [23] Rannamäe, K. (2007). Tartu vangid kaebavad kõva muusika pärast. *Delfi*, 11. detsember. <http://www.delfi.ee/news/paevauudised/eesti/tartu-vangid-kaebavad-kovamuusika-parast?id=17680684> (viimane külastus 11.05.2016)
- [24] Filippov, J. (2009). Vangide kaebused: muna on toores, tatrapudrus elas tõuk! *Delfi*, 19. veebruar. <http://ekspress.delfi.ee/kuum/vangide-kaebused-muna-on-toores-tatrapudrus-elas-touk?id=27686007> (viimane külastus 11.05.2016)
- [25] Niitra, N. (2014). Vangide kaebused ummistavad kohut. *Postimees*, 11. detsember. <http://pluss.postimees.ee/3022203/vangide-kaebused-ummistavad-kohut> (viimane külastus 11.05.2016)
- [26] Otsingumootor, milles saab päringuid teha seaduste ja Riigikohtu lahendite seas. <https://ideelabor.ee/dev/law-n-court/index.php> (viimane külastus 11.05.2016)

# Lisad

## I. Loodud andmebaasi mudel



## II. Kasutamata seaduste loetelu

Selles lisas on välja toodud seadused, millele ei leidu andmebaasis ühtegi viidet. Kõik siin väljatoodud seadused on leitavad Riigi Teataja kodulehel [9].

1. BS - Biotsiidiseadus
2. EAFS - Eesti Arengufondi seadus
3. EKodAS - Euroopa Parlamendi ja nõukogu määruse (EL) nr 211/2011 „Kodanikualgatause kohta” rakendamise seadus
4. ELKTS - Euroopa Liidu liikmesriigist ebaseaduslikult väljaviidud kultuuriväärtuste tagastamise seadus
5. ELS - Eesti lipu seadus
6. EMHÜS - Euroopa Liidu Nõukogu määruse (EMÜ) nr 2137/85 «Euroopa majandushuviühingu kohta» rakendamise seadus
7. ERRS - Eesti Rahvusraamatukogu seadus
8. ErSS - Erakorralise seisukorra seadus
9. ERTS - Ekspordi riikliku tagamise seadus
10. ETKRS - Euroopa Parlamendi ja nõukogu määruse (EÜ) nr 1082/2006 «Euroopa territoriaalse koostöö rühmituse (ETKR) kohta» rakendamise seadus
11. FELS - Finantskriisi ennetamise ja lahendamise seadus
12. GMMOKS - Geneetiliselt muundatud mikroorganismide suletud keskkonnas kasutamise seadus
13. GMOVS - Geneetiliselt muundatud organismide keskkonda viimise seadus
14. HLÜS - Hoiu-laenuühistu seadus
15. HÖNTE - Hea õigusloome ja normitehnika eeskiri
16. HuviKS - Huvikooli seadus
17. IGUS - Inimgeeniuringute seadus
18. KAVS - Krediidandjate ja -vahendajate seadus
19. KeSS - Keskkonnaseire seadus
20. KooS - Kooseluseadus
21. KOÜS - Kohaliku omavalitsuse üksuste ühinemise soodustamise seadus
22. KTTS - Kunstiteoste tellimise seadus
23. KultVS - Kultuuriväärtuste väljaveo, ekspordi ja sisseveo seadus
24. KVEKS - Kunstliku viljastamise ja embrüokaitse seadus
25. MPõS - Mahepõllumajanduse seadus
26. MTKS - Mikrolülituse topoloogia kaitse seadus
27. MTöös - Meretöö seadus
28. MTVS - Maksualase teabevahetuse seadus
29. MuuS - Muuseumiseadus
30. MVS - Majandusvööndi seadus
31. NTS - Noorsootöö seadus
32. RaHS - Rahvahääletuse seadus
33. RiKS - Riigikaitse seadus
34. RKESS - Rakkude, kudede ja elundite hankimise, käitlemise ja siirdamise seadus
35. RKPSS - Riigi kultuuripreemiate ja kultuuristipendiumide seadus
36. RKSKS - Riigikaitseliste sundkoormiste seadus
37. RSanS - Rahvusvahelise sanktsiooni seadus
38. RTsMS - Rahvusvahelisel tsiviilmissioonil osalemise seadus
39. SCEPS - Euroopa Liidu Nõukogu määruse (EÜ) nr 1435/2003 «Euroopa ühistu (SCE) põhikirja kohta» rakendamise seadus
40. SeOS - Seadme ohutuse seadus



41. SEPS - Euroopa Liidu Nõukogu määruse (EÜ) nr 2157/2001 «Euroopa äriühingu (SE) põhikirja kohta» rakendamise seadus
42. STS2004\_2006 - Perioodi 2004–2006 struktuuritoetuse seadus
43. STS2014\_2020 - Perioodi 2014–2020 struktuuritoetuse seadus
44. SÜS - Sotsiaalseadustiku üldosa seadus
45. SVS - Seadus süümevande andmise korra kohta
46. TAS - Eesti Teaduste Akadeemia seadus
47. TFS - Tagatisfondi seadus
48. THS - Tööstusheite seadus
49. TTÜKS - Tallinna Tehnikaülikooli seadus
50. VereS - Vereseadus
51. VSaarS - Püeiasustusega väikesaarte seadus
52. VVR - Vabariigi Valitsuse reglement

### **III. Valminud programmi kood**

Programmi koodi avalik osa on tööle lisatud zip-failina.

#### IV. Litsents

**Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks**

Mina, **Brait Õispuu**,  
(*autori nimi*)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose  
**Eesti Vabariigi seaduste ja kohtulahendite andmebaas ning ristviitamine**,  
(*lõputöö pealkiri*)

mille juhendajad on Mart Randala ja Mare Koit,  
(*juhendaja nimi*)

- 1.1. reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
- 1.2. üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, **16.05.2016**