TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Informaatika eriala

Katrin Jets

Tekstikorpusest kollokatsioonide tuvastaja

Magistritöö (30 EAP)

Juhendaja: Kadri Muischnek

Autor: ..... „ “ 2013
Juhendaja: ..... „ “ 2013

Lubada kaitsmisele
Professor .... „ „ 2013

TARTU 2013

Sisukord

Sissejuhatus

Arvutilingvistilisetes uurimustes on juba ammu järeldusele jõutud, et loomuliku keele automaatsel töötlemisel ei piisa sõnade vaatlemisest ühekaupa. Kuna üksiksõnade tähendus võib olla hoopis midagi muud kui neist moodustuva ühendi tähendus, siis nt masintõlke puhul võib mingi teksti kõigi sõnade ükshaaval tõlkimisel terviku tähendus oluliselt kannatada saada. Niisiis peaks arvestama tekstis ette tulevate mitmest sõnast koosnevate üksustega, need tekstist ära tundma ja edasi töötlema kui semantilist tervikut.

Et selliste ühenditega uue keeletarkvara loomisel arvestada, tuleb need eelnevalt tekstist tuvastada. Kuna suurte tekstikorpuste puhul on mõistlik vähemalt esialgne potentsiaalsete sõnaühendite tuvastamine (mille tulemusi pärast vajadusel käsitsi korrigeerida) teha automatiseeritult, tekib iseenesest mõistetavalt vajadus vastava programmi järele. Sõnaühendeid on automaatsel teel võimalik tekstist üles leida kasutades statistilisi meetodeid. Defineerides otsitavaid püsiühendeid, kui sõnakomplekte, mille komponendid tulevad tekstides üksteise läheduses sagedamamini ette, kui võiks järeldada nende eraldi esinemise sageduste põhjal, saame mõiste kollokatsioon.

Käesoleva uurimuse eesmärgiks ongi välja töötada selline programm, mis leiaks tekstikorpusest etteantud sõna järgi, kõik sõnad, mis moodustavad sellega kollokatsiooni. Selle töö valmimise ajal sai riikliku programmi "Eesti Keeletehnoloogia" projekti "Eesti keele koondkorpuse esituse ja kasutusvõimaluste arendamine" (Eesti ...) raames loodud ja projekti "Vahendid teksti mitmekihiliseks märgendamiseks (rakendatuna Koondkorpusele)" (Vahendid ...) raames täiendatud kollokatsioonide leidmise veebiinfosüsteem (2010). Kuigi nimetatud rakendus täidab sarnast otstarvet – sõna kollokaatide leidmist tekstikorpusest, on käesolevas kirjeldatava programmi eripäraks mistahes korpuse kasutamise võimalikkus. Seetõttu on antud kollokatsioonide tuvastaja suunatud eelkõige keeleteaduse magistrantidele ja doktorantidele, kes soovivad töödelda omaenda poolt kogutud tekstikorpuseid. Antud programmiga on võimalik uurida mistahes korpust, kui see eelnevalt morfoloogiliselt ühestades viia sobivale kujule. Selleks saab kasutada Heiki-Jaan Kaalepi poolt välja töötatud morfoloogiaanalüsaatorit t3mesta (Kaalep 1998, Kaalep, Vaino 1998).

Peatükk 1

Kollokatsioonid

1.1 Püsiühendid

Püsiühenditeks (*multiword expressions*) nimetatakse kahest või enamast sõnast koosnevaid ühendeid, mida tekstis sageli koos kasutatakse ja millel on omaette tähendus. Püsiühenditeks loetakse idioome (nt sääsest elevanti tegema), ühend-ja väljendverbe (nt maha jääma, hea seisma), mitmest sõnast koosnevaid pärisnimesid (nt San Fransisco) ja lahku kirjutatavaid liitsõnu (nt ingl k *car park*). Püsiühendi tähendus võib olla ühendit moodustavate sõnade tähendustest otseselt aimatav (nt kokku põrkama), kaudselt tuletatav (nt keelt kastma) või komponentide tähendustest täiesti sõltumatu (nt pika ninaga jääma). Mida edukamalt võib ühendi tähenduse üle otsustada selle osade tähenduste järgi, seda läbipaistvamaks seda loetakse.

Arvutilingvistikas kasutatakse püsiühendi mõistet sageli koos terminiga kollokatsioon (*collocation*). Kuigi enamasti omistatakse kollokatsioonile mõnevõrra kitsam tähendus, on neid mõisteid tihtipeale üpris keeruline eristada. Erinevad keeleteadlased jagavad püsiühendeid mitmesuguste kriteeriumite põhjal erinevalt ja kasutavad seega ka erinevaid definitsioone.

1.2 Kollokatsiooni mõiste

Nagu mainitud, on kollokatsiooni mõiste definitsioone kirjanduses mitmeid. Eesti keele käsiraamatus on kollokatsiooni mõiste kohta selline sissekanne: „Kollokatsioon on sõna tähendusest sõltuv kalduvus esineda koos kindlate teiste sõnadega. Mõned sõnad kollotseeruvad omavahel, teised mitte. Kõige selgemini väljenduvad kollokatsioonid selles,

missuguse aluse, sihitise, määruse saab enda juurde valida öeldiseks olev verb. Nt määgib puhul saab aluseks olla lammas, mitte kala või pliiats, määruseks võiks olla karjamaal, laudas, aedikus, mitte aga põhjapoolusel või vanglas. Omadussõna muhklik kollotseerub mitme nimisõnaga, nt muhklik pärnapuu, kadakakepp, põrand, laud, muhklikud käed, sõrmed, liigesed, kuid ikkagi arvestades tähenduspiirangut, et see miski saab olla 'kühmuline, konarlik' (ei sobi abstraktmõisteid või elusolendeid väljendavad sõnad, nagu sõprus, peaminister).“(M.Erelt, T. Erelt, K. Ross 2009)

Kadri Jaanits selgitab oma magistritöös „Leksikaalsetest kollokatsioonidest Soome ja Eesti keeles“ (2004) terminit järgmiselt: „Kollokatsioon on mingi sõnade rühma igapäevases kasutuses välja kujunenud sõnade loomulik järjestus. Väljend, millest on võimalik aru saada sõnade üksiktähenduse kaudu; eeldades, et kombinatsiooni kõigil sõnadel on iseseisev tähendus (nt päike tõuseb, küsimust esitada, mõistlik mõte).“ Jaanits rõhutab oma töös kollokatsioonide olulisust võõrkeeleõppes ja eristab neid idioomidest ja metafooridest läbipaistvuse poolest: kollokatsioonide puhul on võimalik komponentide tähenduse järgi mõista ühendi tähendust, teiste puhul mitte. Samuti ei loe ta kollokaatideks liitsõnu, kuna liitsõnade osade vahel on püsivam side, ega vabu sõnaühendeid (nt algus ja lõpp), mis võivad küll olla väga sagedasti koos kasutatavad, kuid millel ei ole eraldi väljakujunenud tähendust.

Heiki-Jaan Kaalepi ja Kadri Muischneki artiklis „Püsiühendite leidmine teksti abil“ (2002) selgitatakse mõistet järgmiselt: “Kollokatsioon on sõnaühend, mis on defineeritud selle järgi, et teda moodustavad sõnad esinevad tekstides koos sagedamini, kui võiks eeldada nende eraldi esinemise sagedustest. Kollokatsioonid võivad olla väga erinevad nii neid moodustavate sõnade arvu poolest kui ka nende sõnade süntaktiliste funktsioonide ja omavaheliste seoste poolest. Nendeks võivad olla nii idioomid (nt hambasse puhuma), mida sõnaraamatud esitavad põhjalikult, kuid mida tekstides harva esineb; ühend- ja väljendverbid, mida samuti sõnaraamatutes tüüpiliselt esitatakse (üle saama, õppust võtma); mitmesugused nimisõnafraasid (nt rohelised mehikesed).“ Selline definitsioon on palju üldisem ja lubab ühendeid kollokatsioonideks määrata sisuliselt ainult komponentide tekstis esinemise sageduste põhjal. Sõnade muude omavaheliste seoste, arvu või tüüpide järgi saab neid küll veel omakorda rühmadeks jagada, kuid kõiki neid loetakse siiski kollokatsioonideks.

Stefan Evert lahkab oma töös „*Corpora and Collocations*“ (2007) püsiühendite definitsioonide rohkuse probleemi üsna põhjalikult. Ta kirjeldab kolme põhilist lähenemist: empiirilist, fraseoloogilist ja arvutilingvistilist. Empiiriline käsitlus näeb kollokatsioone kui sõnade tähendust ja kasutust iseloomustavat vahendit. See tähendab, et sõna kollokaatide (*collocate*, antud sõnaga ühendit moodustavate sõnade) põhjal on teatud määral võimalik kirjeldada sõna ennast. Fraseoloogias vaadeldakse kollokatsioone, kui poolkompositsioonilisi ühendeid, nii et ühe komponendi kollokaatideks on teatud piiratud hulk sõnu, mida antud sõnaga koos kasutatakse, et edastada mingit konkreetset mõtet. See klapib Jaanitsa toodud definitsiooniga. Arvutilingvistikas mõistetakse kollokatsioone üldisemalt, pidades nendeks kõiki sõnakombinatsioone, mille komponentide vahel on mingi konkreetne semantiline või süntaktiline seos ja mida peaks seetõttu keele töötlemisel eraldi käsitlema. Evert väidab ka, et kõik erinevad kollokatsioonide definitsioonid baseeruvad siiski ühisel ettekujutusel, et teatud sõnad kipuvad loomulikus keeles esinema üksteise läheduses.

Antud töös käsitletakse kollokatsioone Kaalepi ja Muischneki definitsiooni kohaselt, mis on kooskõlas Eesti keele käsiraamatu kirjelduse ja Everti tutvustatud arvutilingvistilise lähenemisega. Töös ei arvestata ühendit moodustavate sõnade süntaktilisi funktsioone ega ka tähendusi, kollokatsioone leitakse sõnade korpuses esinemise sageduste põhjal. Kollokatsioonide tuvastamisel on oluliseks kitsenduseks komponentide arv, milleks on valitud kaks. Kaks on minimaalne kollokatsiooni moodustatavate sõnade arv, seejuures aga täiesti piisav sõnadevaheliste seoste leidmiseks. Niisiis vaadeldakse korpuses igat tüüpi sõnu ja sõnapaaride kollokatsioonideks määramisel lähtutakse vaid sõnade tekstis esinemise sagedustest.

1.3 Koosesinemise tüübid

Sõnade koosesinemise tüüpe on kolm: pindmine, tekstiline ja süntaktiline. Nii jaotatakse neid vastavalt sellele, millises kontekstis toimub kollokatsioonide tuvastamise protsess. St, kuidas leitakse tekstist sõnaühendid, mille seast kollokatsioonid välja sõeluda.

1.3.1 Pindmine koosesinemine

Pindmine koosesinemine tähendab seda, et kollokatsioonide leidmisel vaadeldakse sõnu mingis eelnevalt kindlaks määratud vahemikus ehk „aknas“. Aken (*collocational span*) on maksimaalne potentsiaalse kollokatsiooni komponentide vahele jäävate sõnade arv. Näiteks valides lauses „ta läks, istus pingile ja naeratas“ aknaks neli, leiaksime järgmised kollokatsiooni kandidaadid: „ta läks“, „ta istus“, „ta pingile“, „ta ja“, „läks istus“, „läks pingile“, „läks ja“, „läks naeratas“, „istus pingile“, „istus ja“, „istus naeratas“, „pingile ja“, „pingile naeratas“ ja „ja naeratas“. Aken ei pruugi olla sümmeetriline, see tähendab ta võib omada eraldi väärtusi vasakpoolse ja parempoolse vahemiku jaoks vaadeldavast sõnast. Näiteks kui aknaks valida vasakule poole kaks ja paremale poole üks, siis saaksime eelnevas näites sõna „ta“ korral kollokaadiks vaid sõna „läks“, sõna „pingile“ puhul saaksime aga tulemuseks sõnad „läks“, „istus“ ning „ja“.

1.3.2 Tekstiline koosesinemine

Kui kollokatsioone tuvastatakse mingis tekstiüksuses, näiteks lauses või lõigus, on tegemist tekstilise koosesinemisega. Näiteks kui vaadeldakse püsiühendite esinemisi lauses, tähendab see seda, et kõik antud lauses esinevad sõnad võivad teoreetiliselt moodustada kollokatsiooni teise samas lauses esineva sõnaga. Näiteks lauses „See on kõik puhas juhus.“ võib kollokatsioon moodustuda sõnadest „see on“, „on kõik“, „puhas juhus“, „see juhus“ jne Üldse on antud lauses bigramme (kahest komponendist koosnevaid üksusi) kokku kümme, millest teoreetiliselt võivad kõik osutuda kollokatsioonideks. Tekstilise koosesinemise puhul võib vaadeldavaks tekstiüksuseks olla ka osalause. Sellisel juhul võivad kollokatsioonid tekkida vaid osalause piirides, nt lauses „etendus, mida olime oodanud, jäi ära“ oleks lähemalt uuritavateks bigrammideks „mida olime“, „mida oodanud“, „olime oodanud“ ning „jäi ära“.

1.3.3 Süntaktiline koosesinemine

Süntaktiliseks koosesinemiseks loetakse seda, kui antud sõnad on mingis süntaktilises suhtes. Näiteks võib püsiühendik olla mingi nimisõnafraas. Sel juhul on üks kollokatsiooni komponent nimisõna ja teine selle täiend (nt kuum tee). Süntaktiliselt võivad sõnad koos esineda ka näiteks ühendverbina, kus üheks komponendiks on tegusõna ja teiseks abimäärsõna (nt ette ütlema) või verbi ja selle subjektina (nt palli lööma). Nt lauses „kui mängija tabab punase kuuli asemel musta kuuli, saab ta trahvipunktid“ oleks võimalikud kollokatsioonid „punane kuul“, „must kuul“, kui süntaktiliseks mustriks oleks nimisõnafraas ning „kuuli tabama“ ja „trahvipunkte saama“, kui süntaktiliseks seoseks oleks verb ja subjekt.

Antud töös on kombineeritud pindmist ja tekstilist koosesinemist. Tekstiüksuseks on valitud lause, mis tähendab, et antud sõna kollokaate otsitakse temaga samast lausest. Kuna lauses valitsevad sõnade vahel nii grammatilised kui ka semantilised seosed, siis on see sobiv kollokatsioonide leidmiseks. Veel kasutatakse pindmise koosesinemisele omast akent, milleks antud juhul on valitud 4. St, lause piirides moodustatakse bigramme kõikidest selles olevatest sõnadest tingimusel, et nende vahele ei jääks rohkem kui 4 sõna.

1.4 Kollokatsioonide automaatne tuvastamine

Kollokatsioone on küll võimalik tekstist käsitsi tuvastada ja sel teel saadud tulemused oleksid ilmselgelt ka täpsemad, kuid selline lähenemine on ka väiksemate tekstikorpuste korral väga töömahukas. Seepärast eelistatakse püsiühendeid leida automaatselt arvuti abiga ja niimoodi leitud ühendid, mida on juba hulga vähem, kui esialgses korpuses, vajadusel käsitsi üle kontrollida.

Kollokatsioonide automaatsel tuvastamisel tuleb esmalt leida kõik võimalikud kollokatsiooni kandidaadid ning seejärel teatud eelnevalt välja valitud (statistilisel) meetodil kindlaks teha, millised neist tõepoolest püsiühendi moodustavad. Selliseid meetodeid on väga palju erinevaid, kuid nad kõik kasutavad potentsiaalset kollokatsiooni moodustavate sõnade tekstis

esinemise sagedusi, et otsustada, kas neil sõnadel on tõesti kalduvus üksteise läheduses ette tulla või on tegu vaid juhusega. Selleks on vaja teada nii antud sõnade eraldi esinemise sagedust e. marginaalsagedust (*marginal frequency*) kui ka nende koosesinemise sagedust (*cooccurence frequency*). Lisaks veel valimi mahtu (*sample size*) ehk siis seda, kui palju võimalikke esinemisi kogu teksti peale üldse kokku on, st kui mitmel korral oleks teoreetiliselt selline sõnaühend võinud tekstis ette tulla.

Pindmise koosesinemise puhul on valimiks kõigi selliste sõnakomplektide arv, mille komponentide vahele jääb maksimaalselt nii palju sõnu, kui parajasti aknaks valitud on, ja mis võivad seega püsiühendi moodustada. Koosesinemise sageduseks on siis aga kõikide selliste juhtude arv, kui kõik antud ühendis olevad sõnad paiknevad üksteisest mitte kaugemal, kui aknaks valitud väärtus.

Tekstilise koosesinemise puhul otsitakse kollokatsioone mingist tekstiüksusest. Seega oleks lause puhul valimiks lausete arv, osalause korral osalausete arv jne. Koosesinemise sagedus ütleb mitmes sellises tekstiüksuses tuleb ette, et kõik ühendit moodustavad sõnad seal esindatud on. Marginaalsagedus näitab aga iga sõnakomponendi kohta, mitmes erinevas lauses see esineb.

Süntaktilise koosesinemise puhul võetakse sõnad ühendikandidaatidena vaatluse alla siis, kui nad on antud tekstis omavahel mingis süntaktilises seoses. Seega kõikide selliste sõnaühendite (mille vahel selline seos on olemas) hulk moodustabki valimi. Koosesinemise sagedus ütleb konkreetse sõnakomplekti kohta, mitmel erineval juhul, on vaatluse all oleva seose komponentideks just need sõnad.

Kui kõik vajalikud sagedused on käes, on aeg rakendada neid välja valitud valemis, misjärel on võimalik kirjeldada kollokatsiooni komponentide vahelise seose tugevust. Selleks on mitmeid võimalusi. Lihtsaim, kuid vast ka kõige ebatäpsem viis on arvestada vaid sõnade koosesinemise sagedusi, nii et mida tihedamini ühend tekstis esineb, seda kindlamini on tegu kollokatsiooniga. Teine variant on kasutada lihtsaid seose tugevuse mõõdikuid. Sel juhul võetakse arvesse nii sõnakomplekti koosesinemise sagedusi kui selle komponentide

marginaalsageduste põhjal arvutatud oodatava koosesinemise sagedusi. Viimane näitab mitmel korral võiksid sõnad üksteise lähedusse sattuda juhuslikult, st kui kõik tekstis olevad sõnad suvalisse järjekorda panna, siis kui tihti satuksid vaatluse all olevad sõnad üksteise lähedusse. Teiste sõnadega üritatakse lükata ümber hüpotees, et sõnad esinevad tekstis üksteisest sõltumatult. Kui oodatav ja tegelik sagedus on käes, siis rakendatakse valitud valemit, et jõuda konkreetse seose tugevust iseloomustava arvuni (*association score*). Variantideks on nt vastastikuse informatsiooni funktsioon, t-skoor ja z-skoor. Kolmas ja tõenäoliselt kõige täpsem on statistiline meetod. Kasutusel on nt hii-ruut ja log-tõepära statistikud. Statistikute kasutamiseks on vaja luua iga kollokatsiooni kandidaadi jaoks sagedustabelid oodatud ja tegelike sagedustega. Erinevalt lihtsatest seose tugevuse mõõdikutest on statistilise meetodi puhul nimetatud sagedused põhjalikumad, rohkem kaalu pannakse sõnade eraldi esinemise sagedustele. Kui eelmisel juhul piisas ühest koosesinemise oodatavast ja ühest tegelikust sagedusest sõnapaari kohta, siis statistilise meetodi korral on kumbagi neli: kui mõlemad sõnad esinevad koos, selliste paaride hulk, milles on esindatud vaid üks komponent ja juhud, mil pole kumbagi sõna. Seose tugevuse mõõdikutest räägib lähemalt ka Evert oma eelpool mainitud töös.

1.5 Kollokatsioonide roll keeletöötluses

Peale selle, et kollokatsioonid on eriti lingvistide jaoks huvitav keeleline nähtus, on püsiühendite uurimine kasulik ka mitmes arvutilingvistika valdkonnas. Nt masintõlkes on üheks tõlkeprogrammide arendajatele peavalu valmistavaks probleemiks idioomid. Sõna-sõnalt tõlkimise edukus oleneb idioomi läbipaistvusest, st sellest kuivõrd on idioomi tähendust võimalik mõista teades teda moodustavate sõnade tähendusi. Tõenäoline on aga, et sel moel tõlgitud versioon ei anna originaalteksti mõtet edasi ja võib hoopis palju segadust tekitada. Näiteks töödeldes väljendis “nina alla hõõruma“ sõnu eraldi, saaksime tõenäoliselt tulemuseks otsese füüsilise tegevuse, mis kaasab nina kui näoosa, ülekantud tähendus läheks aga kaduma. Rakendades seda masintõlkes, saaksime tulemuseks ebatäpse ja kummalise lause. Selle vältimiseks saaks idioome ja muid sõnaühendeid, mille sõna kaupa tõlkimine ei ole võimalik, neid tervikuna vaadeldes töödelda. Selleks on need aga kõigepealt vaja tekstist ära tunda.

Ka homonüümide korral, võib sõnade ümbruse uurimisest abi olla. Kui mingi sõnavorm on morfoloogiliselt mitmene, siis võib olla raske otsustada, millist antud juhul mõeldakse. Nt sõna „talle“ on mitmene nelja morfoloogilise tõlgenduse vahel. See võib olla nii „temale“ lühike versioon, mitmuse osastav sõnast „tall“, mis tähistab hoonet või ainsuse omastav või osastav sõnast „tall“, mõeldes noort looma. Inimene mõistab, millist sõna mõeldud on, kas häälduse või konteksti põhjal. Seda võib teha ka arvuti. Kui vaadelda ainult kirjapilti, lähevad küll teise ja kolmanda välte erinevused kaduma, kuid sõna ümbruses esinevad teised sõnad on oluliseks vihjeks vaadeldava sõna tähendusele. Uurides tähenduse kandidaatide kollokaate ja neid antud sõna ümbrusest leides, on võimalik suurema tõenäosusega tähenduse üle otsustada.

Püsiühendid on olulised ka keeleõppe seisukohast, mida selgitab oma magistritöös Kadri Jaanits. Kui emakeelt kõneledes ei pöörata väljendite kasutamisel erilist tähelepanu seda moodustavatele sõnadele ja nende iseseisvatele tähendustele, siis võõrkeelt omandades aga hakkab silma, kui väljakujunenud sõnaühendid ei ole justkui üldse loogilised. Nt väljendid „kellegi käest küsima“ ja „silma torkama“ võivad välismaalastele veidrad tunduda. On äärmiselt oluline teada, milliseid sõnu millistega koos tarvitada, et (võõr)keelt õigesti kasutada. Ka sõnaraamatute koostamisel on otstarbeks töö hõlbustamiseks kasutada automaatset püsiühendite tuvastamist ja töötlust.

1.6 Varem avaldatud püsiühendite teemalisi uurimusi

Sissejuhatuses mainitud kollokatsioonide leidja võimaldab analoogselt käesolevale tööle kasutaja poolt sisestatud sõna järgi leida selle kollokaadid. Veebirakendus kasutab 15 miljonit sõna sisaldavat tasakaalus korpust ja võimaldab valida nelja kollokaatide tuvastamise meetodi vahel: esinemissagedus, log-tõepära, vastastikuse informatsiooni väärtus MI ja minimaalne tundlikkus (*minimal sensitivity*). Otsida saab nii lemmasid kui sõnavorme.

Kaalepi ja Muischneki töös „Püsiühendite leidmine teksti abil“, uuritakse, kuidas töötab n-kohaliste sõnaühendite ekstraheerimise tarkvara SENTA ühend- ja väljendverbide (koondnimetusega fraasiverbide) tuvastamisel tekstikorpusest. Saadud tulemusi võrreldakse

sõnastike põhjal koostatud fraasiverbide andmebaasiga ja leitakse, et statistikal põhinev ja eesti keele iseärasusi mitte arvestav SENTA töötab paremini kui oodatud.

Kristel Uiboaed katsetab oma töös „Statistilised meetodid murdekorpuse ühendverbide tuvastamisel“ (2010) erinevate statistikutega kaheliikmeliste ühendverbide määramist. Ta kasutab selleks nelja statistikut (t-skoor, MI, hii-ruut ja log-tõepära) ja kasutab neid kolme murdekorpuse peal (põhjaeesti, lõunaeesti ja rannikumurde murderühm), et neid omavahel võrrelda ja kindlaks teha, milliseid statistikuid edaspidistes uurimustes mõttekas kasutada oleks. Ta jõuab järeldusele, et kuigi log-tõepära annab läbivalt parimaid tulemusi, oleks efektiivsem kasutada enam kui ühte statistikut.

Püsiühendite automaattöötluse kolmest etapist räägib lähemalt Heiki-Jaan Kaalepi ja Kadri Muischneki artikkel „Eesti keele püsiühendid arvutilingvistikas: miks ja kuidas“ (2009). Nendeks etappideks on: esiteks antud tööski vaadeldav püsiühendite tuvastamine, teiseks leitud püsiühenditest leksikoni koostamine ja viimasena selle põhjal püsiühendite märgendamine tekstis.

Kadri Jaanits selgitab oma töös „Leksikaalsetest kollokatsioonidest soome ja eesti keeles“ kollokatsiooni mõistet ja liike, rõhutab nende olulisust võõrkeele õpetamisel ja esitab soome-eesti kollokatsioonisõnaraamatu, kuhu on koondatud umbes 3600 soomekeelset kollokatsiooni koos eestikeelsete vastetega.

Peatükk 2

Kollokatsioonide tuvastaja

2.1 Programmi ülesehitus

2.1.1 Üldine struktuur

Kollokatsioonide tuvastaja koosneb kahest iseseisvast programmist. Esimene neist vaatab läbi tekstikorpuse ja loob sagedustabelid. Teine kasutab neidsamu tabeleid ja leiab sageduste järgi etteantud sõna olulisemad kollokaadid antud korpuses. St esimene programm *SurfaceCooccurrence.pl* saab sisendiks vähemalt ühe morfoloogilise analüsaatoriga t3mesta (Kaalep 1998, Kaalep, Vaino 1998) ühestatud korpusefaili, leiab iga korpuses esineva sõna jaoks tema võimalikud kollokaadid ja neile vastavad sagedused ning kirjutab saadud info andmebaasi. Teine programm C*ollocationFinder.jar* ühendub samasse andmebaasi ning sinna salvestatud andmete ja kasutaja poolt sisestatud otsingusõna põhjal leiab puuduvad sagedused, (st põhimõtteliselt loob iga kollokaadikandidaadi jaoks sagedustabelid), arvutab nende järgi *log-likelihood* statistikut kasutades seose tugevust iseloomustava suuruse ning kuvab kasutajale kõik leitud kollokaadid koos vastava seose tugevusega.

Sagedustabeleid loov programm *SurfaceCooccurrence.pl* eeldab sisendiks vähemalt üht morfoloogilise ühestajaga t3mesta töödeldud korpusefaili, milles laused on piiratud <s> ja </s> märgenditega, iga sõna asub eraldi real ja sõna järele on lisatud morfoloogiline info (Joonis 1). Info allika ja selle autori kohta jääb märgendite <*ignoreeri*> ja </*ignoreeri*> vahele, selle osa jätab programm vaatluse alt välja. Kõigepealt vaadatakse üks lause haaval läbi kõik korpuse failid. Lauset vaadeldakse sõna kaupa ja iga sõna jaoks leitakse akna, milleks on antud projekti puhul neli, piiridesse jäävad sõnad. St iga sõna puhul arvestatakse nelja temast eespool ja nelja tema järel asuvat sõna. Ainult need aknas asuvad sõnad võivad antud lause põhjal olla vaadeldava sõna kollokaatideks. Kui läbi on vaadatud kõik laused kõikidest failidest, moodustubki andmebaasi nimekiri kõikidest sõnapaaridest ja nende ühes aknas koos esinemise sagedustest.

```
<ignoreeri>
|
autor:
Jaan
Kross
|
tervikteos:
Väljakaevamised
Romaan
|
alaosa:
1
.
|
</ignoreeri>
<s>
Nii    nii+0 //_D_ //
et    et+0 //_J_ //
võisin    või+sin //_V_ sin, //
mõnede    mõni+de //_P_ pl g, //
endiste    endine+te //_A_ pl g, //
kolleegide    kolleeg+de //_S_ pl g, //
juttu    jutt+0 //_S_ sg p, //
isegi    isegi+0 //_D_ //
tõsiselt    tõsiselt+0 //_D_ //
võtta    võt+a //_V_ da, //
,    , //_Z_ //
kui    kui+0 //_D_ //    kui+0 //_J_ //
nad    tema+d //_P_ pl n, //
kinnitasid    kinnita+sid //_V_ sid, //
:    : //_Z_ //
mul    mina+l //_P_ sg ad, //
poleks    ole+ks //_V_ neg ks, //
vist    vist+0 //_D_ //
võimatu    võimatu+0 //_A_ sg n, //
õigusteaduskonna    õigus_teaduskond+0 //_S_ sg g, //
juures    juures+0 //_K_ //
uuesti    uuesti+0 //_D_ //
rakendust    rakendus+t //_S_ sg p, //
leida    leid+a //_V_ da, //
.    . //_Z_ //
</s>
```

Joonis 1. Näide korpusefaili struktuurist.

Päringu tegemiseks mõeldud programm ühendub kõigepealt eelpool kirjeldatud programmi abil loodud andmebaasi ja vastavalt kasutaja poolt sisestatud otsingusõnale ja valitud parameetritele, mis näitavad kas otsingusõna on sõnavorm või lemma ja kas kollokaate vaadeldakse sõnavormide või lemmadena, leiab sealt vajalikud andmed. Edasi arvutatakse leitud sageduste põhjal sisestatud sõna iga võimaliku kollokaadi jaoks log-tõepära funktsiooni kasutades seose tugevuse määr ja kuvatakse kasutajale leitud kollokaadid koos vastava skooriga.

2.1.2 Sagedustabelite koostamine

2.1.2.1 „Akna“ piiride leidmine

Programm vaatab ükshaaval läbi kõik etteantud kataloogis asuvad korpusefailid. Iga faili puhul loetakse korraga mällu tekstiosa kuni märgendini </s>. Seejärel eemaldatakse sellest ebavajalik informatsioon nagu allikate kirjeldus, kirjavahemärgid ja lühendid ning tühjad read, kui neid peaks ette tulema. Järelejäänud tekstiosa jagatakse reavahetuse sümboli järgi sõnadeks. Seejuures jäetakse morfoloogilisest infost järele vaid esimene t3mesta poolt pakutud variant. Saadud sõned, mis sisaldavad nii sõnavormi kui infot selle liigi, tüve, käände jms kohta paigutatakse massiivi. Iga loetud lause puhul luuakse uus massiiv, mille elementideks on seega ühes konkreetses lauses olevad sõnad, ning hiljem kui lause on töödeldud, kaotatakse massiiv jälle ära.

Kui järjekordne lause on sõnadeks jaotatud ja massiivi salvestatud, vaadeldakse saadud massiiv elemendi e. sõna kaupa läbi. Iga uue sõna puhul lisatakse see *hash* tüüpi andmestruktuuri, kui seda seal veel ei ole. Seega moodustub sinna unikaalsete sõnade nimekiri. Iga sõna puhul leitakse ka sõna lemma. Siis uuritakse aknasse jäävaid sõnu. Kuna aknaks sai valitud neli, vaadatakse sõnu, mis asuvad antud sõnast kuni neli sõna ees- või tagapool. Lause alguse ja lõpupool asuvate sõnade puhul täismõõdus akent loomulikult ei esine, sel juhul on lihtsalt vähem kollokaadi kandidaate.

2.1.2.2 Sageduste leidmine

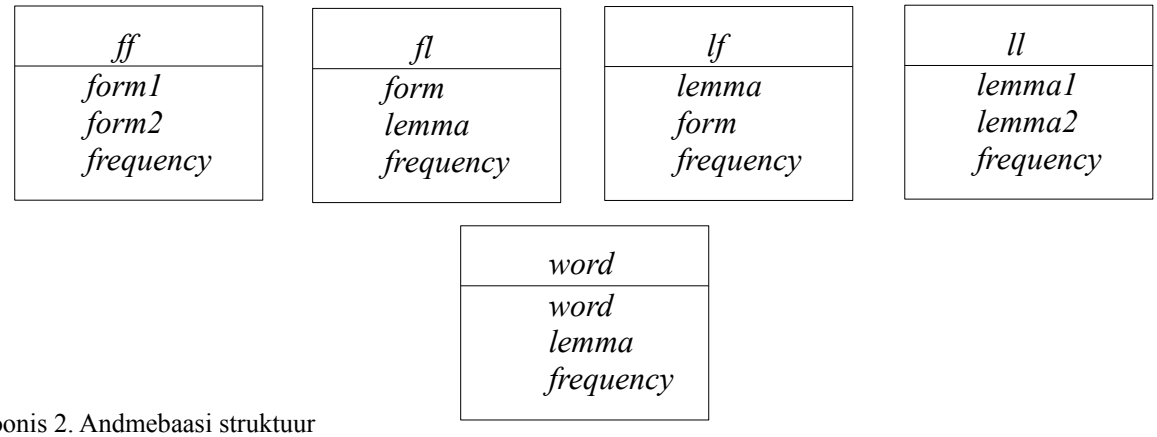
Iga selline potentsiaalne kollokatsioon – parajasti vaatluse all olev w_1 ja järjekordse akna piiridesse jääv sõna w_2 – lisatakse veel nelja erinevasse *hash* nimekirja, mille eristamiseks kasutatakse lühendeid *ff*, *fl*, *lf* ja *ll*. *f* (*form*) tähistab sõnavormi ja *l* (*lemma*) lemmat. Mingi konkreetse kollokatsiooni kandidaadi puhul salvestakse seega w_1 ja w_2 nimekirja *ff* sagedusega 1, kui antud paari seal veel ei ole, ning suurendatakse sagedust, kui see seal juba olemas on. Leides mõlema sõna lemma, lisatakse paar ka tabelitesse *fl*, *lf* ja *ll*, nii et *fl* puhul

on esimeseks komponendiks w_1, teiseks sõna w_2 lemma; *lf* puhul vastupidi: sõna w_1 lemma ja w_2; ning *ll* puhul koosneb paar mõlema sõna lemmast. Kui järjekordse faili lugemine on lõppenud ja sagedustega varustatud paaride nimekirjad on olemas, kirjutatakse potentsiaalsed kollokatsioonid koos vastavate esinemissagedustega ajutistesse tekstifailidesse, kasutades sõnade ja sageduste eraldamiseks semikooloneid ja reavahetust.

2.1.2.3 Sageduste salvestamine andmebaasi

Iga konkreetse faili töötlemise järel tekitatakse seega ajutine andmefail selles korpusefailis esinevate võimalike kollokatsioonide ja neile vastavate esinemissagedustega. Seejärel luuakse andmebaasi ajutised tabelid, kuhu laaditakse kogu info andmefailidest, misjärel tõstetakse tabelite sisu ümber põhitabelitesse. Terve failitäie info korraga sisselugemine kiirendab andmete baasi lisamist. Tekstifaili salvestamist ja andmebaasitabelitesse tõstmist tehakse iga faili läbivaatamise järel, et mälus hoitavad *hash* struktuurid liiga suureks ei läheks. Ka unikaalsete sõnade nimekiri salvestatakse analoogselt esmalt faili (koos sellele vastava lemmaga), siis ajutisse tabelisse ning seejärel vastavasse põhitabelisse.

Programmi töö lõpuks on andmebaasis viis tabelit: *ff*, *fl*, *lf*, *ll* ja *word* (Joonis 2). *ff* sisaldab kaht välja sõnavormide jaoks, *fl* ja *lf* üht sõnavormi, teist lemma jaoks ja *ll* kaht välja lemmade tarbeks. Tabelis *word* on väljad sõnavormi ja sellele vastava lemma jaoks. Kõik viis tabelit sisaldavad ka välja kirje korpuses esinemise sageduse jaoks.



Joonis 2. Andmebaasi struktuur

2.1.3 Päringu tegemine

Kui sagedused on juba andmebaasi tabelitesse saanud, siis võib käivitada päringuprogrammi. Kasutajalt oodatakse sisendina andmebaasi serveri nime, andmebaasi nime, MySQL kasutajat ja parooli ning loomulikult otsisõna (w_1). Lisaks peab ta määrama, kas sisestatud sõna näol on tegu lemma või sõnavormiga ning kas kollokaatidena soovib ta näha lemmasid või sõnavorme.

Kui programmil on kasutajapoolne sisend käes, teeb ta saadud parameetrite põhjal kindlaks, millistes andmebaasi tabelites hoitakse arvutuste jaoks vajalikke sagedusi. Nt kui kasutaja on sisestanud sõnavormi ja soovib kollokaate lemmade kujul, siis vaadeldakse lisaks *word* tabelile tabelit *fl*. Nimetatud tabelites asuvate sageduste põhjal koostatakse iga potentsiaalse kollokatsiooni jaoks mõttelised sagedustabelid tegelike (Joonis 3a) ja oodatud (Joonis 3b) sagedustega.

a)

	w_2	$\neg w_2$	
w_1	O_{11}	O_{12}	$= R_1$
$\neg w_1$	O_{21}	O_{22}	$= R_2$
	$\|\|$	$\|\|$	$\searrow$
	C_1	C_2	N

b)

	w_2	$\neg w_2$
w_1	$E_{11} = \frac{R_1 \cdot C_1}{N}$	$E_{12} = \frac{R_1 \cdot C_2}{N}$
$\neg w_1$	$E_{21} = \frac{R_2 \cdot C_1}{N}$	$E_{22} = \frac{R_2 \cdot C_2}{N}$

Joonis 3. Sagedustabelid

Tegelikud sagedused leitakse andmebaasi tabelitest, kusjuures O_{11} on sõnapaaride tabeli (kas siis *ff*, *fl*, *lf* või *ll*) sagedus, kui tabeli esimesel väljal asub w_1 ja teisel w_2. See kajastab juhtusid, mil w_2 asub w_1 ümbruses. O_{12} leitakse nii, et vaadeldakse sagedusi sõnapaari tabelis juhtudel, kui esimesel väljal asub w_1, aga teisel mitte w_2, see tähistab olukordi, mil w_1 ümbruses asub mõni teine sõna (mitte w_2). O_{21} puhul leitakse tabelist *word* sõna w_2 sagedus ja lahutatakse sellest O_{11}. See näitab juhtumeid, mil w_2 esineb mõne teise sõna (mitte w_1) ümbruses. O_{22} saab arvutada eelnevate põhjal, lahutades valimi suurusest nende summa N - ($O_{11} + O_{12} + O_{21}$). See tähistab kõiki ülejäänud juhtusid st selliseid, mil mõni sõnast w_2 erinev

sõna asub mingi sõna ümbruses, mis pole w_1. Valimi suuruseks on kõikide sõnade arv korpuses, millest on maha lahutatud w_1 sagedus, st kõigi sõnast w_1 erinevate sõnade hulk. Oodatud sagedused arvutatakse tegelike sageduste ja valimi mahu põhjal (Joonis 3b), nii et $R_1 = O_{11} + O_{12}$, $R_2 = O_{21} + O_{22}$, $C_1 = O_{11} + O_{21}$ ja $C_2 = O_{12} + O_{22}$.

Kui sagedustabelid on valmis saab statistikuks valitud log-tõepära funktsiooni kasutades iga potentsiaalse kollokatsiooni jaoks arvutada seose tugevuse. Log-tõepära valem on järgmine:

$$2 \cdot \sum_{ij} O_{ij} \cdot \log \frac{O_{ij}}{E_{ij}}$$

Juhul, kui O_{ij} on võrdne nulliga ja naturaallogaritmi võtmine ebaõnnestuks, võtab programm log 0 väärtuseks nulli. Kui $O_{11} < E_{11}$, st oodatud sagedus ületab tegeliku sageduse ning sõnad „tõukuvad“, tuleb seose tugevus läbi korrutada arvuga -1, et sõnad saaksid negatiivse seose korral ka negatiivse seose tugevuse väärtuse. Seda on tarvis teha sellepärast, et log-tõepära statistik on kahepoolne (*two-sided*) mõõdik ning ei erista tugevaid positiivseid ja negatiivseid assotsiatsioone, vaid annab mõlematele kõrged seose tugevuse väärtused (Evert 2007: 21, 30). Arvuga -1 läbi korrutades saab mõõdiku muuta ühepoolseks (*one-sided*).

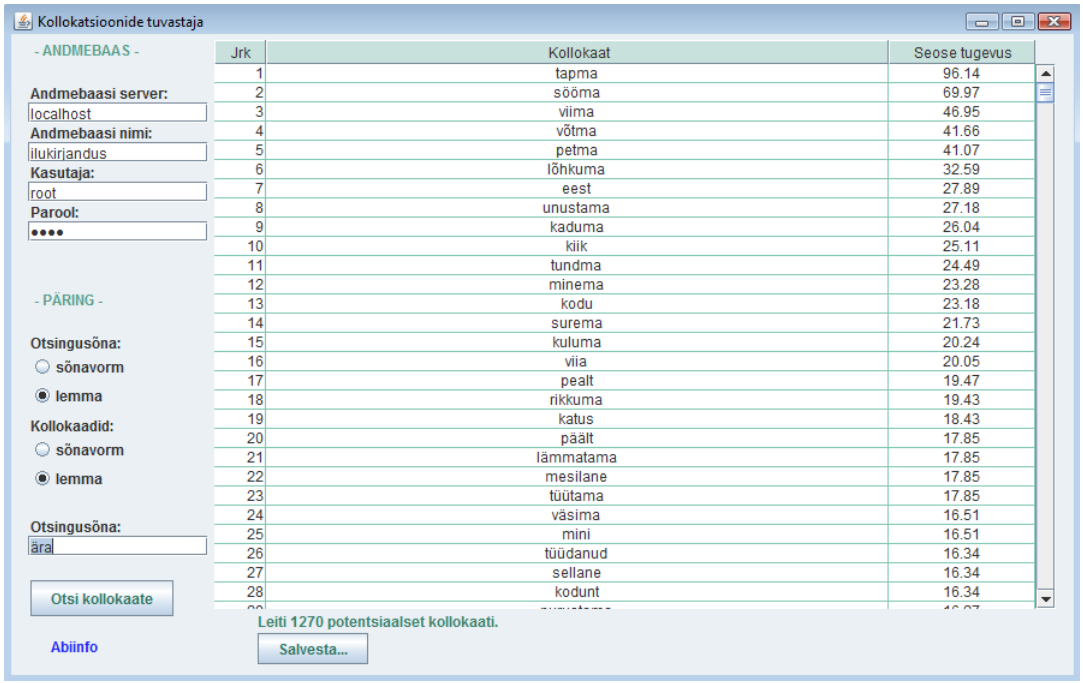
Pärast valemi rakendamist kuvab programm kasutajale sisestatud sõna kõik potentsiaalsed kollokaadid koos vastavate seose tugevuse määradega. Kui kasutaja on tutvunud päringu tulemustega, on tal võimalik salvestada tulemused, alustada kohe uut päringut või väljuda programmist.

2.1.4 Kasutajaliides

Kasutajaliides kujutab endast graafilist programmiakent, millel on vasakus servas tekstiväljad kasutajapoolse sisendi jaoks (Joonis 4). Üleval vasakul asub andmebaasiga ühendumiseks vajalik info, keskel nupud otsingusõna ja kollokaadi kandidaatide soovitud kuju määramiseks ning all väli otsingusõna sisestamiseks ja nupp päringu kinnitamiseks. Kõige all vasakus nurgas on abiinfo programmi kasutamise kohta.

Akna paremale poole kuvatakse tabelina päringu tulemused. Kui tulemusteks on kasutaja soovinud lemmasid, on tulemustabelil kolm lahtrit: jrk number, kollokaat (lemma) ja seose tugevuse määr. Kui väljundiks on valitud sõnavorm, on tabelil viis lahtrit: jrk number, kollokaat (sõnavorm), sellele vastav lemma, morfoloogiline analüüs ja seose tugevuse määr. Morfoloogiline analüüs näitab potentsiaalse kollokaadi sõnaliiki, käänet, arvu jms ehk siis morfoloogilist infot, mis sai t3mesta poolt igale korpuses olevale sõnale lisatud.

Joonis 4. Kasutajaliides enne päringuid



Joonis 5. Kasutajaliides päringu tulemusega

2.2 Valitud tehnoloogiad ja lahenduste põhjendus

Kuna korpus võib olla märkimisväärselt suur ja selle töötlemine ressursikulukas ning kuna päringu tegemise võimaldamine on antud uurimuse põhiline eesmärk, siis on mõeldamatu, et iga kasutaja poolt sisestatud sõna puhul vaadatakse läbi terve korpus, seega tehakse eeltööd ja kogutakse korpusest vajalikud andmed kokku enne kasutajapoolsete päringute tegemist. Piisab, et käia korpusefailid läbi ühe korra ja salvestada vaatlustulemused andmebaasi, mille struktuur on kujundatud jällegi päringuprogrammi silmas pidades.

Aknaks on valitud neli, kuna see on üks tüüpilisemaid akna suurusi püsiühenditega tegelemisel. Kuna eesti keel on vaba sõnajärjega, võivad kollokatsiooni moodustavad sõnad asuda üksteisest suhteliselt kaugel, nii et üks või kaks jätaks tõenäoliselt olulise hulga ühendeid vaatluse alt välja. Liiga suur vahemik aga tooks kaasa liialt palju „müra“, st sõnapaare, mis tegelikult kollokatsiooni ei moodusta. Aken on sümmeetriline, sest antud töös ei keskenduta nimisõna täienditele ega millelegi säärasele, vaid tuntakse huvi sõna ümbruse vastu üldiselt.

Kuna statistiku valik sõltub suuresti korpusest ja sobivaimat on keeruline leida, on antud programmi jaoks, kus korpuse suurus ja valdkond võivad muutuda, selleks valitud log-tõepära funktsioon. Seda seetõttu, et log-tõepära statistik on paljudes uurimustes, k.a eelpool mainitud statistikute võrdluses murdekorpustel (Uiboaed 2010), tunnistatud kõige efektiivsemaks.

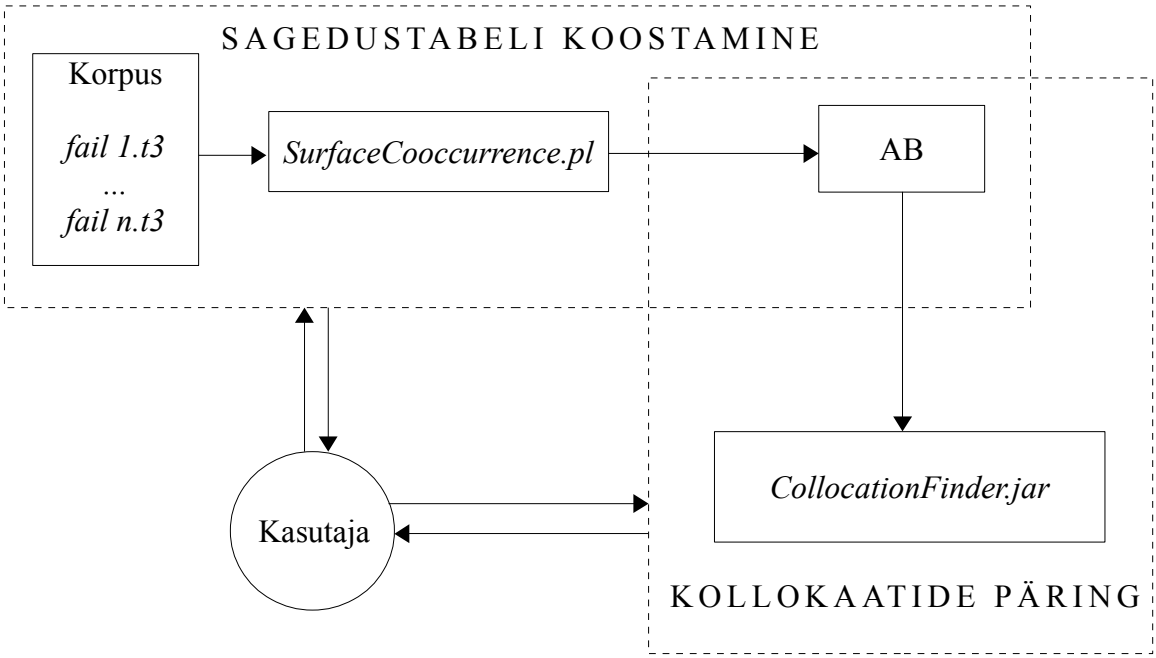
Sagedustabeleid loov programm on kirjutatud programmeerimiskeeles Perl. Seda sel põhjusel, et korpuse lauseteks jaotamisel, lemmade leidmisel jms juures on mõttekas kasutada regulaaravaldisi ja mustrite otsimist (*pattern matching*). Perl keelde on need sisse ehitatud ja nende kasutamine kiire ja mugav. Perli installeerimine ja programmi käivitamine võib küll olla üsna tülikas, kuid antud programmi käivitatakse tõenäoliselt vaid ühe korra ning edasi kasutatakse juba päringuprogrammi koos andmebaasiga.

Päringuprogrammi jaoks on valitud Java programmeerimiskeel, kuna see on laialt levinud ja töötab ühtemoodi hästi erinevates keskkondades. Java tugi (*Java Runtime Environment*) on enamasti arvutites juba olemas ning juhul, kui see puudub, on JRE installeerimine lihtne ja seetõttu ei tohiks ka antud programmi paigaldamine väga keerukas olla. Lisaks sobib Java hästi kasutajaliidese loomiseks.

Andmebaasiks on valitud MySQL. Seda seepärast, et tegemist on populaarseima vabavaralise andmebaasi haldamissüsteemiga, mis töötab mitmesugustes keskkondades ja kasutab üldlevinud SQL päringukeelt.

2.3. Tulemus

Nagu eespool kirjeldatud koosneb kollokatsioonide tuvastaja kahest põhilisest osast: programmist, mis koostab korpuse põhjal indekstabeli ning rakendusest, mis kasutades loodud tabelit, etteantud sõna ja täpsustavaid parameetreid leiab sõna kollokaadid. Rakenduse komponendid ja nendevaheline suhtlus on kujutatud skeemil (Joonis 6).



Joonis 6. Kollokatsioonide tuvastaja ülesehitus

Programmi luues võeti eelduseks, et kasutaja soovib korpusest teha mitmeid päringuid erinevate sõnadega. Suurte korpuste korral oleks mõeldamatu iga otsingusõna puhul terve korpus läbi uurida. Sellepärast ongi loodud eraldi abiprogramm, mis vaatab korpuse ühe korra läbi ja salvestab kasuliku info andmebaasi, nii et hiljem oleks sealt võimalik mistahes otsingusõna korral kõik arvutusteks vajalik üles leida.

Kuna log-tõepära statistiku puhul tuleb iga vaadeldava kollokatsiooni kandidaadi kohta arvutada sagedustabelid oodatavate ja tegelike sagedustega ning selleks eelnevalt andmebaasist ajalik info võtta, tuleb lõppkokkuvõttes suurte korpuste korral teha palju resursikulukat tööd ja seega võib otsing kujuneda üsna aeglaseks. Eriti kui otsitakse mõnda eriti sagedast sõna.

2.4 Edasiarendus

Käesoleva uurimuse käigus loodud programmi oleks kasulik edaspidi täiendada, et seda mitmekülgsemaks ja mugavamaks muuta. Üks selline koht, mida kindlasti parandada saaks, on programmi kiirus. Praeguse lahenduse nõrgaks küljeks on andmebaasi päringute aeglus. Kuna otsingusõna iga kollokaadi kandidaadi jaoks tehakse mitmeid päringuid, et koostada sagedustabel, tuleb andmebaasi poole pöördumisi ühe kasutajapoolse otsingu kohta lausa tuhandeid, olenevalt muidugi korpuse suurusest ja sellest, kui haruldase sõnaga on tegu.

Kollokatsioonide tuvastajat päringutega tegelev osa *CollocationFinder.jar* on hõlpsasti laiendatav. Lisaks akna kasutamisele võiks valikusse lisada ka osalausetest kollokaatide tuvastamise võimaluse. Selleks oleks muidugi vajalik vastav abiprogramm, mis teeks korpusest andmebaasitabelid analoogsel viisil nagu praegune *SurfaceCooccurrence.pl*, kasutades sobivaid algoritme, et sagedused tabelites vastaksid valitud meetodile. Tekstilise koosesinemise puhul võiks valikusse lisada lausa mitmesuguseid tekstiüksusi, nt nii osalause kui terve lause.

Samuti võiks kaaluda statistikute valiku suurendamist, st lisaks log-tõepära funktsioonile kasutada hii-ruut statistikut või lisada hoopis mõni lihtsatest seose mõõdikutest. Log-tõepära on andnud mitmete uurimuste kohaselt kollokatsioonide leidmisel parimaid tulemusi, nii et selle väljavahetamiseks põhjust ei ole, küll aga võiks olla huvitav jälgida, milliseid erinevusi on erinevate mõõdikute otsingutulemustes. Ka mõõdikute kombineerimine, nt kollokatsiooni nimekirja ühisosa leidmine, võiks anda huvitavaid tulemusi.

Veel oleks võimalik kujundada programm nii, et ka ilma otsingusõna sisestamata võiks korpusest leida tugevamini seotud kollokatsioone. Nt terve korpuse peale sada kõige olulisemat kollokatsiooni. See tähendaks muidugi tohutut hulka päringuid, kuid väiksemate korpuste puhul võiks selline variant isegi mõeldav olla.

Väljundiks on praegusel juhul päringutulemuste kasutajale ekraanile kuvamine ja tavalisse tekstifaili salvestamine. Tulemuste salvestamisel saaks failistruktuuri muuta vastavalt vajadusele, nt sobivaks mõne teise programmi sisendiks.

Peatükk 3

Töönäited

Programmi töö testimiseks sai valitud 198 631-sõnaline korpus seitsmes t3 failis, mis koosnes ilukirjanduslikest tekstidest. Unikaalseid sõnavorme esines antud korpuses 41952, lemmasid 20376.

3.1 Sagedustabeli koostamine

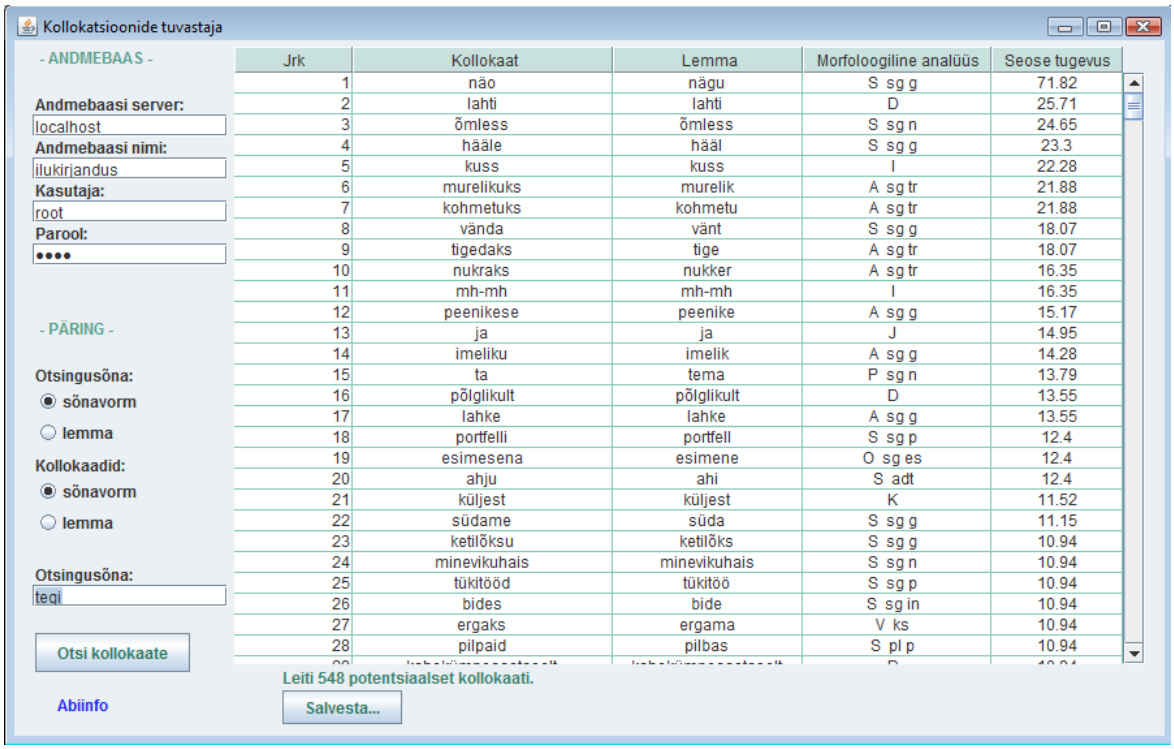
Korpuse esialgne töötlemine programmiga *SurfaceCooccurrence.pl* võttis aega umbes tunni. Selle käigus moodustati andmebaasi viis tabelit: *word*, *ff*, *fl*, *lf*, *ll*. Tabelisse word salvestati kõik unikaalsed sõnavormid koos nende lemmadega ja esinemissagedustega korpuses. Tabelisse *ff* salvestati kõik sõnapaarid sõnavormidena, st kõik bigrammid, mis korpusest leiti sellisel kujul nagu nad seal esinesid. Tabelitesse *fl* ja *lf* salvestati samad paarid ainsa erinevusega, et üks komponent oli viidud lemma kujule. Tabelisse *ll* said kirja kõik samad paarid, milles mõlemad komponendid olid viidud lemma kujule. Seega tabelis *ff* on ridu kõige rohkem, tabelites *fl* ja *lf* vähem ning tabelis *ll* kõige vähem, kuna paljud kirjed hakkasid komponente lemmaks üldistades korduma ja seetõttu need koondati.

3.2 Kollokaatide tuvastamine

3.2.1 Sõnavormi järgi sõnavormide leidmine

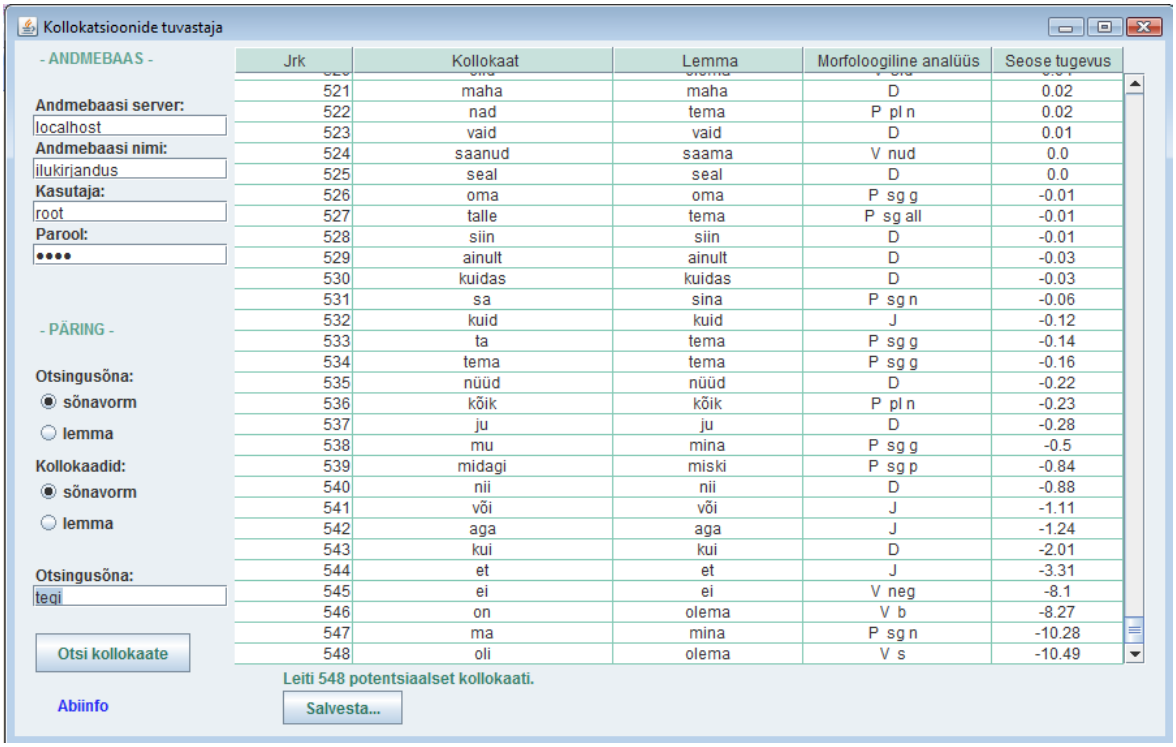
Antud juhul on kasutaja sisestanud otsingusõnaks vormi „tegi“ ja otsingusõna kujuks määranud sisendile vastavalt sõnavormi (Joonis 7). Kollokaadi kujuks on ta määranud

sõnavormi. Päringu tulemusena kuvatakse ekraanile tabel leitud kollokaatide, nende morfoloogilise analüüsi ja lemmadega, reastatud seose tugevuse alusel alates suurimast. Negatiivsete väärtustega antikollokaadid paiknevad tabeli alumises osas (Joonis 8).



Jrk	Kollokaat	Lemma	Morfoloogiline analüüs	Seose tugevus
1	näo	nägu	S sg g	71.82
2	lahti	lahti	D	25.71
3	õmless	õmless	S sg n	24.65
4	hääle	hääl	S sg g	23.3
5	kuss	kuss	I	22.28
6	murelikuks	murelik	A sg tr	21.88
7	kohmetuks	kohmetu	A sg tr	21.88
8	vända	vänt	S sg g	18.07
9	tigedaks	tige	A sg tr	18.07
10	nukraks	nukker	A sg tr	16.35
11	mh-mh	mh-mh	I	16.35
12	peenikese	peenike	A sg g	15.17
13	ja	ja	J	14.95
14	imeliku	imelik	A sg g	14.28
15	ta	tema	P sg n	13.79
16	põlglikult	põlglikult	D	13.55
17	lahke	lahke	A sg g	13.55
18	portfelli	portfell	S sg p	12.4
19	esimesena	esimene	O sg es	12.4
20	ahju	ahi	S adt	12.4
21	küljest	küljest	K	11.52
22	südame	süda	S sg g	11.15
23	ketilõksu	ketilõks	S sg g	10.94
24	minevikuhais	minevikuhais	S sg n	10.94
25	tükitööd	tükitöö	S sg p	10.94
26	bides	bide	S sg in	10.94
27	ergaks	ergama	V ks	10.94
28	pilpaid	pilbas	S pl p	10.94

Joonis 7. Päring: otsingusõna on sõnavorm, otsitavad kollokaadid samuti sõnavormid - „tõmbuvad“ sõnad

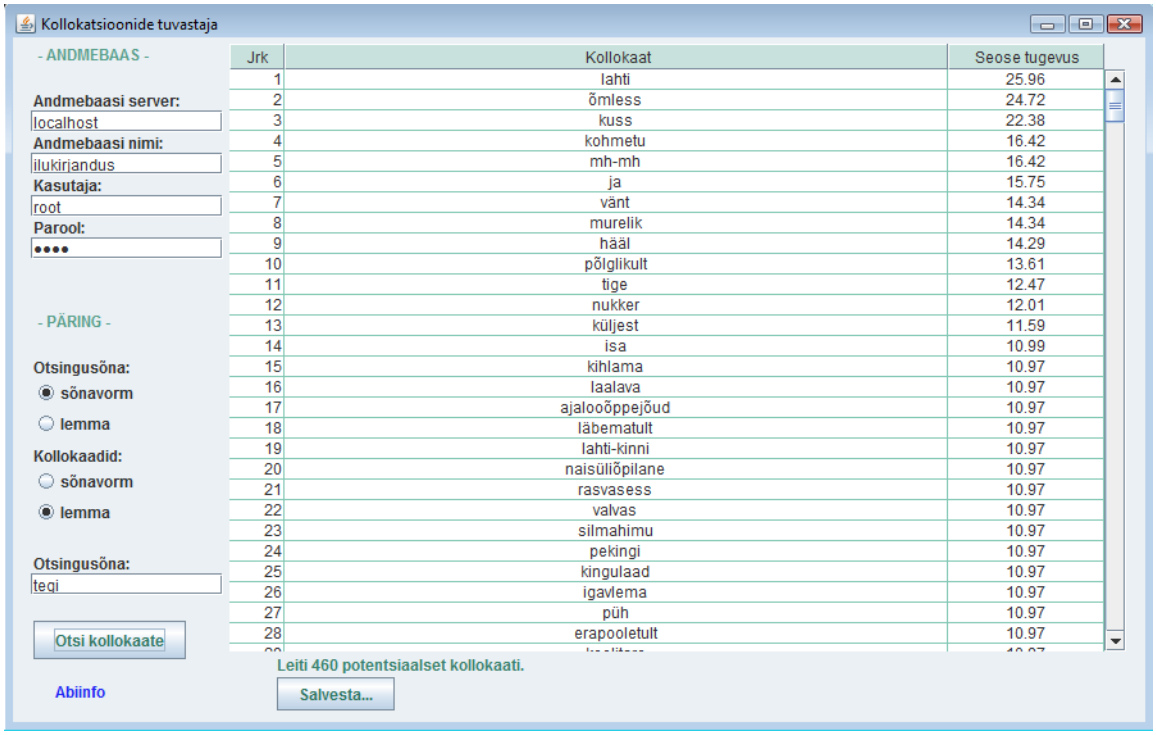


Jrk	Kollokaat	Lemma	Morfoloogiline analüüs	Seose tugevus
521	maha	maha	D	0.02
522	nad	tema	P pl n	0.02
523	vaid	vaid	D	0.01
524	saanud	saama	V nud	0.0
525	seal	seal	D	0.0
526	oma	oma	P sg g	-0.01
527	talle	tema	P sg all	-0.01
528	siin	siin	D	-0.01
529	ainult	ainult	D	-0.03
530	kuidas	kuidas	D	-0.03
531	sa	sina	P sg n	-0.06
532	kuid	kuid	J	-0.12
533	ta	tema	P sg g	-0.14
534	tema	tema	P sg g	-0.16
535	nüüd	nüüd	D	-0.22
536	kõik	kõik	P pl n	-0.23
537	ju	ju	D	-0.28
538	mu	mina	P sg g	-0.5
539	midagi	miski	P sg p	-0.84
540	nii	nii	D	-0.88
541	või	või	J	-1.11
542	aga	aga	J	-1.24
543	kui	kui	D	-2.01
544	et	et	J	-3.31
545	ei	ei	V neg	-8.1
546	on	olema	V b	-8.27
547	ma	mina	P sg n	-10.28
548	oli	olema	V s	-10.49

Joonis 8. Päring: otsingusõna on sõnavorm, otsitavad kollokaadid samuti sõnavormid - „tõukuvad“ sõnad

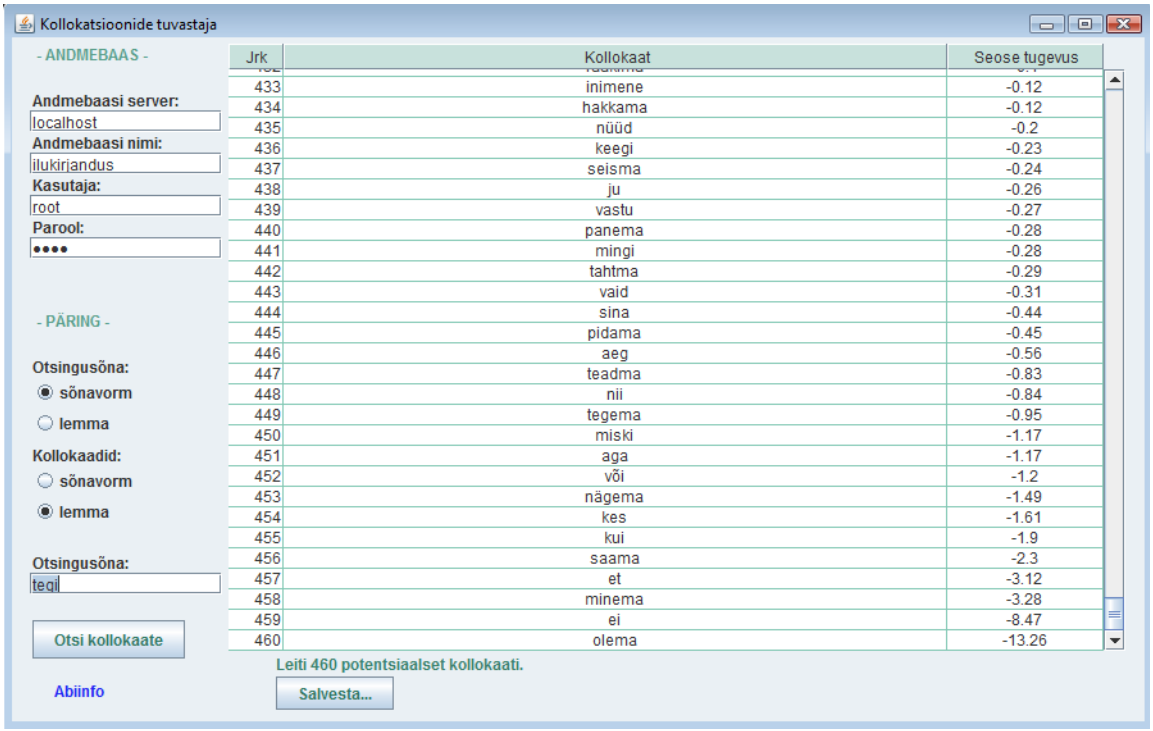
3.2.2 Sõnavormi järgi lemmade leidmine

Kasutaja on sisestanud sõnavormi „tegi“ ning sel korral valinud kollokaatide kujuks lemma. Päringu tulemusena kuvatakse kõik leitud kollokaadid koos vastavate seose tugevustega positiivsetest (Joonis 9) kuni negatiivseteni (Joonis 10).



Jrk	Kollokaat	Seose tugevus
1	lahti	25.96
2	õmless	24.72
3	kuss	22.38
4	kohmetu	16.42
5	mh-mh	16.42
6	ja	15.75
7	vänt	14.34
8	murelik	14.34
9	hääl	14.29
10	põlglikult	13.61
11	tige	12.47
12	nukker	12.01
13	küljest	11.59
14	isa	10.99
15	kihlama	10.97
16	laalava	10.97
17	ajalooõppejõud	10.97
18	läbematult	10.97
19	lahti-kinni	10.97
20	naisüliõpilane	10.97
21	rasvasess	10.97
22	valvas	10.97
23	silmahimu	10.97
24	pekingi	10.97
25	kingulaad	10.97
26	igavlema	10.97
27	püh	10.97
28	erapooletult	10.97

Joonis 9. Päring: otsingusõna on sõnavorm, otsitavad kollokaadid lemmad - „tõmbuvad“ sõnad

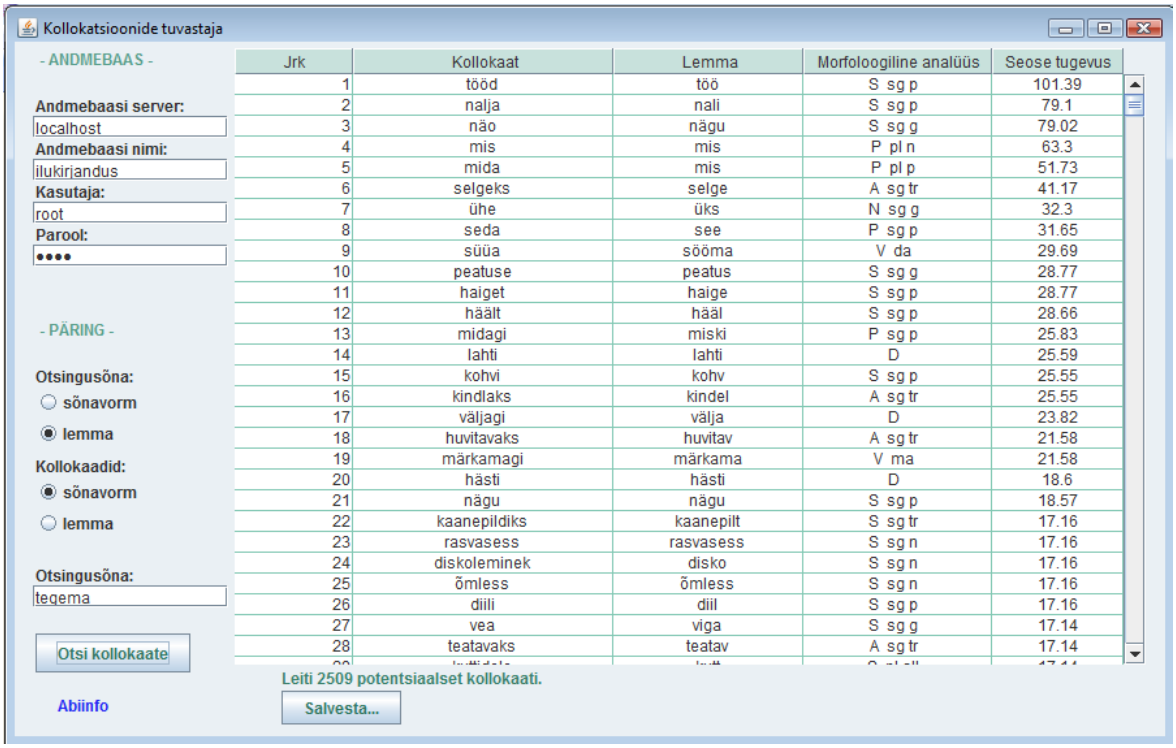


Jrk	Kollokaat	Seose tugevus
433	inimene	-0.12
434	hakkama	-0.12
435	nüüd	-0.2
436	keegi	-0.23
437	seisma	-0.24
438	ju	-0.26
439	vastu	-0.27
440	panema	-0.28
441	mingi	-0.28
442	tahtma	-0.29
443	vaid	-0.31
444	sina	-0.44
445	pidama	-0.45
446	aeg	-0.56
447	teadma	-0.83
448	nii	-0.84
449	tegema	-0.95
450	miski	-1.17
451	aga	-1.17
452	või	-1.2
453	nägema	-1.49
454	kes	-1.61
455	kui	-1.9
456	saama	-2.3
457	et	-3.12
458	minema	-3.28
459	ei	-8.47
460	olema	-13.26

Joonis 10. Päring: otsingusõna on sõnavorm, otsitavad kollokaadid lemmad - „tõukuvad“ sõnad

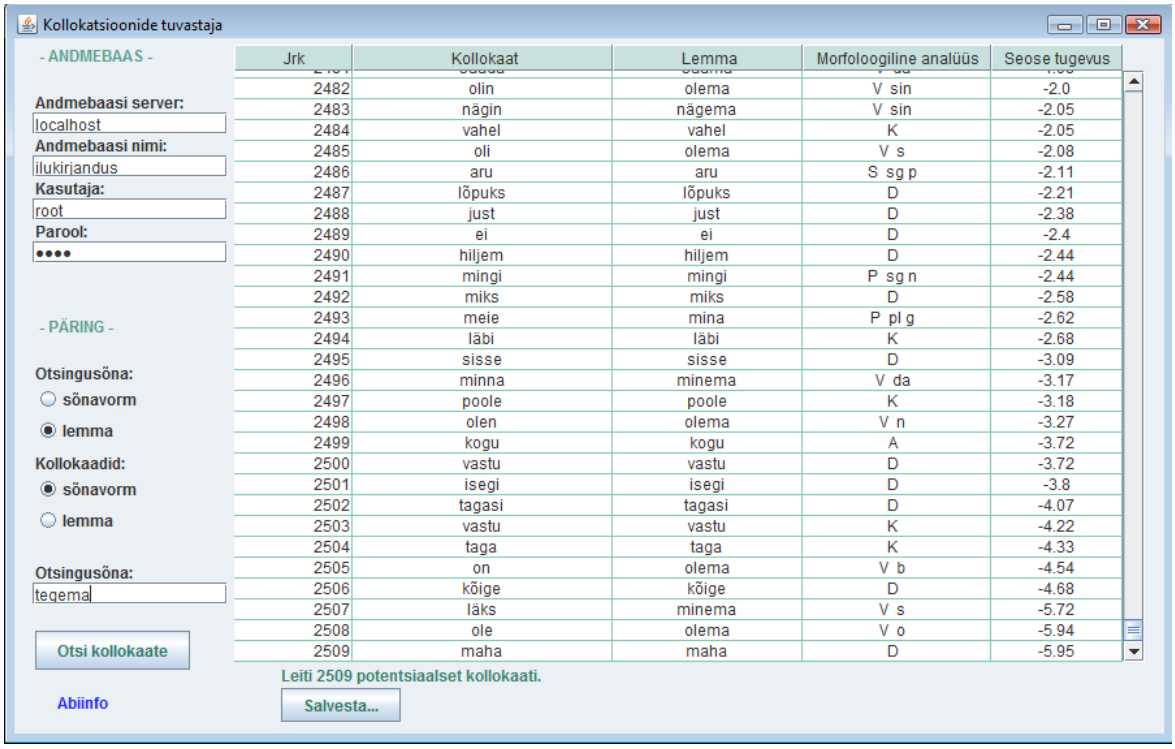
3.2.3 Lemma järgi sõnavormide leidmine

Kasutaja on sisestanud sõna „tegema“ (Joonis 11). Määrates otsingusõna kujuks lemma, laseb ta programmil uurida „tegema“ kõiki vorme, mis korpuses esinevad. Kuna kollokaatide tüübiks on seatud sõnavorm, siis kuvatakse kasutajale kõik sellised vormid, mis „tegema“ mistahes vormidega koos esinevad. Lisaks on tabelis veel leitud vormidele vastavad lemmad, morfoloogiline info ja seose tugevus. Negatiivse seose tugevusega „tõukuvad“ vormid asuvad tabeli allosas (Joonis 12).



Jrk	Kollokaat	Lemma	Morfoloogiline analüüs	Seose tugevus
1	tööd	töö	S sg p	101.39
2	nalja	nali	S sg p	79.1
3	näo	nägu	S sg g	79.02
4	mis	mis	P pl n	63.3
5	mida	mis	P pl p	51.73
6	selgeks	selge	A sg tr	41.17
7	ühe	üks	N sg g	32.3
8	seda	see	P sg p	31.65
9	süüa	sööma	V da	29.69
10	peatuse	peatus	S sg g	28.77
11	haiget	haige	S sg p	28.77
12	häält	hääl	S sg p	28.66
13	midagi	miski	P sg p	25.83
14	lahti	lahti	D	25.59
15	kohvi	kohv	S sg p	25.55
16	kindlaks	kindel	A sg tr	25.55
17	väljagi	välja	D	23.82
18	huvitavaks	huvitav	A sg tr	21.58
19	märkamagi	märkama	V ma	21.58
20	hästi	hästi	D	18.6
21	nägu	nägu	S sg p	18.57
22	kaanepildiks	kaanepilt	S sg tr	17.16
23	rasvasess	rasvasess	S sg n	17.16
24	diskoleminek	disko	S sg n	17.16
25	õmless	õmless	S sg n	17.16
26	diili	diil	S sg p	17.16
27	vea	viga	S sg g	17.14
28	teatavaks	teatav	A sg tr	17.14

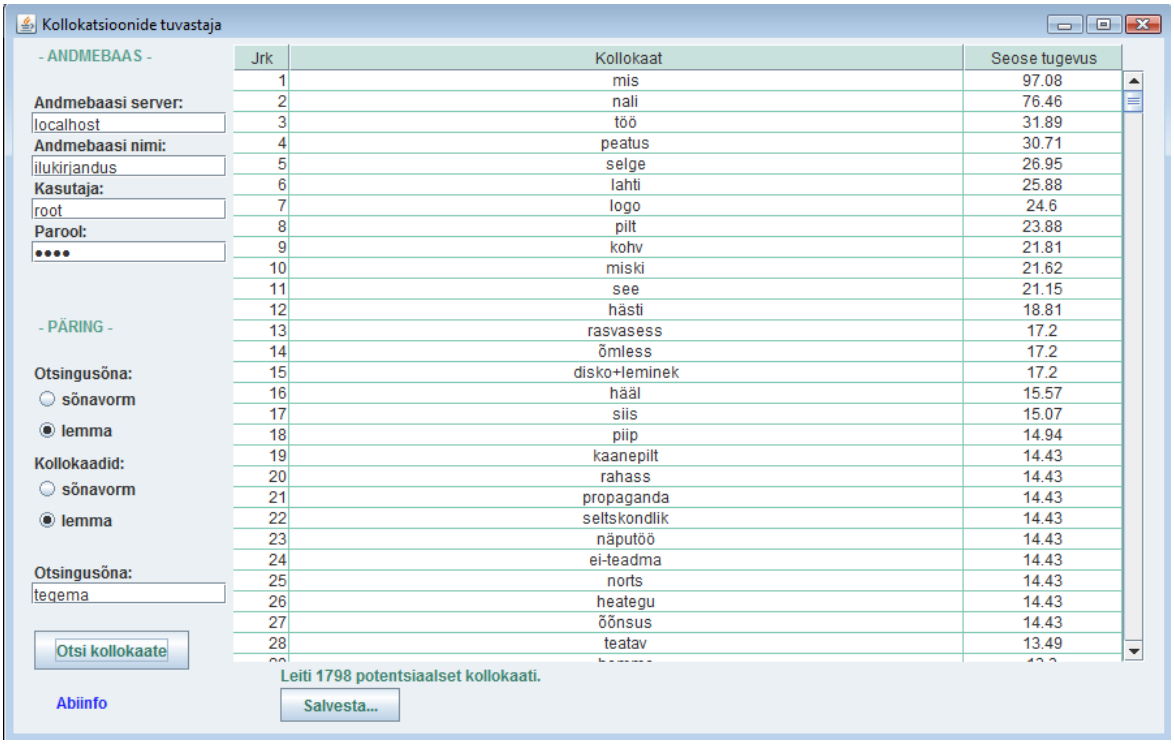
Joonis 11. Päring: otsingusõna on lemma, otsitavad kollokaadid on sõnavormid - „tõmbuvad“ sõnad



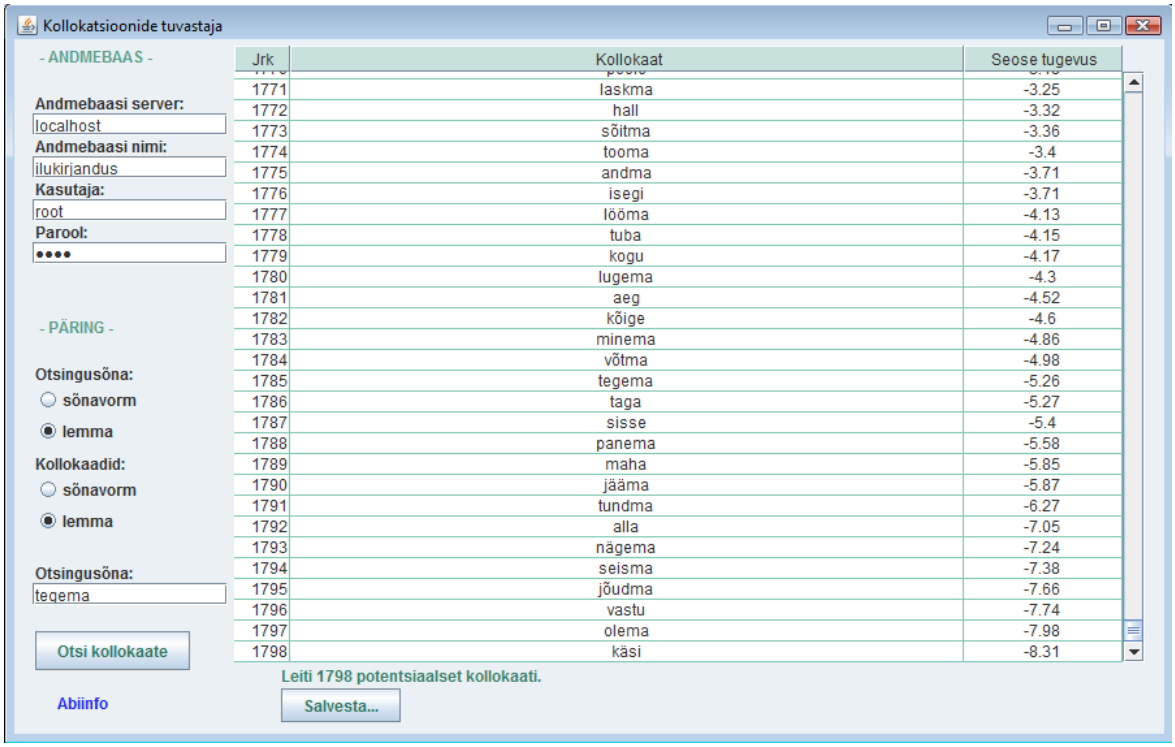
Joonis 12. Päring: otsingusõna on lemma, otsitavad kollokaadid on sõnavormid - „tõukuvad“ sõnad

3.2.4 Lemma järgi lemmade leidmine

Kasutaja sooritab otsingut sõnaga „tegema“ (Joonis 13). Nii otsingusõna kui kollokaatide kujuks on määratud lemma. Päringu tulemusena esitatakse kasutajale tabel kõikidest lemmadest, mis on korpusest leitud sisestatud lemma lähedusest. Kollokaatide kandidaadi on reastatud vastavalt seose tugevuse väärtusele kahanevalt, nii et negatiivse väärtusega antikollokaadid jäävad tabeli lõppu (Joonis 14).



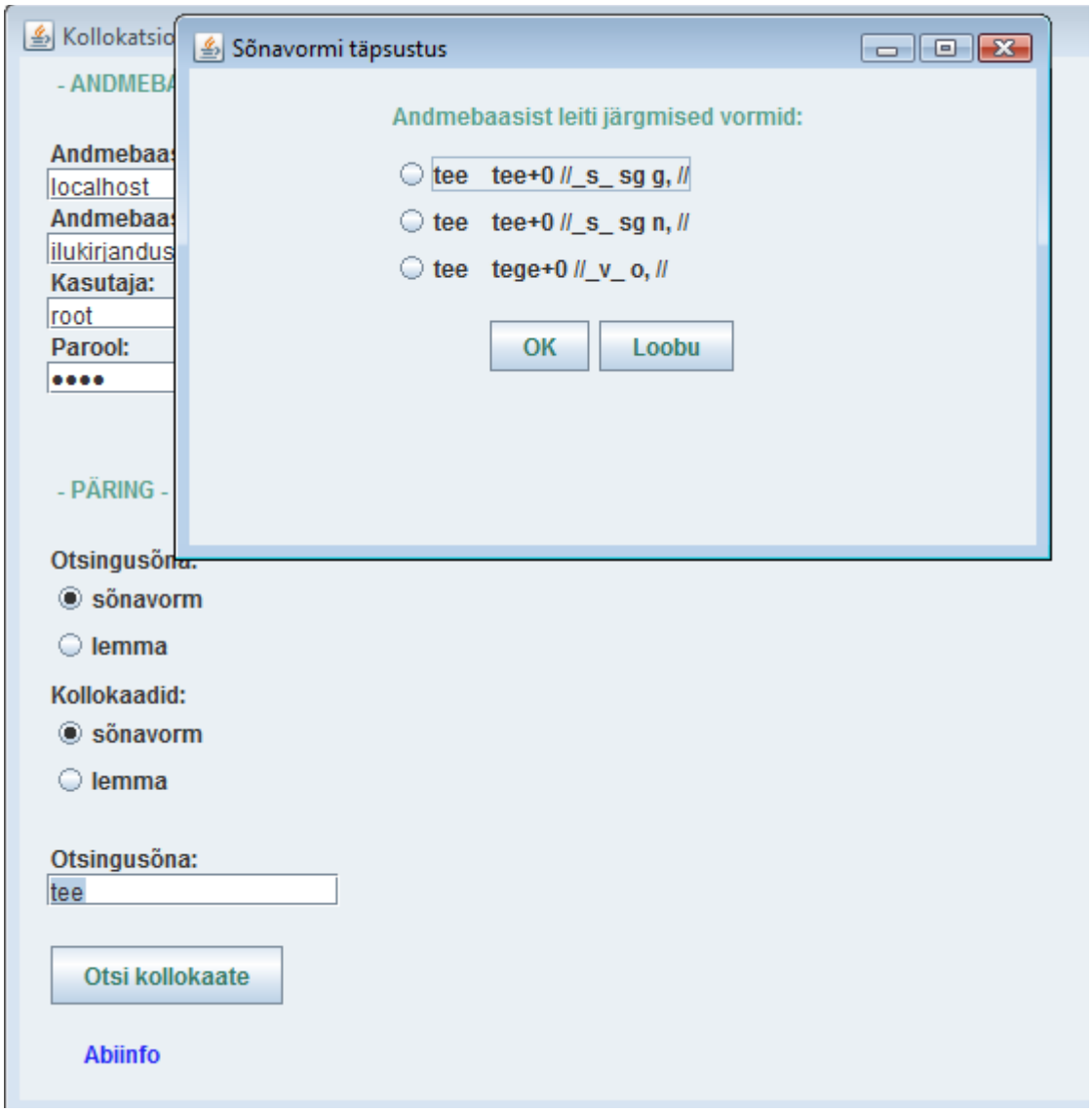
Joonis 13. Päring: otsingusõna on lemma, otsitavad kollokaadid samuti lemmad - „tõmbuvad“ sõnad



Joonis 14. Päring: otsingusõna on lemma, otsitavad kollokaadid samuti lemmad - „tõukuvad“ sõnad

3.2.5 Sõnavormi täpsustamine

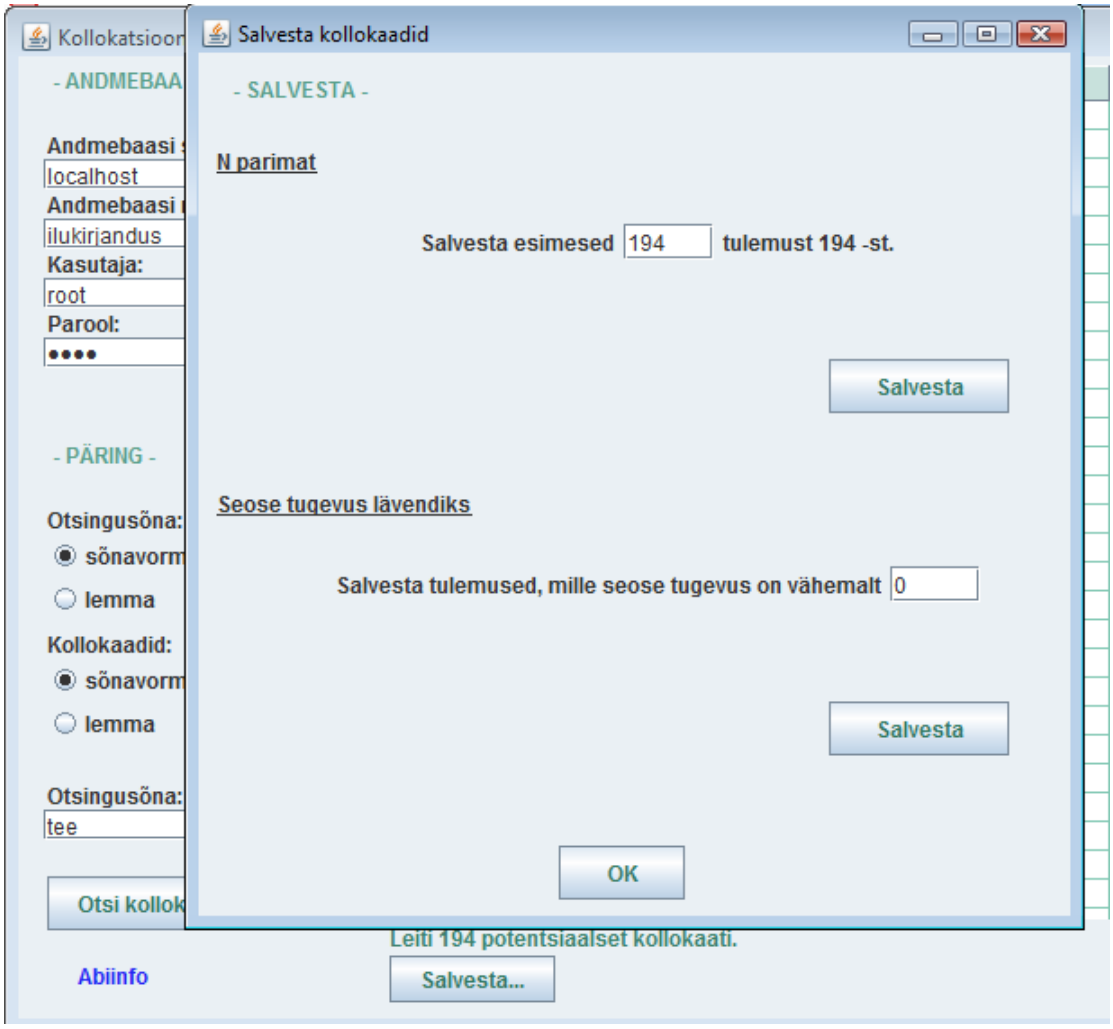
Juhul, kui kasutaja on otsingusõnaks valinud sõnavormi, uuritakse enne kollokaatide otsimist sisestatud sõna mitmesust. Kui korpuses esineb selline sõna mitme erineva morfoloogilise analüüsiga, siis püütakse esmalt kindlaks teha, millist sõnavormi kasutaja silmas pidas. Selleks kuvatakse kasutajale uus aken kõigi leitud morfoloogiliste analüüsidega, millest tal tuleb üks välja valida (Joonis 15). Kui kasutaja on oma valiku teinud, siis jätkatakse kollokaatide päringut kindlaks tehtud sõnaga.



Joonis 15. Sõnavormi täpsustamine

3.2.6 Tulemuste salvestamine

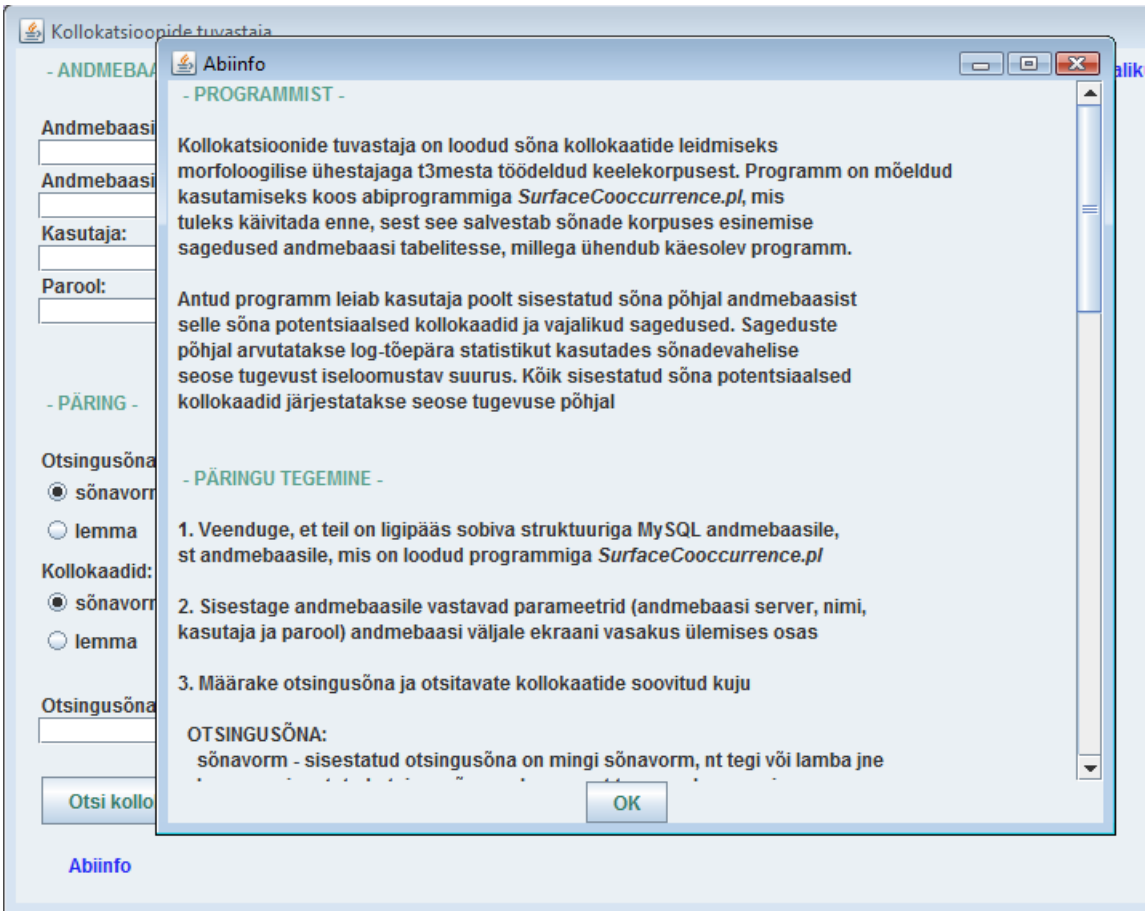
Kui päringu tulemusel on leitud vähemalt üks kollokaat, tekib kollokaatide tabeli juurde nupp „Salvesta“. Selle vajutamisel kuvatakse uus aken, mis lubab kasutajal valida, milliseid kollokaate salvestada (Joonis 16). Salvestada on võimalik kahel viisil. Esimene variant – N parimat – on salvestada teatud arv suurima seose tugevusega kollokaate. Vaikimisi väärtuseks on kogu leitud potentsiaalsete kollokaatide hulk. Teine variant – seose tugevus lävendiks – on sisestada seose tugevuse väärtus, millest alates kollokaate salvestada. St kui kollokaadi puhul seose tugevuse määr on võrdne või ületab sisestatud väärtuse, salvestatakse kollokaat faili. Mõlema variandi puhul viivad vaikimisi sisestatud numbrid kõikide leitud kollokaatide salvestamiseni. Kõikidel juhtudel salvestatakse faili kollokaat koos talle vastava seose tugevusega, nii et seose tugevus on kollokaadist eraldatud tühikuga ning iga kollokaat asub eraldi real.



Joonis 16. Tulemuste salvestamine faili

3.2.7 Abiinfo

Programmi kasutamise kohta saab informatsiooni programmiakna vasakul all servas olevale „Abiinfo“ lingile vajutades. Selle peale kuvatakse kasutajale uus aken, mis kirjeldab päringu tegemist ja selgitab programmi käitumist (Joonis 17). Avatud abiinfo aknaga saab programmi vabalt edasi kasutada.



Joonis 17. Info kuvamine programmi kasutamise kohta

Kokkuvõte

Käesoleva uurimuse tulemusena sai loodud rakendus etteantud sõna kollokaatide tuvastamiseks tekstikorpusest. Kollokatsioonide tuvastaja koosneb kahest eraldiseisvast osast. Esimene neist loob etteantud korpuse põhjal andmebaasi tabelid kollokaatide leidmiseks vajaliku statistikaga. Teine programm kasutab loodud andmebaasi ja kasutaja poolt sisestatud sõna ja sõnakuju valikute (lemma või sõnavorm) põhjal leiab selle sõna kõik potentsiaalsed kollokaadid. Kollokatsioonidena vaadeldakse antud töös samas lauses asuvaid bigramme, mis paiknevad üksteisest kuni nelja sõna kaugusel.

Iga kollokaadi kandidaadi juures on ära toodud kollokatsiooni seose tugevus. Kui sõnad esinevad tekstis sagedamini kui oodatud nö „tõmbuvad“, siis on seose tugevus positiivne arv ning mida suurem, seda tugevamini on antud sõnad omavahel seotud. Negatiivse väärtusega tähistatakse nö „tõukuvaid“ sõnapaare, st neid, mille puhul sõnad esinevad korpuses üksteise läheduses harvemini, kui võiks eeldada nende eraldi esinemise sageduste põhjal.

Programmi kõige vajalikumaks edasiarenduseks oleks vast selle töökiiruse parandamine. Praeguse seisuga võivad suurte korpuste juures päringud muutuda aeglaseks. Samuti võiks edaspidi kohandada kollokatsioonide tuvastajat vastavalt sellele, millisel otstarbel kollokaatide nimekirja edasi kasutada, st muuta salvestatavate andmete struktuuri või lisada päringu parameetreid (nt sõnaliikide valik vms).

Kuna püsiühendeid peetakse üheks loomuliku keele töötlemise peamistest probleemidest, on kollokatsioonide tuvastamine oluline etapp edasiste arvutilingvistiliste uurimuste ja eesti keele automaatse analüüsi ja sünteesiga tegeleva tarkvara väljatöötamisel. Antud uurimuse käigus loodud kollokatsioonide tuvastaja üldisemaks eesmärgiks ongi lihtsustada seda protsessi.

Collocation finder from a text corpus

Master's thesis

Katrin Jets

Summary

Handling multiword expressions is an essential part of succesful natural language processing. Since languages are full of expressions that consist of two or more words and have obtained a meaning very different from the meanings of its components, then it's futile to process a text one word at a time. Compounds need to be recognized from a text and processed as a single semantic unit. This research is focusing on the finding of the multiword expressions from a corpus using statistical measures.

As a result of this research an application was implemented for finding multiword expressions or rather collocations from a text corpus. In this case collocations are defined as bigrams, and the cooccurrence type observed is surface cooccurrence with four as the collocational span. In other words, collocations are word pairs that appear in the same sentence at most four words away from each other. The application gets a word from the user as input, and finds all its collocate candidates from the corpus. For each candidate association score is calculated using log-likelihood statistic. The association score indicates how strongly the two words are connected. Positive association score means that the two words are found in the same sentence within four words more often than they would be, if the words were in a random order. The higher the score, the more connected the word pair is. Negative values indicate the words appear near each other not as often as expected. The application consists of two separate programs. The first one looks through the corpus files, finds the necessary statistics to be able to find collocates and corresponding association scores for any entered word, and stores them in a database for the further use. The second program connects to this database, and based on the entered word, the form of the word (either a word form or a lemma) and the form of its collocates selected by the user, retrieves the collocations candidates form the database and calculates the association score for each potential collocate. It is then up to the user how to further process the list of the potential collocates.

Kirjandus

Eesti keele koondkorpuse esituse ja kasutusvõimaluste arendamine. http://www.keeletehnoloogia.ee/projektid/eesti-keele-koondkorpuse-esituse-ja

Kollokatsioonide tuvastaja. 2010. http://www.rabauti.ee/clc

Vahendid teksti mitmekihiliseks märgendamiseks (rakendatuna Koondkorpusele). http://www.keeletehnoloogia.ee/ekt-projektid/vahendid-teksti-mitmekihiliseks-margendamiseks

Mati Erelt, Tiiu Erelt, Kristiina Ross. Eesti keele käsiraamat. 2009. http://www.eki.ee/books/ekk09/

Stefan Evert. Corpora and Collocations. Extended Manuscript, 2007.
http://cogsci.uni-osnabrueck.de/~severt/PUB/Evert2007HSK_extended_manuscript.pdf

Kadri Jaanits. Leksikaalsetest kollokatsioonidest soome ja eesti keeles. 2004

Heiki-Jaan Kaalep. Tekstikorpuse abil loodud eesti keele morfoloogiaanalüsaator. Keel ja Kirjandus 1 1998, lk 22-29

Heiki-Jaan Kaalep, Kadri Muischnek. Püsiühendite leidmine teksti abil. Kogumikus Tähendusepüüdja „Catcher of the Meaning“, TÜ üldkeeleteaduse õppetooli toimetised 3, 2002, lk 172-184

Heiki-Jaan Kaalep, Kadri Muischnek. Eesti keele püsiühendid arvutilingvistikas: miks ja kuidas. Eesti Rakenduslingvistika Ühingu aastaraamat 5 2009, lk 157-172

Heiki-Jaan Kaalep, Tarmo Vaino. Kas vale meetodiga õiged tulemused? Statistikale tuginev eesti keele morfoloogiline ühestamine. Keel ja Kirjandus 1 1998, lk 30-38

Kristel Uiboaed. Statistilised meetodid murdekorpuse ühendverbide tuvastamisel. Eesti Rakenduslingvistika Ühingu aastaraamat 6 2010, lk 307-326

Lisa

CD kollokatsiooni tuvastaja ja teiste vajalike failidega

Tööle on lisatud CD, mis sisaldab programmi, selle kasutusjuhendit ja programmi töö testimiseks mõeldud korpusefaile. Lisaks on seal päringuprogrammi CollocationFinder.jar lähtekood, mis võib osutuda tarvilikuks programmi töö paremaks mõistmiseks ja võimalikuks edasiarenduseks.