TARTU ÜLIKOOL

MATEMAATIKA-INFORMAATIKATEADUSKOND

Arvutiteaduse instituut

Informaatika eriala

Taavet Kikas

Dialoogiaktide tuvastamine eestikeelsetes dialoogides sufiksipuude abil

Magistritöö (40 AP)

Juhendaja: prof. Mare Koit

Tartu 2007

Sisukord

1 Sissejuhatus 4
2 Dialoogiaktid 6
3 Dialoogiaktide tüpoloogia 8
4 Dialoogiaktide automaatne tuvastamine sufiksipuude abil 12
4.1 Ülevaade ja motivatsioon 12
4.2 Mõisted 13
4.3 Sufiksipuu 15
4.3.1 Sufiksipuu konstrueerimise algoritm 16
4.4 Tõenäosuslik sufiksipuu 19
4.4.1 Tõenäosusliku sufiksipuu konstrueerimise algoritm 24
4.5 Metamärke sisaldav tõenäosuslik sufiksipuu 25
4.5.1 Metamärke sisaldava tõenäosusliku sufiksipuu konstrueerimise algoritm 27
4.6 Sufiksipuude rakendamine 28
5 Eksperimendid 31
5.1 Tulemuste hindamine 31
5.2 Testimismetoodika 34
5.3 Korpuse eeltöötlus 35
5.4 Eksperimentide kirjeldused 37
6 Eksperimentide tulemused 39
6.1 Tuvastamine alamsõnede põhjal 39
6.2 Tuvastamine alamsekventside järgi 41
6.3 Tuvastamine alamsekventside ja diskursuse järgi 43
7 Järeldused 45
8 Edasine töö 46
9 Kokkuvõte 47
10 Recognition of Dialogue Acts in Estonian Dialogues Using Suffix Trees . 48
Kirjandus 50
Lisad 52
Lisa 1. Eesti dialoogiaktide tüpoloogia 52
Lisa 2. Dialoogiaktide tuvastamine alamsõnede põhjal 56

Lisa 3. Dialoogiaktide tuvastamine alamsekventside põhjal 59
Lisa 4. Dialoogiaktide tuvastamine alamsekventside ja diskursuse põhjal..... 62

1 Sissejuhatus

Inimestevahelist suhtlust võib vaadelda kui keele abil tehtavate tegevuste jada. Inimesed tervitavad, tutvustavad ennast, esitavad küsimusi, annavad infot, vabandavad jne. Iga selline tegevus omab suhtluses oma kindlat funktsiooni, mis aitab kõnelejal jõuda lähemale oma lõppeesmärgile, olgu selleks siis info saamine, juhiste andmine või soovi esitamine. Sääraste situatsioonidega peavad toime tulema ka loomulikus keeles suhtlevad dialoogisüsteemid ja tõlkeprogrammid.

Keele abil tehtavad tegevused kannavad koondnimetust dialoogiaktid ning nende automaatne tuvastamine on eelpool nimetatud rakenduste jaoks kriitilise tähtsusega. Automaatse tuvastamise muudab keeruliseks see, et dialoogiakti funktsiooni ja keelelise vormi vahel puudub üksühene seos. Seejuures võivad isegi inimeste arvamused ühe ja sama dialoogiakti funktsiooni osas lahkneda.

Dialoogiaktide tuvastamise meetodid võib jagada üldjoontes kaheks: reeglipõhised meetodid, mis modelleerivad kõneleja kavatsusi ja tuvastavad akte sellest lähtuvalt, ning statistilised meetodid, mis tuvastavad akte märksõnade ja muude pindmiste tunnuste põhjal. Reeglipõhine tuvastamine on võimas meetod, kuid selle puuduseks on aeganõudev reeglite koostamine. Lisaks on see probleem AI-täielik (ingl k *AI-complete*), mis tähendab, et selle täielik lahendamine nõuab täielikku tehisintellekti (mida seni ei ole õnnestunud modelleerida).

Seega eelistatakse tihti lihtsamaid statistilisi meetodeid. Kasutatavamad statistilised meetodid on Bayesi klassifitseerija, Markovi peitmudel, neurovõrgud, otsustuspuud ja teisendusepõhine õppimine.

Käesolev töö esitab veel ühe, dialoogiaktide tuvastamise seisukohalt uudse statistilise meetodi - tõenäosusliku sufiksipuu. Peamiseks motivatsiooniks selle meetodi juurde jõudmisel oli tähelepanek, et suur osa või isegi enamus dialoogiaktidele iseloomulikke tunnuseid esineb alamsõnede või alamsekventsidena. Eesmärgiks oli luua meetod, mis leiab dialoogikorpusest dialoogiaktidele iseloomulikud alamsõned ja alamsekventsid ning kasutab neid seejärel dialoogiaktide tuvastamiseks. Formalismina on seejuures kasutatud tõenäosuslikku sufiksipuud, mis esitab alamsõned ja -sekventsid kompaktsel moel, andes iga mustri jaoks tõenäosuse, millega see ühele või teisele dialoogiaktile viitab.

Antud töö teine osa keskendub sufiksipuude rakendamisele eestikeelsete dialoogiaktide tuvastamiseks. Ühe olulise kõrvalsaadusena annab meetod ülevaate

eestikeelsetele dialoogiaktidele iseloomulikest keelelistest mustritest. Antud tulemused võivad pakkuda lingvistilist huvi või leida edasist kasutust meetodi edasiarendustes või teistes meetodites. Mustrite täielik loetelu on esitatud antud töö elektroonsetes lisades.

Töös antakse ülevaade sufiksipuudest, nende konstrueerimisest ning kasutamisest, tutvustatakse eestikeelsete dialoogide märgendamisel kasutatavat dialoogiaktide tüpoloogiat ning viiakse läbi rida eksperimente dialoogiaktide automaatseks tuvastamiseks. Sel otstarbel on antud töö raames loodud sufiksipuude konstrueerimise ja testimise programm (Python'is). Eesti dialoogiaktide tüpoloogia on toodud lisas 1 ning läbiviidud eksperimentide tähtsamad tulemused lisades 2 kuni 4. Programm ning eksperimentide tulemused täielikul kujul on esitatud töö elektroonsetes lisades.

2 Dialoogiaktid

Üksteisega suheldes väljendavad inimesed keele abil erinevaid asju: esitavad küsimusi või palveid, annavad millegi kohta infot, tervitavad, tänavad jne. Selliseid keele abil tehtavaid tegevusi nimetatakse dialoogiaktideks.

Veel mõni aeg tagasi valitses keelefilosoofias arusaam, et laused on üksnes faktide väljaütlemised (“*taevas on sinine”*, “*koerad hauguvad”*), mis võivad olla kas tõesed või väärad. Seda vaadet ründas Austin [Traum 1999], kes väitis, et lausete peamiseks ülesandeks ei ole mitte faktide edastamine, vaid maailma muutvate tegevuste läbiviimine.

Näiteks, öeldes “*ma määran sulle eluaegse vanglakaristuse”*, sooritab rääkija kellegi vangisaatmise akti. Tegu pole lihtsalt fakti nentimisega, vaid teoga, mis muudab maailma olekut. Tagajärjed ei pea alati nii radikaalsed või ilmsed olema. Enamik lauseid mõjutavad üksnes rääkijat või kuulajat, muutes nende meelteseisundit. Näide sellest on lause “*vaata ette*”, mille eesmärgiks on muuta kuulaja meelteseisundit selliselt, et too muutuks ettevaatlikuks.

Austin eristas mitut tegevuste kategooriat, mida kõneledes sooritatakse. Peamised kategooriad on lokutiivsed, illokutiivsed ja perlokutiivsed aktid (ingl k vastavalt *locutionary acts*, *illocutionary acts* and *perlocutionary acts*) [Jurafsky & Martin 2000],[Traum 1999].

Lokutiivne akt on millegi ütlemise akt. See viitab helide ja sõnade tekitamisele, et öelda midagi tähenduslikku. Näiteks, öeldes “*heida pikali”*, sooritab *kõneleja lokutiivse akti, mis* informeerib kuulajat sellest, et too pikali heidaks.

Lokutiivne akt võib samal ajal olla ka illokutiivne akt. Viimane viitab aktile, millel on teatud kindel funktsioon: informeerimine, palve, hoiatamine jne. Näiteks öeldes “*heida pikali*”, sooritab kõneleja illokutiivse akti, kus ta hoiatab kedagi (hoiatus) või soovitab kellelgi puhata (soovitus). Suurem osa kõneaktidega tehtavast tööst keskendub just illokutiivsetele aktidele [Traum 1999]. Nii ka antud töö.

Kolmandaks peamiseks aktiliigiks on perlokutiivne akt, mis viitab mõjule, mida lause kuulajale omab. Näiteks öeldes “heida pikali”, võib see mõjutada kuulaja meelteseisundit selliselt, et too heidabki pikali.

Austin nimetas kõiki eelpoolnimetatud tegevusi koondnimega kõneakt (ingl k *speech act*). Peale kõneaktide on kasutusel veel mitu terminit, mis viitavad üldjoontes samale nähtusele. Nende terminite seas on dialoogiaktid, suhtlusaktid (ingl k

communicative acts), vestlusaktid (ingl k *conversation acts*) ja suhtluskäigud (conversational moves) [Traum 2000]. Antud töös viitame sellele nähtusele kui dialoogiaktidele. Dialoogiaktide märgendamise korral kasutatakse tihti ka terminit funktsionaalne märgendamine, mis viitab üldiselt illokutiivsete aktide märgendamisele.

3 Dialoogiaktide tüpoloogia

Dialoogiaktide eristamiseks on loodud mitmeid tüpoloogiaid. Antud töö põhineb eesti dialoogiaktide tüpoloogial (lühendatult EDiT), mille aluseks on mitme projekti raames kogutud eestikeelsete dialoogide kogum, mida kutsutakse Eesti Dialoogikorpuseks. 2006. aasta detsembri seisuga sisaldas see 172410 sõnast koosnevat 962 litereeritud teksti, millest suurema osa moodustasid telefoni teel peetud infodialoogid. Dialoogide täpne jaotus on järgmine [EDiC].

- Telefonivestlused 846
  - informatsioon 472
  - reisibüroo 86
  - polikliiniku registratuur 97
  - avalik teenus (pank, kino, spordiklubi, külalistemaja jne) 50
  - telefonimüük 45
  - taksopark 23
  - bussijaam 16
  - riigiasutus 10
  - pood 19
  - ülikool 7
  - raamatukogu 5
  - haigla 3
  - arsti kabinet 1
  - muu 12 (helistamised raadiosaatesse, valed numbrid jne)
- Silmast silma vestlused 116
  - pood 63
  - teejuhiste andmine tänaval 20
  - avalik teenus (postkontor, kinnisvarabüroo, juuksur jne) 15
  - reisibüroo 10
  - muu (polikliiniku registratuur, ülikool, bussijaam, veterinaarkeskus jne) 8

Kuna EDiT on loodud Eesti Dialoogikorpuse põhjal, siis on ka nimetatud tüpoloogia mõeldud ennekõike infodialoogide jaoks. Tüpoloogia koostamisel on

lähtutud kahest üldisest printsiibist [Hennoste & Rääbis 2004]. Esiteks, aktide kategooriad peavad olema selgelt eristatavad ning teiseks, aktide kategooriad peavad olema ammendavad, st. iga dialoogiakti jaoks peab leiduma kategooria, millesse ta kuulub. EDiT-is on ühtekokku 126 erinevat dialoogiakti.

Nagu on tavaks, eristatakse ka antud tüpoloogias akte ennekõike nende funktsiooni, mitte keelelise vormi põhjal. Erandiks on vaid küsimused, mis on määratud just keelelise vormiga. Väga selgete eristuskriteeriumite puudumise tõttu leidub akte, mida ei saagi üheselt määrata. Kui samade dialoogide aktitüüpe määrab korraga kaks inimest, siis kasutatakse aktimäärangute usaldusväärsuse hindamiseks nii-nimetatud κ-koefitsienti, mis arvutatakse valemiga

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)},$$

kus $P(A)$ on tegelike nõustumiste ja $P(E)$ juhuslike samameelsuste osakaal [Carletta 1996].

Eesti Dialoogikorpuse puhul on saadud κ väärtuseks 0,74. Seejuures loetakse heaks tulemuseks väärtusi, mis on suuremad kui 0,8. Mengel jt on andnud mitme hästi tuntud tüpoloogia võrdluse, sealhulgas ka nende usaldusväärsuse hinnangud [Mengel et al. 2000]. Nõustumiste osakaal jääb neis 77% ja 98% vahele ning κ väärtus 0,56 ja 0,85 vahele.

Kuna dialoogiaktide piirid ei pruugi alati kattuda kirjaliku keele lausega, siis on EDiT-i puhul võetud analüüsiüksuseks lausung. Lihtsustatult öeldes on lausung terviklik suhtlusüksus, mille järel läheb vestlusjärg potentsiaalselt teisele vestluspartnerile üle.

Laias laastus jagab EDiT dialoogiaktid kaheks: naaberpaare moodustavateks aktideks ja üksikaktideks. Esimeste alla kuuluvad aktid, mis esinevad vestluses tavaliselt koos, näiteks tervitus ja vastutervitus. Naaberpaare moodustavatest aktidest esimest nimetatakse eesliikmeks ning talle järgnevat akti järelliikmeks. Üksikaktide alla kuuluvad aktid, mille esinemine dialoogis ei ole otseselt mõne teise aktiga seotud.

Naaberpaare moodustavad aktid ja üksikaktid jagunevad omakorda makro- ja mikrotasandi aktideks. Makrorühma moodustavad aktid, mis jagavad dialoogi väiksemateks alamüksusteks. Siia alla kuuluvad nii teemavahetused kui ka rituaalid.

Üldine dialoogiaktide tüpoloogia on järgmine [Hennoste & Rääbis 2004]:

1. Naaberpaare moodustavad aktid
    1.1. Makrotasandi aktid
        1.1.1. Naaberpaare moodustavad rituaalid
        1.1.2. Teemavahetus
    1.2. Mikrotasandi aktid
        1.2.1. Probleemide lahendamise aktid
            1.2.1.1.Partneri algatatud parandused
            1.2.1.2.Vastuse tingimuste täpsustamine
            1.2.1.3.Kontakti kontroll
        1.2.2. Naaberpaare moodustavad infoaktid
            1.2.2.1.Direktiivid
            1.2.2.2.Küsimused
            1.2.2.3.Seisukohavõtud
2. Üksikaktid
    2.1. Makrotasandi aktid
        2.1.1. Üksikrituaalid
    2.2. Mikrotasandi aktid
        2.2.1. Probleemide lahendamise aktid
            2.2.1.1.Üksi tehtavad parandused
        2.2.2. Üksik-infoaktid
            2.2.2.1.Infoaktid (sh. Mitteinterpreteeritavad aktid)
            2.2.2.2.Infolisad
            2.2.2.3.Vabatahtlikud reaktsioonid

Tüpoloogias on antud igale aktirühmale ja aktile omaette nimi. Iga akti puhul on määratud rühm, millesse ta kuulub. Näiteks tähistatakse tähekombinatsioonidega KYE ja RIJ vastavalt küsimuse eesliiget ja rituaali järelliiget. Lisaks antakse igale aktile tema täpset funktsiooni kajastav nimi. Näiteks tähistab nimi KYE: AVATUD avatud küsimust ning RIJ: VASTUTERVITUS rituaalset akti vastutervitus. Märgendatud dialoog ise näeb välja selline:

((457b11 infotelefon))
((ühtlustas Andriela Rääbis 28.11.2004))

((kutsung)) | RIE: KUTSUNG |
V: info`telefon= | RIJ: KUTSUNGI VASTUVÕTMINE | | RY: TUTVUSTUS |
Ker[sti= | RY: TUTVUSTUS |
tere] | RIE: TERVITUS |
H: [tere= | RIJ: VASTUTERVITUS |
palun]=`öelge kuskohas Pärnus on ilusalong Di`one. | KYE: AVATUD |
(.)
V: üks=`hetk? | KYJ: EDASILÜKKAMINE |
(...) .hh e=`Strand o`tellis. | KYJ: INFO ANDMINE |
H: jaa? | VR: NEUTRAALNE VASTUVÕTUTEADE |
aitäh=teile? | RIE: TÄNAN |
V: palun= | RIJ: PALUN |
H: =kõike=head | RIE: SOOVIMINE |

Põhjalikumalt võib eestikeelsete dialoogiaktide tüpoloogia kohta lugeda allikast [Hennoste & Rääbis 2004]. Dialoogiaktide tüpoloogia EDiT on toodud ka lisas 1.

4 Dialoogiaktide automaatne tuvastamine sufiksipuude abil

4.1 Ülevaade ja motivatsioon

Dialoogiaktide tuvastamise meetodid võib jagada üldjoontes kaheks: reeglipõhised meetodid (ingl *k inference models*) ja statistilised meetodid. Reeglipõhised meetodid teevad järeldusi kõneleja kavatsuste kohta ning tuvastavad dialoogiaktid sellest lähtuvalt. Kuigi tegu on võimsa meetodiga, nõuab see aeganõudvat reeglite käsitsi kirjapanekut.

Statistilised meetodid tuvastavad dialoogiakte pindmiste struktuuride, nagu näiteks võtmesõnad või bi- ja trigrammid, abil. Tuntumad statistilised meetodid on Bayes'i klassifitseerija, Markovi peitmudel, neurovõrgud, transformatsioonipõhine õppimine (ingl k *transformation-based learning*) ning otsustuspuud. Tegu on juhendatavate meetoditega, mis tähendab, et meetodid vajavad käsitsi märgendatud korpust.

Stockle jt annavad ülevaate dialoogiaktide tuvastamise eksperimentidest [Stockle et al. 2000]. Tuvastamise täpsus varieerub neis 40% ja 81% vahel. Seejuures on tegu on tüpoloogiatega, mis sisaldavad 6 kuni 42 dialoogiakti.

Antud töö autor on varem tegelenud eestikeelsete dialoogiaktide tuvastamisega otsustuspuude abil [Kikas 2005][Fishel & Kikas 2006]. Otsustuspuudega saavutatud saagis jääb seejuures 32% ja 45% vahele sõltuvalt tunnuste valikust: kasutatud on võtmesõnu, morfoloogilise analüüsi tulemusi, bi- ja trigramme ning diskursuse tunnuseid.

Üks eksperimentide käigus tehtud tähelepanek oli, et paljud tunnused esinevad pikemate alamsõnede või alamsekventsidena[1]. Sellest tekkis idee konstrueerida meetod, mis leiaks korpusest dialoogiakte iseloomustavad alamasõned ja –sekventsid ning kasutaks seda informatsiooni dialoogiaktide tuvastamiseks. Käesoleva töö edasine osa keskendubki selle meetodi kirjeldamisele ning Eesti Dialoogikorpusel läbiviidud eksperimentide tulemustele.

Selles peatükis esitame kolm statistilist mudelit: sufiksipuu, tõenäosuslik sufiksipuu ja metamärke (ingl k *wildcards*) sisaldav tõenäosuslik sufiksipuu. Esimene

[1] Alamsekventsi täpne definitsioon on antud paragrahvis 4.2.

neist on antud ennekõike sufiksipuu mõiste illustreerimiseks. Kaks ülejäänut omavad aga ka praktilist rolli dialoogiaktide automaatsel tuvastamisel. Kõige olulisem neist on siiski viimane, metamärke sisaldav sufiksipuu, mis üldistab kahte esimest mudelit. Mudelid on esitatud alustades lihtsaimast ja lõpetades keerukaimaga. Terminoloogilise segaduse vältimiseks rõhutame juba siinkohal, et antud töös kasutatud sufiksi mõiste erineb sufiksi mõistest lingvistikas. Selle täpne definitsioon ja vaatluse all olevate mudelite definitsioonid on esitatud järgnevalt.

4.2 Mõisted

Enne sufiksipuu mõiste defineerimist anname esmalt mõned abimõisted, sealhulgas ka sufiksi mõiste [Sufiks].

Definitsioon. Tähestikuks nimetatakse lõplikku hulka Σ, selle elemente nimetatakse sümboliteks.

Definitsioon. *Sõneks* üle tähestiku Σ nimetatakse lõplikku jada $S = s_1 s_2 s_3 \ldots s_n$, kus iga $i \in \{1,2,\ldots n\}$ korral $s_i \in \Sigma$.

Definitsioon. *Sõne pikkuseks* nimetatakse sõnele vastava sümbolite jada elementide arvu.

Definitsioon. *Tühi sõne* on sõne, mille pikkus on 0, seda tähistatakse ε.

Definitsioon. Olgu antud sõne $S = s_1 s_2 s_3 \ldots s_n$. Sõne S *alamsõneks* nimetatakse sõnet $S_{i\ldots j} = s_i s_{i+1} s_{i+2} \ldots s_j$, kus $1 \le i \le j \le n$.

Definitsioon. Olgu antud sõne $S = s_1 s_2 s_3 \ldots s_n$. Sõne S *alamsekventsiks* nimetatakse sõnet $S_{i_1 \ldots i_j} = s_{i_1} s_{i_2} s_{i_3} \ldots s_{i_j}$, kus $1 \le i_1 < i_2 < \ldots < i_j \le n$.

Definitsioon. Olgu antud sõne $S = s_1 s_2 s_3 \ldots s_n$. Sõne S *i-ndaks sufiksiks* nimetatakse alamsõnet $S_i = S_{i\ldots n} = s_i s_{i+1} s_{i+2} \ldots s_n$, kus $1 \le i \le n$.

Definitsioon. Olgu antud sõne $S = s_1 s_2 s_3 \ldots s_n$. Sõne S *i-ndaks prefiksiks* nimetatakse alamsõnet $P_i = S_{1\ldots i} = s_1 s_2 s_3 \ldots s_i$, kus $1 \le i \le n$.

Toodud mõistete illustreerimiseks anname järgmise näite. Olgu antud tähestik $\Sigma = \{t, ö\}$ ja sõne $S = ööt öö$. Antud sõnele vastab kuus sufiksit:

$S_1 = ööt öö = S$,

$S_2 = öt öö$,

$S_3 = t öö$,

$S_4 = öö$,

$S_5 = ö$,

$S_6 = \varepsilon$.

Nagu näha, on toodud sufiksi definitsioon laiem kui sufiksi mõiste lingvistikas, kus see tähendab sõna tüvele liituvat osa. Vastavas kirjanduses on küll sagedamini kasutatav samatähenduslik järelliite mõiste.

Ka tähestiku mõiste on võib-olla laiem kui oleme tavaliselt harjunud. Tähestik ei pruugi ilmtingimata koosneda tavalisest tähemärkidest, nagu *a, b, c* jne. Analoogselt eelmisele näitele võime konstrueerida sõne tervetest sõnadest. Näiteks, võttes tähestikuks $\Sigma = \{ainult, on, pole, viga, üks, ülesandes\}$ võime koostada sõne $S = pole\ viga\ ülesandes\ on\ ainult\ üks\ viga$, millele vastavad sufiksid

$S_1 = pole\ viga\ ülesandes\ on\ ainult\ üks\ viga = S$,

$S_2 = viga\ ülesandes\ on\ ainult\ üks\ viga$,

$S_3 = ülesandes\ on\ ainult\ üks\ viga$,

$S_4 = on\ ainult\ üks\ viga$,

$S_5 = ainult\ üks\ viga$,

$S_6 = üks\ viga$,

$S_7 = viga$.

$S_8 = \varepsilon$

4.3 Sufiksipuu

Sufiksipuu[2] (ingl k *suffix tree*) ja selle laiendused on antud töö jaoks keskse tähtsusega andmestruktuurid. Lühidalt öeldes on sufiksipuu puukujuline andmestruktuur, mis esitab sõne või sõnede sufikseid kompaktsel moel.

Sufiksipuu täpne definitsioon võib kirjanduses varieeruda. Meie oleme lähtunud suhteliselt vabast sufiksipuu definitsioonist, mis on järgmine.

Definitsioon. Sõne $S = s_1 s_2 s_3 \ldots s_n$ sufiksipuuks nimetatakse puud, millel on järgmised omadused:

- Puul on juur
- Puu on suunatud
- Puus leiduvad tipud märgenditega[3] $1,2,\ldots,n$
- Iga serv on märgendatud sõne S mittetühja alamsõnega
- Tipust väljuvatele servadele peavad vastama erinevad märgendid
- Servade märgendid juurest tipuni i, $i \in \{1,2,\ldots,n\}$, moodustavad sõne S i-nda sufiksi $S_i = s_i s_{i+1} s_{i+2} \ldots s_n$

Juhime tähelepanu sellele, et viimane tingimus garanteerib, et kui kahel sufiksil on ühine prefiks, siis leidub puus täpselt üks sellele prefiksile vastav juurest lähtuv tee. Sufiksipuu definitsioonist on näha, et sufiksipuu esitab sõne *S* kõiki sufikseid.

---

[2] Ingliskeelses kirjanduses viidatakse antud töös sufiksipuuks nimetatud andmestruktuurile ka kui atomaarsele sufiksipuule (ingl k *atomic suffix tree*) [Giegerich & Kurtz 1995] või kasutatakse ka terminit *suffix-trie*. Viimasel juhul võidakse (aga ei pruugi) eristada kahte andmestruktuuri: *suffix tree* ja *suffix-trie*. *Suffix-trie* on selline sufiksipuu, mille igal seesmisel tipul, mis ei ole juur, on vähemalt kaks järglast. (Sõna *trie* on tuletatud sõnast *retrieval*.)

Definitsioon garanteerib, et tavalises *sufiks-tries* muidu ühe järglasega olevad tipud "liidetakse kokku". Selline esitus on kompaktsem ning see on eelistatud lähenemisviis paljudes rakendustes. Miinuseks on keerukamad puu konstrueerimise algoritmid ning võimetus siduda iga üksiku sufiksi sümboliga lisainformatsiooni.

Terminil *suffix-trie* ei ole otsest vastet eesti keeles, kuna *trie* on tõlgitud lihtsalt kui prefiksipuu, mis antud konteksti loomulikult ei sobi.

[3] Puu tippude märgendamine annab meile mugava võimaluse sufiksipuu definitsiooni esitamiseks, kuid vähemalt antud töö raames ei oma märgendatud tipud praktilist tähtsust.

Ilmne, kuid oluline omadus on, et ühtlasi sisaldab sufiksipuu ka kõiki sõne *S* alamsõnesid, kuna iga alamsõne on mõne sufiksi prefiksiks. Seda teadmist ära kasutades anname üldistatud sufiksipuu definitsiooni mitme sõne jaoks (näide sellisest sufiksipuust on toodud joonisel 4.2).

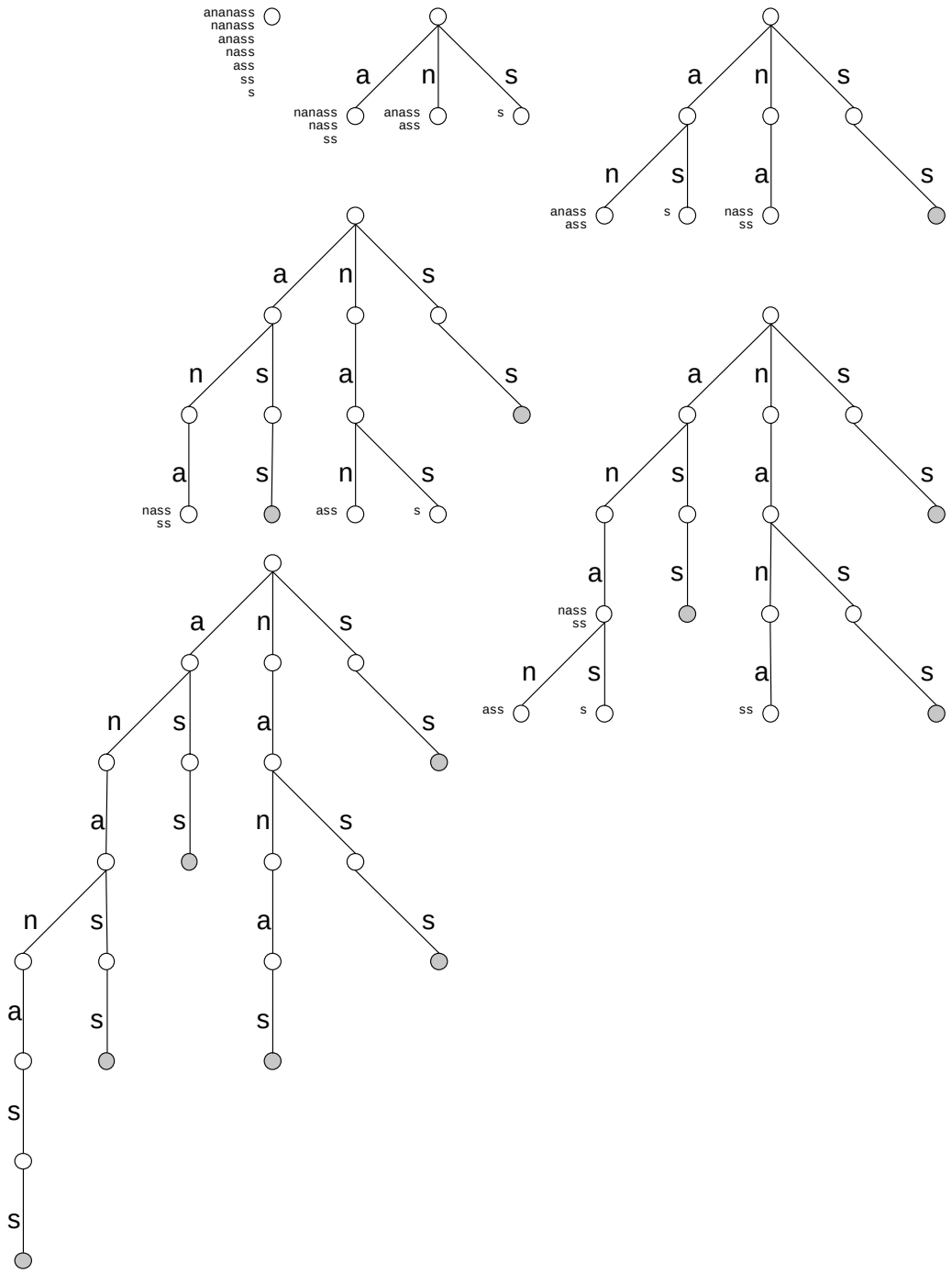
Definitsioon. Olgu antud sõnede hulk $C_1 = \{S_{1_1}, S_{1_2}, ..., S_{1_n}\}$ ning sellele vastav alamsõnede hulk $T = \{T_1, T_2, ..., T_k\}$. Sõnede hulga C_1 sufiksipuuks nimetame puud, millel on järgmised omadused

- Puul on juur
- Puu on suunatud
- Puus leiduvad tipud märgenditega $1,2,\ldots,k$
- Iga serv on märgendatud sümboliga hulgast $\Sigma/\{\varepsilon\}$
- Tipust väljuvatele servadele peavad vastama erinevad märgendid
- Servade märgendid juurest tipuni i, $i \in \{1,2,\ldots,k\}$, moodustavad alamsõne T_i

Sufiksipuude peamiseks kasutusalaks on tekstiotsing. Otsingupõhimõte on iseenesest lihtne. Esiteks koostatakse otsitavast tekstist sufiksipuu, millest toimub seejärel meid huvitava sõne otsing. Otsingut alustatakse puu juurest, võrreldes otsingusõne esimest sümbolit juurest väljuvate servade märgenditega. Kui leidub vastavus, liigutakse järgmisesse tippu ja võrreldakse otsingusõne teist sümbolit sellest tipust väljuvate servade märgenditega. Otsing lõpeb, kui kõik otsingusõned on läbi vaadatud või on enne jõutud lõpusümbolini. Esimene juht tähendab otsingu õnnestumist, teine juht otsingu ebaõnnestumist.

4.3.1 Sufiksipuu konstrueerimise algoritm

Sufiksipuude konstrueerimiseks on olemas mitmeid algoritme. Neist tuntumad on Weineri [Weiner 1973], McCreight' [McCreight 1976], Ukkoneni [Ukkonen 1995] ning Giegerichi ja Kurtzi [Giegerich & Kurtz] algoritmid. Antud töös oleme lähtunud just viimasest. Seda ennekõike selle arusaadavuse ja laiendatavuse pärast.



Joonis 4.1. Sufiksipuu konstrueerimine sõne *ananass* jaoks.

Kuigi algoritmi keskmine ajaline keerukus on $O(n \log n)$ (kus n on teksti pikkus) ning halvimal juhul isegi $O(n^2)$, on Giegerich ja Kurtz näidanud, et algoritm võib praktikas võistelda ka teiste, lineaarajas töötavate algoritmidega [Giegerich & Kurtz 1999].

Seletame sufiksipuu konstrueerimise algoritmi näite varal. Olgu antud sõne *ananass* ja ühe tipuga (juurega) puu (joonis 4.1). Sõnele vastab sufiksite hulk $\{ananass, nanass, anass, nass, ass, ss, s, \varepsilon\}$. Algoritmi esimeseks sammuks on jagada antud hulk sufiksite esimeste sümbolite põhjal alamhulkadeks. Selle operatsiooni tulemuseks on alamhulgad $\{ananass, anass, ass\}$, $\{nanass, nass\}$, $\{ss, s\}$ ja $\{\varepsilon\}$. Järgmise sammuna lisame puusse kolm uut tippu ning ühendame need puu juurega, kasutades märgenditega *a*, *n* ja *s* servi. Seejärel kustutame alamhulkade elementide esimesed sümbolid ning saame hulgad $\{nanass, nass, ss\}$, $\{anass, ass\}$, $\{s\}$.

Toimime nüüd iga saadud hulgaga samuti nagu esialgse sufiksite hulgaga, st jagame hulga $\{nanass, nass, ss\}$ hulkadeks $\{nanass, nass\}$ ja $\{ss\}$. Hulki $\{anass, ass\}$ ja $\{s\}$ ei saa jagada ning need jäävad samaks. Seejärel lisame puusse neli uut tippu ning servad märgenditega *n*, *s*, *a* ja *s* selliselt, et servade märgendid alates juurest kuni viimati lisatud tippudeni moodustaksid sõned *an*, *as*, *na*, *ss*. Nüüd kustutame viimati saadud alamhulkade elementide esimesed sümbolid ning kordame kirjeldatud protseduuri uuesti. Konstrueerimine lõpeb, kui alles on jäänud üksnes tühje sõnesid sisaldavad hulgad.

Järgnevalt esitame algoritmi üldistatud sufiksipuu konstrueerimiseks, kasutades selleks Python'ile sarnanevat pseudokoodi.

```
1    rootNode = Node()
2    queue.append(rootNode)
3    while queue:
4        node = queue.pop(0)
5        if node == rootNode:
6            nextWords = allWords()
7        else:
8            nextWords = []
9            for word in node.words:
10               nextWord = word.nextWord
11               if nextWord:
12                   nextWords.append(nextWord)
13       subsets = getSubsets(nextWords)
14       for subset in subsets:
15           child = Node()
16           child.label = subset[0].string
17           child.words = subset
18           node.children.append(child)
19           queue.append(child)
```

Algoritmi tööpõhimõte on järgmine.

1. Loo juurtipp *rootNode* (rida 1).
2. Lisa juurtipp *rootNode* järjekorda *queue* (rida 2).
3. Kuni järjekord *queue* pole tühi, võta sealt esimene tipp *node* (read 3-4).
4. Loo sõnade massiiv *nextWords*.
   4.1. Kui *node* on juurtipp, moodusta *nextWords* kõikidest korpuses esinevatest sõnadest (rida 6).
   4.2. Kui *node* ei ole juurtipp, leia *node.words* sõnadele vahetult järgnevad sõnad ja loo neist massiiv *nextWords* (read 8-12).
5. Jaga *nextWords* leksikograafiliselt võrdsete sõnade alamhulkadeks ning loo alamhulkadest massiiv *subsets* (rida 13).
6. Iga *subsets* massiivi kuuluva mittetühja alamhulga *subset* jaoks tee järgmist:
   6.1. Loo uus tipp *child* (rida 15).
   6.2. Pane tipule *child* märgendiks *subset* hulgale vastav sõna (rida 18).
   6.3. Seo sellega sõnade alamhulk *subset* (rida 17).
   6.4. Muuda tipp *child* tipu *node* järglaseks (rida 18).
   6.5. Lisa tipp *child* järjekorda *queue*. (rida 19).
7. Mine 3. sammule.

Üks asjaolu, millele algoritmi juures tähelepanu juhime, on see, et erinevalt eelpool toodud sufiksipuu definitsioonist, ei ole me kasutanud servadele vastavaid olemeid. Selle asemel lisatakse märgendeid tippe esitavatele olemitele. Seda selleks, et vältida liigseid olemeid ja hoida algoritmi võimalikult lihtsana. Tipule *child* antud märgendit tuleks seega interpreteerida kui antud tippu siseneva serva märgendit. Kuna tegu on puukujulise andmestruktuuriga, ei saa selliseid servi olla rohkem kui üks ning seega on interpretatsioon ühene.

4.4 Tõenäosuslik sufiksipuu

Antud töö eesmärgiks on konstrueerida sobiv mudel dialoogiaktide automaatseks tuvastamiseks. Juhul kui meie käsutuseks oleks lõpmatult suur korpus, mis sisaldaks kõikvõimalikke dialooge (just dialooge, mitte üksikuid akte), siis oleks sufiksipuu ilmselt sobiv mudel dialoogiaktide automaatseks tuvastamiseks. Sellisel

juhul saaksime konstrueerida iga aktiklassi jaoks omaette sufiksipuu. Akti tuvastamine seisneks siis lihtsalt aktile vastava sufiksipuu ülesotsimises.

Reaalsuses on meil kasutada piiratud dialoogikorpus, mis tähendab, et me ei saa loota üksnes näidetele, vaid peame tegema ka üldistusi, mis võimaldaksid laiendada korpuses olevat infot seni esinemata juhtudele. Lisaks ei maksa ka unustada, et korpuses esineb juhte, kus vormiliselt sama lausung võib viidata erinevatele aktidele (näiteks EDiT-is eristatakse tervitust ja vastutervitust). Sellest tulenevalt konstrueerime dialoogiaktide automaatse tuvastamise mudeli tõenäosuslikuna.

Tõenäosusliku sufiksipuu kontseptsiooni pakkusid välja [Bejerano & Yona 1999], toetudes varasemale tõenäosusliku sufiksiautomaadi kontseptsioonile (ingl k *probabilistic suffix automaton* ehk PSA) [Ron *et al.* 1996].

Samas märgime, et antud töös kasutatud meetodi esialgne kontseptsioon, mis on sufiksipuu laiendus, tekkis käesoleva töö autoril sõltumatult nimetatud kirjandusest ja sellelaadsest kirjandusest üldse, olles peamiselt motiveeritud varasematest Eesti Dialoogikorpusel läbiviidud otsustuspuude eksperimentide tulemustest [Kikas 2005], mille järeldusena leiti, et märksõnad ning bi- ja trigrammid pole piisavad, et haarata kogu olulist informatsiooni.

Tõenäosuslik sufiksipuu sarnaneb oma põhiosas tavalisele sufiksipuule. Erinevus on selles, et tõenäosuslikus sufiksipuus on servade ja tippudega seotud tõenäosused, mis iseloomustavad alamsõnede esinemise ja klassidesse kuulumise tõenäosusi. Tõenäosuslikud sufiksipuud on esitatud joonistel 4.2 ja 4.3.

Definitsioon. Olgu antud tähestik Σ, sõnede multihulgad[4] $C_1, C_2, ..., C_n$ ning neile vastavad kõikvõimalike alamsõnede multihulgad

$$D_1 = \left\{T_1^{m_{1_1}}, T_2^{m_{2_1}}, ..., T_k^{m_{k_1}}\right\}, D_2 = \left\{T_1^{m_{1_2}}, T_2^{m_{2_2}}, ..., T_k^{m_{k_2}}\right\}, ..., D_n = \left\{T_1^{m_{1_n}}, T_2^{m_{2_n}}, ..., T_k^{m_{k_n}}\right\}$$

Sõnede multihulkade $C_1, C_2, ..., C_n$ tõenäosuslikuks sufiksipuuks nimetame puud, millel on järgmised omadused

- Puul on juur
- Puu on suunatud
- Puus leiduvad tipud märgenditega $1,2,\ldots,k$

[4] Multihulk on hulga mõiste üldistus, mille korral elemendid ei tarvitse olla üksteisest erinevad [Kaasik 2003]. Multihulka tähistatakse kujul $\left\{a_1^{m_1}, a_1^{m_2}, ..., a_k^{m_k}\right\}$, kus $a_1, a_2, ..., a_k$ on hulga elemendid ning $m_1, m_2, ..., m_k$ nende eksemplaride arvud multihulgas.

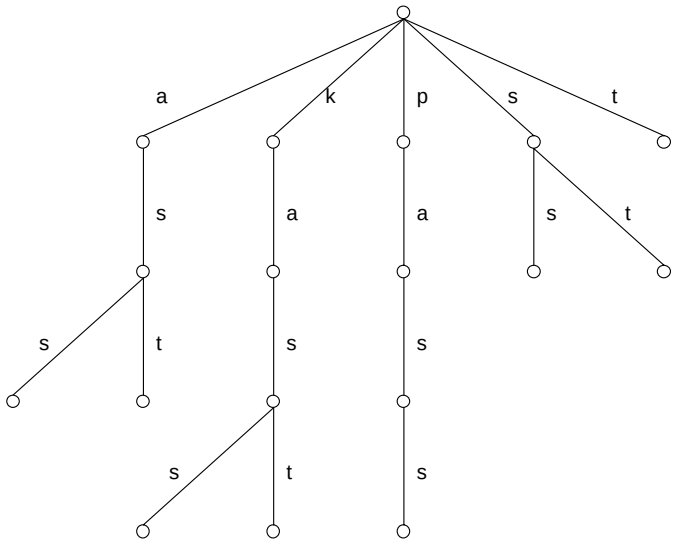
- Iga serv on märgendatud sümboliga hulgast $\Sigma/\{\varepsilon\}$
- Tipust väljuvatele servadele peavad vastama erinevad märgendid
- Servade märgendid juurest tipuni i, $i \in \{1,2,\ldots,k\}$, moodustavad alamsõne T_i
- Iga servaga on seotud tõenäosus, nii et juurest tippu i viivate servade tõenäosuste korrutis on võrdne alamsõne T_i esinemistõenäosusega multihulkade $C_1, C_2, ..., C_n$ ühendis
- Iga tipuga i on seotud tõenäosused $p_{i_1}, p_{i_2}, \ldots, p_{i_k}$, nii et

$$p_{i_j} = \frac{m_{i_j}}{m_{i_1} + m_{i_2} + ... + m_n}$$

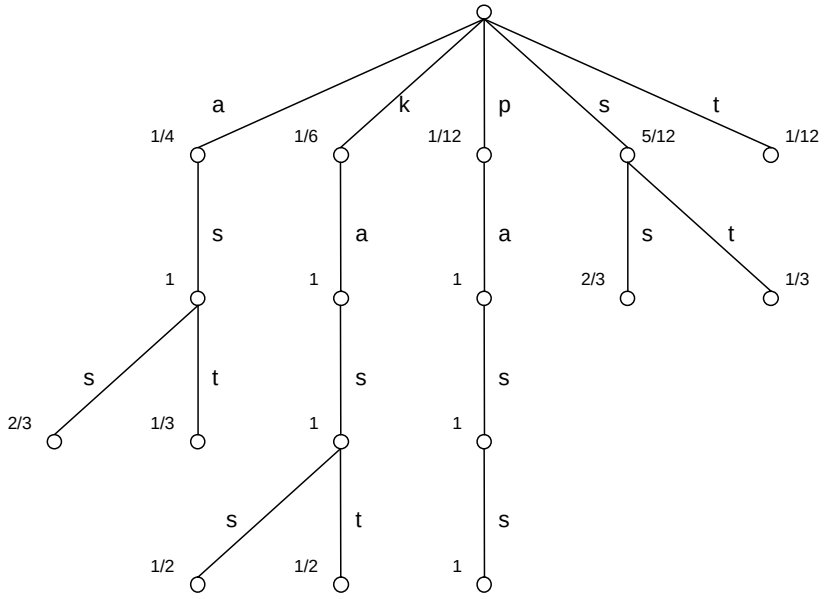
Definitsioonis on kõige olulisemad kaks viimast punkti. Eelviimane punkt ütleb, et suvalise alamsõne jaoks saab leida puust tema esinemise tõenäosuse. Viimasest punktist tuleneb, et suvalise sõne ja suvalise sõnede multihulga jaoks on antud tõenäosus, millega sõne kuulub multihulka. Mudel sarnaneb seega mõneti Markovi peitmudelile, seetõttu nimetame Markovi peitmudeli eeskujul esimest nimetatud tõenäosust *ülemineku tõenäosuseks* (ingl k *transition probability*) ja teist tõenäosust *väljundi tõenäosuseks* (ingl k *output probability*).

Dialoogiaktide tuvastamise korral on multihulkade $C_1, C_2, ..., C_n$ rollis dialoogiaktide klassid, mille elementideks on üksikud dialoogiaktid. Näiteks võib C_1 olla kõikide tervitusaktide hulk ja C_2 kõikide küsimusaktide hulk. Hulga D_1 elementideks on sellisel juhul kõik tervitustes leiduvad alamsõned ja D_2 elementideks kõik küsimustes leiduvad alamsõned.

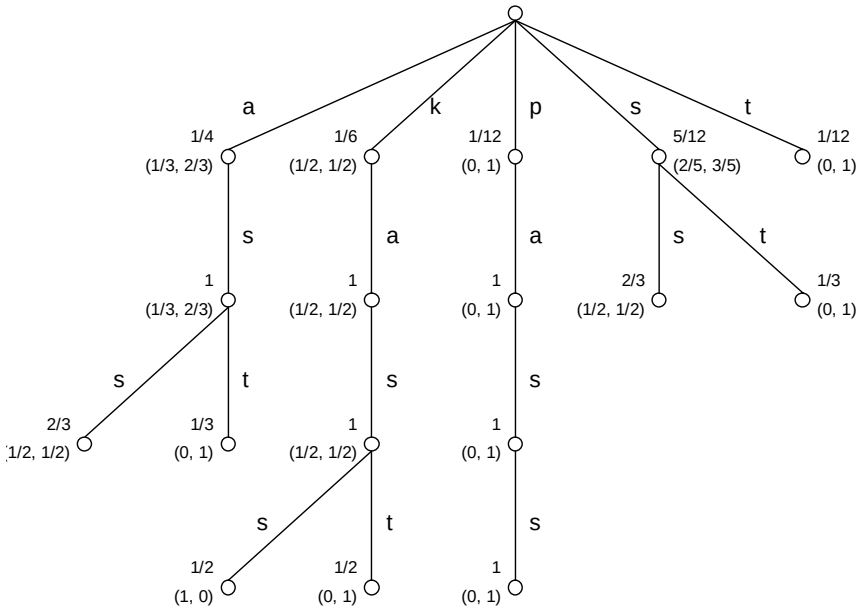
Toome ka ühe näite. Tõenäosuslik sufiksipuu multihulkade $C_1 = \{kass\}$ ja $C_2 = \{pass, kast\}$ jaoks on esitatud joonisel 3.4.



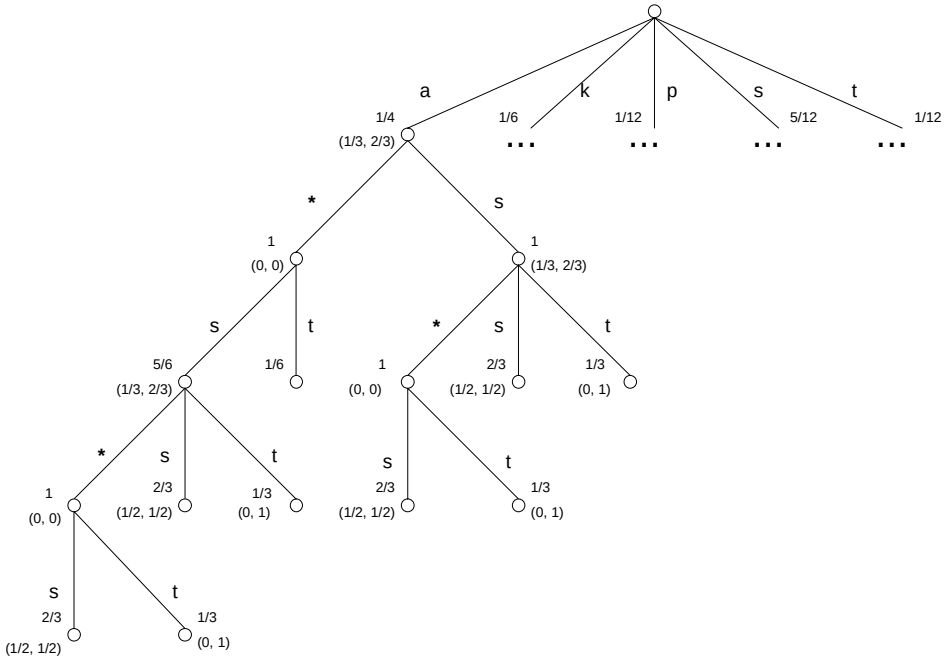
Joonis 4.2. Sõnede hulga $C_1 = \{kass, pass, kast\}$ sufiksipuu.



Joonis 4.3. Multihulga $C_1 = \{kass, pass, kast\}$ tõenäosuslik sufiksipuu.



Joonis 4.4. Multihulkade $C_1 = \{kass\}$ ja $C_2 = \{pass, kast\}$ tõenäosuslik sufiksipuu. Servade juures on antud üleminekutõenäosused ja tippude juures väljunditõenäosuste paarid, kus esimene arv on klassi C_1 kuulumise tõenäosus ja teine arv klassi C_2 kuulumise tõenäosus.



Joonis 3.5. Multihulkade $C_1 = \{kass\}$ ja $C_2 = \{pass, kast\}$ metamärke sisaldav tõenäosuslik sufiksipuu. Paneme tähele, et metamärgile * vastavad servad omavad ühega võrdset üleminekutõenäosust.

4.4.1 Tõenäosusliku sufiksipuu konstrueerimise algoritm

Järgnevalt esitame tõenäosusliku sufiksipuu konstrueerimise algoritmi, mille põhiosa on sama, mis tavalise sufiksipuu konstrueerimise algoritmi puhul. Vahe on selles, et nüüd lisatakse sufiksipuusse ülemineku- ja väljunditõenäosused.

```
1    rootNode = Node()
2    queue.append(rootNode)
3    while queue:
4        node = queue.pop(0)
5        if node == rootNode:
6            nextWords = allWords()
7        else:
8            nextWords = []
9            for word in node.words:
10               nextWord = word.nextWord
11               if nextWord:
12                   nextWords.append(nextWord)
13       subsets = getSubsets(nextWords)
14       for subset in subsets:
15           child = Node()
16           child.label = subset[0].string
17           child.words = subset
18           child.transitionProbability = len(subset)/len(nextWords)
19           child.outputProbabilities = getOutputProbabilities(subset)
20           node.children.append(child)
21           queue.append(child)
```

Algoritmi tööpõhimõte on järgmine.

1. Loo juurtipp *rootNode* (rida 1).
2. Lisa juurtipp *rootNode* järjekorda *queue* (rida 2).
3. Kuni järjekord *queue* pole tühi, võta sealt esimene tipp *node* (read 3-4).
4. Loo sõnade massiiv *nextWords*.
   4.1. Kui *node* on juurtipp, moodusta *nextWords* kõikidest korpuses esinevatest sõnadest (rida 6).
   4.2. Kui *node* ei ole juurtipp, leia *node.words* sõnadele vahetult järgnevad sõnad ja loo neist massiiv *nextWords* (read 8-12).
5. Jaga *nextWords* leksikograafiliselt võrdsete sõnade alamhulkadeks ning loo alamhulkadest massiiv *subsets* (rida 13).
6. Iga *subsets* massiivi kuuluva mittetühja alamhulga *subset* jaoks tee järgmist:

6.1. Loo uus tipp *child* (rida 15).

6.2. Pane tipule *child* märgendiks *subset* hulgale vastav sõna (rida 16).

6.3. Seo sellega sõnade alamhulk *subset* (rida 17).

6.4. Arvuta tõenäosus, et üleminekutõenäosus, et tipust *node* jõutakse tippu *child*. (rida 18)

6.5. Arvuta tipule vastava sõne klassidesse kuulumise tõenäosused (rida 19).

6.6. Muuda tipp *child* tipu *node* järglaseks (rida 20).

6.7. Lisa tipp *child* järjekorda *queue*. (rida 21).

7. Mine 3. sammule.

4.5 Metamärke sisaldav tõenäosuslik sufiksipuu

Senini oleme andnud sufiksipuu ja tõenäosusliku sufiksipuu mõisted. Tavalise sufiksipuu puudustele juhtisime juba eelnevalt tähelepanu. Tõenäosuslik sufiksipuu on dialoogiaktide tuvastamiseks tunduvalt parem mudel, siiski on ka sellel olulisi puudusi. Nimelt rajaneb dialoogiaktide iseloomustamine endiselt üksnes alamsõnedele. Loomuliku keele puhul ei ole selline eeldus põhjendatud. Üsna ilmselgelt võivad olulised tunnused esineda sõnede alamsekventsidena. Vaatame järgmist näitelauset.

Kas kärbseseened, kui nad on keedetud, on ohutud?

Lause vormilised tunnused viitavad küsimusele. Ennekõike on selliseks tunnuseks alamsekvents *kas on*. Samas on aga võimatu leida piisavalt lühikest ja ilmekat alamsõnet, mis viitaks sellele, et tegu on küsimusega. Seega teeme veel ühe laienduse ja asendame tõenäosuslikus sufiksipuus alamsõned alamsekventsidega.

Selleks, et säilitada puus ka alamsõnesid, võtame sekventside tähistamisel kasutusele metamärgi *, mis kahe sümboli vahele kirjutatuna tähendab, et sümbolite vahel võib olla veel null või enam sümbolit. Näiteks vastavad sõnele “Mis kell on?” alamsekventsid

*Mis * kell on?*
*Mis kell * on?*
*Mis * kell * on?*

*Mis * kell*

*kell * on?*

*Mis * on?*

Metamärke sisaldava tõenäosusliku sufiksipuu definitsioon on järgmine.

Definitsioon. Olgu antud tähestik Σ, sõnede multihulgad $C_1, C_2, ..., C_n$ ning neile vastavad kõikvõimalike alamsekventside multihulgad

$D_1 = \left\{T_1^{m_{1_1}}, T_2^{m_{2_1}}, ..., T_k^{m_{k_1}}\right\}, D_2 = \left\{T_1^{m_{1_2}}, T_2^{m_{2_2}}, ..., T_k^{m_{k_2}}\right\}, ..., D_n = \left\{T_1^{m_{1_n}}, T_2^{m_{2_n}}, ..., T_k^{m_{k_n}}\right\}$

Sõnede multihulkade $C_1, C_2, ..., C_n$ metamärkidega tõenäosuslikuks sufiksipuuks nimetame puud, millel on järgmised omadused

- Puul on juur
- Puu on suunatud
- Puus leiduvad tipud märgenditega $1,2,\ldots,k$
- Iga serv on märgendatud sümboliga hulgast $\Sigma \cup \{*\} / \{\varepsilon\}$
- Tipust väljuvatele servadele peavad vastama erinevad märgendid
- Servade märgendid juurest tipuni i, $i \in \{1,2,\ldots,k\}$, moodustavad alamsekventsi T_i
- Iga servaga on seotud tõenäosus, nii et juurest tippu i viivate servade tõenäosuste korrutis on võrdne alamsekventsi T_i esinemistõenäosusega multihulkade $C_1, C_2, ..., C_n$ ühendis
- Iga tipuga i on seotud tõenäosused $p_{i_1}, p_{i_2}, \ldots, p_{i_k}$, nii et

$$p_{i_j} = \frac{m_{i_j}}{m_{i_1} + m_{i_2} + ... + m_n}$$

4.5.1 Metamärke sisaldava tõenäosusliku sufiksipuu konstrueerimise algoritm

Ühe võimalusena saaksime metamärke sisaldava sufiksipuu konstrueerida kasutades täpselt sama algoritmi, mis tavalise tõenäosusliku sufiksipuu puhul. See eeldaks, et esmalt genereeritakse kõikvõimalikud sekventsid ning seejärel rakendatakse eelpool toodud algoritmi, käsitledes metamärke kui tavalisi sümboleid. Selline lähenemisviis on aga praktikas äärmiselt ebaefektiivne.

Seega täiustame tõenäosusliku sufiksipuu konstrueerimise algoritmi. Täiustatud algoritmi erinevus on selles, et nüüd vaadatakse lisaks sümbolile ja sellele vahetult järgnevale sümbolile tervet sümbolile järgnevat alamsõnet. Kõige paremini selgitab seda ilmselt algoritmi kirjeldus.

```
1    rootNode = Node()
2    queue.append(rootNode)
3    while queue:
4        node = queue.pop(0)
5        if node == rootNode:
6            nextWords = allWords()
7        else:
8            nextWords = []
9            for word in node.words:
10               nextWord = word.nextWord
11               if nextWord:
12                   nextWords.append(nextWord)
13       subsets = getSubsets(nextWords)
14       for subset in subsets:
15           child = Node()
16           child.label = subset[0].string
17           child.words = subset
18           child.transitionProbability = len(subset)/len(nextWords)
19           child.outputProbabilities = getOutputProbabilities(subset)
20           node.children.append(child)
21           queue.append(child)
22       wildcardNode = Node()
23       wildcardNode.label = '*'
24       wildcardNode.transitionProbability = 1.0
25       node.children.append(wildcardNode)
26       nextWords = []
27       for word in node.words:
28           nextWords.extend(word.nextWords())
29       subsets = getSubsets(nextWords)
30       for subset in subsets:
31           child = Node()
```

```
32            child.label = subset[0].string
33            child.words = subset
34            child.transitionProbability = len(subset)/len(nextWords)
35            child.outputProbabilities = getOutputProbabilities(subset)
36            node.children.append(child)
37            queue.append(child)
```

Algoritmi esimene osa on täpselt sama, mis tavalise tõenäosusliku sufiksipuu puhul. Seega esitame ainult uue osa kirjelduse, mis on seejuures eelnevaga paljuski sarnane.

1. Loo tipp *wildcardNode* märgendiga *, kusjuures tippu siseneva serva tõenäosus on üks (read 22-25).
2. Muuda tipp *wildcardNode* tipu *node* alamtipuks (rida 25).
3. Leia kõik *node.words* sõnadele järgnevad sõnad ja loo neist massiiv *nextWords* (read 26-28)
4. Toimi analoogselt ridadele 13-21 (read 29-37).
5. Mine 3. sammule.

4.6 Sufiksipuude rakendamine

Senini oleme rääkinud sufiksipuude konstrueerimisest Nüüd anname meetodi dialoogiaktide tuvastamiseks sufiksipuude abil.

Nagu me teame, võimaldavad sufiksipuud leida sarnasusi etteantud sõne ja treeningkorpuse näidete vahel. Teiste sõnadega saame leida kõik sõne alamsõned ja alamsekventsid (mustrid), mis esinevad sufiksipuus.

Vaatame klassifitseerimisprobleemi, kus on antud kaks klassi, C_1 ja C_2, ning neile vastavad näidete multihulgad $C_1 = \{kass\}$ ja $C_2 = \{pass, kast\}$. Meie ülesandeks on otsustada, kas sõne *laps* kuulub klassi C_1 või C_2. Jätame siinkohal kõrvale semantilised põhjendused, miks *laps* võiks neist ühte või teise kuuluda, ning vaatame, kuidas klassifitseeritakse sõne sufiksipuu abil.

Esimese sammuna tuleb leida kõik ühised sekventsid sõne *laps* ja joonisel 4.5 esitatud sufiksipuu vahel. Tulemuseks on sekventsid *a*, *a***s*, *p*, *p***s* ning *s*.

Ühtlasi huvitab meid, kui tugevalt üks või teine sekvents mingile klassile viitab. Selleks arvutame iga sekventsi jaoks üleminekutõenäosuste ja väljunditõenäosuste korrutise (tabel 4.1).

Tabel 4.1. Sõne *laps* alamsekventside tõenäosused vastavalt joonisel 3.5 esitatud sufiksipuule.

Sõne	Üleminekutõenäosuste korrutis	Väljunditõenäosused	Koondtõenäosused
a	$\frac{1}{4}$	$\left(\frac{1}{3};\frac{2}{3}\right)$	$\left(\frac{1}{12};\frac{1}{6}\right)$
*a*s*	$\frac{1}{4}\cdot 1\cdot\frac{5}{6}=\frac{5}{24}$	$\left(\frac{1}{3};\frac{2}{3}\right)$	$\left(\frac{5}{72};\frac{5}{36}\right)$
p	$\frac{1}{12}$	$(0;1)$	$\left(0;\frac{1}{12}\right)$
*p*s*	$\frac{1}{12}\cdot 1\cdot\frac{2}{3}=\frac{1}{18}$	$(0;1)$	$\left(0;\frac{1}{18}\right)$
s	$\frac{5}{12}$	$\left(\frac{7}{17};\frac{10}{17}\right)$	$\left(\frac{35}{204};\frac{25}{102}\right)$

Üleminekutõenäosuste ja väljunditõenäosuste korrutis on suurim alamsekventsi *s* ja klassi C_1 puhul. Sellest lähtuvalt võiksime liigitada sõne *s* klassi C_1. Siiski on ilmne, et ühest sümbolist koosnevad sekventsid võivad olla eksitavad. Võtame kasvõi sõna *ja*, mida esineb dialoogiaktides sagedasti, kuid mis pole ilmselgelt hea klassifitseerijana.

Seetõttu lähtume klassifitseerimisel hoopis pikimast alamsekventsist[5]. Antud näite korral on selliseid sekventse kaks, *a*s* ning *p*s*, kusjuures suurima üleminekutõenäosuste ja väljunditõenäosuste korrutise annab sekvents *a*s* klassi C_2 korral. Seega liigitame sõne *laps* alamsekventsi *a*s* esinemise põhjal klassi C_2.

Siinkohal võib tekkida küsimus, miks toimub klassifitseerimine sekventsi *a*s* põhjal, samas kui sekvents *p*s* on klassi C_2 jaoks unikaalsem. Vastus sellele küsimusele on järgmine. Kuigi *p*s* on tõepoolest unikaalsem tunnus, ei ole selle esinemises seaduspära ning antud tunnusest lähtudes teeksime järelduse üksikjuhu põhjal, mis ei ole üldistusvõimele püüdleva mudeli puhul kindlasti õigustatud.

Eelnevat kokku võttes toimub lausungile vastava dialoogiakti tuvastamine järgnevalt:

1. Leia kõik sufiksipuus esinevad lausungi alamsekventsid
2. Iga alamsekventsi jaoks tee järgmist:

[5] Sekventsi pikkuse arvestamisel ei võeta metamärke arvesse, st sekventse *ab* ja *a*b* loetakse sama pikkadeks.

    a. Arvuta sekventsile vastavate servade tõenäosuste korrutis ehk sekventsi esinemistõenäosus.
    b. Arvuta esinemistõenäosuse ja väljunditõenäosuste korrutised ehk koondtõenäosused.
3. Vali välja pikim sekvents suurima koondtõenäosusega ning anna lausungile vastav määrang.

5 Eksperimendid

5.1 Tulemuste hindamine

Järgnevalt kirjeldatud eksperimentide eesmärgiks on anda hinnang automaatsele dialoogiaktide tuvastamisele tõenäosusliku sufiksipuu abil. Seda ennekõike Eesti Dialoogikorpuse raames, kuid samas püüame anda ka hinnangu antud meetodile laiemas kontekstis. Järgnevalt anname lühikese ülevaate automaatse aktituvastuse laiemalt levinud kvaliteedimõõtudest ning probleemidest, mida nende kasutamine endaga kaasa toob.

Enimlevinud mõõdud dialoogiaktide tuvastamise kvaliteedi hindamiseks on saagis (ingl k *recall*) ja täpsus (ingl k *precision*). Nende päritolu ulatub ennekõike infootsingusse (ingl k *information retrieval*), kus need kirjeldavad leitud info hulka ja asjakohasust. Üldine infootsingu probleem on leida mingist dokumentide hulgast meile huvipakkuvad dokumendid. Sisuliselt on see klassifitseerimisprobleem, kus dokumendid jagatakse kahte klassi: olulised ja ebaolulised. Ka dialoogiaktide tuvastamine on vaadeldav infootsingu probleemina, ainult siin esinevad dokumentide rollis aktid, mis on vaja jagada tavaliselt rohkem kui kahte klassi.

Täpsus on suurus, mis iseloomustab korrektsete klassimäärangute ja kõikide klassimäärangute suhet. Tähistame korrektselt määratud aktide hulka tähega *A* ning valesti määratud aktide hulka tähega *B* (tabel 4.1). Täpsus arvutatakse valemiga

$$täpsus = \frac{|A|}{|A| + |B|}.$$

Tabel 4.1 Saagise ja täpsuse arvutamisel kasutatavad hulkade tähised.

	Oluline	Ebaoluline
Leitud	*A*	*B*
Leidmata	*C*	*D*

Saagis on suurus, mis iseloomustab saadud korrektsete määrangute ja kõikvõimalike korrektsete määrangute suhet. Saagise defineerimiseks tähistame tähega *C* kõiki olulisi akte, mis jäid ilma määranguta. Saagis ise arvutatakse valemiga

$$saagis = \frac{|A|}{|A| + |C|}.$$

Eelpool nimetatud suurustel on mitmeid häid omadusi. Esiteks on nad lihtsasti arvutatavad, teiseks on neist lihtne aru saada ning kolmandaks on need kujunenud üldiseks standardiks. Viimane annab potentsiaalse võimaluse võrrelda antud meetodit teiste sarnaste meetoditega. Siiski peame märkima, et antud suurused üksinda võetuna annavad meetodist subjektiivse pildi.

Kirjanduses rõhutatakse küllaltki sageli ainult ühte eelpoolnimetatud mõõtudest, kas saagist või täpsust. Samas ütleb juba põgus pilk valemitele, et selline hinnang on subjektiivne. Näiteks võime kunstlikult saavutada saja protsendi lähedase saagise, kui piirdume määrangute andmisel vaid juhtudega, milles võime sajaprotsendiliselt kindlad olla.

Oletame, et meil on vaja märgendada sajast aktist koosnevat testkorpust. Üldiselt ootame, et märgendamise tulemusena kannaksid kõik aktid õigeid märgendeid. Samas, kui märgendame sajast aktist ainult ühe ja teeme seda õigesti, on märgendamise täpsus, mis väljendab õigete märgendite ja kõikide märgendite suhet, automaatselt 100%. Siiski ei ole see ilmselt see, mida me märgendajalt ootaksime.

Eelnev kehtib üldjuhul ka saagise kohta. Kui anname igale aktile kõikvõimalikud määrangud, saavutame situatsiooni, kus igal aktil on vähemalt üks õige määrang. Kuna saagise arvutamisel valesid määranguid arvesse ei võeta, saavutatakse sellega ka automaatselt sajaprotsendiline saagis.

Eelnevat kokku võttes võime öelda, et piisav informatiivsus on tagatud üldjuhul vaid siis, kui esitatud on nii saagis kui täpsus. Väidame siiski, et teatud tingimuste täidetuse korral võib ka ainuüksi saagis küllaltki informatiivne olla. Saagise lubab üldjuhul kõrgeks tõsta tõsiasi, et ühele aktile on lubatud anda lõpmatult palju määranguid. Dialoogiaktide tüpoloogiad on aga valdavalt sellised, kus aktile on lubatud anda maksimaalselt üks määrang, nii ka EDiT[6].

Lähtudes nimetatud tingimusest, ei saa määrangute arv olla kunagi suurem kui aktide arv:

[6] Teatud harvadel juhtudel lubab EDiT anda dialoogiaktile ka rohkem kui ühe määrangu [Hennoste & Rääbis 2004].

$$|A|+|B| \le |A|+|C|$$

Lihtne arutluskäik näitab, et sellisel juhul ei saa ka saagis olla suurem kui täpsus:

$$\left.\begin{array}{l} täpsus = \dfrac{|A|}{|A|+|B|} \\ saagis = \dfrac{|A|}{|A|+|C|} \\ |A|+|B| \le |A|+|C| \end{array}\right\} \Rightarrow \frac{|A|}{täpsus} \le \frac{|B|}{saagis} \Rightarrow saagis \le täpsus .$$

Seega, juhul kui igale aktile on lubatud anda maksimaalselt üks määrang, on saagis täpsuse alumiseks tõkkeks. Näiteks, kui teame, et saagis on 50%, teame ühtlasi ka, et täpsus on vähemalt 50%. Vastupidine kahjuks ei kehti – ka sajaprotsendilise täpsuse korral võib saagis endiselt olla nullilähedane. Märgime siiski, et seos kehtib üksnes juhul kui räägime kõikide aktide korraga märgendamisest. Üksikute aktiklasside märgendamise korral ei pruugi see seos kehtida.

Eelnevat kokku võttes väidame, et juhul kui igale aktile on lubatud anda maksimaalselt üks määrang, on saagis informatiivsem suurus kui täpsus, olles viimase alumiseks tõkkeks. Lisaks ei ole sellistel eeltingimustel saagis kunstlikult suurendatav. Juhul, kui igale aktile antakse täpselt üks määrang, on saagis ja täpsus võrdsed.

Üldlevinud arusaama kohaselt toob täpsuse kasv kaasa saagise vähenemise ning vastupidi, saagise kasv toob kaasa täpsuse vähenemise. Alvarez viitab sellele kui "rahvateoreemile" (ingl k *folk theorem*) ja märgib, et teatud juhtudel on selle mittekehtimine ilmne [Alvarez 2002]. Selliste juhtudena nimetab ta süsteemi, mis leiab üksnes ebaolulisi kirjeid ja omab seega nulliga võrdset saagist ja täpsust, ning ideaalset süsteemi, mille täpsus ja saagis on sajaprotsendilised. Samuti esitab Alvarez näite samaaegselt ühes suunas muutuvast saagisest ja täpsusest.

Esitame veel mõned dialoogiaktide märgendamise eriomadused. Tüüpilise infootsingu probleemi korral on oluliste kirjete hulga võimsus $|A \cup C|$ märgatavalt väiksem ebaoluliste kirjete hulga võimsusest $|B \cup D|$. Dialoogiaktide märgendamise korral on pigem vastupidi, ebaolulisi kirjeid peaaegu ei olegi või on neid üsna vähesel

määral. EDiT-i puhul on näiteks lausungite vahel olevad pausid sellised, mis ei kuulu ühtegi aktiklassi.

Automaatse märgendamise puhul räägitakse tihti baasjoonest (ingl k *baseline*) või detailsemalt baastäpsusest ja baassaagisest. Tavaliselt defineeritakse need kui täpsus ja saagis, mis saavutatakse, andes kõikidele aktidele kõige sagedamini esineva akti märgendi. Eesti dialoogikorpuses on selleks KYJ: INFO ANDMINE, mille suhteline sagedus on ligikaudu 11%. Seega on ka baastäpsus ning baassaagis sama suured. Selleks, et meetod omaks praktilist väärtust, peab see andma paremaid tulemusi kui sagedaima akti järgi märgendamine.

Üldiselt on aga erinevaid meetodeid küllaltki raske, kui mitte võimatu, võrrelda, eriti kui kasutatud on erinevaid korpuseid ja märgendamisskeeme. Jättes korpuste suuruste erinevuse hetkeks kõrvale, toome Eesti Dialoogikorpuse juurde näiteks lihtsustatud SWBD-DAMSL tüpoloogia järgi märgendatud Switchboard korpuse [Stolcke 2002]. Nimetatud märgendusskeemis on 42 üksteist välistavat märgendit. Kolm enim esinevat märgendit on järgmised: STATEMENT (36%), BACKCHANNEL/ACKNOWLEDGE (19%), OPINION (13%). Eesti Dialoogikorpusega võrreldes on teatud aktiklasside sagedused märgatavalt suuremad ning seega on ka baasjoon oluliselt kõrgem. Ühtlasi tähendab see seda, et Eesti Dialoogikorpuse puhul on vaja sama saagise ja täpsuse saavutamiseks eristada oluliselt rohkem aktiklasse. Kindlat järeldust ei saa muidugi teha, sest teatud lausungite suurem sagedus ei näita veel, kas ka nende tuvastamine on proportsionaalselt parem. Siiski on mitmeid häid põhjusi selle oletamiseks. Näiteks on tagasiside aktid eristatavad küllaltki lihtsate atribuutidega. Switchboardi korpuse puhul on sellisteks tunnusteks näiteks *uh-huh* (38% tagasiside aktidest) ja *yeah* (34%). Seega annab Switchboardi korpuse puhul juba ainuüksi tagasiside aktide tuvastamine lihtsate märksõnade abil ilmselt parema saagise ja täpsuse kui Eesti Dialoogikorpuse baasjoon.

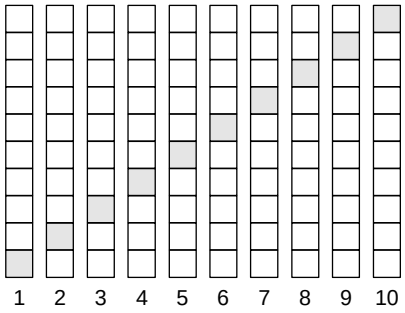
5.2 Testimismetoodika

Tavaline statistiliste masinõppesüsteemide testimismetoodika näeb ette korpuse jagamise kaheks. Esimest osa nimetatakse tavaliselt treeningkorpuseks (treeninghulgaks) ning teist osa testkorpuseks (testhulgaks). Nagu ka nimetused vihjavad, kasutatakse esimest osa masinõppesüsteemi sisendina, teist poolt aga

treeningu tulemusel saadud statistilise mudeli testimiseks. Mudeli testimist ja treenimist samal näidete hulgal loetakse üldiselt veaks. Teatud situatsioonides võib see siiski õigustatud olla. Seda näiteks siis, kui näidete hulk on täielik ning mudel peab üksnes olemasolevaid näiteid kajastama, mitte neid üldistama.

Esmapilgul lihtne ülesanne, jagada näited treeninghulgaks ja testhulgaks, ei ole praktikas aga alati väga lihtne. Üldiselt kehtib statistiliste mudelite puhul seaduspärasus, et mida rohkem treeningnäiteid, seda parem on ka tulemus. Seega, kui ohverdada oluline osa näidetest testimiseks, langeb üsna tõenäoliselt ka mudeli saagis/täpsus. Teisest küljest ei ole liiga väike testkorpus jällegi piisavalt esinduslik, et saada usaldusväärseid tulemusi.

Probleemi lahendamiseks on välja pakutud meetod, mis kannab nime *k-kordse ristkontrolli meetod* (ingl k *k-fold cross-validation*). Nimetatud meetodi puhul jagatakse näidete hulk *k* alamhulgaks, kasutades seejärel ühte hulka testimiseks ja ülejäänud hulki treenimiseks. Protseduuri korratakse *k* korda, kuni iga alamhulk on esinenud *k*-1 korda treeninghulgas ning üks kord testhulgana. Ilmselt kõige tavalisem on jagada näidete hulk kümneks alamhulgaks. Sellisel juhul räägitakse 10-kordsest ristkontrollist (joonis 5.1). Arv 10 pole siiski kohustus ning kasutusel on ka teistsugused jaotused.



Joonis 5.1. Kümnekordne ristkontroll. Näidete hulk on jagatud kümneks võrdseks alamhulgaks. Treeninghulgal sooritakse 10 erinevat testimist ja treeningut. Treeninghulgaks võetakse iga kord 9 alamosa ning testimiseks üks alamosa selliselt, et testhulk ei kordu üheski iteratsioonis.

5.3 Korpuse eeltöötlus

Eesti Dialoogikorpus koosneb üksikutest dialoogidest, kusjuures iga dialoog on antud omaette failis. Eksperimentide seisukohast on mugavam, kui aktid on jaotatud

aktiklasside järgi eraldiseisvatesse failidesse. Seega on korpuse eeltöötluse esimeseks sammuks dialoogiaktide jagamine aktiklasside failidesse.

Antud sammu käigus eemaldatakse korpusest ühtlasi kõik read, millele ei vasta ühtegi märgendit. Sellisteks ridadeks on tavaliselt dialoogi identifikaator dialoogi alguses ja pausid dialoogiaktide vahel.

Teiseks sammuks on dialoogide puhastamine üleliigsest informatsioonist. Selle sammu käigus asendatakse lausungite alguses olevad kõnelejale viitavad tunnused, nagu *A:*, *H:* kõneleja järjekorranumbriga. Dialoogi alustanud kõnelejale vastab null, teine kõneleja saab numbriks ühe, kolmandana dialoogiga liitunud kõneleja saab numbriks kahe jne. Kuna kõnelejate märkimiseks on dialoogikorpuses kasutatud erinevaid tunnuseid, on arvude kasutuselevõtt vajalik korpuse ühtlustamiseks. Teise võimalusena võiksime kõnelejate tunnused lihtsalt ära kustutada. Siiski eeldame, et kõnelejate järjekorra ja dialoogiaktide vahel valitsevad teatud seaduspärasused. Eriti arvestades, et tegu on infodialoogidega, kus üks osapool on tavaliselt infopäringu algatajaks ja teine info andjaks.

Dialoogide puhastamise käigus eemaldatakse ka sõnadega kokku kirjutatud intonatsioonimärgid. Näiteks *`Leenu=kuuleb* asendatakse sõnadega *Leenu kuuleb*. Samuti eemaldatakse kommentaarid ja muu sarnane. Sisuliselt eemaldatakse peaaegu kõik, mis ei ole esitatav tähtede ja numbritega. Erandiks on vaid pausid ja ebaselgused, mille puhul pole tuvastatud teksti (*{-}*, *{--}* ja *{---}*).

Pärast puhastamist lisatakse iga lausungi algusse spetsiaalne märgend *$S*, mis tähistab lausungi algust. Lausungi lõppu lisatakse analoogne märgend *$E*. Piiride märkimine on vajalik, et hiljem eristada lausungi algust ja lausungi lõppu hõlmavaid mustreid lausungi keskel olevatest mustritest.

Dialoogide eeltöötluse peamiseks eesmärgiks on andmete organiseerimine ning nende hõreduse vähendamine. Näiteks viiakse esialgne akt

```
V: neil ei ole `antud `Põlvas.= | SEJ: MUU | | DIJ: INFO PUUDUMINE |
```

kujule

```
0 $S neil ei ole antud põlvas $E.
```

Seejuures kirjutatakse tulemus eraldiseisvasse faili (SEJ_MUU.txt). Ühe olulise eeltöötluse sammuna heidetakse kõrvale kõik kahekordsed märgendid, jättes alles vaid lausungile vahetult järgneva märgendi. Eelneva lausungi korral tähendab see

seda, et antud lausungit vaadeldakse edaspidi üksnes kui akti SEJ: MUU. Topeltmärgendite lubamine muudaks hilisema aktide tuvastamise oluliselt keerulisemaks. Topeltmärgendite eemaldamine on ka kooskõlas EDiT-iga, mis lubab ühele lausungile üldjuhul vaid ühte märgendit.

Lisaks eelpool kirjeldatud viisil töödeldud korpusele luuakse ka teine paralleelne korpus, millesse on lisatud aktile eelneva ja järgneva akti märgend. Väljavõte sellest korpusest võib välja näha nii:

```
KYJ_INFO_PUUDUMINE 0 $S te peate sealt küsima $E  DIE_PAKKUMINE.
```

Lausungi lõpu- ja algusmärgendi vahele jääv tekst on antud juhul akt DIE: ETTEPANEK. Algusmärgendile eelnev osa tähistab sellele dialoogiaktile eelnevat akti ning lõpumärgendile järgnev osa sellele aktile järgnevat akti. Selliselt töödeldud korpus võimaldab leida mustreid dialoogiaktide järgnevuse ja aktiklasside vahel.

5.4 Eksperimentide kirjeldused

Meie eksperimentide seeria koosneb kolmest eksperimendist, milles keskendutakse erinevatele sufiksipuu variatsioonidele ning erinevatele tunnustele. Esimeseks eksperimendiks on dialoogiaktide tuvastamine alamsõnede abil. Antud eksperimenti võib vaadata ka kui baaseksperimenti, kus sufiksipuud kasutatakse selle kõige lihtsamas vormis. Selline sufiksipuu võib sisaldada näiteks mustreid *1 $S aga see* või *(0.5) ja*, kuid ei sisalda metamärkidega mustreid. Erinevalt järgmistest eksperimentidest ei sea me mustritele sageduslävendit.

Teises eksperimendis tuvastatakse dialoogiakte sufiksipuude abil, mis sisaldavad lisaks tavalistele alamsõnedele ka alamsekventse ehk metamärke sisaldavaid mustreid. Kuna selliste mustrite arv võib äärmiselt suureks kasvada, oleme seadnud piirangu, et sufiksipuuse valitakse üksnes mustrid, mida esineb treeningkorpuses vähemalt kolmel korral. Samuti oleme piiranud metamärkide maksimaalse arvu ühega. Seega kasutatakse teises eksperimendis lisaks alamsõnedele selliseid mustreid nagu näiteks *0 $S * h $E* või *$S * ei ole.*

Kolmandas eksperimendis laiendame dialoogiaktide tuvastamiseks kasutatavate tunnuste hulka naaberaktidele. Kasutusele tulevad sellised mustrid nagu näiteks *PPE_ÜLEKÜSIMINE 1 $S jah $E* või *0 $S * $E RIE_SOOVIMINE.*

Nimetatud mustreid tuleb interpreteerida selliselt, et algusmärgendile *$S* eelnev osa vastab lausungile eelnevale dialoogiaktile, märgendite *$S* ja *$T* vahel olev osa vastab dialoogiaktile, mida püütakse tuvastada, ning lõpumärgendile *$T* järgnev osa vastab lausungile järgnevale dialoogiaktile. Nii nagu teise eksperimendi korral, on ka nüüd seatud piirid metamärkide arvule ning mustrite sagedusele.

Eksperimentide läbiviimiseks on kasutatud antud töö raames loodud tarkvara, mis võimaldab sufiksipuid konstrueerida ning testida. Vastav programm on kirjutatud Python-is ning koosneb ligikaudu 1000 koodireast. Programmi võib leida antud töö elektroonsetest lisadest.

6 Eksperimentide tulemused

Järgnevalt anname ülevaate eksperimentide tulemustest. Olulisemad tulemused on esitaud antud töö lisades nr 2-4 ning täies mahus töö elektroonsetes lisades.

6.1 Tuvastamine alamsõnede põhjal

Meie esimeseks eksperimendiks oli dialoogiaktide tuvastamine alamsõnede abil. Väljavõte tulemustest on toodud lisas 2. Dialoogiaktide tuvastamise keskmine saagis ja täpsus on 45,31%. Saagise ja täpsuse ühtelangemine pole juhuslik, vaid tuleneb sellest, et iga lausung sai täpselt ühe märgendi.

Tulemused näitavad suurt erinevust eri klassidesse kuuluvate dialoogiaktide tuvastamises. Näiteks tuvastatakse akti RIE: KUTSUNG ligi 97%-lise täpsuse ja 99%-lise saagisega, samas jääb aga mitme dialoogiakti puhul saagis nulliks. Üheks olulisemaks faktoriks on siinjuures andmete hõredus (joonis 5.1). Tervelt 32 aktiklassi jaoks on korpuses olevate näidete arv alla kümne, mis teeb oodatavaks treeningnäidete arvuks samuti alla kümne ning testimisnäidete arvuks alla ühe. Samas leidub ka akte, mida ei suudeta hoolimata suurest näidete arvust edukalt tuvastada. Sellisteks aktideks on IL: TÄPSUSTAMINE, YA: INFO ANDMINE, DIJ: NÕUSTUMINE, VR: MUU, IL: ÜLERÕHUTAMINE ja PPJ: LÄBIVIIMINE.

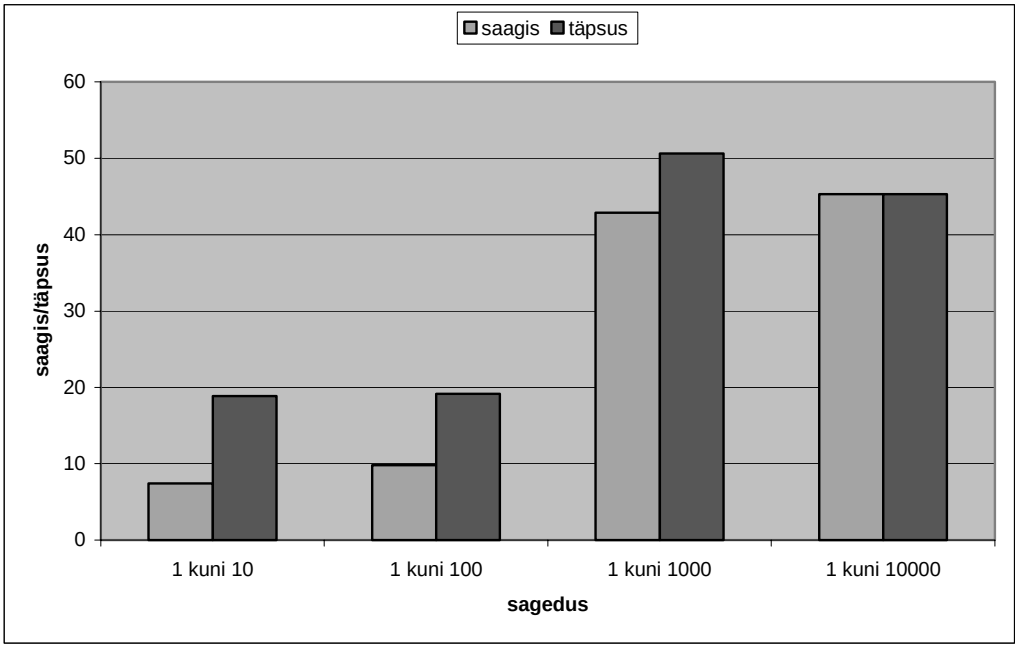
Kuigi treeningnäidete arvu ja tuvastamise saagise ning täpsuse vahel valitseb positiivne seos, on vajalik treeningnäidete arv dialoogiakti jaoks erinev. Seda iseloomustab ilmekalt alamsõnede arv, mida ühe või teise aktiklassi näidete tuvastamiseks kasutatakse. Näiteks tuvastati 376 aktist RIE: TERVITUS 313, kusjuures selleks kasutatud mustrite arv oli üksnes 15. Seevastu KYJ: INFO ANDMINE puhul tuvastati 1486 aktist 655, kasutades selleks 572 mustrit. See teeb ühe mustri kasutamise sageduseks RIE: TERVITUS puhul keskmiselt 25,07 korda ning KYJ: INFO ANDMINE puhul 1,15 korda.

Siiski märgime ära, et kasutatud sufikspuud pole kaugeltki minimaalsed ning ilmselt saaks mõlemal juhul mustrite arvu veel mõnevõrra vähendada. Kuna dialoogiaktid klassifitseeritakse pikima mustri alusel, ei ole kasutatud mustrid alati optimaalsed. Näiteks on akti RIE: SOOVIMINE tuvastamiseks kasutatud nelja mustrit: *1 $S kõike head $E, 0 $S kõike head $E, 0 $S kõike ead $E ja $S kõike head $E*. Kolmel juhul neljast sisaldab muster alamsõne *$S kõike head $E,* kusjuures

nimetatud alamsõne ei esine ühegi teise aktiklassi tuvastamisel kasutatud mustrites. Seega võiksime piirduda akti RIE: SOOVIMINE tuvastamisel ka üksnes mustriga *$S kõike head $E*. Teisest küljest ei oleks aga ka liiga lühikesed mustrid, nagu *$S kõike või head $E* sobivad.

Eksperimendid näitavad, et kõnelejate järjekorra märkimine on teatud aktide korral ülimalt oluline. Näiteks kasutatakse akti RIE: TERVITUS tuvastamiseks 348 juhul mustrit *0 $S tere $E*. Seevastu akti RIJ: VASTUTERVITUS puhul kasutatakse 170 korral mustrit *1 $S tere $E*. Mõlemal juhul annavad nimetatud mustrid õige tulemuse täpsusega ligikaudu 85%. On ilmne, et ilma kõnelejate järjekorda arvesse võtmata oleks tervituste eristamine oluliselt raskendatud. Kõnekas fakt on ka see, et 3352 erinevast mustrist, mida dialoogiaktide tuvastamisel kasutati, sisaldas 2336 mustrit kõneleja järjekorranumbrit.

Tulemused näitavad, et akte, mis on vormiliselt sarnased või isegi eristamatud, kuid kuuluvad eri klassidesse, on tegelikult väga palju. Sellised on näiteks naaberpaare moodustavad aktid RIE: TÄNAN ja RIJ: TÄNAN või info andmise aktid DIJ: INFO ANDMINE ja KYJ: INFO ANDMINE. Nimetatud aktid ei ole tihti määratavad ühe lausungi kontekstis, vaid nõuavad teadmist naaberaktide kohta. Selle probleemiga on põhjalikumalt tegeletud kolmandas eksperimendis.



Joonis 5.1. Saagise ja täpsuse sõltuvus dialoogiakti sagedusest korpuses.

Järgnevalt anname mõned näited alamsõnedest, mida kasutati dialoogiaktide tuvastamiseks. Eelnevalt mainitud RIE: KUTSUNG tuvastatakse kõikidel juhtudel alamsõne *$S $T* põhjal. Tegelikult ongi see ainus alamsõne, mille põhjal kutsungit tuvastada, kuna pärast eeltöötlust jäävad kutsungist alles vaid lausungi lõpu- ja algusmärk. Selliseid aktiklasse on veel - näiteks DIJ: TEGEVUS ja KYJ: TEGEVUS. Kutsungit eristab neist üldjuhul see, et kutsungiga ei ole dialoogi transkriptsioonis seotud kõnelejat.

Mõneti eriline akt on KYJ: INFO ANDMINE. Korpuses kõige sagedasema aktina saavad KYJ: INFO ANDMINE märgendi paljud lausungid, millel puudub märkimisväärne ühisosa sufiksipuuga. Seda iseloomustab ka tõsiasi, et kõige enam kasutatud KYJ: INFO ANDMINE muster on *0 $S*, mida rakendati 476 korral. On ilmne, et antud mustrit sisaldavad ka paljud teised aktid. Sellest tulenevalt tehakse ka palju vigu.

6.2 Tuvastamine alamsekventside järgi

Meie teine eksperiment seisnes dialoogiaktide tuvastamises metamärke sisaldava sufiksipuu abil. Piiranguna oli lubatud kasutada kuni ühte metamärki sisaldavaid mustreid, kusjuures sufiksipuusse valimiseks pidi muster esinema treeningkorpuses vähemalt kolmel korral. Eksperimendi tulemused on esitatud lisas 3. Võrreldes eelmise eksperimendiga on saagis ja täpsus kasvanud 46,75%-le. Samuti on aktide tuvastamiseks kasutatud mustrid märkimisväärselt erinevad.

Saagise ja täpsuse kasv ei ole siiski üldine. Tegelikult näeme täpsuse ja saagise langust isegi valdava osa aktiklasside korral. Märkimisväärne tõus esineb vaid suhteliselt väheste aktide korral, nagu näiteks DIJ: INFO ANDMINE, DIJ: INFO PUUDUMINE, KYJ: INFO ANDMINE ja KYJ: INFO PUUDUMINE. Kuna nimetatud akte esineb aga sagedasti, on see siiski piisav, et tõsta üldist saagist ja täpsust.

Harvemini esinevate aktide saagise langus on vähemalt osaliselt seotud sellega, et sufiksipuust on välja jäetud treeningkorpuses harvemini kui kolm korda esinevad mustrid. Teisest küljest näeme, et antud eksperimendis kasutatud sufiksipuu on suurema üldistusvõimega. Kui esimeses eksperimendis tuvastati akte 3352 mustri abil, siis teises eksperimendis on see arv langenud 2665-le.

Mõnevõrra üllatav on, et nii esimeses kui ka teises eksperimendis andis suhteliselt häid tulemusi akti DIJ: INFO ANDMINE tuvastamine. Tulemus on ootamatu, kuna akt DIJ: INFO ANDMINE märgib mingi fakti esitamist. Üldiselt ei ole põhjust oodata, et faktid sisaldaksid sagedasti korduvaid mustreid. Eksperimentide tulemused viitavad aga vastupidisele.

Põhjus saab selgeks, kui vaadata mustreid, mille abil kõneallolevaid akte tuvastati. Rõhuvas enamuses on tegu erinevate arvuliste mustritega. Näiteks annavad mustrid *0 $S seitse kolm * $E* ja *0 $S null null $E* kumbki saagiseks 1,48%.

Tulemused vihjavad sellele, et vähemalt teatud aktiklasside tuvastamisel on meie poolt kasutatav mudel seotud tihedalt ainevaldkonnaga. Antud juhul tuleneb arvuliste mustrite suur osatähtsus sellest, et korpus sisaldab hulgaliselt dialooge, milles antakse infot telefoninumbrite kohta.

Siiski leidub ka DIJ: INFO ANDMINE puhul mustreid, mis võivad viidata üldisematele seaduspärasustele. Näiteks annab teises eksperimendis muster *0 $S (...) hh * $E* saagise 0,82%. Tõsi küll, selle täpsus on kõigest 25%.

Toome illustratsioonina veel mõned näited kasutatud mustritest. Dialoogiakti KYE: AVATUD korral on küllaltki sagedasteks mustriteks *0 $S mis* ja *1 $S mis*. Mustrid kokkuvõetuna annavad saagiseks veidi alla kolme protsendi ning seda üle seitsmekümneprotsendilise täpsusega.

Nagu me näeme, pole saagis ja täpsus oluliselt paranenud, või on isegi halvenenud, selliste problemaatiliste dialoogiaktide puhul nagu IL: TÄPSUSTAMINE, YA: INFO ANDMINE, DIJ: NÕUSTUMINE, VR: MUU, IL: ÜLERÕHUTAMINE ja PPJ: LÄBIVIIMINE.

IL: TÄPSUSTAMINE madala saagise ja täpsuse peamiseks põhjuseks on selle suur sarnasus infoaktidele KYJ: INFO ANDMINE ja DIJ: INFO ANDMINE. Ligikaudu pooled kõikidest täpsustustest tuvastatakse kui info andmised. IL: TÄPSUSTAMINE problemaatilisuse on ära märkinud ka Hennoste ja Rääbis, öeldes, et vahet tuleks teha teema põhjal - täpsustamine jätkab sama teemat ja info andmine alustab uut teemat või alateemat [Hennoste & Rääbis 2004]. Selline vahetegemine ei kuulu aga antud meetodi võimetesse.

Teine problemaatiline akt on YA: INFO ANDMINE, mis tuvastati 111 juhul valesti kui KYJ: INFO ANDMINE. Aktide sarnasuse on ära märkinud ka Hennoste ja Rääbis [Hennoste & Rääbis 2004]. Erinevus seisneb selles, et kui KYJ: INFO ANDMINE on reaktsioon naaberpaari eesliikmele, siis YA: INFO ANDMINE ei ole

eesliikmega seotud, vaid toob sisse uue teema, probleemi või objekti [Hennoste & Rääbis 2004].

Akt DIJ: NÕUSTUMINE tuvastatakse 103 juhul valesti kui VR: NEUTRAALNE JÄTKAJA. Põhjus saab selgeks, kui vaadata kummagi aktiklassi lausungeid. Akti DIJ: NÕUSTUMINE korral domineerivad selgelt ühesõnalised lausungid nagu *jah*, *jaa* ning *mhmh*. Täpselt samasugused lausungid vastavad üldjoontes ka aktile VR: NEUTRAALNE JÄTKAJA. Kuna viimasesse aktiklassi kuuluvate lausungite sagedus on suurem, saab oluline osa DIJ: NÕUSTUMINE aktidest märgendiks hoopis VR: NEUTRAALNE JÄTKAJA.

Akt IL: ÜLERÕHUTAMINE tuvastatakse kõige sagedamini kui KYJ: INFO ANDMINE. Juhul kui diskursusega ei arvestata, on see loomulik viga, kuna ülerõhutamine tähendab eelnevalt öeldud info kordamist.

Akti PPJ: LÄBIVIIMINE süntaks on äärmiselt sarnane aktidele DIJ: NÕUSTUMINE ja VR: NEUTRAALNE JÄTKAJA. Nii, nagu nimetatud aktide korral, on ka PPJ: LÄBIVIIMINE puhul kõige sagedamini korduvaks alamsõneks $S jah $E. Probleemile on püütud leida lahendust järgmises eksperimendis.

6.3 Tuvastamine alamsekventside ja diskursuse järgi

Nagu me eelmise eksperimendi tulemustest nägime, leidub küllaltki palju keeleliselt sarnaseid või koguni eristamatuid akte. Antud eksperimendi eesmärgiks oli rakendada sufiksipuud, mis arvestab lisaks lausungis leiduvatele mustritele ka lausungile eelneva ja järgneva aktiga. Tuvastamine toimus eeldusel, et eelneva ja järgneva akti märgendid on varem täpselt määratud.

Võrreldes eelmise eksperimendiga kasvasid saagis ja täpsus ligi 10%, saavutades väärtuse 56,49%. Koondtulemused on esitatud lisas 4. Toome siinkohal välja mõned olulisemad muutused mustrites ja dialoogiaktide tuvastamises.

Näiteks kasutati akti DIJ: EDASILÜKKAMINE tuvastamiseks mustrit

VR_NEUTRAALNE_VASTUVÕTUTEADE 0 $S üks hetk $E DIJ_INFO_ANDMINE,

mis andis 100%-lise täpsuse juures 28,24%-lise saagise. Eelmises eksperimendis andis põhiosa saagisest (58,02%) muster *0 $S üks hetk $E*, kuid seda oluliselt madalama täpsusega 67,86%.

Näiteks paranes ka akti KYE: VASTUST PAKKUV tuvastamine. Kui eelmises eksperimendis oli antud akti tuvastamise saagis 10,94% ning täpsus 22,62%, kusjuures tuvastamiseks puudusid "tugevad" tunnused, siis nüüd kasvasid saagis ja täpsus vastavalt 57,01%-ni ja 52,47%-ni. Seejuures andis 7,29% saagisest muster *0 $S * jah $E KYJ_JAH* ning seda 100%-lise täpsusega. Sarnaseid näiteid leidub veelgi. Keskendume aga nüüd eelpool väljatoodud problemaatilistele aktidele.

Akti IL: TÄPSUSTAMINE korral saavutatakse mõne protsendi võrra kõrgemad tulemused kui eelnevates eksperimentides, kuid samas ei ole naaberaktide analüüs suutnud info andmise ja täpsustamise eristamise probleemi lahendada – sagedasemad vead on endiselt samad. Probleem pole lahenenud ka akti YA: INFO ANDMINE jaoks, kuigi ka siin on saagis ja täpsus mõne protsendi võrra kasvanud.

Seevastu akti DIJ: NÕUSTUMINE korral näeme me juba märgatavat tulemuste paranemist. Kui eelnevalt aeti kõnealune akt segi aktiga VR: NEUTRAALNE JÄTKAJA, siis nüüd tehakse kahel aktil oluliselt paremini vahet. Peamiseks eristajaks on nõustumisele eelnev direktiivi eesliige, mis neutraalse jätkaja korral puudub.

Veel ühe problemaatilise akti, IL: ÜLERÕHUTAMINE, tuvastamine pole aga paranenud. Kasutatud mustritest näeme, et vähemalt pindmisel tasemel puudub antud aktil märgatav seos eelneva või järgneva aktiga.

Akti PPJ: LÄBIVIIMINE tuvastamine on seevastu jällegi oluliselt paranenud. Kui eelnevalt valmistas probleeme antud akti eristamine aktidest, mille peamine iseloomulik tunnus oli samuti *jah*, siis nüüd eristatakse akti PPJ: LÄBIVIIMINE ennekõike sellele eelneva akti, PPE: ÜLEKÜSIMINE, põhjal.

7 Järeldused

Võtame nüüd eksperimentide tulemused kokku. Dialoogiaktide tuvastamine alamsõnede põhjal andis saagiseks ja täpsuseks 45,31%. Dialoogiaktide tuvastamine alamsekventside põhjal andis saagiseks ja täpsuseks 46,75%. Tuvastamine alamsekventside ja eelneva ning järgneva akti põhjal andis saagiseks ja täpsuseks 56,49%.

Tulemused on piisavalt kõrged selleks, et sufiksipuude meetod pakuks praktilist huvi. Võrreldes baasjoonega on tulemused paranenud 35-45%. Seejuures oleme piirdunud küllaltki lihtsate algoritmidega. Statistiliste meetodite puhul on tavaks kasutada pärast mudelite treenimist veel pügamis- ja häälestamismeetodeid, mis tulemusi mõnevõrra parandavad. Antud töös pole selliseid meetodeid kasutatud.

Samuti oleme suhteliselt pealiskaudselt käsitlenud andmete ületäpsustamise probleemi (ingl k *overfitting*). Ainsaks selle vastu kasutatud abinõuks on alampiiri seadmine mustrite sagedusele. Sellest hoolimata saavutati arvestatavad tulemused.

Sufiksipuu meetodi oluliseks plussiks on selle arusaadavus – dialoogiaktid tuvastatakse treeningkorpusest leitud alamsõnede ja -sekventside abil, mis on inimese jaoks hästi loetavad. Tulemusena saadakse tuvastamise tagamaadest tunduvalt parem ülevaade kui tüüpilise “musta kasti” meetodi korral. Seda illustreerib ka tulemuste peatükis esitatud tulemuste analüüs, kus tuvastamisel kasutatud mustrid toovad selgelt esile tuvastamisel tekkivate probleemide põhjused ning loovad ühtlasi võimaluse nende likvideerimiseks. Selline lähenemisviis osutub eriti kasulikuks juhul, kui kasutajal puudub ülevaade konkreetsest dialoogiaktide tüpoloogiast ja dialoogiaktide omadustest.

Eksperimendid näitavad aga ka selgelt, et dialoogiaktide tuvastamisega kaasneb probleeme, mida antud meetod ei suuda lahendada ning mis võivad nõuda reeglipõhist lähenemist. Samuti toovad eksperimendid esile andmete hõreduse probleemi, mis seab statistiliste meetoditele mõningased piirid.

8 Edasine töö

Antud töös piirdusime suhteliselt lihtsakujuliste sufiksipuudega – ainsaks kasutatud metamärgiks oli *, mis võimaldas lisaks alamsõnedele esitada ka alamsekventse. Üheks edasiarenduse võimaluseks on laiendada sufiksipuid veel teiste metamärkidega. Selliseid laiendusi on uurinud Vilo [Vilo 2002].

Oluliseks võib osutuda tagasiviidete (ingl k *backreferences*) kasutuselevõtt. Seda näiteks akti IL: ÜLERÕHUTAMINE korral, milles toimub eelpoolöeldu kordamine, või aktide YA: INFO ANDMINE ning KYJ: INFO ANDMINE korral, mida eristatakse selle põhjal, kas alustatakse uut teemat või jätkatakse vanaga.

Võrreldes senisega tuleb oluliselt enam rõhku panna “parima” mustri leidmisele. Antud töös kasutatud meetodi korral tuvastatakse dialoogiakt pikima sufiksipuule vastava mustri põhjal. Juhul kui selliseid mustreid on mitu, valitakse neist suurema tõenäosusega muster.

Sellisel lähenemisviisil on ilmselgelt puudusi: eelistuse saavad pikad loetelud, mis üldiselt ei viita akti funktsioonile; vähetõenäolist mustrit eelistatakse ühe sümboli võrra lühemale tõenäolisele mustrile; arvestamata jäetakse juhud, kus aktile viitab korraga rohkem kui üks muster. Antud töö autor on astunud ka esimesed sammud “parima” mustri leidmise meetodi täiustamiseks. Olulisi tulemusi pole siiski veel saavutatud.

Kui antud töö keskendus ennekõike sufiksipuu meetodi esmasele katsetamisele, siis tulevikus võib kaaluda puu konstrueerimise algoritmide tõhustamist ning puude suuruste vähendamist, mis lisaksid meetodile atraktiivsust.

9 Kokkuvõte

Üksteisega suheldes esitavad inimesed küsimusi või palveid, annavad millegi kohta infot, tervitavad, tänavad jne. Selliseid keele abil tehtavaid tegevusi nimetatakse dialoogiaktideks. Dialoogiaktide automaatne tuvastamine on oluline loomulikus keeles suhtlevates dialoogisüsteemides, kus see võimaldab siduda dialoogiakti pindmise struktuuri selle sisulise funktsiooniga.

Antud töö pakub uudse lähenemisviisi dialoogiaktide automaatseks tuvastamiseks. Meetod leiab treeningkorpusest dialoogiakte iseloomustavad mustrid ning konstrueerib neist tõenäosusliku andmestruktuuri, sufiksipuu, mida kasutatakse seejärel dialoogiaktide tuvastamiseks.

Meetod annab Eesti Dialoogikorpusel tulemuseks saagise ning täpsuse 45%-56%. Seejuures kasutatakse kolme erinevat lähenemisviisi: dialoogiaktide tuvastamine alamsõnede põhjal, tuvastamine alamsekventside põhjal ning tuvastamine alamsekventside ja dialoogiaktile eelneva ning järgneva akti põhjal. Saagis ja täpsus on võrreldavad muude, hästi tuntud ja laialt kasutatavate tuvastusmeetoditega. Antud meetodi eeliseks teiste meetodite ees on selle hea arusaadavus, kuna dialoogiaktid tuvastatakse treeningkorpusest leitud alamsõnede ja -sekventside põhjal. Antud töö peamisteks tulemusteks on:

- uudne meetod dialoogiaktide tuvastamiseks (dialoogiaktide jaoks kohandatud sufiksipuu),
- programm dialoogiaktide märgendamiseks sufiksipuude abil,
- Eesti Dialoogikorpuses kasutatavasse dialoogiaktide tüpoloogiasse kuuluvad dialoogiakte iseloomustavad keelelised mustrid (esitatud elektroonsetes lisades).

Antud töö loob piisavad alused edasisteks uuringuteks.

10 Recognition of Dialogue Acts in Estonian Dialogues Using Suffix Trees

MSc Thesis

Taavet Kikas

Abstract

People use language to express different things: ask questions, give information, greet each other, wish something etc. Actions like these are called dialogue acts.
One of the natural language processing tasks is to recognize these acts. The importance of automated dialogue act recognition lays in the fact that it relates the surface form of the dialogue act to its function. This makes possible the development of natural language dialogue systems.

This MSc thesis presents a novel method for dialogue act recognition. The recognition is made on the basis of dialogue substrings and subsequences, which are automatically extracted from an annotated corpus. This is accomplished by constructing a data structure called suffix tree.

An overview of suffix trees, their construction and usage is given. We also become acquainted with the typology of Estonian dialogue acts and standard methods for estimating dialogue act recognition quality. Finally, a set of experiments with suffix trees is conducted.

The suffix trees achieve average recall and precision from 45% to 56% on the Estonian Dialogue Corpus. This is comparable with well-known dialogue recognition techniques. The advantage of the current method is its good comprehensibility as recognitions are made on the basis of actual substrings and subsequences found in the training corpus.

Three different suffix tree models are used. The first model classifies dialogue acts by means of simple substrings. The second model uses subsequences in addition to substrings and the third model uses subsequences, substrings and discourse information in the form of preceding and following act. The results improve with each model.

The method shows a good promise for further improvements as the model can be extended with additional dialogue information. The current work gives a sufficient basis for future studies.

Kirjandus

[Alvarez 2002] Sergio A. Alvarez. An exact analytical relation among recall, precision, and classification accuracy in information retrieval. Technical Report BCCS-02-01, Computer Science Department, Boston College, 2002.

[Bejerano & Yona 1999] Gill Bejerano, Golan Yona. Modeling protein families using probabilistic suffix trees. Proceedings of the3rd Annual International Conference on Computational Molecular Biology (RECOMB), 1999.

[Carletta 1996] Carletta, J. C. Assessing agreement on classification tasks: the kappa statistic. Computational Linguistics, 22 (2), 249-254, 1996.

[EDiC] http://math.ut.ee/~koit/Dialoog/EDiC.html. Kasutatud 14. mai 2007.

[Fishel & Kikas 2006] Mark Fishel, Taavet Kikas. Dialoogiaktide automaatne tuvastamine. Tartu Ülikooli üldkeeleteaduse õppetooli toimetised 6: Keel ja arvuti, 233-245, 2006.

[Fishel 2006] Mark Fishel. Dialogue Act Recognition Techniques. 2006.

[Giegerich & Kurtz 1995] A Comparison of Imperative and Purely Functional Suffix Tree Construction. Science of Computer Programming, 1995.

[Giegerich & Kurtz 1999] Efficient Implementation of Lazy Suffix Trees. Proceedings of the 3rd International Workshop on Algorithm Engineering, 30-42, 1999.

[Hennoste & Rääbis 2004] Tiit Hennoste, Andriela Rääbis. Dialoogiaktid eesti infodialoogides: tüpoloogia ja analüüs. Tartu Ülikooli Kirjastus, 2004.

[Jurafsky & Martin 2000] Daniel Jurafsky, James H. Martin. Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall, 2000

[Kikas 2005] T. Kikas. Dialoogiaktide tuvastamine eestikeelsetes dialoogides otsustuspuude abil. Bakalaureusetöö. Tartu Ülikool, 2005.

[McCreight 1976] E. M. McCreight. A Space-Economical Suffix Tree Construction Algorithm. Journal of the ACM (JACM), 23(2), 262-272, 1976.

[Mengel et al. 2000] A. Mengel, L. Dybkjaer, J.M. Garrido, U. Heid, M. Klein, V. Pirrelli, M. Poesio, S. Quazza, A. Schiffrin, C. Soria. MATE Dialogue Annotation Guidelines. http://www.ims.uni-stuttgart.de/projekte/mate/mdag/, kasutatud 14. mai 2007.

[Ron et al. 1996] Dana Ron, Yoram Singer, Naftali Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. Machine Learning, 25, 1996.

[Stockle et al. 2002] Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg et al. Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. Computational Linguistics, 26 (3), 339-373, 2000.

[Sufiks] http://en.wikipedia.org/wiki/Suffix_%28computer_science%29. Kasutatud 14. mai 2007.

[Tarum 1999] D. R. Traum, D. R. Speech Acts for Dialogue Agents, Foundations and Theories of Rational Agents. Foundations And Theories Of Rational Agents, Kluwer Academic Publishers, 69-201, 1999.

[Traum 2000] D. R. Traum. 20 Questions on Dialogue Act Taxonomies. Journal of Semantics, 17 (1), 7-30, 2000.

[Ukkonen 1995] E. Ukkonen. On-line construction of suffix-trees. Algorithmica 14, 249-260, 1995.

[Weiner 1973] P. Weiner. Linear Pattern Matching Algorithms. Proceedings of the 14th IEEE Annual Symposium on Switching and Automata Theory, The University of Iowa, 1-11, 1973.

[Vilo 2002] Jaak Vilo (2002). Pattern Discovery from Biosequences. PhD Thesis, Department of Computer Science, University of Helsinki, 2002.

Lisad

Lisa 1. Eesti dialoogiaktide tüpoloogia

I. Naaberpaare moodustavad aktid
1. Rituaalid
Tervitamine RIE: Tervitus RIJ: Vastutervitus
Hüvastijätmine RIE: Hüvastijätt RIJ: Vastuhüvastijätt
Soovimine RIE: Soovimine RIJ: Tänamine RIJ: Vastusoovimine
Viisakusküsimus RIE: Viisakusküsimus RIJ: Viisakusvastus
Tänamine RIE: Tänan RIJ: Palun
Palumine RIE: Palun RIJ: Tänan
Vabandamine RIE: Vabandus RIJ: Vabanduse vastuvõtmine
Esitlemine RIE: Esitlus RIJ: Vastuesitlus RIJ: Hinnang
Kutsesignaal RIE: Kutsung RIJ: Kutsungi vastuvõtmine
Lõpetamine RIE: Lõpusignaal RIJ: Lõpetamise vastuvõtmine RIJ: Lõpetamise tagasilükkamine
Rituaalid: muu RIE: Muu RIJ: Muu
2. Teemavahetus
TVE: pakkumine TVE: muu
TVJ: vastuvõtmine TVJ: tagasilükkamine

TVJ: muu
3. Partneri algatatud parandused
PPE: ümbersõnastamine PPE: üleküsimine PPE: mittemõistmine PPE: muu
PPJ: läbiviimine PPJ: muu
4. Kontakti kontroll
KKE: algatus KKE: muu
KKJ: kinnitamine KKJ: muu
5. Direktiivid
DIE: soov DIE: ettepanek DIE: pakkumine DIE: palve oodata DIE: muu
DIJ: info andmine DIJ: info puudumine DIJ: keeldumine DIJ: kahtlev DIJ: nõustumine DIJ: mittenõustumine DIJ: piiratud nõustumine DIJ: tegevus DIJ: edasilükkamine DIJ: muu
6. Küsimused
KYE: suletud kas KYE: jutustav kas KYE: alternatiiv KYE: avatud KYE: vastust pakkuv KYE: täpsustav KYE: vastuse tingimuste täpsustamine KYE: muu
KYJ: jah KYJ: ei KYJ: nõustuv ei KYJ: muu kas-vastus KYJ: alternatiiv: üks KYJ: alternatiiv: mõlemad KYJ: alternatiiv: kolmas valik

KYJ: alternatiiv: eitav KYJ: alternatiiv: muu KYJ: tegevus KYJ: info andmine KYJ: info puudumine KYJ: keeldumine KYJ: edasilükkamine KYJ: vastus alternatiivina KYJ: kahtlev KYJ: muu
7. Seisukohavõtud
SEE: väide SEE: arvamus SEE: muu
SEJ: nõustumine SEJ: mittenõustumine SEJ: piiratud nõustumine SEJ: keeldumine SEJ: muu
II. Üksikaktid
1. Rituaalsed üksikaktid
RY: üleandmine RY: tutvustus RY: äratundmine RY: kontakteerumine RY: kutsumine RY: muu
2. Infolisad
IL: täpsustamine IL: seletamine IL: põhjendamine IL: järeldamine IL: kokkuvõtmine IL: ülerõhutamine IL: pehmendamine IL: hinnang IL: muu
3. Vabatahtlikud reaktsioonid
VR: hinnanguline jätkaja VR: neutraalne jätkaja VR: hinnanguline vastuvõtuteade VR: neutraalne vastuvõtuteade VR: hinnanguline info osutamine uueks VR: neutraalne info osutamine uueks

VR: hinnanguline piiritleja VR: neutraalne piiritleja VR: paranduse hindamine VR: muu
4. Parandused
PA: eneseparandus PA: partneri parandus PA: muu
5. Üksikaktid
YA: eelteade YA: jutustamine YA: lubadus YA: info andmine YA: jutu piiride osutamine YA: retooriline küsimus YA: retooriline vastus YA: referaat YA: muu YA: mitteverbaalne akt YA: praak

Lisa 2. Dialoogiaktide tuvastamine alamsõnede põhjal

Dialoogiakt	Sagedus	Olulisi leitud	Ebaolulisi leitud	Saagis	Täpsus
KYJ_INFO_ANDMINE.txt	1486	655	1607	44,08	28,96
VR_NEUTRAALNE_JÄTKAJA.txt	1019	750	1182	73,6	38,82
VR_NEUTRAALNE_VASTUVÕTUTEADE.txt	707	110	247	15,56	30,81
KYJ_JAH.txt	628	262	385	41,72	40,49
DIJ_INFO_ANDMINE.txt	609	331	488	54,35	40,42
KYE_AVATUD.txt	567	295	596	52,03	33,11
KYE_VASTUST_PAKKUV.txt	521	69	250	13,24	21,63
VR_NEUTRAALNE_INFO_OSUTAMINE_UUEKS.txt	493	462	50	93,71	90,23
IL_TÄPSUSTAMINE.txt	424	20	223	4,72	8,23
RIE_TÄNAN.txt	381	357	22	93,7	94,2
RIE_TERVITUS.txt	376	313	98	83,24	76,16
RIE_KUTSUNG.txt	369	365	12	98,92	96,82
RIJ_KUTSUNGI_VASTUVÕTMINE.txt	366	269	10	73,5	96,42
RIJ_VASTUTERVITUS.txt	346	269	100	77,75	72,9
DIE_SOOV.txt	332	146	210	43,98	41,01
RIJ_PALUN.txt	300	295	25	98,33	92,19
KYE_JUTUSTAV_KAS.txt	292	71	229	24,32	23,67
RY_TUTVUSTUS.txt	275	228	9	82,91	96,2
YA_INFO_ANDMINE.txt	239	4	82	1,67	4,65
KYE_SULETUD_KAS.txt	224	39	115	17,41	25,32
VR_NEUTRAALNE_PIIRITLEJA.txt	210	144	108	68,57	57,14
YA_MUU.txt	208	46	62	22,12	42,59
DIJ_NÕUSTUMINE.txt	163	3	23	1,84	11,54
RIE_HÜVASTIJÄTT.txt	153	124	35	81,05	77,99
KYJ_EDASILÜKKAMINE.txt	137	28	63	20,44	30,77
DIJ_EDASILÜKKAMINE.txt	131	92	72	70,23	56,1
KYJ_EI.txt	128	78	117	60,94	40
YA_PRAAK.txt	127	90	52	70,87	63,38
VR_MUU.txt	126	3	12	2,38	20
IL_PÕHJENDAMINE.txt	124	21	61	16,94	25,61
IL_ÜLERÕHUTAMINE.txt	124	1	76	0,81	1,3
PPJ_LÄBIVIIMINE.txt	106	0	14	0	0
KYJ_INFO_PUUDUMINE.txt	93	12	64	12,9	15,79
RIJ_VASTUHÜVASTIJÄTT.txt	90	45	27	50	62,5
DIE_PAKKUMINE.txt	89	19	56	21,35	25,33
KYE_ALTERNATIIV.txt	85	5	57	5,88	8,06
DIE_ETTEPANEK.txt	73	4	47	5,48	7,84
KYJ_MUU.txt	66	0	15	0	0
IL_KOKKUVÕTMINE.txt	65	1	37	1,54	2,63
PPE_ÜLEKÜSIMINE.txt	60	1	37	1,67	2,63
RIE_LÕPUSIGNAAL.txt	56	1	21	1,79	4,55
KYJ_ALTERNATIIV_ÜKS.txt	55	2	10	3,64	16,67
RIJ_LÕPETAMISE_VASTUVÕTMINE.txt	53	0	2	0	0
KYJ_NÕUSTUV_EI.txt	51	2	10	3,92	16,67
SEJ_NÕUSTUMINE.txt	51	0	10	0	0

YA_JUTU_PIIRIDE_OSUTAMINE.txt	47	6	14	12,77	30
KYE_MUU.txt	45	5	18	11,11	21,74
DIJ_INFO_PUUDUMINE.txt	40	4	18	10	18,18
SEE_ARVAMUS.txt	40	0	10	0	0
YA_EELTEADE.txt	39	9	14	23,08	39,13
KYE_VASTUSE_TINGIMUSTE_TÄPSUSTAMINE.txt	38	0	16	0	0
KKE_ALGATUS.txt	34	12	10	35,29	54,55
PPE_ÜMBERSÕNASTAMINE.txt	34	0	17	0	0
IL_SELETAMINE.txt	33	0	16	0	0
SEE_VÄIDE.txt	31	0	10	0	0
DIE_PALVE_OODATA.txt	30	8	17	26,67	32
IL_HINNANG.txt	27	0	19	0	0
KYE_TÄPSUSTAV.txt	27	0	9	0	0
KKJ_KINNITAMINE.txt	26	0	13	0	0
VR_PARANDUSE_HINDAMINE.txt	25	0	1	0	0
VR_HINNANGULINE_VASTUVÕTUTEADE.txt	24	3	11	12,5	21,43
DIJ_PIIRATUD_NÕUSTUMINE.txt	20	0	2	0	0
KYJ_KEELDUMINE.txt	20	0	10	0	0
DIJ_MITTENÕUSTUMINE.txt	18	2	16	11,11	11,11
RIJ_TÄNAN.txt	17	0	1	0	0
RIE_SOOVIMINE.txt	16	14	3	87,5	82,35
RIJ_MUU.txt	16	0	3	0	0
PA_ENESEPARANDUS.txt	15	0	5	0	0
IL_JÄRELDAMINE.txt	14	0	15	0	0
IL_MUU.txt	14	0	9	0	0
VR_HINNANGULINE_INFO_OSUTAMINE_UUEKS.txt	13	0	4	0	0
DIJ_MUU.txt	12	0	6	0	0
PA_PARTNERI_PARANDUS.txt	12	0	4	0	0
YA_REFERAAT.txt	12	0	4	0	0
KYJ_MUU_KAS-VASTUS.txt	11	0	7	0	0
YA_LUBADUS.txt	11	4	2	36,36	66,67
YA_RETOORILINE_KÜSIMUS.txt	11	0	6	0	0
DIJ_TEGEVUS.txt	10	7	5	70	58,33
RIE_PALUN.txt	10	2	0	20	100
DIJ_KEELDUMINE.txt	9	0	6	0	0
IL_PEHMENDAMINE.txt	9	0	6	0	0
PPE_MITTEMÕISTMINE.txt	9	2	1	22,22	66,67
KYJ_ALTERNATIIV_MÕLEMAD.txt	8	0	1	0	0
RIE_VABANDUS.txt	8	3	3	37,5	50
VR_HINNANGULINE_JÄTKAJA.txt	7	0	2	0	0
RY_KONTAKTEERUMINE.txt	6	3	2	50	60
SEJ_MITTENÕUSTUMINE.txt	6	0	1	0	0
SEJ_MUU.txt	6	0	1	0	0
SEJ_PIIRATUD_NÕUSTUMINE.txt	6	0	0	0	
KYJ_ALTERNATIIV_EITAV.txt	5	2	3	40	40
TVE_PAKKUMINE.txt	5	0	2	0	0
VR_HINNANGULINE_PIIRITLEJA.txt	5	0	3	0	0
DIE_MUU.txt	4	0	4	0	0

KYJ_ALTERNATIIV_KOLMAS_VALIK.txt	4	0	1	0	0
KYJ_ALTERNATIIV_MUU.txt	4	0	0	0	
TVJ_VASTUVÕTMINE.txt	4	0	1	0	0
KYJ_TEGEVUS.txt	3	0	0	0	
RIE_VABANDUSE_VASTUVÕTMINE.txt	3	0	0	0	
RIJ_LÕPETAMISE_TAGASILÜKKAMINE.txt	3	0	0	0	
SEJ_KEELDUMINE.txt	3	0	1	0	0
KYJ_VASTUS_ALTERNATIIVINA.txt	2	0	1	0	0
RIJ_VASTUSOOVIMINE.txt	2	0	1	0	0
RY_ÄRATUNDMINE.txt	2	0	0	0	
RY_ÜLEANDMINE.txt	2	0	1	0	0
VR_JÄTKAJA.txt	2	0	0	0	
YA_RETOORILINE_VASTUS.txt	2	0	0	0	
DIJ_PAKKUMINE.txt	1	0	0	0	
IL_TÄPSUSTAMNE.txt	1	0	0	0	
KYJ_KAHTLEV.txt	1	0	0	0	
RIE_MUU.txt	1	0	2	0	0
SEE_NÕUSTUMINE.txt	1	0	0	0	
Kokku	**13504**	**6118**	**7386**	**45,31**	**45,31**

Lisa 3. Dialoogiaktide tuvastamine alamsekventside põhjal

Dialoogiakt	Sagedus	Olulisi leitud	Ebaolulisi leitud	Saagis	Täpsus
KYJ_INFO_ANDMINE.txt	1486	811	1820	54,58	30,82
VR_NEUTRAALNE_JÄTKAJA.txt	1019	735	1096	72,13	40,14
VR_NEUTRAALNE_VASTUVÕTUTEADE.txt	707	107	209	15,13	33,86
KYJ_JAH.txt	628	290	533	46,18	35,24
DIJ_INFO_ANDMINE.txt	609	392	548	64,37	41,7
KYE_AVATUD.txt	567	298	663	52,56	31,01
KYE_VASTUST_PAKKUV.txt	521	57	195	10,94	22,62
VR_NEUTRAALNE_INFO_OSUTAMINE_UUEKS.txt	493	465	48	94,32	90,64
IL_TÄPSUSTAMINE.txt	424	12	131	2,83	8,39
RIE_TÄNAN.txt	381	364	34	95,54	91,46
RIE_TERVITUS.txt	376	313	99	83,24	75,97
RIE_KUTSUNG.txt	369	365	12	98,92	96,82
RIJ_KUTSUNGI_VASTUVÕTMINE.txt	366	268	20	73,22	93,06
RIJ_VASTUTERVITUS.txt	346	270	114	78,03	70,31
DIE_SOOV.txt	332	138	197	41,57	41,19
RIJ_PALUN.txt	300	295	51	98,33	85,26
KYE_JUTUSTAV_KAS.txt	292	65	234	22,26	21,74
RY_TUTVUSTUS.txt	275	245	8	89,09	96,84
YA_INFO_ANDMINE.txt	239	0	38	0	0
KYE_SULETUD_KAS.txt	224	35	91	15,63	27,78
VR_NEUTRAALNE_PIIRITLEJA.txt	210	157	100	74,76	61,09
YA_MUU.txt	208	41	39	19,71	51,25
DIJ_NÕUSTUMINE.txt	163	3	17	1,84	15
RIE_HÜVASTIJÄTT.txt	153	120	40	78,43	75
KYJ_EDASILÜKKAMINE.txt	137	31	56	22,63	35,63
DIJ_EDASILÜKKAMINE.txt	131	89	69	67,94	56,33
KYJ_EI.txt	128	72	138	56,25	34,29
YA_PRAAK.txt	127	95	59	74,8	61,69
VR_MUU.txt	126	1	2	0,79	33,33
IL_PÕHJENDAMINE.txt	124	7	41	5,65	14,58
IL_ÜLERÕHUTAMINE.txt	124	0	27	0	0
PPJ_LÄBIVIIMINE.txt	106	0	2	0	0
KYJ_INFO_PUUDUMINE.txt	93	26	67	27,96	27,96
RIJ_VASTUHÜVASTIJÄTT.txt	90	46	35	51,11	56,79
DIE_PAKKUMINE.txt	89	16	59	17,98	21,33
KYE_ALTERNATIIV.txt	85	8	34	9,41	19,05
DIE_ETTEPANEK.txt	73	3	39	4,11	7,14
KYJ_MUU.txt	66	0	3	0	0
IL_KOKKUVÕTMINE.txt	65	2	38	3,08	5
PPE_ÜLEKÜSIMINE.txt	60	0	4	0	0
RIE_LÕPUSIGNAAL.txt	56	0	8	0	0

KYJ_ALTERNATIIV_ÜKS.txt	55	1	1	1,82	50
RIJ_LÕPETAMISE_VASTUVÕTMINE.txt	53	0	2	0	0
KYJ_NÕUSTUV_EI.txt	51	0	4	0	0
SEJ_NÕUSTUMINE.txt	51	0	0	0	
YA_JUTU_PIIRIDE_OSUTAMINE.txt	47	9	15	19,15	37,5
KYE_MUU.txt	45	4	8	8,89	33,33
DIJ_INFO_PUUDUMINE.txt	40	9	16	22,5	36
SEE_ARVAMUS.txt	40	0	0	0	
YA_EELTEADE.txt	39	6	11	15,38	35,29
KYE_VASTUSE_TINGIMUSTE_TÄPSUSTAMINE.txt	38	0	14	0	0
KKE_ALGATUS.txt	34	14	10	41,18	58,33
PPE_ÜMBERSÕNASTAMINE.txt	34	0	2	0	0
IL_SELETAMINE.txt	33	0	7	0	0
SEE_VÄIDE.txt	31	0	5	0	0
DIE_PALVE_OODATA.txt	30	5	13	16,67	27,78
IL_HINNANG.txt	27	0	11	0	0
KYE_TÄPSUSTAV.txt	27	0	0	0	
KKJ_KINNITAMINE.txt	26	0	3	0	0
VR_PARANDUSE_HINDAMINE.txt	25	0	0	0	
VR_HINNANGULINE_VASTUVÕTUTEADE.txt	24	0	3	0	0
DIJ_PIIRATUD_NÕUSTUMINE.txt	20	0	2	0	0
KYJ_KEELDUMINE.txt	20	0	1	0	0
DIJ_MITTENÕUSTUMINE.txt	18	0	8	0	0
RIJ_TÄNAN.txt	17	0	0	0	
RIE_SOOVIMINE.txt	16	15	3	93,75	83,33
RIJ_MUU.txt	16	0	0	0	
PA_ENESEPARANDUS.txt	15	0	0	0	
IL_JÄRELDAMINE.txt	14	0	9	0	0
IL_MUU.txt	14	0	8	0	0
VR_HINNANGULINE_INFO_OSUTAMINE_UUEKS.txt	13	0	0	0	
DIJ_MUU.txt	12	0	0	0	
PA_PARTNERI_PARANDUS.txt	12	0	0	0	
YA_REFERAAT.txt	12	1	5	8,33	16,67
KYJ_MUU_KAS-VASTUS.txt	11	0	0	0	
YA_LUBADUS.txt	11	0	0	0	
YA_RETOORILINE_KÜSIMUS.txt	11	0	0	0	
DIJ_TEGEVUS.txt	10	1	1	10	50
RIE_PALUN.txt	10	0	0	0	
DIJ_KEELDUMINE.txt	9	0	1	0	0
IL_PEHMENDAMINE.txt	9	0	2	0	0
PPE_MITTEMÕISTMINE.txt	9	0	1	0	0
KYJ_ALTERNATIIV_MÕLEMAD.txt	8	0	0	0	
RIE_VABANDUS.txt	8	3	1	37,5	75
VR_HINNANGULINE_JÄTKAJA.txt	7	0	0	0	
RY_KONTAKTEERUMINE.txt	6	0	0	0	
SEJ_MITTENÕUSTUMINE.txt	6	0	0	0	

SEJ_MUU.txt	6	0	0	0	
SEJ_PIIRATUD_NÕUSTUMINE.txt	6	0	0	0	
KYJ_ALTERNATIIV_EITAV.txt	5	0	3	0	0
TVE_PAKKUMINE.txt	5	0	0	0	
VR_HINNANGULINE_PIIRITLEJA.txt	5	0	0	0	
DIE_MUU.txt	4	0	5	0	0
KYJ_ALTERNATIIV_KOLMAS_VALIK.txt	4	0	0	0	
KYJ_ALTERNATIIV_MUU.txt	4	0	1	0	0
TVJ_VASTUVÕTMINE.txt	4	0	0	0	
KYJ_TEGEVUS.txt	3	0	0	0	
RIE_VABANDUSE_VASTUVÕTMINE.txt	3	0	0	0	
RIJ_LÕPETAMISE_TAGASILÜKKAMINE.txt	3	0	0	0	
SEJ_KEELDUMINE.txt	3	0	0	0	
KYJ_VASTUS_ALTERNATIIVINA.txt	2	0	0	0	
RIJ_VASTUSOOVIMINE.txt	2	0	0	0	
RY_ÄRATUNDMINE.txt	2	0	0	0	
RY_ÜLEANDMINE.txt	2	0	0	0	
VR_JÄTKAJA.txt	2	0	0	0	
YA_RETOORILINE_VASTUS.txt	2	0	0	0	
DIJ_PAKKUMINE.txt	1	0	0	0	
IL_TÄPSUSTAMNE.txt	1	0	0	0	
KYJ_KAHTLEV.txt	1	0	0	0	
RIE_MUU.txt	1	0	0	0	
SEE_NÕUSTUMINE.txt	1	0	0	0	
Kokku	**13504**	**6310**	**7194**	**46,73**	**46,73**

Lisa 4. Dialoogiaktide tuvastamine alamsekventside ja diskursuse põhjal

Dialoogiakt	Sagedus	Olulisi leitud	Ebaolulisi leitud	Saagis	Täpsus
KYJ_INFO_ANDMINE.txt	1486	886	1347	59,62	39,68
VR_NEUTRAALNE_JÄTKAJA.txt	1019	846	674	83,02	55,66
VR_NEUTRAALNE_VASTUVÕTUTEADE.txt	707	354	321	50,07	52,44
KYJ_JAH.txt	628	475	308	75,64	60,66
DIJ_INFO_ANDMINE.txt	609	413	440	67,82	48,42
KYE_AVATUD.txt	567	270	479	47,62	36,05
KYE_VASTUST_PAKKUV.txt	521	297	269	57,01	52,47
VR_NEUTRAALNE_INFO_OSUTAMINE_UUEKS.txt	493	437	97	88,64	81,84
IL_TÄPSUSTAMINE.txt	424	54	214	12,74	20,15
RIE_TÄNAN.txt	381	373	81	97,9	82,16
RIE_TERVITUS.txt	376	351	53	93,35	86,88
RIE_KUTSUNG.txt	369	366	0	99,19	100
RIJ_KUTSUNGI_VASTUVÕTMINE.txt	366	327	4	89,34	98,79
RIJ_VASTUTERVITUS.txt	346	324	67	93,64	82,86
DIE_SOOV.txt	332	208	195	62,65	51,61
RIJ_PALUN.txt	300	297	43	99	87,35
KYE_JUTUSTAV_KAS.txt	292	56	125	19,18	30,94
RY_TUTVUSTUS.txt	275	249	39	90,55	86,46
YA_INFO_ANDMINE.txt	239	10	41	4,18	19,61
KYE_SULETUD_KAS.txt	224	27	67	12,05	28,72
VR_NEUTRAALNE_PIIRITLEJA.txt	210	137	104	65,24	56,85
YA_MUU.txt	208	12	16	5,77	42,86
DIJ_NÕUSTUMINE.txt	163	52	53	31,9	49,52
RIE_HÜVASTIJÄTT.txt	153	145	22	94,77	86,83
KYJ_EDASILÜKKAMINE.txt	137	50	44	36,5	53,19
DIJ_EDASILÜKKAMINE.txt	131	94	120	71,76	43,93
KYJ_EI.txt	128	65	89	50,78	42,21
YA_PRAAK.txt	127	70	21	55,12	76,92
VR_MUU.txt	126	11	43	8,73	20,37
IL_PÕHJENDAMINE.txt	124	2	30	1,61	6,25
IL_ÜLERÕHUTAMINE.txt	124	0	18	0	0
PPJ_LÄBIVIIMINE.txt	106	40	6	37,74	86,96
KYJ_INFO_PUUDUMINE.txt	93	18	36	19,35	33,33
RIJ_VASTUHÜVASTIJÄTT.txt	90	76	10	84,44	88,37
DIE_PAKKUMINE.txt	89	22	55	24,72	28,57
KYE_ALTERNATIIV.txt	85	16	34	18,82	32
DIE_ETTEPANEK.txt	73	5	30	6,85	14,29
KYJ_MUU.txt	66	0	5	0	0
IL_KOKKUVÕTMINE.txt	65	1	32	1,54	3,03
PPE_ÜLEKÜSIMINE.txt	60	25	22	41,67	53,19
RIE_LÕPUSIGNAAL.txt	56	11	7	19,64	61,11
KYJ_ALTERNATIIV_ÜKS.txt	55	26	18	47,27	59,09

RIJ_LÕPETAMISE_VASTUVÕTMINE.txt	53	21	11	39,62	65,63
KYJ_NÕUSTUV_EI.txt	51	14	17	27,45	45,16
SEJ_NÕUSTUMINE.txt	51	7	11	13,73	38,89
YA_JUTU_PIIRIDE_OSUTAMINE.txt	47	10	21	21,28	32,26
KYE_MUU.txt	45	3	3	6,67	50
DIJ_INFO_PUUDUMINE.txt	40	8	10	20	44,44
SEE_ARVAMUS.txt	40	3	3	7,5	50
YA_EELTEADE.txt	39	4	7	10,26	36,36
KYE_VASTUSE_TINGIMUSTE_TÄPSUSTAMINE.txt	38	0	10	0	0
KKE_ALGATUS.txt	34	16	9	47,06	64
PPE_ÜMBERSÕNASTAMINE.txt	34	0	0	0	
IL_SELETAMINE.txt	33	0	5	0	0
SEE_VÄIDE.txt	31	2	9	6,45	18,18
DIE_PALVE_OODATA.txt	30	7	7	23,33	50
IL_HINNANG.txt	27	0	20	0	0
KYE_TÄPSUSTAV.txt	27	0	0	0	
KKJ_KINNITAMINE.txt	26	4	2	15,38	66,67
VR_PARANDUSE_HINDAMINE.txt	25	3	8	12	27,27
VR_HINNANGULINE_VASTUVÕTUTEADE.txt	24	0	0	0	
DIJ_PIIRATUD_NÕUSTUMINE.txt	20	6	7	30	46,15
KYJ_KEELDUMINE.txt	20	0	1	0	0
DIJ_MITTENÕUSTUMINE.txt	18	0	9	0	0
RIJ_TÄNAN.txt	17	3	0	17,65	100
RIE_SOOVIMINE.txt	16	13	3	81,25	81,25
RIJ_MUU.txt	16	0	0	0	
PA_ENESEPARANDUS.txt	15	0	0	0	
IL_JÄRELDAMINE.txt	14	0	8	0	0
IL_MUU.txt	14	0	2	0	0
VR_HINNANGULINE_INFO_OSUTAMINE_UUEKS.txt	13	0	0	0	
DIJ_MUU.txt	12	0	1	0	0
PA_PARTNERI_PARANDUS.txt	12	0	1	0	0
YA_REFERAAT.txt	12	0	2	0	0
KYJ_MUU_KAS-VASTUS.txt	11	0	0	0	
YA_LUBADUS.txt	11	0	0	0	
YA_RETOORILINE_KÜSIMUS.txt	11	0	0	0	
DIJ_TEGEVUS.txt	10	1	1	10	50
RIE_PALUN.txt	10	5	1	50	83,33
DIJ_KEELDUMINE.txt	9	0	1	0	0
IL_PEHMENDAMINE.txt	9	0	0	0	
PPE_MITTEMÕISTMINE.txt	9	0	0	0	
KYJ_ALTERNATIIV_MÕLEMAD.txt	8	0	0	0	
RIE_VABANDUS.txt	8	0	0	0	
VR_HINNANGULINE_JÄTKAJA.txt	7	0	0	0	
RY_KONTAKTEERUMINE.txt	6	0	0	0	
SEJ_MITTENÕUSTUMINE.txt	6	0	0	0	
SEJ_MUU.txt	6	0	0	0	
SEJ_PIIRATUD_NÕUSTUMINE.txt	6	0	0	0	

KYJ_ALTERNATIIV_EITAV.txt	5	0	1	0	0
TVE_PAKKUMINE.txt	5	0	0	0	
VR_HINNANGULINE_PIIRITLEJA.txt	5	0	0	0	
DIE_MUU.txt	4	0	5	0	0
KYJ_ALTERNATIIV_KOLMAS_VALIK.txt	4	0	0	0	
KYJ_ALTERNATIIV_MUU.txt	4	0	0	0	
TVJ_VASTUVÕTMINE.txt	4	0	0	0	
KYJ_TEGEVUS.txt	3	0	0	0	
RIE_VABANDUSE_VASTUVÕTMINE.txt	3	0	0	0	
RIJ_LÕPETAMISE_TAGASILÜKKAMINE.txt	3	0	0	0	
SEJ_KEELDUMINE.txt	3	0	0	0	
KYJ_VASTUS_ALTERNATIIVINA.txt	2	0	0	0	
RIJ_VASTUSOOVIMINE.txt	2	0	0	0	
RY_ÄRATUNDMINE.txt	2	0	0	0	
RY_ÜLEANDMINE.txt	2	0	0	0	
VR_JÄTKAJA.txt	2	0	0	0	
YA_RETOORILINE_VASTUS.txt	2	0	0	0	
DIJ_PAKKUMINE.txt	1	0	0	0	
KYJ_KAHTLEV.txt	1	0	0	0	
RIE_MUU.txt	1	0	0	0	
SEE_NÕUSTUMINE.txt	1	0	0	0	
Kokku	**13503**	**7628**	**5875**	**56,49**	**56,49**

ELEKTROONILISED LISAD

- Dialoogiaktide tuvastamise programm
- Dialoogiaktide tuvastamise tulemused
    - Eksperimentide tulemused
        - alamsõned.txt
        - alamsekventsid.txt
        - alamsõned_ ja_ diskursus.txt
    - Eesti Dialoogikorpuse dialoogiakte iseloomustavad mustrid