Loomuliku keele töötlus lõplike automaatidega

Süntaksiteooriad ja -mudelid 2005/06

Kaili Müürisep

ATI

11. mai 2006

Loomuliku keele töötlus lõplike automaatidega

Süntaksiteooriad ja -mudelid 2005/06

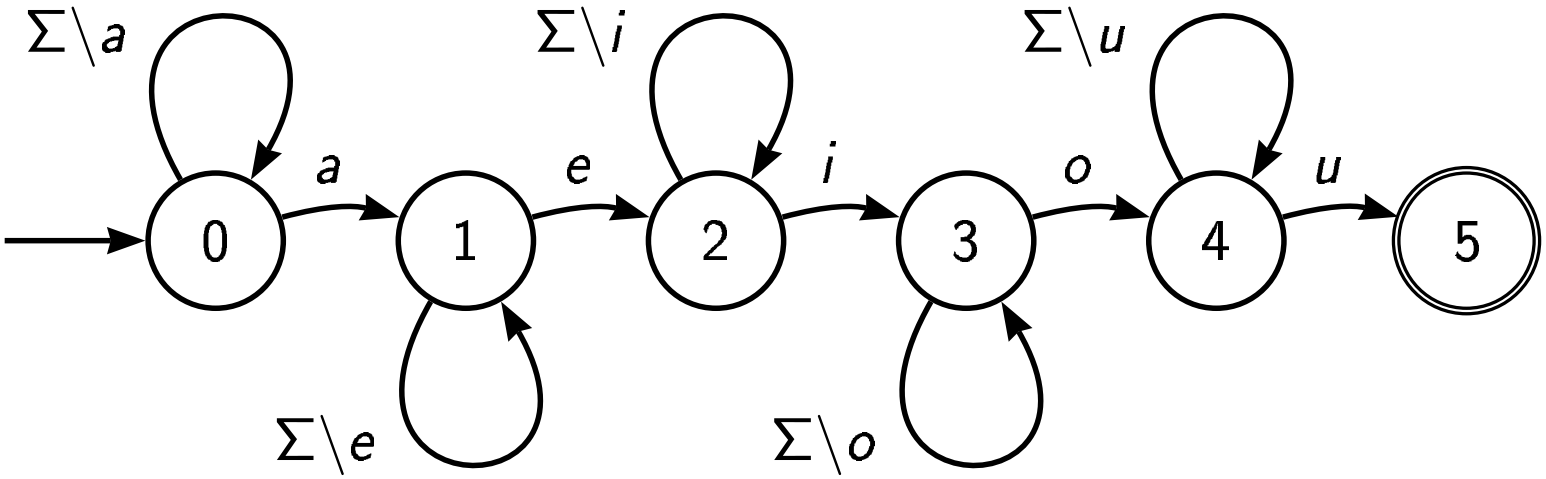
Kaili Müürisep

ATI

11. mai 2006

Automaat

Automaat- programm, mis modelleerib situatsiooni, mis on kirjeldatav olekutega ja üleminekutega ühest olekust teise.
Algoritm esitatakse enamasti diagrammina, mis koosneb tippudest ehk olekutest ja märgendatud kaartest ehk siiretest.
Lõplikud automaadid (finite automaton) leiavad kasutust olukordades, kus arvutusprobleemid saavad olla vaid teatud lõplikus arvus olekutes.



Automaat ja selle genereeritud keel

Me ütleme et automaat tunneb ära teatud keele. St, sellise keele, ehk Σ^* alamhulga, mille puhul iga selle keele sõna aktsepteeritakse antud automaadi poolt.
$L(M) = \{w | w \in \Sigma^*, M(w) returns TRUE\}$

Automaat ja selle genereeritud keel

Me ütleme et automaat tunneb ära teatud keele. St, sellise keele, ehk Σ^* alamhulga, mille puhul iga selle keele sõna aktsepteeritakse antud automaadi poolt.
$L(M) = \{w | w \in \Sigma^*, M(w) returns TRUE\}$

Keele esitamiseks saab kasutada ka nn. produktsiooni reegleid
$S0 \rightarrow aS1|\Sigma \backslash aS0$
$S1 \rightarrow eS2|\Sigma \backslash eS1$
$S2 \rightarrow iS3|\Sigma \backslash iS2$
$S3 \rightarrow oS4|\Sigma \backslash oS3$
$S4 \rightarrow uS5|\Sigma \backslash uS4$
$S5 \rightarrow \Sigma S5|\varepsilon$

Automaat ja selle genereeritud keel

Me ütleme et automaat tunneb ära teatud keele. St, sellise keele, ehk Σ^* alamhulga, mille puhul iga selle keele sõna aktsepteeritakse antud automaadi poolt.
$L(M) = \{w | w \in \Sigma^*, M(w) returns TRUE\}$

Keele esitamiseks saab kasutada ka nn. produktsiooni reegleid
$S0 \rightarrow aS1|\Sigma \backslash aS0$
$S1 \rightarrow eS2|\Sigma \backslash eS1$
$S2 \rightarrow iS3|\Sigma \backslash iS2$
$S3 \rightarrow oS4|\Sigma \backslash oS3$
$S4 \rightarrow uS5|\Sigma \backslash uS4$
$S5 \rightarrow \Sigma S5|\varepsilon$

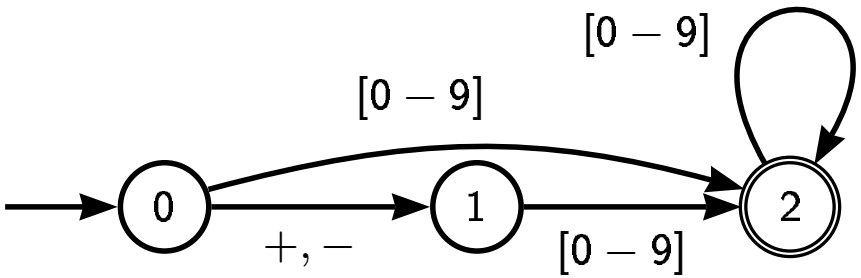
Selliseid produktsioone saab kasutada ka keelde kuuluvate sõnade genereerimiseks:
S0 $\rightarrow$ a S1 $\rightarrow$ a b S1 $\rightarrow$ a b e S2 $\rightarrow$ a b e i S3 $\rightarrow$ a b e i o S4 $\rightarrow$ a b e i o u S5 $\rightarrow$ a b e i o u ε

Determineeritud lõplik automaat

Definition

(DFA) Lõplik determineeritud automaat (ingl. finite automaton) on viisik M=(Q, Σ, δ, q0, F), kus

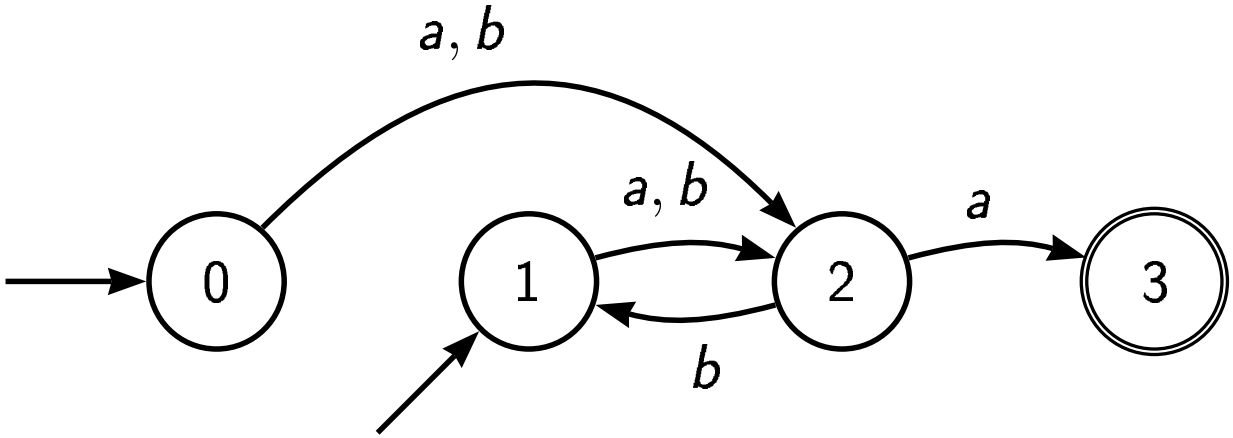
- Q on automaadi olekute (ingl. states) lõplik hulk
- Σ on sisendtähestik (input alphabet)
- $\delta : Q \times \Sigma \rightarrow Q$ on automaadi üleminekuseos (siirdefunktsioon) (transition function)
- q0 $\in$ Q on automaadi algolek (initial state)
- F $\subseteq$ Q on (aktsepteerivate) lõppolekute hulk (accepting final states)

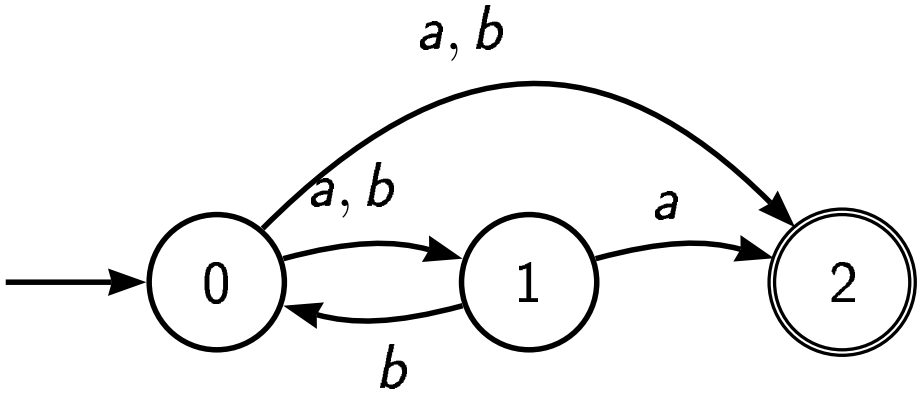


Deterministliku automaadi minimiseerimine

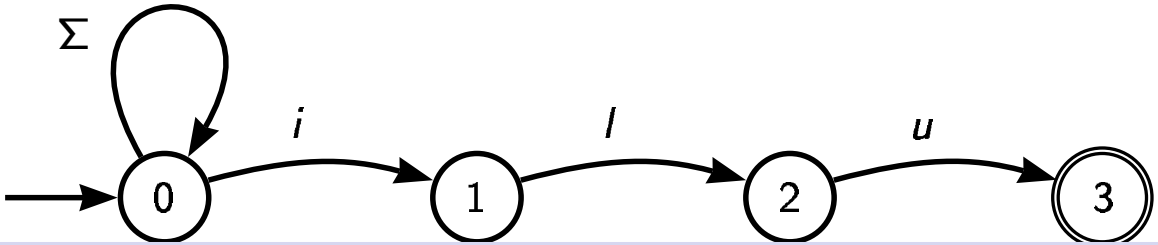
- Kaks automaati mis tunnevad ära sama keele on ekvivalentsed
- Lõplik automaat on minimaalne kui see on väikseima olekute arvuga ekvivalentsete automaatide klassis
- Automaat kus on rohkem olekuid on redundantne
- Automaate moodustavad algoritmid ei tee alati minimaalset automaati
- Lihtsam on aru saada väikse mitteredundantse automaadi tööst
- Ilma asjata pole vaja hoida üleliigseid olekuid
- minimaalse automaadi töötlemine on efektiivsem

Näide ekvivalentsetest automaatidest





Mittedeterminееritud lõplik automaat



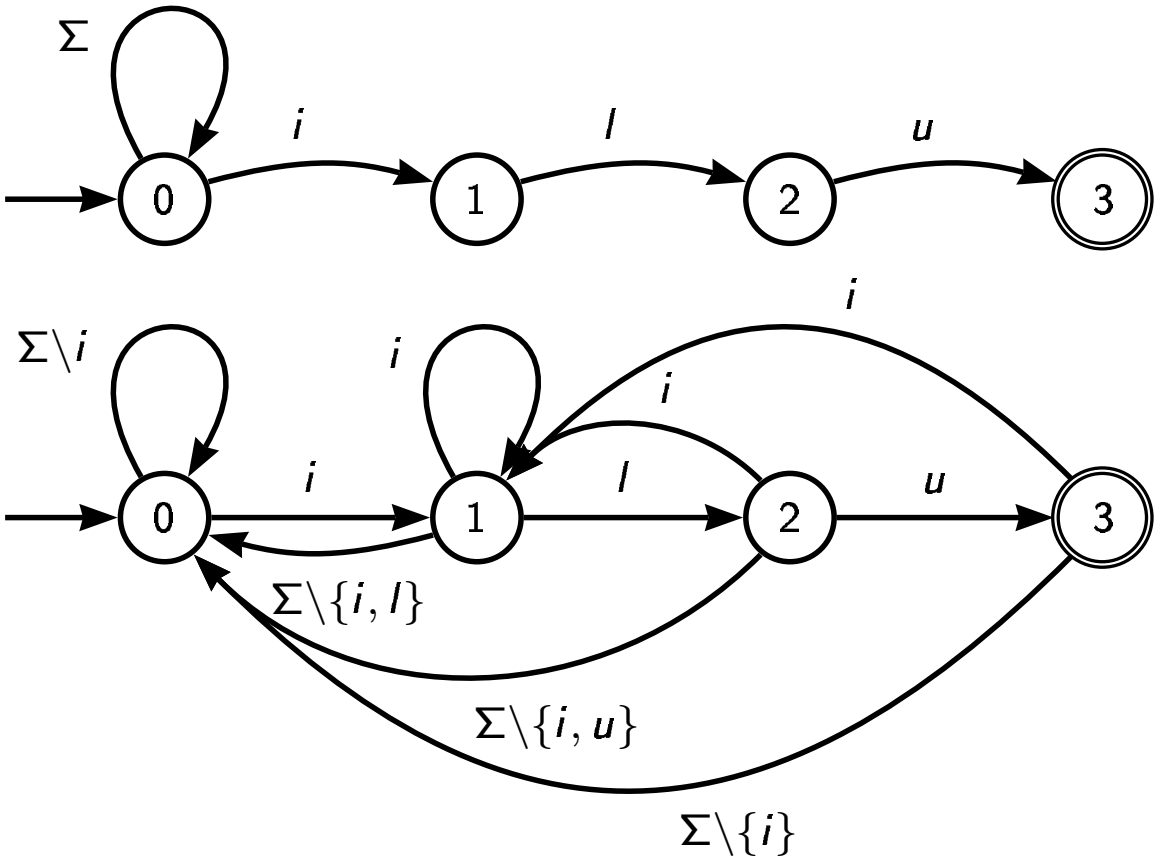
Definition

(NFA) Lõplik mittedeterminееritud automaat on viisik M=(Q, Σ, δ, q_0, F), kus

- Q on automaadi olekute lõplik hulk
- Σ on sisendtähestik
- $\delta : Q \times \Sigma \cup \varepsilon \rightarrow P(Q)$ on automaadi üleminekuseos
- $q_0 \in Q$ on automaadi algolek
- $F \subseteq Q$ on aktsepteerivate lõppolekute hulk

DFA teisendamine NFAks

Mittedetermineeritud automaadi saab muuta determineerituks



Regulaaravaldised

Definition

Regulaaravaldised (RE) on

1. Øon RE; Øon tühi sõnede hulk
2. Iga $a \in \Sigma$ on RE
3. Kui A ja B on RE, siis on regulaaravaldised ka:
   1. (A + B), ühend
   2. (AB), konkatenatsioon
   3. (A*), Kleene'i sulund

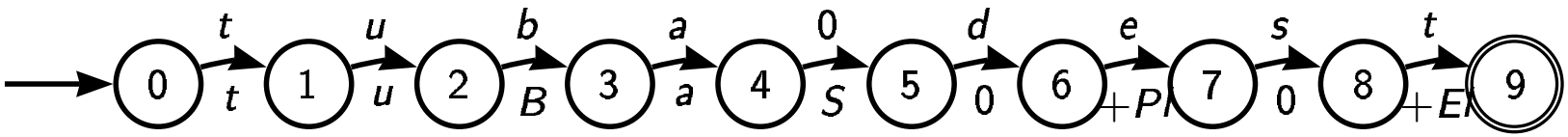
Regulaaravaldise põhjal on võimalik koostada NFA ja sellest DFA
Regulaarsed avaldised defineerivad keele,mida saab töödelda lõpliku automaadiga.

Veel regulaaravaldistest

Regulaaravaldisi saab kasutada tekstilise mustri tuvastamiseks ehk mallvõrdlemise (võrdlus näidisega, pattern-matching) vahendina, mida rakendatakse mingile tekstile. Iga teksti jaoks antakse vastus, kas tekst sobis avaldisega või mitte. Regulaaravaldistes on oluline sümbolite omavaheline järjestus.
Matemaatiliselt, iga regulaaravaldis esindab keelt (sõnade hulka). Sõnad, mis sobivad regulaaravaldise mustriga, kuuluvad sinna keelde, sõnad, mis ei sobi mustriga, ei kuulu.
Mustri tuvastamist võib vaadelda kui masinat, mis saab sisendiks sümbolite jada (sõna) ning tulemuseks on sõna aktsepteerimine või mitteaktsepteerimine. Kõikide sisendsõnade läbi proovimisel saamegi keele, mille see masin defineerib (ära tunneb).

Muundurid

Muundurid e Transductorid e Transducers Lõpilikud automaadid, mille igal kaarel on sisendsümbol ja väljundsümbol



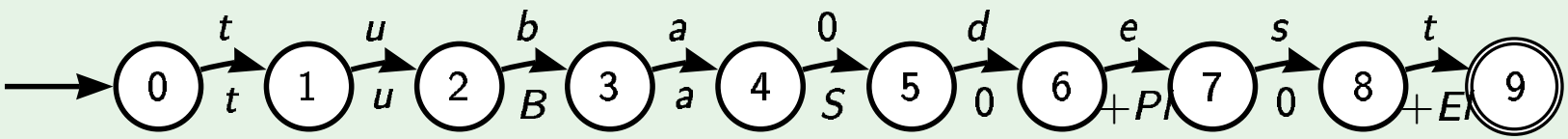
Lõplike automaatide kasutamine keeletöötluses

- morfoloogiaanalüsaator
- morfosüntaktiline ühestaja
- pindsüntaktiline analüsaator

Lõplikel automaatidel põhinev morfoloogiaanalüsaator

Transduktoris seab mistahes tee algolekust lõppolekusse omavahel vastavusse mingi sõnavormi (surface form) ja tema lemma+ morfoloogilise info (lexical form).

Olekudiagramm



Kompaktne esitus

t u B:b a +S:0 0:d +Pl:e 0:s +El:t

Morfoloogiaanalüsaator

Morfoloogias tuleb modelleerida kaks põhilist protsessi:

1. Morfotaktika (kuidas kombineeritakse morfeemidest sõnavormid)
   - prefiksid ja sufiksid, liitsõnamoodustus - konkatenatsioon
   - reduplikatsioon, infiksatsioon, interdigitatsioon - mittekonkatenatiivsed protsessid
2. Fonoloogilised/ortograafilised alternatsioonid
   - assimilatsioon (hind : hinna)
   - lisandumine (jooksma : jooksev)
   - kadu (number : numbri)
   - geminatsioon (tuba : tuppa)

Morfoloogiline ühestamine

Morfoloogilisel ühestamisel kasutatakse lõplikke automaate eelkõige statistilistes meetodites:

- Brilli märgendaja õpib reeglid eelmärgendatud korpusest, need reeglid saab teisendada lõplikeks automaatideks
- Peidetud Markovi mudel -automaadiga on seotud tõenäosused

Brilli märgendaja

- Leksikaalne märgendaja - lisab kõige tõenäolisema märgendi
- Tundmatute sõnade mõistataja
- Kontekstipõhine märgendaja

Brilli märgendaja näide

Algsed laused

1. Chapman /np killed /vbn John /np Lennon /np
2. John /np Lennon /np was /bedz shot /vbd by /by Chapman /np
3. He /pps witnessed /vbd Lennon /np killed /vbn by /by Chapman /np

Brilli märgendaja näide

Algsed laused

1. Chapman /np killed /vbn John /np Lennon /np
2. John /np Lennon /np was /bedz shot /vbd by /by Chapman /np
3. He /pps witnessed /vbd Lennon /np killed /vbn by /by Chapman /np

Reeglid

1. vbn vbd PREVTAG np
2. vbd vbn NEXTTAG by

Brilli märgendaja näide

Algsed laused

1. Chapman /np killed /vbn John /np Lennon /np
2. John /np Lennon /np was /bedz shot /vbd by /by Chapman /np
3. He /pps witnessed /vbd Lennon /np killed /vbn by /by Chapman /np

Reeglid

1. vbn vbd PREVTAG np
2. vbd vbn NEXTTAG by

Analüüsitud laused -reegel 1

1. Chapman /np killed /vbd John /np Lennon /np
2. John /np Lennon /np was /bedz shot /vbd by /by Chapman /np
3. He /pps witnessed /vbd Lennon /np killed /vbd by /by Chapman /np

Brilli märgendaja näide

Algsed laused

1. Chapman /np killed /vbn John /np Lennon /np
2. John /np Lennon /np was /bedz shot /vbd by /by Chapman /np
3. He /pps witnessed /vbd Lennon /np killed /vbn by /by Chapman /np

Reeglid

1. vbn vbd PREVTAG np
2. vbd vbn NEXTTAG by

Analüüsitud laused -reegel 2

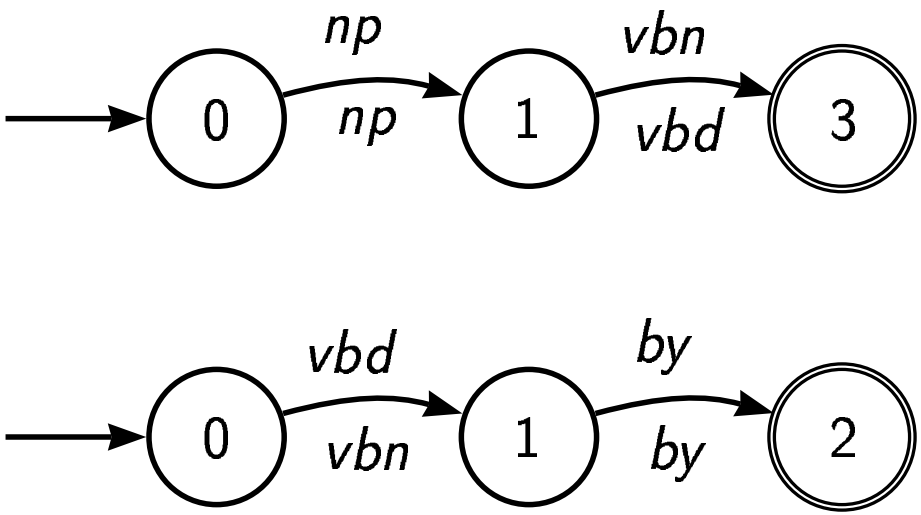
1. Chapman /np killed /vbd John /np Lennon /np
2. John /np Lennon /np was /bedz shot /vbn by /by Chapman /np
3. He /pps witnessed /vbd Lennon /np killed /vbn by /by Chapman /np

Reeglite liigid

- A B PREVTAG C
- A B PREV1OR2OR3TAG C
- A B PREV1OR2TAG C
- A B NEXTTAG C
- A B NEXT1OR2TAG C
- A B SURROUNDTAG C D
- A B NEXTBIGRAM C D
- A B PREVBIGRAM C D

Keerukus RKn, kus R - reeglite arv, K - konteksti pikkus, n -sõnade arv

Reeglite teisendamine muunduriks



Ajavõit: 500 sõna/s $\longrightarrow$ 10800 sõna/s

NPpiiride leidmine

Lõplikke automaate ja regulaaravaldisi kasutatakse palju just lause fraasideks jagamisel Reeglid, mis leiavad chunke:

```
<DT>?<JJ>*<NN.?>
 the/DT little/JJ cat/NN sat/VBD on/IN the/DT mat/NN
 [the/DT little/jj cat/NN] sat/VBD on/IN [the/DT mat/NN]
```

Positiivne lähenemine:

```
   (<DT>?<JJ>*<NN.?>) -->{\1}
```

Välistav lähenemine:
Kogu tekst sulgudesse:

```
 (<.*>*) --> {\1}
```

Chinkide abil tükeldamine

```
((<VB.?>|<IN>)+ --> }\1{
   {the/DT little/jj cat/NN} sat/VBD on/IN {the/DT mat/NN}
```

Lõplike olekutega markerid ja filtrid

Notatsioonist

x:x a:y*
[]:x
y:[]

- Lõplike olekutega marker on muundur, mis lisab uue sümboli.
- Lõplike olekutega filter on muundur, mis väljastab ainult osa sisendsõnest.

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid

```
NounGroup =
[ [ Art  => _ [ Noun ]] &
  [ Noun => _ [ PAdj | Prep | .#. ]] &
  [ PAdj => _ [ PAdj | Prep | .#. ]] &
  [ Prep => _ [ Art | Noun ]] ] &
[ [ Art | Noun ] [ Art | Noun |Padj |Prep ]* ] ;

MarkNGroup = NounGroup @-> "<NG" ... "NG>" ;
```

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid

```
NounGroup =
[ [ Art  => _ [ Noun ]] &
  [ Noun => _ [ PAdj | Prep | .#. ]] &
  [ PAdj => _ [ PAdj | Prep | .#. ]] &
  [ Prep => _ [ Art | Noun ]] ] &
[ [ Art | Noun ] [ Art | Noun |Padj |Prep ]* ] ;

MarkNGroup = NounGroup @-> "<NG" ... "NG>" ;
```

Sisend

Administration/NN of/IN 10/CD per/IN cent/NN oxygen/NN to/IN the/AT ewe/NN for/IN 1/CD hour/NN prior/NN to/IN delivery/NN did/DOD not/NOT alter/VB the/AT surfactant/JJ properties/NNS of/IN the/AT fetal/JJ tracheal/JJ fluid/NN

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid

```
NounGroup =
[ [ Art  => _ [ Noun ]] &
  [ Noun => _ [ PAdj | Prep | .#. ]] &
  [ PAdj => _ [ PAdj | Prep | .#. ]] &
  [ Prep => _ [ Art | Noun ]] ] &
[ [ Art | Noun ] [ Art | Noun |Padj |Prep ]* ] ;

MarkNGroup = NounGroup @-> "<NG" ... "NG>" ;
```

Väljund

<**NG** Administration/NN of/IN 10/CD per/IN cent/NN oxygen/NN to/IN the/AT ewe/NN for/IN 1/CD hour/NN prior/NN to/IN delivery/NN **NG**> <**VG** did/DOD not/NOT alter/VB **VG**> <**NG** the/AT surfactant/JJ properties/NNS of/IN the/AT fetal/JJ tracheal/JJ fluid/NN **NG**>

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid
2. Markeeritakse fraaside põhjad

Marker

```
HeadNouns =
  "*HeadN" -> [] || [ "<NG" | TAG ] _ [ NOUN ] [ ~$ NOUN ]
                [ INGVERB | PPART | PREP | COMMA | CC | "NG>" ] ;
PrepNouns =
  "*PrepN" -> "*HeadN"
                || [ [$ PREP] & ~$[ PREP ?* [PREP |COMMA | "NG>" ]] _ ;
LabelNounFST = [ PrepNouns .o. HeadNouns ]
```

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid
2. Markeeritakse fraaside põhjad

Väljund

<NG Significiant/JJ *HeadN correlations/NNS NG> <VG were/BED *PasV obtained/VBV VG> <NG between/IN the/AT maternal/JJ and/CC fetal/JJ glucose/NN *PrepN levels/NNS and/CC the/AT maternal/JJ and fetal/JJ ffa/JJ *PrepN levels/NNS NG> ./SENT

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid
2. Markeeritakse fraaside põhjad
3. Süntaktilised filtrid

Skeem

```
[ ]:LEFTCONTEXT
              token
                [ ]:MIDDLE
                        token
                           relation:[ ]
                                [ ]:RIGHTCONTEXT
```

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid
2. Markeeritakse fraaside põhjad
3. Süntaktilised filtrid

Näide

<NG *HeadN corticosteroid/NNS NG> <VG did/DOD not/NOT appear/VB to/TO *ActV affect/VB VG> <NG the/AT *HeadN progress/NN of/IN the/AT *PrepN disease/NN NG>

```
FilterSubj=
  [ ]:? * [ ]:"*HeadN"
       Token
          []:[ ~$ ["<NG"|"VG>"]] [ ]:"ActV"
              Token
                "<SUBJ":[ ]
                    [ ]:? *;
```

corticosteroids/NNS affect/VB <SUBJ

Parsimine filtreerimise teel

1. Tuvastatakse ja tähistatakse markerite abil nimisõna- ja verbigruppide piirid
2. Markeeritakse fraaside põhjad
3. Süntaktilised filtrid

Näide 2

<NG *HeadN amyloidosis/NN NG> <VG was/BEDZ *PasV found/VBN VG>

```
FilterPassDobj =
  [ ]:? * [ ]:"HeadN"
        Token
            [ ]:[ ~$ ["<NG"|"VG>"]] [ ]:"*PasV"
                Token
                   " <PDOBJ":[ ]
                          [ ]:? *;
```

amyloidosis/NN found/VBN <PDOBJ

Filtreerimise tulemused

- 10000 sõna minutis (1999. a)
- 76% SUBJ-seostest olid korrektsed
- 80% PDobj-seostest oldi korrrektsed

Sõltuvuste analüüs kasutades lõplikke automaate

(1) *O adam bir elma yedi*
see mees (üks) õun sõi
'see mees sõi õuna'

```
0. <(o)> <(adam)> <(bir)> <(elma)> <(yedi)>
1. <0(o)d> <D(adam)0> <0(bir)d> <D(elma)0> <0(yedi)0>
2. <00(o)0d> <D0(adam)s0> <01(bir)1d> <D1(elma)10> <0S(yedi)00>

[ LR [ML IGMiddle MR]* RL ] (->) "{Rel" ... "Rel}" || IGDep IGHead

[ LR [ ML AnyIG MR ]* RL ] (->) "{SBJ" ... "SBJ}" ||
                          NominativeNominalA3pl _ FiniteVerbA3sgA3pl;
```