From Grammar Checker to Syntactic Trees: Estonian CG

Kaili Müürisep
Krista Liin
University of Tartu

Outline

- Developing grammar checker for Estonian
- Converting morph disambiguation rules to CG3
- Spoken language and corpus of dialects
- Building trees
- Particle verbs of Estonian
- Future plans

Grammar checker

- Main error types of Estonian orthography
  - foreign words, h in beginning of words, length of consonants, compound words, typos ⇨ spell checker
  - intrasentential punctuation
  - bad style
  - errors of language learners/foreigners

Detector of comma errors

- Goals:
  - avoid false alarms
  - focus on error detection, not correction
  - only comma errors, except commas in parentheses (non-verbal clauses) and semantically/pragmatically motivated commas
  - uses output of morphological disambiguator / syntactic analyser

How to collect comma errors?

- Commentariums of online news portals, internet forums – spontaneous writings of different users
- Plan: to use first versions of bachelor theses of students of mathematical-informatics department

The main idea

- Labels @OK and @ERR have been attached to conjunction words and finite verbs.
- Constraints have been applied to check whether there are other finite verbs in left or right contexts.
- The input is automatically morphologically disambiguated.

The errors

- non-finite clauses
- verb is erroneously disambiguated as noun
- some word form has ambiguous undeleted verb reading

The numbers

- Korrektselt tuvastatud komavigade osakaal kõigist grammatikakorrektori poolt antud komavigade märgenditest ehk grammatikakorrektori täpsus testkorpusel on 93,8%.
- Saagis ehk leitud komavigade suhe korpuses leidunud (ja käsitsi märgendatud) komavigadesse on 94,1%.

Morphological disambiguation

- Original rules written by Tiina Puolakainen 1995-2001
- In the style of Karlsson's first version of CG
- Maybe the semantic of rules is little bit different
- Program is implemented by her, C, designed for os windows
- 1400 rules

Conversion

- manually
- I tried to understand the idea of the rule
- hard to find one-to-one translation

Sample rule

```
(@w =0 (_S_ sg nom) (NOT -1 Terve) \
        (NOT *1 PrVerb) (NOT *-1 PrVerb) (NOT *1 DaInf) \
        (NOT *-1 DaInf) (NOT *1 Imper) (NOT *-1 Imper) \
        (*-2C SPSgNom W)(NOT W AdjPron W)\
        (NOT W LisaNom W) \
        (NOT W ASPron L-1) (NOT L-1 Terve W) \
        (NOT W Prep R+2) (NOT R+2 Postp *R+0) \
        (NOT *R+0 Konj&Kvm) CLB)
```

Spoken language

Adaption of grammar

- New POS – special particles – *ahah*, *mhmh*, *hurraa*, *jess*, *ee*, *õõ*, *noh* etc.
- New syntactic labels:
  @B – syntactically independent uninflected words;
  @T – unknown syntactic function.
- compile new rules for the sentence internal clause boundary detection
- fix the syntactic constraints (slight modifications of less than 100 rules from 1200)

Self-repairs (2)

```
väga [ADVL]
    väga+0 //_D_ //     **CLB @ADVL
!!!nor- [REP]
!!!    nor //_T_  #- //
!!!väga [REP]
!!!    väga+0 //_D_ //
normaalne [PRD]
    normaalne+0 //_A_ pos sg nom //    @PRD
noh [B]
    noh+0 //_B_ //    @B
väga [ADVL]
    väga+0 //_D_ //    @ADVL
```

Self-repairs (3)

väga
    väga+0 //_D_ //   **CLB @ADVL
nor-
    nor //_T_ #- // @REP
väga
    väga+0 //_D_ //   @REP
normaalne
    normaalne+0 //_A_ pos sg nom //   @PRD
noh
    noh+0 //_B_ //   @B
väga
    väga+0 //_D_ //   @ADVL

Results

	Normalized Prec/Rec	**Original** Prec/Rec	**Now** Prec/Rec
Repairs	87.6/96.4	84.9/94.6	85.5/95.0
Repeats	91.8/98.6	90.7/98.2	92.1/98.6
False starts	93.8/98.9	90.0/97.4	91.1/98.1
Institutional calls			86.8/96.9
Everyday conversation			91.8/97.6

Types of errors

Error type	Information dialogue	Everyday conversation
Ellipsis	19	13
Synt apocope	13	20
Synt epanorthosis	3	3
Anacoluthon	2	1
Anaphoric constructions	2	4
Agreement	6	5
Vocabulary	4	5
Clause boundaries	13	14
Syntactic rules	36	33

Dialects

```
!!!u who AU
!!!>a=kui pal'lu nad pidid `maksma siss selle `ves'ki=pealt (...)
$</s>
$<s>
!!!u who KJ
ma
    mina+0 //_P_ pers ps1 sg nom //  **CLB @SUBJ
!!!=
i
    ei+0 //_V_ aux neg #FinV //  @NEG
mäletta
    mäleta+0 //_V_ main ps indic pres neg #FinV #Part-P #InfP //  @+FMV
seda
    see+0 //_P_ dem sg part //  @OBJ
$.$.$.
    $. //_Z_ Fst //
ind
    hind+0 //_S_ com sg nom //  **CLB-C @SUBJ
ol'i
    ole+0 //_V_ main ps indic impf sg ps3 #FinV #Intr //  @+FMV
ikke
    ikka+0 //_D_ //  @ADVL
```

Trees

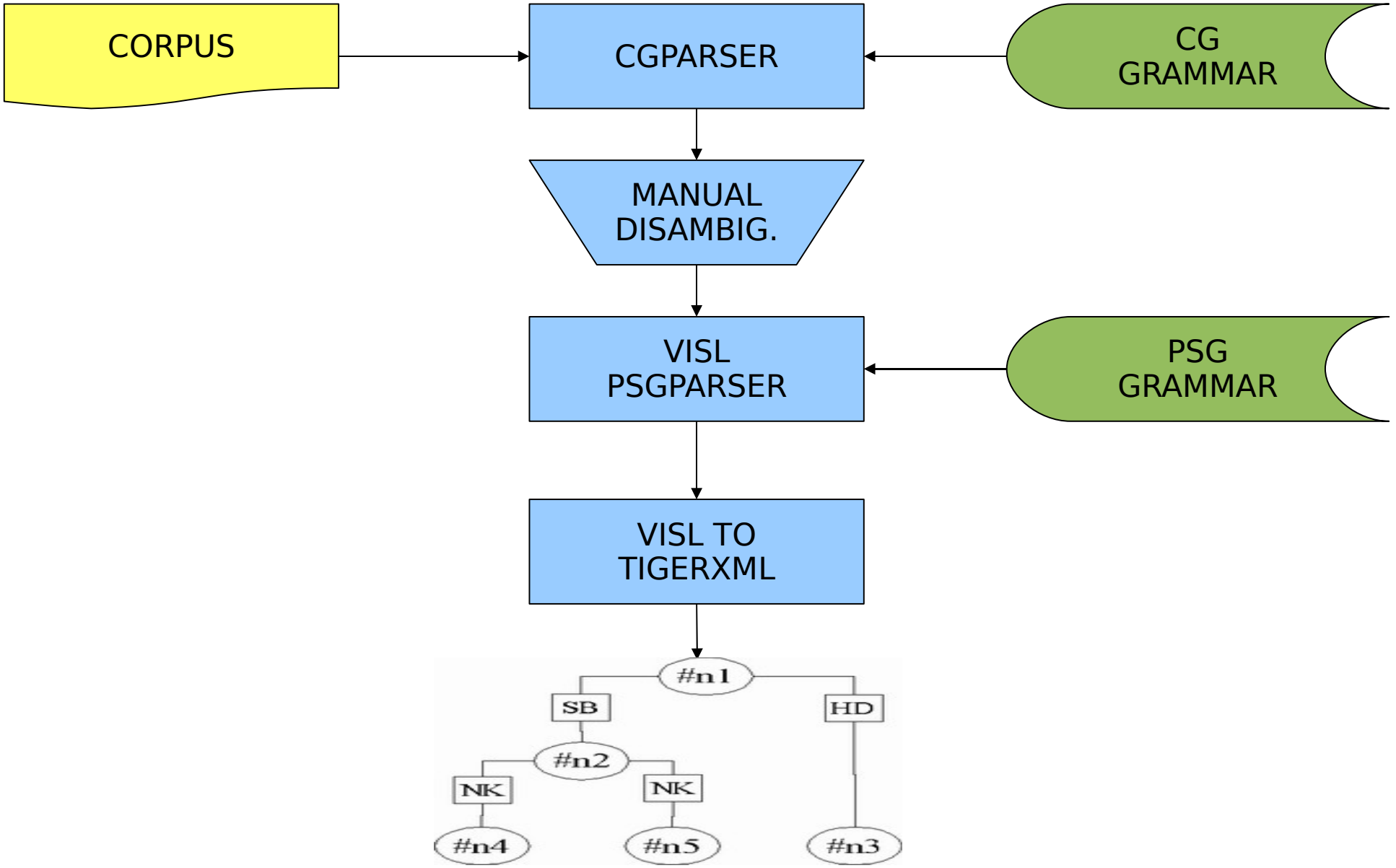
Description of Corpus

370 simple sentences with verbs that indicate motion from „Types of Simple Sentences in Estonian” by H. Rätsep.

Originally these are constructed sentences in order to describe syntactic contexts of the verbs.

- *Tankid lähenesid tee poolt kõrgendikule*
  Tanks approached from the road to the hill
- *Uurija lähenes küsimusele oskuslikult*
  Researcher approached to the problem skilfully

Towards deeper syntactic analysis



Shallow syntactic analysis

```
$<s>
 Peeter                                                    ;; Peeter
    Peeter+0 //_S_ prop sg nom #cap //  @SUBJ
hiilis                                                     ;; sneaked
    hiili+s //_V_ main indic impf ps3 sg ps af #FinV //  @+FMV
linnaääri                                                  ;; suburb
    linna_äär+i //_S_ com pl part //  @P>
mööda                                                      ;; through
    mööda+0 //_K_ post #part //  @ADVL
koosolekult                                                ;; meeting
    koos_olek+lt //_S_ com sg abl //  @ADVL
koju                                                       ;; home
    kodu+0 //_S_ com sg adit //  @ADVL @NN> @<NN
püssi                                                      ;; gun
    püss+0 //_S_ com sg gen //  @P>
järele                                                     ;; for
    järele+0 //_K_ post #gen //  @ADVL
.
    . //_Z_ Fst //
$</s>
 !<hiilima#67.3.>
```

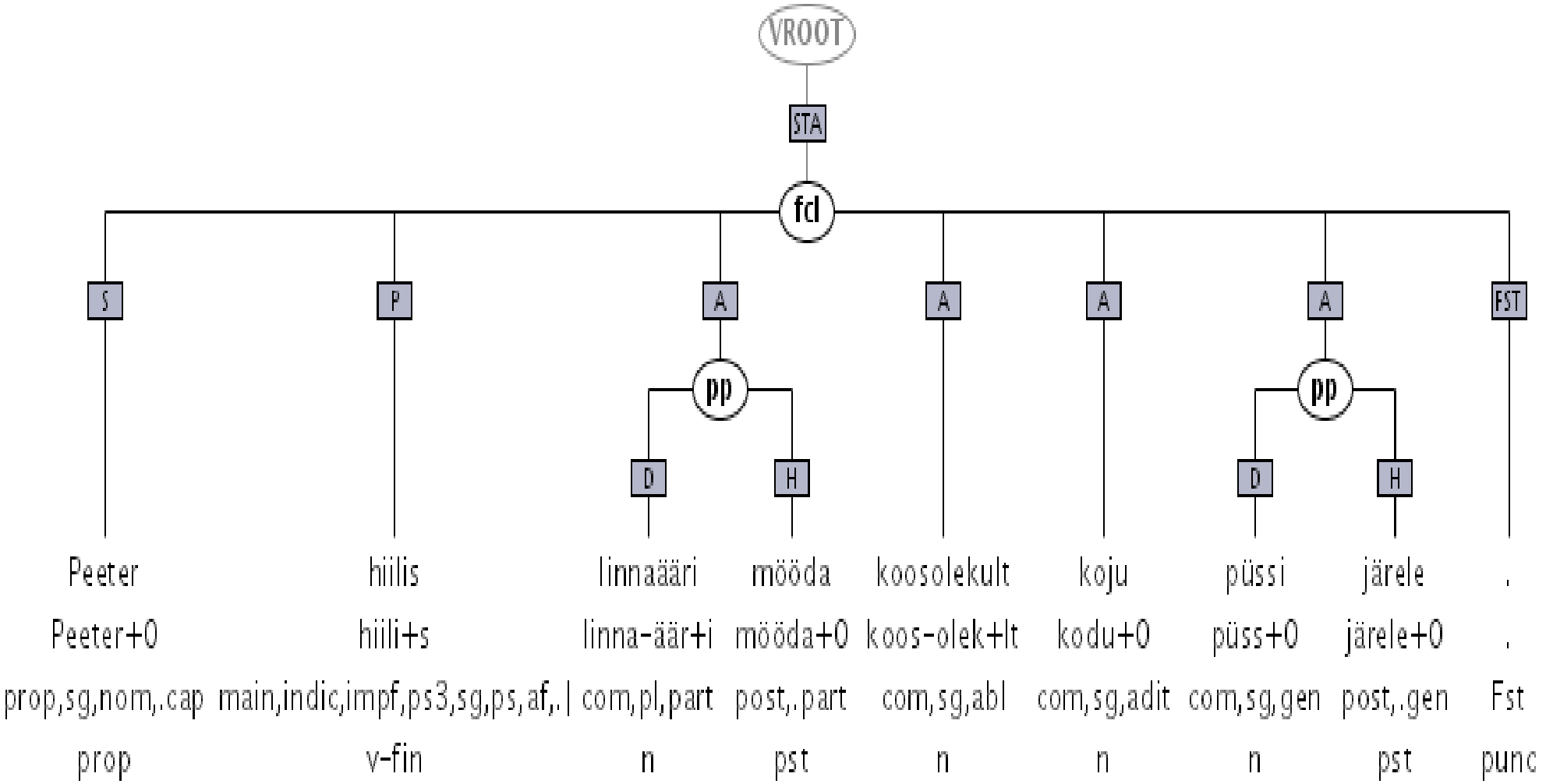
VISL PSGparser

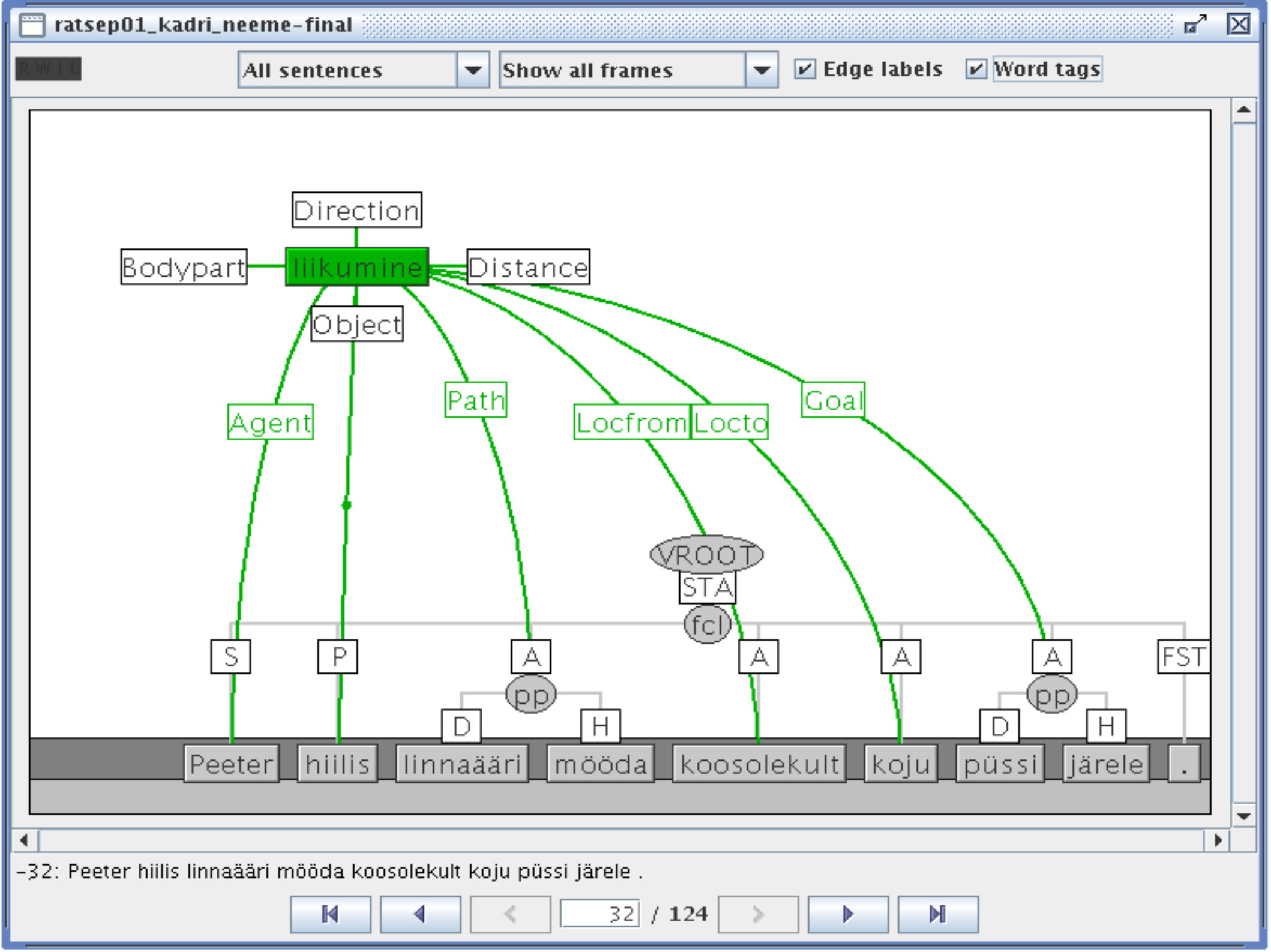
```
ADJP:ap = ADVL[->D]:adv AN>[->H];

ATRNP:np = {AN>,ADJP}[->D] NN>[->H];

S:np = {AN>,ADJP,NN>,ATRNP}[->D]  S[->H];

S:np = S[->H] <NN[->D];

PREP:np = {AN>,ADJP,NN>,ATRNP}[->D] P<[->H];

A:pp = A[->H]:prp <P[->D];

A:pp = A[->H]:prp PREP[->D]:np;

STA:fcl = S P A O A A FST;

STA:fcl = S P O A FST;

STA:fcl = S P A O A FST;

STA:fcl = S P O A A FST;

STA:fcl = S P A A A FST;

STA:fcl = S P A A FST;
```

Visl format

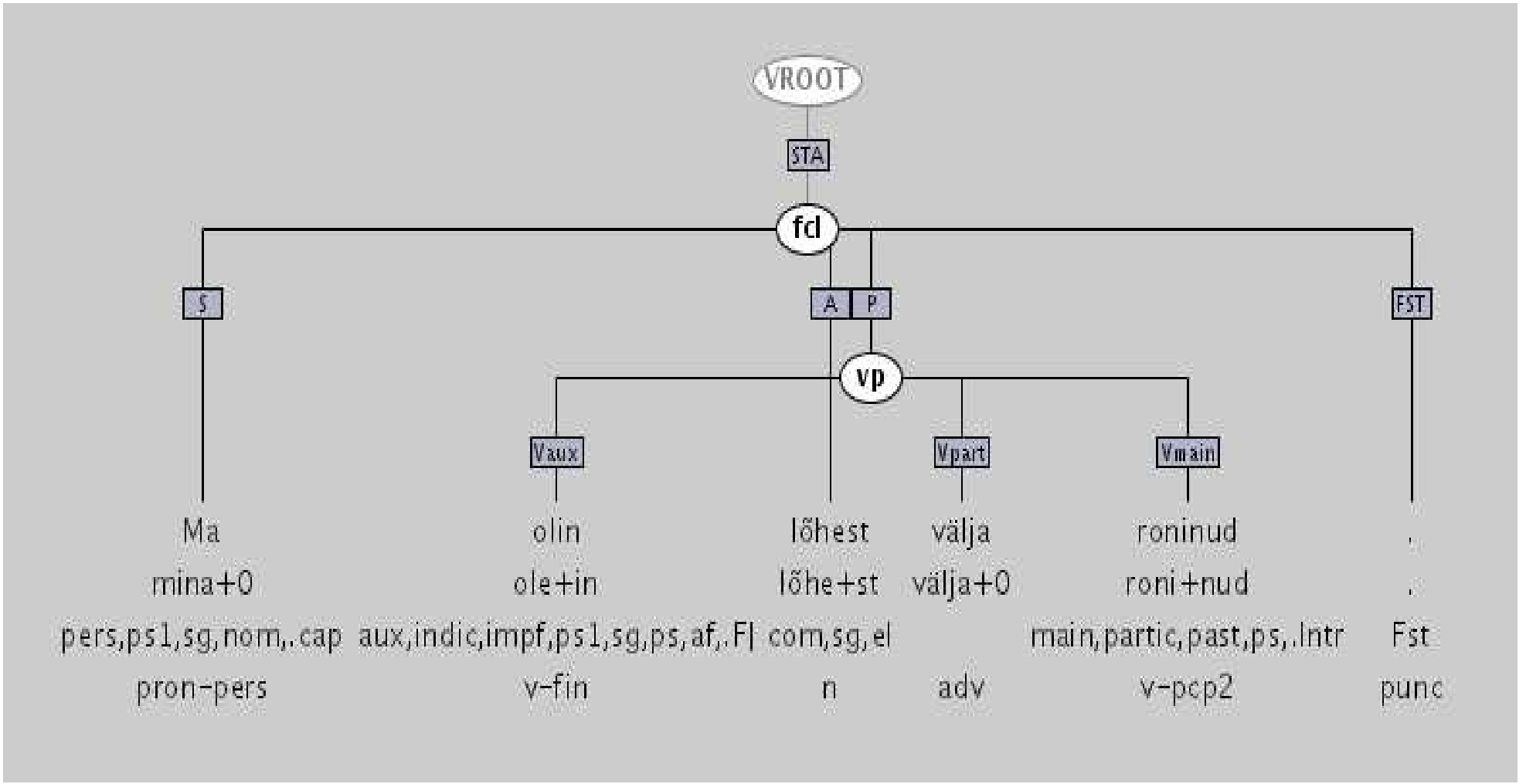
```
STA:fcl
S:prop('Peeter+0',prop,sg,nom,.cap)     Peeter
P:v-fin('hiili+s',main,indic,impf,ps3,sg,ps,af,.FinV)   hiilis
A:pp
=D:n('linna-ää;r+i',com,pl,part)        linnaääri
=H:pst('mööda+0',post,.part)    mööda
A:n('koos-olek+lt',com,sg,abl)  koosolekult
A:n('kodu+0',com,sg,adit)       koju
A:pp
=D:n('püss+0',com,sg,gen)       püssi
=H:pst('järele+0',post,.gen)    järele
FST:punc('.',Fst)       .
```

TigerXML output




ratsep01_kadri_neeme-final
All sentences
Show all frames
Edge labels
Word tags
Direction
Bodypart
liikumine
Distance
Object
Agent
Path
Locfrom
Locto
Goal
VROOT
STA
fcl
S
P
A
pp
D
H
A
A
A
pp
D
H
FST
Peeter
hiilis
linnaääri
mööda
koosolekult
koju
püssi
järele
.
-32: Peeter hiilis linnaääri mööda koosolekult koju püssi järele .
32 / 124

Trees with particle verbs



Ambiguities with particle verbs

Mari heitis mulle laiskust ette.

m[nom] throw.3sg.past I.all laziness.PART PRT

‘Mary reproached me for my laziness.’

Muri heitis ukse ette matile.

m[nom] throw.3sgpast door.gen PRT mat.all

‘Muri lay down on the mat in front of the door.’