Arborest – a Growing Treebank of Estonian

by

Eckhard Bick

Institute of Language and Communication
University of Southern Denmark
lineb@hum.au.dk

Heli Uibo and Kaili Müürisep

Institute of Computer Science
University of Tartu, Estonia
{heli.uibo, kaili.muurisep}@ut.ee

1. Introduction

Treebank creation is a very labor-consuming task, especially if the applications intended include machine learning, gold standard parser evaluation or teaching, since only a manually checked syntactically annotated corpus can provide optimal support for these purposes. There are, however, possibilities to make the annotation process (partly) automatic, saving (manual) annotation time and/or allowing the creation of larger corpora. Whenever possible, existing resources – both corpora and grammars – should be reused.

In the case of the Estonian treebank project Arborest, we have therefore opted to make use of existing technology and experiences from the VISL project[1], where two-stage systems including both Constraint Grammar (CG)- and Phrase Structure Grammar (PSG)-parsers have been used to build treebanks for several languages (Bick, 2003 [1]). Moreover, the VISL annotation scheme has been

[1] URL: http://visl.sdu.dk

adopted as a standard for tagging the parallel corpus in Nordic Treebank Network[2]. For Estonian, there already exists a shallow syntactically annotated – and proof-read – corpus, allowing us to bypass the first step in treebank construction (CG-parsing).

This paper describes how a VISL-style hybrid treebank of Estonian has been semi-automatically derived from this corpus with a special Phrase Structure Grammar, using as terminals not words, but CG function tags. We will analyze the results of the experiment and look more thoroughly at adverbials, non-finite verb constructions and complex noun phrases.

The questions we will try to answer are:

- How much can we automatize the process of treebank creation on the basis of the existing morphologically and shallow syntactically tagged corpus?
- What kind of additional information could the PSG rules obtain from morphological analysis, if implemented in the compiler formalism?
- What kind of information is principally missing in the Estonian CG corpus and what kind of enrichment of categories is needed to facilitate the automatic treebank creation?

2. Estonian Constraint Grammar Corpus

The shallow syntactically annotated corpus was considered necessary for training and evaluation of the Constraint Grammar based shallow syntactic parser of Estonian, the detailed description of which is given in the subsection 2.1. The development of the corpus started in 1998 with the gold standard corpus, consisting of 20 000 words of Estonian original fiction from 1980s. During 1999-2003 the corpus has been extended to ca 200 000 words, including 177 000 words of fiction, 10 000 words of newspaper texts and 6 000 words of legal texts. The process of creation of Estonian CG Corpus is described in (Uibo, 2004 [12]). 65 000 words of newspaper texts from 1996-1999 and 20 000 words of sample sentences for different sentence templates (Rätsep, 1978 [11]) will be added by the end of the year 2004.

2.1. Estonian Constraint Grammar Parser

The Estonian Constrain Grammar parser (Müürisep et al, 2003 [8]) has been developed in 1996-2000 by T. Puolakainen and K. Müürisep. It is the first attempt to automate the syntactic analysis of Estonian.

---

[2] URL: http://w3.msi.vxu.se/~nivre/research/nt.html

The main idea of the Constraint Grammar (Karlsson et al, 1995 [5]) is that it determines the surface-level syntactic analysis of the text which has gone through prior morphological analysis. The process of syntactic analysis consists of three stages: morphological disambiguation, identification of clause boundaries, and identification of syntactic functions of words. Grammatical features of words are presented in the forms of tags which are attached to words. The tags indicate the inflectional and derivational properties of the word and the word class membership, the tags attached during the last stage of the analysis indicate its syntactic functions. The underlying principle in determining both the morphological interpretation and the syntactic functions is the same: first all the possible labels are attached to words and then the ones that do not fit the context are removed by applying special rules or constraints. Constraint Grammar consists of hand written rules which by checking the context decide whether an interpretation is correct or has to be removed.

A number of rules are clearly of a heuristic nature – the rule might not be 100 % true but its proficiency rate is very high, compared to the number of errors. Several rules have been compiled solely on the statistical information about the word order in the sentence. The rules are grouped in such a way that the most reliable ones or those that cause least errors are in the main part of the grammar; the heuristic rules in turn have been divided into groups based on their reliability.

The grammar consists of 1,240 morphological disambiguation rules, 47 clause boundary detection rules, 180 morphosyntactic mapping rules and 1,118 syntactic constraints. The morphological disambiguation rules are commented in detail in (Puolakainen, 2001 [10]) and syntactic constraints in (Müürisep, 2000 [7]).

Evaluation of the morphological disambiguator show the recall 86.6 % and the error rate 1.8 %. The results of the full syntactic analysis show the ambiguity rate of 17 % (83 % of all wordforms are unambiguous) and the error rate of 3.5 %. (Müürisep et al, 2003 [8]).

2.2. Estonian Constraint Grammar Tagset

Estonian Constraint Grammar (EstCG) uses the following set of syntactic tags:

@+FMV – finite main verb
@-FMV – non-finite main verb
@+FCV – finite modal or auxiliary verb
@-FCV – non-finite modal or auxiliary verb
@NEG – negator (particles *ei, ära* as a part of a negative verb-form)
@SUBJ – subject

@OBJ – object
@PRD – predicative complement
@ADVL – clause level adverbial or modifier of an adverb or an adjective
@AN> or @<AN – an adjective or ordinal as a modifier
@NN> or @<NN – noun as a modifier (of a noun)
@AD> or @<AD – adverb as a modifier (of a noun)
@VN> or @<VN – participle as a modifier (of a noun)
@INF_N> or @<INF_N – infinitive as a modifier (of a noun)
@PN> or @<PN – adposition (more precisely: the adpositional phrase as a whole) as a modifier (of a noun)
@<P or @P> – noun belonging to the adpositional phrase (*on the table*)
@<Q or @Q> – noun belonging to the quantifier (*five men*)
@J – conjunction
@I – interjection

**CLB marks a very likely clause boundary and **CLB-C a less likely clause boundary. The analysis is performed inside the clause (sentential clause) boundaries only. No attempt is made to connect the clauses.

2.3 Representation Formats of EstCG Corpus

Part of EstCG Corpus is available as a directory of text files in the web[3]. In these files one word-form occupies two lines: the word-form itself is on the first line and the lemma+inflectional endings, morphological analysis and syntactical tag are on the second line (cf. Figure 1).

```
Mälestustes
   mälestus+tes //_S_ com pl in #cap //  **CLB @ADVL
muutus
   muutu+s //_V_ main indic impf ps3 sg ps af #FinV #Intr //  @+FMV
kõik
   kõik+0 //_P_ det sg nom //  @SUBJ
vapustavalt
   vapustavalt+0 //_D_ //  @ADVL
kauniks
   kaunis+ks //_A_ pos sg tr //  @ADVL
$.$.$.
   $.$.$. //_Z_ Ell //
```

Figure 1: Example sentence from EstCG Corpus.
(*Everything became strikingly beautiful in the memories*...)

[3] URL: http://lepo.it.da.ut.ee/~heli_u/SA.html

EstCG Corpus has also been converted to NEGRA export format (Brants, 1997 [2]) by Kaarel Kaljurand[4], thus now it can be searched and visualized with the TIGERSearch tool (Lezius, 2002 [6]). However, the trees are very flat – the smallest unit for grouping is a subclause and all the subclauses are at one and the same level. It is because CG markup includes clause boundary tags only; it does not contain information about the hierarchy of subclauses.

3. VISL-style treebanks

Acknowledging the need for a common set of grammatical categories for the annotation of its multilingual teaching treebanks, the VISL group of researchers at the Institute of Language and Communication (University of Southern Denmark) has held a large number of terminological workshops over several years, and agreed upon a set of annotation principles and grammatical labels, known as the Cafeteria Categories. Throughout the system, each VISL language and each VISL annotator have striven to make use of existing Cafeteria core categories wherever possible, even at the price of slight remaining differences in category definitions, adding subcategory extensions where necessary, rather than coining new labels from scratch. Like the Nordic Treebank Network in general, the Arborest treebank project has chosen, wherever possible, to adhere to VISL style categories in its syntactic annotation, adopting the following principles:

- Each node in a syntactic tree is annotated with both a function and a form label.
- Optimally, only branching nodes are used, i.e. the form of the daughter in a non-branching node is raised and expressed as the mother's function.
- **Function labels** have upper case key letters, **form labels** have lower case key letters. A complete node label in constituent grammar notation fuses form and function with a colon, e.g. S:np (subject noun phrase).
- **Subcategories** are attached to function labels in lower case, and to form labels with a hyphen. The distinction between adjunct and argument can be optionally marked with a 'b' (bound) or 'f' (free) in front of the upper case function label.
- In constituent grammar notation, if crossing branches are unwanted, **discontinuous constituents** (crossing branch nodes) are marked with hyphens pointing towards the constituent's other part(s), e.g. P:vp- fA -P:vp.

The core categories for clause level function are the following:

- **S** Subject, subcategories e.g.: **Ss** Situative subject, **Sf** Formal subject
- **P** Predicator or Verbal constituent (function of "small vp")

[4] URL: http://psych.ut.ee/~kaarel/Programs/Treebank/EstCG2Negra

- **O** Object, subcategories, e.g.: **Od/Oacc** direct (accusative) object **Oi/Odat** indirect (dative) object, **Op** prepositional object, **Ogen** genitive object
- **C** Predicative or complement, subcategories: **Cs** Subject complement, **Co** Object complement, **fC** free (subject) complement
- **A** Adverbial, subcategories e.g.: **fA** Free adverbial, **As** Subject-bound adverbial, **Ao** Object-bound adverbial

Form categories are divided into complex forms and word class forms. Complex forms are clauses **(cl)**, groups **(g)** and paratagmata or compound units **(par)**. Core categories are:

- **fcl** Finite clause, **icl** Non-finite clause, **acl** Averbal (verb-elliptic) clause
- **np** Noun phrase, **adjp** Adjective phrase and **advp** Adverb phrased, **pp** Prepositional phrase, **vp** Verb phrase
- **par** Paratagma (Coordinated unit)
- At the group level, the minimal annotation is dependency based, with one **H** (head) and one or more **D** (dependent) constituents. Dependents can optionally be subclassified as to valency:
- **Darg** Argument dependent
- **Dmod** Modifier dependent
- Dependent function in groups is defining for group form, and can thus be subdivided accordingly:
- **DN** Adnominal dependent (in np's, possibly specified as **DNarg** or **DNmod**), with subclasses like e.g.: **DNapp** Apposition, **DNc** Predicative adnominal dependent
- **DA** Adverbial dependent (in **adjp**'s and **advp**'s, can be **DAarg** or **DAmod**), subclass example: **DAcom** Argument of comparator
- **DP** Argument or modifier of preposition
- **DC** Modifier of conjunction
- The **vp** ("little vp") has special constituents, rather than head and dependent, since a syntactic/dependency view and a semantic "main verb" view can't agree on what the head is:
- **Vm** Main verb
- **Vaux** Auxiliary
- **Vpart** Verb integrated particle

Finally, word class form operates with a cafeteria consisting of **n**, **prop**, **v** (**v-fin**, **v-inf**, **v-pcp**), **adj**, **adv**, **pron** (with subclasses), **prp**, **art**, **num**, **conj** (**conj-s**, **conj-c**) and **intj**. The syntactic top-node receives the default function of **UTT** (utterance), but may be subdivided into **STA** statement, **QUE** question, **COM** command, **EXC** exclamation, **PER** performative.

For undefined or unclear functions, (uppercase) **X** is used, undefined or unclear forms are (lower case) **x**. These are also used to handle coordination of parts of constituents (e.g. shared subject, coordinated object-adverbial pairs), where the paratagma receives X-function, while its daughter conjuncts receive x-form, delegating specific function and form to the conjuncts' daughters. On top of the above, VISL has introduced certain experimental function categories, such as **TOP** (topic), **FOC** (focus), **VOC** (vocative) and **fAsta** (statement apposition).

4. Conversion of EstCG Corpus to Arborest

4.1. The cg2tree compiler

The automatic creation of Arborest analyses is handled by a context free PSG, using VISL's open source cg2tree compiler. The formalism allows mother-from-daughters rewriting rules, addressing function and form tags, as well as word forms and base forms (lexemes), all of which can be combined among themselves (sets and negated sets) or with each other (conditioned nodes). Each rule can be conditioned by additional operators, like '!' (not as top node) or '+' (at least 2 daughters). Each daughter node expression can be suffixed by regular expression style existential operators (?, *, +). Since cg2tree grammars typically expect CG-annotated input, terminals will typically be function:form expressions, making use of word or base forms only as form restrictors.

```
FM:fm = A:a.{'w1', 'w2', ...} B[->B2]:b[->b2] .... C*/+/? .... {D1, D2, ...}:^{d1,d2 ...}
```

Figure 2: Example PSG rule.

In the rule given on Figure 2, FM and fm are the mother node's function and form, respectively, rewritten as a chain of daughters A ... D, where A is conditioned by a specific set of words, and D is given as a set of functions and a negated (^) set of forms. For B, tags are rewritten as B2 and b2, if the rule is instantiated, and C is an example of regular expression operators allowing 0 or more (*), 1 or more (+) or 0/1 (?) repetitions.

While the compiler formalism is language independent and has successfully been used to create CG-to-PSG grammars in a number of other languages (Danish, German, English, French, cf. Bick, 2003 [1]), the grammar rules themselves have to be more language specific, and obviously also depend on the kind of CG input they receive – its tag granularity, level of dependency specification etc. Finally, the grammar will depend on the descriptive linguistic tradition it is set to implement (small or large VP, use of non-finite clauses etc). Luckily, since all Constraint Grammars so far share most of their core function tags and all adhere to the same structural paradigm (flat dependency grammar), at least rule **types**

can be ported from one language to another, especially for lower level constituents. Thus, it is possible to adapt certain rules rather than write them from scratch. For Estonian, for instance, pp-rewriting is basically the same as for English, but left hand arguments have to be provided for, since the language uses adpositions rather than (only) prepositions.

4.2. The PSG grammar

The first two examples handle ordinary finite statement clauses, while the second two example rules create object subclauses from underspecified input by drawing on complementizer words (the conjunctions "et+0" and "kas+0").

```
STA:fcl = CLB? OBJ-QUOTE? {ADVL,OBJ,PRD}:^{d-rel}* SUBJ {ADVL,OBJ,PRD}
* P {ADVL,OBJ,PRD}* ARGS? {$.,$;}? ; # SOV, SVO, OSV

STA:fcl = CLB? OBJ-QUOTE? {ADVL,OBJ,PRD}:^{d-rel,p-rel}* P
{ADVL,OBJ,PRD}* SUBJ {ADVL,OBJ,PRD}* ARGS? {$.,$;}? ; # OVS, VSO, VOS

OBJ:fcl = $,? CLB ADVL:d? {SUB,ADVL}.{"et+0","kas+0"} {ADVL,OBJ,PRD}*
SUBJ? {ADVL,OBJ,PRD}* P {ADVL,OBJ,PRD}* ARGS? CLB? ; # SOV, SVO -- also
without subject: kas ei saaks tund aega (not marked as 'v-imps'!! --- also adverbial
between CLB and SUB: U¤ Mitte et Rootsi kapital on halb.

OBJ:fcl = $,? CLB ADVL:d? {SUB,ADVL}.{"et+0","kas+0"} {ADVL,OBJ,PRD}* P
{ADVL,OBJ,PRD}* SUBJ {ADVL,OBJ,PRD}* ARGS? CLB? ; # OVS, VSO, VOS
(only OSV lacking!)
```

Individual tags can be rewritten one-to-one inside a rule, if and when it is instantiated. Thus, object functions [OBJ] in the co-ordination rule below are rewritten as conjuncts [CJT];

```
OBJ:cu = ADVL:d.{'nii'}? OBJ[->CJT]:^{cu}+ CO OBJ[->CJT]:^{cu} ;
```

Rules allow both function and form variables (X and x, respectively), which are, however, in the current formalism not unified across the right hand side of a rewriting rule.

```
X:np =+ {AN>,NN>,VN>}:^{np,s-gen,prop-gen}* X[->H]:{s,num,p}
{<AN,<NN,<VN,<CN,<PN,<INFN,<AD}* ;
```

The current PSG grammar comprizes 110 rules, roughly a quarter of which are finite clause rules, another quarter are phrase (group) rules, and a third quarter covers coordination patterns. With variable unification, the number of

coordination rules could be reduced by using general rules like: X:cu = X+ CO X, which now have to be individually listed for all constituent types.

In other VISL grammars, notably Germanic ones, the uniqueness principle has been implemented by specifying allowed constituent orders individually. For Estonian, however, which has a much freer word order, clause level constituent chains have to accommodate for all S-V-O combinations but the infamous OSV. Therefore, possible constituent chains have been lumped by using {ADVL,OBJ,PRD} or similar sets with the *-operator. As a result, current rules have a laxer uniqueness constraint, at clause level basically limited to subordinators, predicator and subject. Since the Estonian CG does not provide dependency direction markers for clause level constituents, its grammar design decision of *-lumping constituents, would have been risky, were it not for the clause boundary markers (CLB) supplied by the CG-grammar and used as delimiters in the PSG.

Though linguistic theory treats auxiliaries and verb chains in various ways, one descriptional convention had to be favoured over another, and for the sake of notational compatibility, the VISL treebank convention of “small vp” was adopted, with a predicator constituent (P) consisting of finite and non-finite main verbs (MV), chain verb “auxiliaries” (CV) and negation particles, leaving objects and other verb complements outside the vp, and not recognizing “argument of auxiliary/CV” constituents either:

P:vp = NEG? {FCV,FMV} ICV* IMV; # FMV allowed due to: ... *jääb püsima füüsiline töö*
P:vp = NEG? IMV ICV* FCV ; # inverse vp: *näidata sai*

Not least in newspaper text, embedded sentences occur fairly frequently, often marked by parenthesis or pairs of quotes or hyphens. In order to reduce the complexity of the grammar, such punctuation is not ignored but rather used to delimit embedded sentences, which are then rewritten as themselves, but including the otherwise isolating boundaries:

X:x = $" X:x $" ; # ..., *kõikesuutvate masinate ajastul, ...*
X:x = $(X:x $) ;

5. Results of Conversion

We have examined and manually revised 149 trees – the corpus *Estonian-best*, containing articles from an issue of the Estonian weekly newspaper "Eesti Ekspress" (from August, 1996). 61 trees were correct, i.e. had both correct branching structure and correct labels for forms as well as for functions. Among

the correct sentences the following subclause structures were represented (unified):

(1) (A) S (A) P A*
(2) S P (A) C (A)
(3) S P (A) O A*
(4) O S A P
(5) O P S A
(6) A O A P A+ (no subject)
(7) A+ P (A)S A*
(8) (A) P A*(S)A*O A*
(9) A P O S
(10) A* P C A O S
(11) C P A S

Generalizing, we could add A* everywhere in between S, P, O and C in the structures.

Estonian is a free-word-order language and that has been taken into account in the rules. Simple sentences with the word order S-P-O, S-O-P and P-S-O plus maybe A* everywhere have been correctly parsed. The predicative complement (C) can occur either after or before predicate.

The structure (4), where the predicate is in the end, occurred in subordinated clauses only. However, a predicate may also occur at anterior positions in subordinated clauses.

The subject is not an obligatory clause constituent in Estonian, and the subject is "inflexion-included" in the verb form (1rd or 2rd person verb forms).

In Estonian discontinuous verb phrases where object or adverbial(s) occur in the middle of the verb phrase are quite common. There is a convenient way to represent discontinuous structures in the VISL tag set and a comprehensible format to represent it graphically (cf. Figure 3).

The trees for composite sentences (subclauses bound with *ja, ning (and)*, *või*, *ehk* (*or*) or comma) and complex sentences with subordinated clauses in the function of adverbial (*kui ... siis* (*if ... then*)) or object (beginning with the subordinating conjunction *et* or an interrogative-relative pronoun *kes*, *mis*) have also been correctly built. An example of a correctly analyzed complex sentence is given in figure 4 (complex sentence with a subordinated clause).

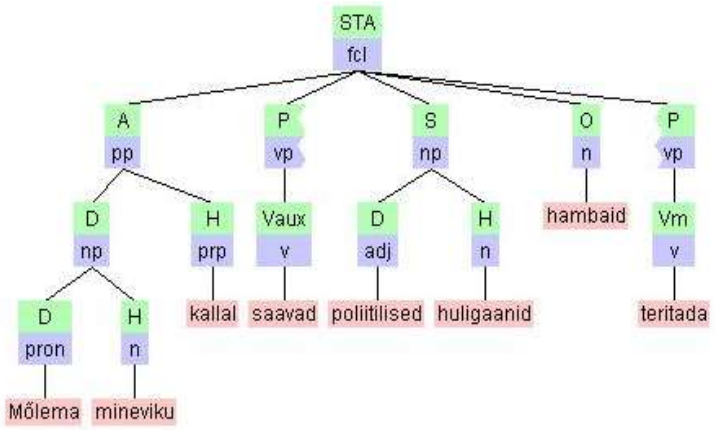

Figure 3: Tree of a sentence with a discontinuous verb phrase (*saavad teritada*). (*Political hooligans can sharpen their teeth on the past of both persons*.)

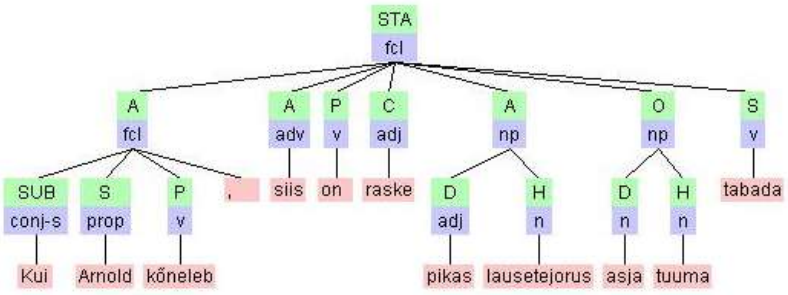

Figure 4: Tree of a sentence with the subclause in the function of adverbial. *Kui Arnold kõneleb, siis on raske pikas lausetejorus asja tuuma tabada*. (*If Arnold talks, then it is difficult to get the point in the long row of senteces*.)

In the subsections 5.1 – 5.3 the entities that caused the largest numbers of false structures will be analyzed – adverbials, non-finite clausal constructions and complex noun phrases.

5.1. Adverbials

The family of adverbial constituents is represented by only two tags in EstCG – @AD> / @<AD – as adverbial modifiers of nouns (mostly state adverbials) and @ADVL – for all other adverbials (including adjective-phrase-internal adverbial modifiers, like "very big"). Therefore, it is sometimes unclear, where to attach adverbs. We have seen in the corpus *Estonian best* that an adverb modified an adjective only in two sentences out of 149, but it was erroneously attached to the NP in more than 10 sentences (e.g. sentence 52 which is visualized in figure 5).

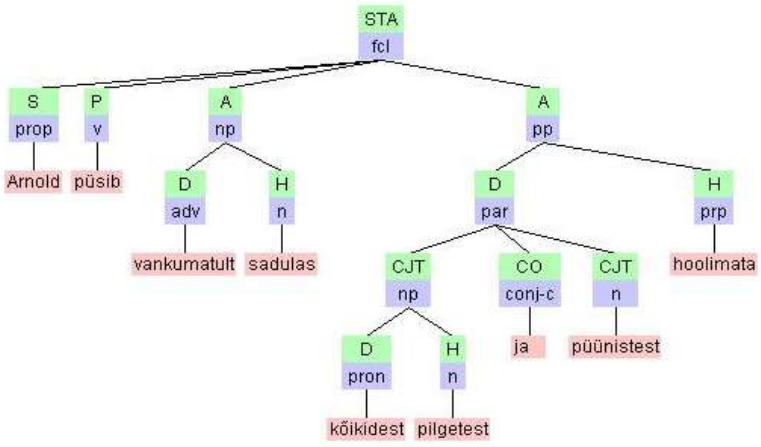

Figure 5: Tree of a sentence where an adverb is wrongly attached to a NP. The adverb *vankumatult* (*immovably*) actually is a free adverbial. *(Arnold sits immovably on his horse, regardless of all gibes and traps*.)

Thus, the adverbial attachment rules are overgenerating and should be revised. Some PSG errors occurred, because a correcting rule turning ADVL into group dependents like DN or DA, overgenerated. Provided a 99% consistent adverbial tagging in the CG source corpus, such rules could and should, of course, be abolished, and the risk of overgeneration be reduced as a consequence.

There is a list of adverbs that can be only phrase-attached: *kõige, liiga, üpris, üsna*, which can be exploited by PSG rules, but there is a considerably longer and open list of adverbs that can act both as free adverbials and adverbal modifiers.

Another possible solution to the adverbial problem is to subcategorize the ADVL tag. There are at least two different principles of classification of adverbials – by semantics and by syntactic function. For example, in Functional Dependency Grammar (Järvinen & Tapanainen 1998, [4]) tagset there are twenty different adverbial tags, classified by the semantic role of the adverb, corresponding to the single ADVL tag in EstCG. Alternatively, we could divide the adverbials according to their syntactic functions, e.g. as follows:

1. AdjP or AdvP-dependent adverb (*very big*, *too quickly*) [VISL: DA]
2. predicate-dependent adverbials (*He painted the wall green*) [VISL: Co (adjectival object complement), As, Ao (subject- or object-bound adverbial). In Estonian syntax (Erelt et al, 1993 [3]) this is called "dependency adverbial" or "valency adverbial", as in Estonian syntax the object can be only in nominative, genitive or partitive case.]
3. non-predicate verb dependent adverbials (*Walking in the park was his favorite hobby*.) [VISL: fA, but a part of a non-finite rather than a finite clause]
4. free adverbial (*It is raining outside*.) [VISL: fA]

As one of the motivations for building Estonian treebank is the research on predicate-argument structures it is significant to distiguish at least between verb-dependent and free adverbials.

5.2. Non-finite clauses

Non-finite clausal constructions (infinitival and averbal clauses, short clauses with participles as a predicate, ma-supine infinitival clauses, participles as noun modifiers) are not easy to recognize in Estonian, especially when they are not separated by a comma. Moreover, there are no infinitival markers (like *to* in English) in Estonian. This problem caused 8 errors in the corpus *Estonian-best*.

The solution can be to add an explicit CG-tag for the start word of such clauses. However, the automatic detection of non-finite clause boundaries is far from trivial. At the same time, for the level of semantics it would be very useful to have all the dependent objects and adverbials determined not only for finite but also for non-finite verbs (which often take arguments similarly to finite verbs). Example of a unidentified infinitival clause is given on the figure 6. Here, *kohe vabastada teletorni* is a infinitival subordinate clause, which should be separately grouped in the sentence tree and which is having an non-finite predicate *vabastada*.

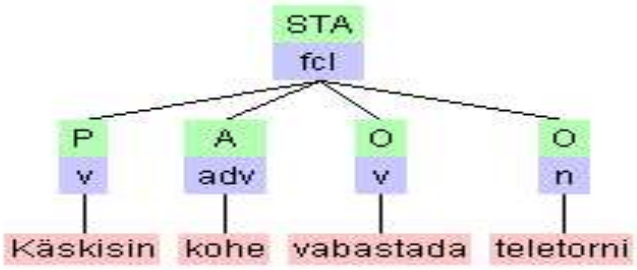

Figure 6: Tree for the sentence where non-finite subclause has been not identified. *Käskisin kohe vabastada teletorni*.
(*(I) gave an order to vacate the television tower immediately*.)

5.3. Noun phrases

It is quite difficult to guess the structure of a complex NP relying on the CG tags @NN> and @<NN, because we only know the direction, in which the head is situated but we don't know, which word exactly is the head (sometimes a word, tagged as @NN> can be a head for another word tagged @NN>, etc.

Sometimes the head can be determined relying on the morphological information. If an NP consists of a proper or common noun in genitive case + adjective + substantive, with the latter two agreeing in case, e.g. "Ida-Virumaa raskest olukorrast" the structure is:

```
A:np
=D:prop("Ida-Virumaa+0" prop sg gen %cap)     Ida-Virumaa
=H:np
==D:adj("raske+st" pos sg el)     raskest
==H:n("olu_kord+st" com sg el)                olukorrast
```

but not:

```
=A:np
==D:adjp
===D:prop("Ida-Virumaa+0" prop sg gen %cap)                Ida-Virumaa
===H:adj("raske+st" pos sg el)  raskest
==H:n("olu_kord+st" com sg el)                olukorrast
```

However, the present version of the open source VISL psg-compiler does not allow explicit reference to morphological features (even where they are known from CG input), unless cumbersome new 'word classes' are 'invented' for only this purpose (e.g. n-acc, n-gen, etc.). The necessary changes in the compiler formalism have been discussed in the VISL user community, but not yet implemented.

With CG-to-PSG rules we have gained quite good results in noun phrase extraction. We have compared the list of NP-s that were determined by CG-to-PSG rules against the correct list of noun phrases from a part of the corpus *Estonian best*. The number of NP-s in the correct NP list was 253. The rules gave 93,3 % for recall and 92,5 % of precision on noun phrase extraction.

The errors in NP extraction by CG-to-PSG rules were caused by false adverbial attachment analyzed in the section 4.3.1. (e.g. *Koos kaadrisse*, *truualamlikult viina*, *kolinal ämbrisse*). There was also a number of the errors in the NP-internal structure but this is actually not the matter of the NP extractor, thus these errors are not counted. Thus, as a side product, we have got quite a good noun phrase recognizer.

6. Comparison of (the expressive power of) CG and PSG

We can bring forth the following principal differences between CG and PSG (specifically, Arborest) which make it difficult to automatically convert the CG annotated corpus to PSG annotated corpus:

- CG: syntactic function and morphological form of each word determined
  Arborest: In addition, complex forms (phrases, subclauses, co-ordinated units) are established and their syntactic function annotated
- Attachment uncertainty
  CG: no explicit dependencies, directional dependency markers only for group-level modifiers, not clause level dependents (e.g. @AN> and @<NN looking for right and left noun-heads, but not @<ADVL or @ADVL> looking for left or right main verbs). Arborest, on the other hand, has to resolve all attachments, at least implicitly, in connection with its constituent bracketing.
- CG: finite clause boundaries are determined but not non-finite clause boundaries. PSG-rules can therefore address the former, but not the latter, and has here to rely on functional relations, uniqueness principle etc.
- Attachment of subclauses

CG: The hierarchy of subclauses is not expressed, and subclause function is not annotated. As implemented in the VISL family of CGs, such information could be added to head verbs or complementizer words. So far, however, we have used a partial solution, exploiting a list of subordinating conjunctions and pronouns typical of, for instance, adverbial, relative or averbal constructions.

7. Conclusions and Future Developments

The experiment to derive a hybrid form+function treebank from Estonian Constraint Grammar corpus has been quite successful. The semi-automatic procedure is usable for treebank creation, although in the present stage it is still time-consuming. The revision of the corpus *Estonian best* (149 trees) took one week of full-time linguist's work (including the learning of the category set and textual representation format of the trees). The manual correction job could be made significantly easier with a graphical interactive tree editing tool (like *Annotate* or a planned interactive version of VISL's tree visualiser). We believe that a particular strength of out method is that it, to a certain degree, processes function and structure separately, exploiting the robustness of syntactic-function tagging at the CG-level (and in this case, pre-existing manual revision), while adding structural information through a separate (PSG) grammar, allowing a more focused linguistic revision. It may be of interest to point out, that our approach differs from other hybrid methods not only by employing a Constraint Grammar base, but also with respect to the order of steps, inverting the maybe more traditional progression from chunking to parsing to function labelling.

The CG-to-PSG conversion rules have been most accurate on noun phrase detection and simple sentence analysis consisting of the usual sentence constituents subject, object, predicate, predicative complement and adverbials in any order. The composite sentences and subordinate clauses have also been well analyzed, using the condition that a subordinate clause begins with one of the subordinating conjunctions or interrogative-relative pronouns given in the lexicon.

We can see three possibilities to improve the CG-to-PSG treebank conversion results, best, if combined:

- revise CG-to-PSG rules taking into account the results of the current evaluation
- refine CG markup (subcategorize adverbials, add infinite and averbal clause boundaries)
- use more morphological (especially case) information in the PSG rules

During 2004–2008, it is planned to create a larger treebank using existing Estonian text corpora. Thus, we plan to turn the Estonian CG corpus (200.000 words) into a treebank using the CG-to-PSG grammar. A kernel of 1000 sentences will be hand-corrected at the gold-standard level and used for documentation and exemplification. Part of the remaining treebank will also be revised, but in a somewhat looser fashion (for instance, no cross-revision), relying on the fact that at least with regard to syntactic function, the corpus has already be revised at the CG-level.

The main research plans connected to the Estonian treebank include the examination of the predicate-argument structures in the corpus and to revise Rätsep's sentence templates (Rätsep, 1978 [11]) in the light of corpus data. That's why it is important to determine verbs' arguments both in finite and infinite subclauses. In perspective, the annotation will also be enriched by semantic information – adding semantic category information to the terminal nodes. It isintended to build a syntactic-semantic treebank of Estonian by integration of Arborest and Estonian Wordnet (Orav & Vider, 2000 [9]), containing 10 000 synsets. We are also planning to work on phrase level alignment of Estonian-Swedish-German parallel treebank to provide material for experiments on machine translation.

References

[1] E. Bick. *A CG & PSG Hybrid Approach to Automatic Corpus Annotation*, in Kiril Simow & Petya Osenova: Proceedings of SProLaC2003 (at Corpus Linguistics 2003, Lancaster), pp. 1-12

[2] T. Brants. *The Negra Export Format for Annotated Corpora*, Version 3. Techinal report. Department of Computational Linguistics, University of Saarland.

[3] M. Erelt, R. Kasik, H. Metslang, H. Rajandi, K. Ross, H. Saari, K. Tael, S. Vare *Eesti keele grammatika II. Süntaks*. (The Grammar of Estonian II: Syntax) Institute of Estonian Language. Tallinn 1993.

[4] T. Järvinen and P. Tapanainen.Towards an implementable dependency grammar**.** In *Proceedings of the Workshop "Processing of Dependency-Based Grammars"*, (eds.) Sylvain Kahane and Alain Polgučre, Université de Montréal, Quebec, Canada, 15th August 1998, pp. 1-10.

[5] F. Karlsson, A. Anttila, J. Heikkilä, A. Voutilainen. *Constraint Grammar: a Language-Independent System for Parsing Unrestricted Text*. Mouton de Gruyter, 1995.

[6] W. Lezius. TIGERSearch – Ein Suchwerkzeug für Baumbanken (in German). In: S. Busemann (editor): *Proceedings der 6. Konferenz zur Verarbeitung natürlicher Sprache (KONVENS 2002)*. Saarbrücken, 2002.

[7] K. Müürisep. *Computer Grammar of Estonian: Syntax*. Dissertationes Mathematicae Universitatis Tartuensis – 22. Tartu, 2000.

[8] K. Müürisep, T. Puolakainen, K. Muischnek, M. Koit, T. Roosmaa, H. Uibo A New Language for Constraint Grammar: Estonian.*International Conference "Recent Advances in Natural Language Processing". Proceedings*. Borovets, Bulgaria, 10-12 September 2003, pp. 304-310.

[9] H. Orav, K. Vider. Estonian WordNet. In: C*ongressus Nonus Internationalis Fenno-Ugristarum. 7.-13.8.2000 Tartu. Pars V*. Dissertationes sectionum: Linguistica II. pp. 490-497

[10]T. Puolakainen *Computer Grammar of Estonian: Morphological Disambiguation*. Dissertationes Mathematicae Universitatis Tartuensis – 27. Tartu, 2001.

[11]H. Rätsep. *Eesti keele lihtlausete tüübid.* (The templates of Estonian simple sentences) Tallinn, 1978.

[12]H. Uibo. Syntactically annotated corpora of Estonian. In *The First Baltic Conference "Human Language Technology – the Baltic Perspective*“, Riga, Latvia, April 21-22, 2004, pp. 45-48.