Estonian Dependency Treebank: from Constraint Grammar Tagset to Universal Dependencies

Kadri Muischnek, Kaili Müürisep, Tiina Puolakainen
University of Tartu
Estonia
Email: kadri.muischnek@ut.ee, kaili.muurisep@ut.ee, tiina.puolakainen@ut.ee

Abstract

This paper presents the first version of Estonian Universal Dependencies Treebank which has been semi-automatically acquired from Estonian Dependency Treebank and comprises ca 400,000 words (ca 30,000 sentences) representing the genres of fiction, newspapers and scientific writing. Article analyses the differences between two annotation schemes and the conversion procedure to Universal Dependencies format. The conversion has been conducted by manually created Constraint Grammar transfer rules. As the rules enable to consider unbounded context, include lexical information and both flat and tree structure features at the same time, the method has proved to be reliable and flexible enough to handle most of transformations.

The automatic conversion procedure achieved LAS 95.2%, UAS 96.3% and LA 98.4%. If punctuation marks were excluded from the calculations, we observed LAS 96.4%, UAS 97.7% and LA 98.2%.

Still the refinement of the guidelines and methodology is needed in order to re-annotate some syntactic phenomena, e.g. inter-clausal relations. Although automatic rules usually make quite a good guess even in obscure conditions, some relations should be checked and annotated manually after the main conversion.

Keywords: universal dependencies, dependency grammar, Estonian language

1. Introduction

Estonian Dependency Treebank (EDT) is a dependency-annotated treebank of written Estonian[1]. It comprises ca 400,000 words (ca 30,000 sentences); the texts represent the genres of fiction, newspapers and scientific writing (Muischnek et al., 2014b).

The treebank is annotated semi-manually for lemma, part of speech, morphological categories, syntactic functions and dependency relations.

The original morphological tagset[2] is a language-specific local standard, whereas the set of syntactic relations is based on Constraint Grammar (Karlsson, 1990; Karlsson et al., 1995) and coding of dependency relations is based on an expansion of Constraint Grammar (Bick & Didriksen, 2015).

Several large projects with the aim of creating treebanks in multiple languages have emerged during the the past few years. These projects try to introduce homogeneous or universal dependency annotations (McDonald et al., 2013) as such annotation enables building better syntax-based machine translation systems and other language technology applications.

The general aim of Universal Dependencies[3] (UD) project is to provide a cross-linguistically and typologically consistent inventory of categories and guidelines in order to facilitate consistent annotation of similar constructions across languages and thus promote cross-linguistically relevant methods and tools in natural language processing (e.g. Nivre, 2015).

There are currently available Universal Dependencies for more than 40 languages. The list of UD-related publications[4] contains articles about Finnish (Pyysalo et al., 2015) and Swedish (Nivre, 2014) Universal Dependencies.

It seems that the conversion of Finnish (Turku Dependency) Treebank and Swedish Treebank to Universal Dependencies was an easier or at least less fuzzy task than converting the Constraint Grammar based EDT annotation. The original Turku Dependency Treebank annotation was derived from Stanford Dependencies, which is also the predecessor of UD. Although the original Swedish Treebank uses a local standard (MAMBA, Teleman, 1974) for dependency annotation, the majority of grammatical constructions are annotated the same way according to both UD and MAMBA annotation schemes.

At the moment UD repository contains a smallish Estonian UD treebank that has been created as a part of automatic multiple treebank conversion effort (Rosa et al., 2014).

1 The EDT is freely available at https://github.com/EstSyntax/EDT

2 The documentation is available at http://www.cl.ut.ee/korpused/morfliides/seletus.php?lang=en

3 http://universaldependencies.org/

4 http://universaldependencies.org/introduction.html

Having just finished compiling the first version of Estonian Dependency Treebank (Muischnek et al., 2014b) the obvious next step would be converting it to UD format. In this paper, we report work in progress which aims at converting the Constraint Grammar style treebank annotation to Universal Dependencies' annotation.

Automatically converted Estonian UD treebank described in this article is freely available at the repository: https://github.com/EstSyntax/EstUD.

The rest of this paper is organized as follows. Section 2 gives a brief description of Constraint Grammar style syntactic annotation of the Estonian Dependency Treebank and Section 3 compares the latter with the Universal Dependencies annotation scheme. Section 4 gives a step-by-step description of the conversion process and estimates the quality of the conversion. Section 5 discusses the future tasks and Section 6 concludes.

2. CG Syntactic Annotation

The annotation is word-based or, more precisely, every token delimited by white-spaces is annotated as an autonomous word.

The Constraint Grammar style annotation of EDT has three layers: morphological, surface-syntactic and dependency layers. The morphological annotation layer contains information about lemma, part of speech and grammatical categories (e.g. case and number for nominals; mood, tense, person and number for verbs) for every word-form in text.

Surface-syntactic layer contains the labels for syntactic relations. According to our annotation scheme, the members of the verbal chain can have labels `FMV` (finite main verb), `IMV` (infinite main verb), `FCV` (finite chain verb), `ICV` (infinite chain verb). Particles as parts of a particle verb are tagged `Vpart`, and if the particle verb is a nominalization, then the particle has a tag `VpartN`. The verb negator is labelled as `NEG`. The arguments of the verb are labelled as subject `SUBJ`, object `OBJ`, predicative `PRD` or adverbial `ADVL`; the adjuncts also get the adverbial `ADVL` label.

The attributes of a nominal are tagged according to their word-class: `AN` stands for adjectival attribute, `NN` for nominal attribute and apposition, `DN` for adverb attribute, `INFN` for infinitival attribute and `KN` for an adpositional phrase as an attribute (label is attached to the adposition as it is considered to be the governor of the adpositional phrase, the noun governed by an adposition receives a label `P`). A word-form governed by a quantor is labelled as `Q`. The premodifying and postmodifying labels have been distinguished by adding arrow symbols to them (`AN>` is premodifying adjectival attribute, `<NN` is postmodifying nominal attribute). `J` stands for conjunctions and `I` for interjections.

Dependency layer gives information about the governor of every word-form in text; this layer is shallow, meaning that no virtual nodes are postulated.

```
"<s>"
"<öö>"
    "öö" L0 S com sg gen @P> #1->2
"<jooksul>"
    "jooksul" L0 K post @ADVL #2->8
"<olid>"
    "ole" Lid V aux indic impf ps3 pl af @FCV #3->8
"<hundid>"
    "hunt" Ld S com pl nom @SUBJ #4->8
"<kolm>"
    "kolm" L0 N card sg nom l @OBJ #5->8
"<lammast>"
    "lammas" Lt S com sg part @<Q #6->5
"<maha>"
    "maha" L0 D @Vpart #7->8
"<murdnud>"
    "murd" Lnud V main partic past ps @IMV #8->0
"<.>"
    "." L0 Z Fst CLB #9->9
"</s>"
```

Figure 1: CG analysis of sentence (1).

(1) Öö jooksul olid hundid kolm lammast
night during be-AUX wolf-PL three sheep-PART
maha murdnud
down kill-PCP
'The wolves had killed three sheep during the night.'

The sentence (1) starts with an postpositional phrase *öö jooksul* 'during the night'. The verbal chain olid *maha murdnud* 'had killed' is split so the auxiliary *olid* 'had' occupies the second position in the clause and the rest of the verbal chain is situated at the end of the clause after the object *kolm lammast* 'three sheep', a typical word order of multiword predicates in Estonian. Main verb of the clause is a particle verb *maha murdma* 'kill down'; the particle *maha* 'down' functioning as a perfective marker.

Figure 1 illustrates the CG annotation of example sentence (1) and Figure 2 depicts the same annotation in a graphical view.

As depicted on Figure 1, the word forms are in separate rows followed their morphological and syntactic description. The morphologival description consists of the lemma, ending, POS, morphological information, and valency information. The syntactic description consists of a syntactic label (starting with @) and dependency information (starting with #).

The first word form *öö* ('night') is substantive (S), common noun (com), singular (sg), genitive (gen). It belongs to postpositional phrase (@P>) and depends on the word form in the position 2 (#1->2). The second word form *jooksul* ('during') is adposition (K), postposition (post), it is functioning as an adverbial (@ADVL) and it depends on the word form in the position 8 (#2->8). Figure 2 depicts visualization of CG annotation. Analyses visualized using BRAT (Stenetorp et al., 2012).

As said above, the used set of syntactic relations derives from Constraint Grammar, but the definitions of syntactic relations (i.e. what word-forms under which conditions are analysed as a subject or an adverbial, for example) are based on an academic description of Estonian grammar (Erelt et al., 1993). As it is often the case, this descriptive grammar is rooted in the local grammatical tradition established over long time.

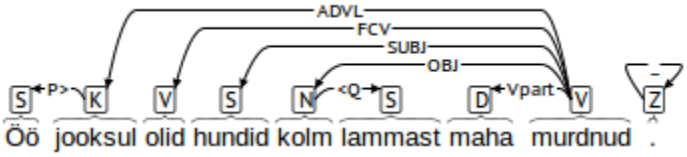

Figure 2: Graphical view of EDT annotation for sentence (1).

In general, our annotation scheme is quite fine-grained for annotating intra-clausal phenomena, but although we annotate the dependency relations that hold between the clauses, our scheme does not distinguish the names of those relations, the annotation only shows that there is a dependency relation between the clauses. That should be considered one of the main shortcomings of EDT annotation scheme.

While creating EDT, the texts are first parsed with rule-based Estonian Dependency Constraint Grammar parser (Muischnek et al., 2014a), then checked and disambiguated by two parallel independent human annotators, parallel annotations compared and discrepancies solved by a so-called super-annotator.

3. Main differences between EDT and UD annotation schemas

Although both EDT and UD syntactic annotations are based on dependency grammar, they employ different sets of syntactic relations and analyse or annotate several linguistic phenomena (e.g. coordination, verbal chain) differently. So, the following subsections contain a brief comparison of EDT and UD annotation schemes.

Figures 1 and 3 present EDT and UD annotations of a sentence Öö jooksul olid hundid kolm lammast maha murdnud ‘The wolves had killed three sheep during the night’, Figures 2 and 4 present visualizations of EDT and UD syntactic annotations respectively.

3.1. POS Tags and Syntactic Relation Labels

As for POS tags, we don't use the determiner (`DET`) tag as Estonian has no true articles. The same decision, at least in the current version of UD, has been made for Finnish, a close relative of Estonian. Also we don't make use of the part of speech tag `PART` for particles (defined as “function words that must be associated with another word or phrase to impart meaning and that do not satisfy definitions of other universal parts of speech”). The reason for that is practical: the word-forms that should be tagged as particles according to the UD guidelines, are currently tagged as adverbs or pronouns and it needs a special effort to re-tag them.

We do not go into details of annotating morphological features here.

As for syntactic relation labels, the current Estonian UD scheme does not use the relations of determiner (`det`), indirect object (`iobj`), expletive (`expl`), passive nominal subject (`nsubjpass`) and clausal passive subject (`csubjpass`), also relations `mwe`, `goeswith`, `dislocated`, and `reparandum`.

We do not annotate recipients of ditransitive verbs or benefactives as indirect objects, in fact no grammatical description of Estonian uses the notion “ditransitive verb” or “indirect object”; the morphological case used for coding recipients and benefactives is allative - one of the locative cases with the basic or prototypical meaning “onto”.

Estonian is a null-subject language and thus doesn't use any expletives.

Similarly to English, German and Swedish UDs, we use the label `compound:prt` for particle components of particle verbs, which are common phenomenon in Estonian.

The EDT annotation scheme differentiates between finite and non-finite clauses. The head of a non-finite clause is labelled according to its syntactic relation, but for the main verb of a finite clause only its direct governor is indicated and not the syntactic relation the finite clause has in respect to the main clause. So annotating the governors of finite dependent clauses with the right labels is a task of annotation, not just task of conversion.

According to the EDT annotation scheme, quite a large range of modal verb constructions and phase verb constructions were annotated as multi-word predicates, whereas the UD annotation scheme recognizes only a small set of auxiliary verbs as eligible for being annotated as a part of the verbal predicate; the rest of the verb + verb constructions should be annotated so that the second verb is labelled as `xcomp` or `ccomp`.

1	Öö	öö	NOUN	S	Case=Gen|Number=Sing	8	nmod	_	_
2	jooksul	jooksul	ADP	K	_	1	case	_	_
3	olid	olema	AUX	V	InfForm=Fin|Mood=Ind|Number=Plur|Person=3|Tense=Past|VerbForm=Fin	8	aux	_	_
4	hundid	hunt	NOUN	S	Case=Nom|Number=Plur	8	nsubj	_	_
5	kolm	kolm	NUM	N	Case=Nom|Number=Sing|NumForm=Letter|NumType=Card	6	nummod	_	_
6	lammast	lammas	NOUN	S	Case=Par|Number=Sing	5	dobj	_	_
7	maha	maha	ADV	D	_	8	compound:prt	_	_
8	murdnud	murdma	VERB	V	Tense=Past|VerbForm=Part|Voice=Act	0	root	_	_
9	.	.	PUNCT	Z	_	8	punct	_	_

Figure 3: UD analysis of the sentence (1).

The UD syntactic labels contain a separate set of labels for various multi-word units and unanalyzable tokens (labels `compound`, `mwe`, `goeswith`, `name` and `foreign`). None of them is present in EDT annotation scheme. A subtype of compounds, particles as part of particle verbs are annotated using the label `Vpart` (cf the word-form *maha* in the example). Most of the phenomena that are annotated as parts of compounds, multi-word expressions or names, were annotated as some kind of attributes in EDT.

The UD annotation scheme also contains a special set of syntactic relation labels for loose joining relations (labels `list`, `parataxis`, `remnant`, `dislocated` and `reparandum`), which are not distinguished in the EDT annotation scheme and therefore the annotation of these complex constructions cannot be converted fully automatically, still automatics may offer a good guess.

3.2. Primacy of Content Words in UD

The primacy of content words means that the syntactic dependencies hold primarily between content words and are not mediated by function words; the function words depend on content words and not vice versa. In EDT annotation, function words were allowed to have dependents of their own, permitting e.g. chains of auxiliary verbs.

For example, according to UD schema an adposition phrase should be annotated so that the adposition depends on the noun. Although Estonian has a rich system of morphological cases, both pre- and postposition phrases are also used. In EDT annotation scheme the adpositional phrases were annotated so that the noun depends on the adposition and the justification for that was that the case form of the noun was assigned by the adposition, e.g. *öö jooksul* 'during the night' in Figure 2. Another example of structures that need re-annotating because of the principle of primacy of content words are the numeral and quantifier phrases which were annotated as headed by numerals or quantors in EDT, e.g. *kolm lammast* 'three sheep' in Figure 2. Again, the justification for that was that the case form of the noun in a quantor phrase is assigned by the numeral or other quantifier. The corresponding dependency tree in UD representation is given in Figure 4.

The primacy of content words in UD also means that the lexical verb is the head of the verbal chain, not the auxiliary. The principle that function words do not take dependents also means that multiple function words related to the same content word appear as siblings. So, in case of multiple auxiliaries (e.g. eng *could have done*) both *could* and *have* are attached to *done*. Copular verbs are also counted as auxiliaries in this respect and in copular constructions the UD annotation scheme instructs to attach auxiliaries to predicates that are not verbs.

The verbal chain is annotated differently in EDT: verbal chains (like *could have done*) were annotated in a chain-like manner and in copular constructions copular verb was treated as the head. The compound tense form of a particle verb *olid maha murdnud* 'had killed' in Figure 2 illustrates annotation of dependency relations of multi-word predicates.

3.3. Coordination

The notion of dependency is not quite feasible for describing coordination. UD scheme treats coordinate structures asymmetrically, so the head of the relation is the first conjunct and all the other conjuncts and coordinating conjunctions depend on it. In EDT annotation scheme, we have annotated each following coordinated element as a dependant of the previous one, and the coordinating conjunction as the dependent of the coordinated element following the conjunction.

4. Conversion to UD Schema

Morphological annotation could be converted in a rather straightforward way, but the conversion from EDT dependency annotation to that of UD often requires not only relabeling of types, but also changes to the tree structure.

4.1. Conversion Procedure

The conversion process is preceded by some technical preprocessing steps like splitting multi-word proper names (e.g *New York*) and marking clause boundaries.

The conversion process itself consists of the following steps:

1. The main procedure of rearranging subtrees and finding correspondences between EDT and UD relations.
2. Conversion from CG3 (used in EDT) to CONLL-U format, including already established correspondences between EDT and UD syntactic relation labels and converting POS and morphological features with simple table lookup.
3. Formal checks for existent and only one root in the sentence, missing cycles and presence and validity of all obligatory field values.

The main step of conversion procedure (step 1 above) is realised as a VISL-CG3 (Bick & Didriksen, 2015) script, which is a native manipulation tool for EDT corpus in CG3 syntactically annotated format. It conveniently enables to use long-distance conditions and accounts both

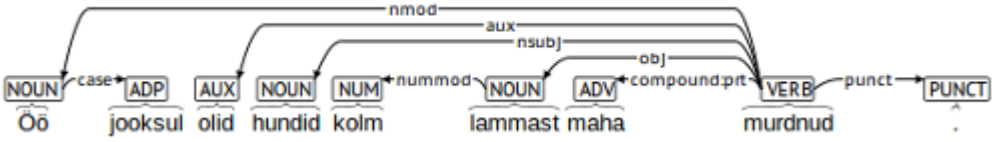

Figure 4: Graphical view of UD representation.

for flat and tree structure features in the conditions of rearranging and conversion rules. Based on the context, rules find the respective UD relation names and change the dependencies of particular constructions, namely:

1) Invert the dependencies in adposition and quantor phrases: noun becomes the head and adpositions and quantors its dependents as modifiers. Previous dependents of adposition or quantor are re-attached to the noun, but with an exception: if numeral quantor is modified by some approximative adverb as *peaaegu* 'nearly', *umbes* 'circa', *ligikaudu* 'approximately' in EDT, then these adverbs should not be re-attached. An example of conversion of a phrase *Alates 1980. a keskelt oli…* 'From the middle of the year 1980 was...' is depicted in Figure 5 and Figure 6 The phrase contains two adpositions: a preposition *alates* 'from the beginning' and its dependant postposition *keskelt* 'from the middle', which applies before preposition, in this case projective annotation in CG3 format become non-projective in UD.

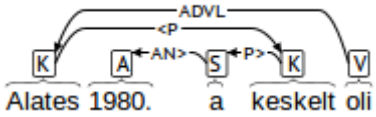

Figure 5: CG analysis of adpositional phrase.

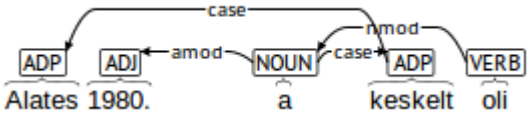

Figure 6: UD analysis of adpositional phrase.

In the UD annotation of the phrase also the direction of the relation between prepositions is reversed showing now that first the *keskelt* 'from the middle' should be applied to the a 'year' and after that the *alates* 'from the beginning', not vice versa, so the function word here is modified by another function word. So we cannot add the adpositions to the UD class of pure function words which cannot take the modifiers other than negation as otherwise logical structure of the phrase would be lost.

2) Rise a predicative to the head of clause with corresponding rearrangement of other dependencies in the clause. According to UD copula subject is attached directly to the predicative with the copular verb becoming a dependent of the predicative; while in EDT annotation copula is the head and predicative its dependent. Also elliptical sentences with missing predicate or subject are considered here. This "predicative raising" type of change also influences the inter-clausal connections as other clauses should be attached to the predicative instead of earlier head of the clause.

3) Invert dependencies in verbal chain as, differently from EDT, the finite main verb becomes the head and infinitives that are its dependants get labels of `csubj`, `csubj:cop`, `ccomp`, `xcomp` or `advcl`. One should bear in mind that according to EDT annotation scheme quite a large number of phase verb (verbs of starting/beginning and finishing) constructions (e.g. eng *start digging*) were treated as multiword predicates, but the UD annotation scheme regards only a small class of modal verb constructions as multi-word predicates. So a considerable part of EDT multi-word predicates has to be re-labeled.

The main rules for assigning new relation names for members of the verbal chain are quite straightforward: infinitival subjects get `csubj` label; infinitival subjects in copula clauses get `csubj:cop` label; infinitival predicatives get `ccomp` label; infinitival objects, infinitival components of multi-word predicates and infinitival adverbial complements get `xcomp` label, and finally the remaining infinitival adverbials get `advcl` label. In addition, infinitives and participles modifying a noun get the label `acl`.

4) Make subsequent conjuncts and conjunctions dependents of the first conjunct. If first conjunct is not present in the sentence (in case the sentence begins with a conjunction), then second conjunct takes its place. Find and label all occurrences of preconjunctions. This step needs to take into account structural changes in coordinated phrases which can result in a quite different tree structure.

5) Invert the dependencies between clauses in sentences with direct speech. In EDT the reporting verb is always considered to be the root, but according to the UD annotation scheme the predicate (or other head) of the first main clause should be the root of the whole sentence.

6) Attach correct relation labels and dependency information to text in parentheses, which was not annotated in EDT.

7) Find correct relation names and dependencies for punctuation, which remained unattached in EDT. This appears to be a non-trivial task, especially fulfilling two conditions: that paired punctuation marks should have the same governor and that crossing arcs should be avoided. The rule set for annotating punctuation in UD is still under development.

8) Suggest the inter-clausal relation names, which are not present in EDT as the EDT annotation scheme does not recognize different inter-clausal relations. The relation labels for heads of the clauses in UD represent the role of the whole clause in respect to the head word in the main clause which is different from EDT approach where labels indicate the clause internal surface-syntactic functions. This difference influences and complicates the most the annotation of predicates (e.g verbal chain) and choosing the right inter-clausal relation labels differentiating between clausal complements and modifiers. This step is not planned to be accomplished fully automatically and therefore these labels should also undergo manual post-checking.

The possible labels for inter-clausal relations are `csubj`, `csubj:cop` and `ccomp` for complement clauses, `acl`, `acl:relcl` and `advcl` for modifier clauses and finally `conj` and `parataxis` for connecting coordinated and

other loosely related non-subordinated clauses. In general, it is not feasible to make a confident decision on a clause being complement or modifier automatically, but it can be suggested by some clues in particular sentences.

`csubj` label and correspondent `csubj:cop` label in copula clauses are suggested to certain types of subordinated clauses with a main clause with missing subject. Analogously `ccomp` label are suggested for subordinated clauses with a main clause with a predicate from a certain subset of (reporting) verbs and missing object. The most reliable mapping is the assignment of the `acl:relcl` relation to relative clauses which modify a nominal, since both the nominal head and relative correlate can be easily validated. On the other hand, the assignment of more general `acl` relation is not always correct.

Elliptical constructions were treated in quite a robust way in EDT, implementing mainly two following strategies. As first possibility, one component of the elliptical construction was annotated as the governor of the construction and the other word-forms were annotated as its dependents (2). The other possibility was to annotate elliptical constructions using coordination relation (3). The elliptical constructions were impossible to convert to UD automatically and they are waiting for manual post-editing.

(2) Lehvitas talle, nägu naerul.
Waved s/hel face smiling.
'S/he waved to him/her with a smiling face.'

(3) Mari õppis keeleteadust ja Jüri füüsikat.
Mari studied linguistics and Jüri physics.
'Mari studied linguistics and Jüri physics.'

The Table 1 shows possible resulting UD relation labels after conversion based on EDT syntactic functions. The table does not indicate that everything except FCV, ICV, NEG, INFN, DN and KN may be mapped also to root and everything except FCV, ICV, NEG, KN, J and B may be additionally mapped to `conj` or `parataxis`.

EDT label	Possible UD labels
SUBJ	nsubj, nsubj:cop, csubj, nummod, csubj:cop, advmod:quant, nmod, compound, advcl, case, cop, acl:relcl, ccomp
OBJ	dobj, xcomp, nummod, advmod:quant, compound, advcl, case
PRD	advcl, ccomp, acl:relcl, nummod, csubj, acl, compound, advmod:quant, case, cop
ADVL	nmod, advmod, case, xcomp, nummod, advcl, mark, amod, list, cc:preconj, advmod:quant, discourse, compound, cop, cc
FMV	(root), cop, advcl, acl:relcl, ccomp, acl, csubj
IMV	(root), advcl, xcomp, acl:relcl, cop, ccomp, acl, csubj
FCV	aux, auxpass
ICV	aux
NEG	neg
INFN	acl, cop
NN	nmod, det, name, amod, appos, nummod, compound, advmod:quant, advcl, cc:preconj, discourse, case
AN	amod, acl, advcl, nummod, name
KN	case
DN	advmod, cc:preconj, mark, acl
P	nmod, nummod, amod, compound, advmod:quant, advcl, xcomp, cc:preconj
Q	nmod, nsubj, dobj, nummod, advmod:quant, advcl, nsubj:cop, compound, advmod, amod, xcomp, acl:relcl, appos
J	cc, mark, cc:preconj
B	discourse
NONE	list, foreign, nummod, nmod

Table 1: Dependency relation type mapping accordingly to EDT surface-syntactic function. The Constraint Grammar labels are explained in Section 2.

4.2. Quality Estimation of the Conversion

The evaluation experiments were run on a small manually annotated corpus, consisting of 3,428 tokens (4/5 newspaper texts, 1/5 fiction). We used MaltEval tool for evaluation metrics (Nilsson & Nivre, 2008). The preliminary results yielded the labeled attachment score (LAS, both labels and relations are correct) of approximately 95.2%. Also, we observed the unlabeled attachment score (UAS) of 96.3% and the label accuracy (LA) of 98.4%. The main source of errors was the annotation of punctuation marks.

EDT annotation scheme did not contain detailed rules for attaching punctuation, and it remained unattached. UD annotation scheme provides quite detailed instructions for labelling punctuation, but re-attaching it is still a difficult task. On the other hand, Estonian punctuation rules are rather strict and interpunctuation marks are used extensively. Therefore, we decided to focus on syntactic annotation of words and deal with the analysis of punctuation marks later.

The accuracy scores for the same corpus without punctuation marks are the following: LAS 96.4%, UAS 97.7% and LA 98.2%.

The detection of dependency relations of subclauses is a challenging task as there is no relevant prior information in the EDT annotation. In other words, it is not a conversion task but rather an annotation task.

Although we have somewhat simplified the division of clausal complements into the classes of `ccomp` (clausal complement) and `xcomp` (open clausal complement) for the time being, `ccomp` remains the most error-prone dependency relation. According to the UD guidelines, clausal complement of a verb is a dependent clause which is a core argument. That is, it functions like an object of the verb. A typical clausal complement is an Estonian version of the that-clause containing the reported speech. But as the conjunct *et* 'that' can also be used in the beginning of other clause types there are no formal clues to make a clear-cut distinction between *et*-clauses that should get the `ccomp` label and other types of *et*-clauses. E.g. a multiword construction *vaatamata sellele* 'despite' is always followed by a that-clause, that nevertheless should not get the `ccomp` label.

The performance measure for annotating `ccomp` was the lowest. Its treebank accuracy[5] was 64.3%. Some of these errors can be fixed by adding more precise conversion rules.

Apart from aforementioned labels, another label that involves a complex conversion procedure is `cc` - coordinating conjunction.

In EDT, the coordinating conjunction is the dependent of the coordinated element following the conjunction, but according to the UD annotation scheme first conjunct is the head of whole coordinated construction, all other conjuncts and conjunctions are its direct dependents. In our benchmark corpus, every fifth `cc` was assigned an incorrect head, but this issue can also be addressed by improving conversion rules.

4.3. UD and Projectivity

It seems that the UD principle of the primacy of content words helps to avoid non-projectivity, at least in some contexts. For example, in the case a syntactic structure including a postposition is situated at a clause boundary and is followed by a relative clause modifying the noun that is governed by the postposition according to EDT annotation scheme or that, according to UD annotation scheme, itself governs this postposition. So EDT-style analysis of a sentence fragment (5) is depicted on Figure 7 and its UD-style variant on Figure 8.

(5) Kirik kandis hoolt trükikoja eest,
church take care print-house-GEN of
mis asus vanas koolimajas.
which locate-PST.3SG old-INE schoolhouse-INE
'Church took care of the printing house,
which was located in the old schoolhouse.'

5 Percentage of ccomp tags in the gold standard correctly annotated after the conversion.

In the EDT version, the word-form *trükikoja* 'printing house-gen' is governed by the postposition *eest* 'of', which in turn is governed by the main verb of the clause, which appears to be a particle verb *kandis hoolt* 'took care'. On the other hand, the word-form *trükikoja* 'printing house-gen' itself is modified by the main verb *asus* 'was located' of the relative clause, resulting in crossing arcs. In the UD version the crossing arcs are avoided as the postposition is governed by the noun. Figures 7 and 8 illustrate the difference.

5. Future work

There were a few remarks about future tasks in the previous sections - that inter-clausal relations, clausal complements, elliptical constructions and coordinating conjunctions need some more effort. Also it would be beneficial to annotate parts of multi-word expressions with `mwe` label and extend the set of constructions that are annotated using the name label.

Talking about far-reaching plans and tasks, it would be useful to add dependency relation `nmod:own` into Estonian UD annotation scheme. This relation was introduced in the Finnish version of UD and it is needed in a special clause type called possessive clause in Estonian (and in Finnish). It is syntactically structured so that the thing possessed or the stimulus cognized is coded in nominative case and is thus the subject of the clause while possessor or cognizer is coded in allative case and has been annotated as an adverbial in EDT. So, in the current version of automatic conversion, it is simply re-labelled as nominal modifier (`nmod`) and the important semantic information about cognizers and possessors as actually the most "agent-like" participants in those clauses gets lost. As for the annotation procedure of owners and cognizers, apparently we have no better solution than to annotate clauses following the possible possessive pattern manually as the frequent locative clauses follow same syntactic pattern; so literally *Child has a book* in Estonian would be *At child is a book*.

In the case we have to do some manual re-annotation, it would also make sense to annotate the argument in allative case in recipient or benefactive constructions (i.e. recipient or beneficient) as a special subtype of nominal modifier `nmod:rec`.

It would be also very important to annotate reflexive verbs to make this aspect comparable with information in other languages.

6. Conclusion

This article presented a work in progress: converting the Estonian Dependency Treebank (EDT) to Universal Dependencies format. Primarily Constraint Grammar (CG) rules were used to map EDT annotation labels onto UD labels. As the CG rules enable to consider unbounded context, include lexical information and both flat and tree structure features at the same time, the method has proved to be reliable and flexible enough to handle most of the transformations.

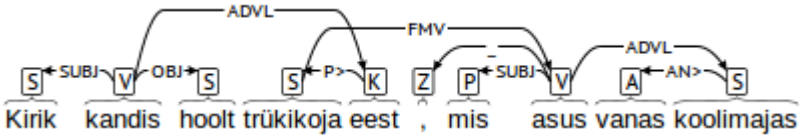

Figure 7: Non-projective tree of sentence (5) in EDT format.

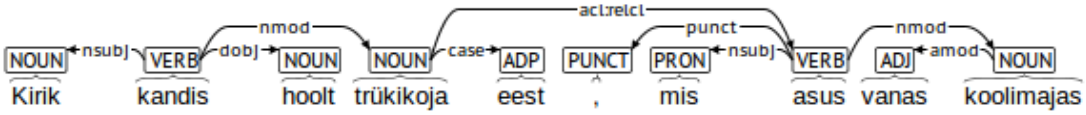

Figure 8: Projective UD tree of sentence (5).

The mapping system achieved LAS 95.2%, UAS 96.3% and LA 98.4%. If punctuation marks were excluded from the calculations, we observed LAS 96.4%, UAS 97.7% and LA 98.2%.

Although the overall quality is not bad, the methodology for annotating some syntactic phenomena, for example, inter-clausal connections, should be reviewed and reconsidered. As some distinctions, especially while labelling clausal complements and modifiers, were not present in the original EDT annotation and are largely based on semantic knowledge, they are very difficult to make automatically, so some manual post-editing of the UD treebank will be necessary.

7. Acknowledgements

This work has been supported by Estonian Ministry of Education and Research (grant IUT 20-56 “Eesti keele arvutimudelid (Computational Models for Estonian)” and Norwegian-Estonian Research Cooperation Programme (grant EMP160 “SAMEST - Sami-Estonian language technology cooperation - similar languages, same technologies”).

8. Bibliographical References

Bick, E. and Didriksen, T. (2015). CG3 Beyond Classical Constraint Grammar. In Proceedings of the 20th Nordic Conference of Computational Linguistics NODALIDA 2015, pages 31–40.

Erelt, M., Kasik, R., Metslang, H., Rajandi, H., Ross, K., Saari, H., Tael, K., and Vare, S. (1993). Eesti keele grammatika II. Süntaks. Eesti TA Keele ja Kirjanduse instituut.

Karlsson, F. (1990). Constraint Grammar as a Framework for Parsing Running Text. Proceedings of Coling-90. Vol. 3, pages 168-173.

Karlsson, F., Voutilainen, A., Heikkilä, J., and Anttila, A. (1995). Constraint Grammar: A Language- Independent System for Parsing Unrestricted Text. Mouton de Gruyter, Berlin.

McDonald, R. T., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K. B., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Castelló, N. B., and Lee, J. (2013). Universal Dependency Annotation for Multilingual Parsing. In ACL (2), pages 92–97. The Association for Computer Linguistics.

Muischnek, K., Müürisep, K., and Puolakainen, T. (2014a). Dependency Parsing of Estonian: Statistical and Rule-based Approaches. In Andrius Utka, et al., editors, Baltic HLT, volume 268 of Frontiers in Artificial Intelligence and Applications, pages 111–118. IOS Press.

Muischnek, K., Müürisep, K., Puolakainen, T., Aedmaa, E., Kirt, R., and Särg, D. (2014b). Estonian Dependency Treebank and its annotation scheme. In Verena Henrich et al., editors, Proceedings of the Thirteenth International Workshop on Treebanks and Linguistic Theories (TLT13), pages 285–291. University of Tübingen.

Nilsson, J., & Nivre, J. (2008). MaltEval: an Evaluation and Visualization Tool for Dependency Parsing. In Proceedings of LREC 2008.

Nivre, J. (2014). Universal Dependencies for Swedish. In Proceedings of the Fifth Swedish Language Technology Conference (SLTC 2014).

Nivre, J. (2015). Towards a universal grammar for natural language processing. In Alexander Gelbukh, editor, Computational Linguistics and Intelligent Text Processing, pages 3-16. Springer.

Pyysalo, S., Kanerva, J., Missilä, A., Laippala, V., and Ginter, F. (2015). Universal Dependencies for Finnish. In Proceedings of Nodalida 2015.

Rosa, R., Masek, J., Mareček, D., Popel, M., Zeman, D., Žabokrtský, Ž. (2014) HamleDT 2.0: Thirty Dependency Treebanks Stanfordized. In Proceedings of LREC 2014.

Stenetorp, P., Pyysalo, S., Topic, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: a web-based tool for nlp-assisted text annotation. In Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp 102–107.

Teleman, U. (1974). Manual for grammatisk beskrivning av talad och skriven svenska. Lund: Studentlitteratur.