

Arbobanko - A Treebank for Esperanto

Eckhard Bick

Institute of Language and Communication, University of Southern Denmark
Campusvej 55, DK-5230 Odense M, Denmark
`eckhard.bick@mail.dk`

Abstract. In this paper we describe and evaluate Arbobanko, a syntactic treebank for the artificial language Esperanto, as well as methods and tools used to produce the treebank. For an under-resourced language, the quality of automatic syntactic pre-annotation is of obvious importance, and by evaluating the parser associated with the treebank, we try to answer the question whether the language's extremely regular morphology and low lexical ambiguity carry over into a more regular syntax and higher parsing accuracy. On the linguistic side, the treebank allows us to address and quantify the typological issue of (free) word order in Esperanto.

Keywords: Treebanks, Esperanto, Dependency Grammar, Constraint Grammar, Syntactic parsing, Free word order languages

1 Introduction

Syntactic treebanks satisfy important needs in both language technology and descriptive linguistics, allowing the latter to identify and quantify linguistic patterns, and the former to train and evaluate machine-learned parsers. With a general change of focus from the latter to the former, dependency treebanks have become more prevalent at the expense of constituent treebanks, driven by methodological considerations such as implementability in mathematical models (graphs).

Historically, dependency syntax has roots in the description of slavic languages, one of its strengths being the handling of free word order and discontinuities, while constituent grammar was linked to English with its fixed word order and reliable subject-predicate pairs. Thus, the first and largest dependency treebank was built for Czech (Böhmová et al. 2003, Bejček et al. 2013), while major English treebanks like the Penn Treebank were originally annotated with phrase structure and converted to the dependency format only later (Johansson & Nugues 2007), by the machine-learning (ML) community. A third approach was used for the Danish Arboretum treebank (Bick 2003), where a rule-based Constraint Grammar (CG) parser was used

to create shallow dependency trees that were then converted to constituent trees, using manual revision at both stages.

As an artificial language with a sizable living speaker community, Esperanto is a linguistically interesting language, albeit under-resourced in terms of both development/research funding and existing NLP resources. Our treebank project intends to address this issue at both the linguistic and NLP levels. We decided on a dependency format not only because of the current focus of the research community, but also because of the purported free word order-characteristics of the language, and because the only available parser was a CG dependency parser, and we needed to minimize (unfunded) human revision labor.

2 The corpus

Arbobanko is a news corpus, covering the period 1997-2003. It is based on journalistic material from the Esperanto journal *Monato*, published by Flandra Esperanto Ligo, and compiled and TEI-encoded by Bertil Wennergren. The overall text corpus contains ca. 579,000 words, and is available for search and download at <http://tekstaro.com>. For the *Arbobanko* treebank a 50.000 word section of the corpus was tokenized and morphosyntactically annotated with the *EspGram* parser (Bick 2007 and 2009) and manually revised at all levels. Like the source corpus, this annotated subcorpus will be made available on-line.

Annotation was carried out with what could be called a recursive boot-strapping method, where corrections learned from manual revision were fed back into the parser in the form of rule changes or additions, that would then increase the accuracy of further automatic parses. By logging all manual corrections, it was also possible to establish an estimate of global and category-specific parser performance. Ultimately, knowledge of category-specific error margins should allow the use of a much larger treebank with only automatic annotation, that would still allow linguistic research with a reasonable level of reliability.

3 Annotation levels

The treebank contains linguistic annotation at four primary levels: lemma, part-of-speech (POS) and inflexion, syntactic function ("edge labels") and dependency-head id's (attachment links). In addition, there is some secondary, lexical information about morpheme structure and POS-subclass, as well as some semantic class information,

mostly for nouns. All information is strictly token-based and contained in the following ordered tag fields, with '@' used as a marker for the syntactic label, and '#' for a numbered dependency relation:

Wordform lemma <subclass> ... POS inflexion @syntactic_function #head_id

Apart from the linguistic annotation, most of the original TEI meta-information, such as topic, titles and paragraph id's, is retained in the treebank on separate xml lines. In the example below, token lines were indented according to tree depth to increase readability. Apart from the native format, we also provide Tiger xml and the CoNLL tab field format with feature-attribute pairs.

Post	[post] <*> PRP @ADVL> #1->14	<i>After</i>
12	[12] <card> <cif> NUM P @>N #2->3	<i>12</i>
jaroj	[jaro] <dur> <per> N P NOM @P< #3->1	<i>years</i>
da	[da] PRP @N< #4->3	<i>of</i>
reformoj	[reformo] <PREF:re%form o> <sem-c> <act> N P NOM @P< #5->4	<i>reform</i>
la	[la] ART @>N #6->7	<i>the</i>
efikeco	[efikeco] <N:efik%ec o> <f> N S NOM @SUBJ> #7->14	<i>efficiency</i>
de	[de] PRP @N< #8->7	<i>of</i>
la	[la] ART @>N #9->11	<i>the</i>
ĉehxa	[ĉehxa] <jnat> <Du> ADJ S NOM @>N #10->11	<i>czech</i>
ekonomio	[ekonomio] <domain> N S NOM @P< #11->8	<i>economy</i>
ne	[ne] <amod> <setop> ADV @>A #12->13	<i>not</i>
signife	[signife] ADV @ADVL> #13->14	<i>significantly</i>
transpaŝas	[transpaŝi] <PRP:trans+paŝ i> <mv> V PR VFIN @FS-STA #14->0	<i>surpasses</i>
la	[la] ART @>N #15->16	<i>the</i>
nivelon	[nivelo] <ac> N S ACC @<ACC #16->14	<i>level</i>
atingitan	[atingi] <mv> V PCP PAS IMPF ADJ S ACC @ICL-N< #17->16	<i>achieved</i>
en	[en] PRP @<ADVL #18->17	<i>in</i>
la	[la] ART @>N #19->20	<i>the</i>
jaro	[jaro] <dur> <per> N S NOM @P< #20->18	<i>year</i>
1989	[1989] <year> <card> <cif> NUM S @N< #21->20	<i>1989</i>
\$.	[.] PU @PU #22->14	<i>.</i>

[N=noun, ADJ=adjective, ADV=adverb, NUM=numeral, ART=article, PRP=preposition, V=verb, S=singular, P=plural, NOM=nominative, ACC=accusative, VFIN=finitive verb, PR=present, IMPF=past, PCP=participle, @SUBJ=subject, @ACC=direct object, @ADVL=adverbial, @FS-STA=statement, @>N=prenominal, @N<=postnominal, @>A=pre-adject, @P<=argument of preposition, @ICL-N<=postnominal non-finite clause]

3.1 Morphological Annotation

A low degree of morphological ambiguity is a planned design feature of Esperanto, and together with its regular inflexion and affixation system, meant to make the language easy to learn. As a result, automatic annotation is very reliable at this level, and few ambiguity classes exist, with little need for human revision. The only systematic POS ambiguity is between proper nouns and other word classes because of upper-casing (especially in sentence-initial position), and in connection with tokenization errors. Thus, the otherwise reliable vowel coding for POS (e.g. -o = noun, -a = adjective, -i = infinitive, -e = adverb) breaks down in the face of foreign names in (a) and (b). Another type of ambiguity arises from the syntactic, rather than morphological, nature of some non-inflecting word classes (c1-3).

- (a) Durrës-Varna --> adjective -a
- (b) Verdi kaj Ĉajkovskij --> verb -i
- (c1) ĝis la mateno [until morning], --> preposition
- (c2) ĝis ili subskribis [until they signed], --> conjunction
- (c3) ĝis kvar gastoj [up to four guests], --> adverb
- (d) DNA, RNA --> proper?/noun
- (e) i.a. [among other things] --> noun?/adverb

Sometimes, abbreviations can also present problems, because of upper-casing and lack of endings: type (d) is sometimes mis-tagged as e.g. company proper nouns, and dot-shortened abbreviations may default to a (wrong) noun reading.

A final, rare type of ambiguity concerns morpheme structure, and is a source of puns in Esperanto. Although this ambiguity class will not be visible at the lemma/POS/inflexion level, it does affect the meaning of a word, and the *EspGram* parser tries to resolve it (f-g).

- (f) altiri <*ADJ:alt+ir|i> ("go high" [high+go]) vs. <PRP:al+tir|i> ("attract" [to-draw])
- (g) diamante <*N:di+amante> ("God-lovingly") vs. <*ADJ:di|a+mante> ["godly-mantis-ly"] vs. uncompounded "diamond-like"

In principle, there is no inflectional ambiguity in Esperanto. However, foreign proper nouns that have not been assimilated into the language, often retain their original spelling and will rarely receive the accusative case marker -n, unless they happen to end in -o (the noun-marking vowel). Therefore, such proper nouns are nominative/accusative-ambiguous and a theoretical source of errors for *EspGram*'s disambiguation.

3.2 Syntactic Annotation

Syntactic annotation is of course, what a treebank is really about. Thus, the linguistic motivation for creating *Arbobanko* is to allow descriptive studies of Esperanto syntax, addressing topics such as word order and structural complexity. It is for such linguistic reasons, that the relatively fine-grained syntactic tag inventory of *EspGram* is maintained in the treebank. For instance, what could have been one adverbial class, is subdivided into free adverbials (@ADVL), bound adverbials (@SA), object-bound adverbials (@OA), free predicatives (@PRED) and prepositional objects (@PIV). In noun phrases, a distinction is made between *identifying* (@APP) and *predicating* (@N<PRED) appositions. However, we try to avoid unnecessary tag complexity by not introducing different syntactic tags, where POS already contains the distinction. Thus, phrase-level modifiers are only attachment-tagged as prenominals (@>N) and postnominals (@N<) nominals, or pre-adjects¹ (@>A) and post-adjects (@A<), not for what the modifier itself is (e.g. hypothetical @nmod for a modifier that is a nouns), because that would just be duplicated information.

In the same vein, a strict form-function distinction is maintained for dependency heads. For instance, adjectives are not re-tagged as nouns, just because they appear as the head of a noun phrase. In English translation, "sick" stays ADJ in "the sick flocked to him", in spite of it being the head of the subject np. This way, there will be no conflict in it taking an adverb modifier ("the very sick flocked to him"), because "very" still can see necessary ADJ head to attach to. The "noun-ness" of "sick" in "the sick" will thus be expressed solely at the function level, by it carrying a noun function (subject) and an article dependent.

While the above adjective-noun duality is often avoided in Esperanto by adding POS-changing suffixes (mal-san-ul-o = un-healthy-person-noun), another word class, participles, is more problematic, having both adjectival and verbal aspects. Esperanto adjectival participles inflect in gender and number, but are also marked for tense/aspect [aio] and passive/active [\pm n], and often function as non-finite predictors with one or several verb arguments. Therefore, even though there is only one morphological ("form") analysis, the ambiguity manifests at the syntactic function level and needs to be resolved contextually (a-b).

(a) *numeritaj* biletoj [numbered tickets] --> @>N (prenominal)

(b) transportkoridoroj *numeritaj* per romaj ciferoj [traffic corridors numbered with Roman numerals] --> @ICL-N< (postnominal [N<] non-finite [I] clause [CL])

¹ adjects are defined as adverbial modifiers in adjp's and advp's, i.e. of adjectives and adverbs.

3.3 Dependency Annotation

In a typical Constraint Grammar parsing chain, each linguistic level will receive its own grammar module, and disambiguated output from one will be used as input to the next. Classical CG (Karlsson 1990) recognizes three levels: Morphological/POS disambiguation, syntactic function mapping (e.g. based on case or position), and syntactic disambiguation. Syntactic form (structural tags) was addressed only rudimentarily, with arrows indicating attachment direction (e.g. @N< pointing left to a noun head). The state-of-the-art CG3 compiler (Bick & Didriksen 2014) does expand the formalism to allow the creation and use of dependency links, but with pre-existing morphosyntactic parsers this will mean a dependency module that is run *after* function labels have already been assigned - a design different from most machine-learning (ML) approaches, such as the ones described in the CoNLL conference joint tasks on dependency parsing (e.g. Nivre et al. 2007), that will perform the two tasks simultaneously or in the opposite order. This function-first architecture of our automatic annotation system means that dependency attachment rules can exploit existing syntactic information (including attachment direction!), but it also means that many attachment errors need to be fixed in *EspGram* itself, rather than in the add-on dependency module.

In descriptive terms, our native dependency annotation is syntactically motivated rather than semantic, minimizing the dependency distance between a governing head and the token it controls in terms of agreement or valency. Thus, prepositions are treated as heads of pp's, because the verbs and nouns governing the pp may have preposition-specific valency (e.g. *rilati al* [*refer to*], *amikeco kun* [*friendship with*]). In the same vein, auxiliaries are regarded as (syntactic) heads of verb chains, because they control the form of the main verb (infinitive, participle), rather than vice versa. We are aware of the Universal Dependencies (UD) initiative (McDonald et al. 2013) that uses semantic head relations instead (i.e. prepositions and auxiliaries as dependents of main verbs and pp-nouns, respectively), but have chosen to keep syntactic and semantic levels strictly separate in *Arbobanko*. Semantic argument links will thus be added only in a future version with full semantic role and frame annotation. That said, we provide an automatically UD-converted version of the treebank in CoNLL format to further comparability and to allow compatibility with UD-based NLP tools.

Two notoriously difficult issues for a dependency grammar are coordination and ellipsis, both because dependency grammar does not allow empty nodes, forcing either (a) parallel attachment with a loss of structural information or (b) some kind of "dependent nexus", where one dependent attaches to another rather than the common

antecedent. (a) provides short semantic paths for all constituents, but we opted for (b) in the default version of the treebank, again giving priority to syntactic concerns and expliciting the special relation between conjuncts and the parts of an elliptic nexus, respectively. However, sequential attachments of second and later conjuncts to the first conjunct can easily, and automatically, be raised to parallel attachment, if corpus users wish to use the latter format.

Finally, we have chosen to include punctuation in our dependency mark-up. Paired punctuation (e.g. parentheses) will attach to the highest node in the enclosure, and clause and phrase separators attach left to the highest node of the preceding clause or phrase.

3.4 Secondary Tags

Secondary tags are marked with <...> brackets and comprise secondary grammatical and semantic information. They can be mapped either from a lexicon file or by contextual CG rules. Grammatical tags will be disambiguated contextually, if more than one of the same type is possible for a given word, for instance the distinction between <rel> (relative) and <interr> (interrogative) for adverbs and pronouns, or <mv> (main verb) and <aux> (auxiliary) for verbs. A third type of secondary tag helps with coordination conversion: <cjt-first> (first conjunct), <cjt> (second or later conjuncts) and <co-arg> for coordinating conjunctions, with "arg" specifying the syntactic tag of the coordinated material, e.g. <co-subj> for subject coordination.

Semantic tags at this level are not functional (semantic roles), but lexical (ontology-derived), and they are not currently disambiguated in the treebank. Most tags belong to a shallow noun ontology of about 200 classes² and help *EspGram's* CG rules to e.g. choose subject readings over other possible syntactic tags, if a word belongs to a human class (e.g. <Hprof> professional, <Hnat> nationality term, <Hideo> ideology-follower).

4 Parser Evaluation

The focus of this paper is on the creation of a treebank for a language, where there was none, i.e. the resource side rather than the performance side of NLP. So we have evaluated the underlying parser not for its own sake, but in order to be able to improve

² For Esperanto, we have adopted the "semantic prototype" ontology described at http://visl.sdu.dk/semantic_prototypes_overview.pdf.

it and thereby speed up further manual revision of the treebank. Also, category-specific accuracy is useful when interpreting linguistic results from larger, unrevised treebanks made with the same parser.

Our evaluation is based on the change log from the manual revision of the first 16300 tokens of the treebank. Because attachment errors were counted separately, attachment direction arrows at the clause level were ignored when evaluating function tags (i.e. @<SUBJ and @SUBJ> were both counted as just @SUBJ, subject). This evaluation method is clearly more lenient than an independent gold corpus or an independent manual annotation of the same corpus would have been, because when in doubt, a reviewer-annotator will simply choose to do nothing and leave the automatic tag unchanged. A positive side effect of this parser bias, however, is a certain consistency with regard to the resolution of dubious cases, derived from the reproducibility of the automatic choice, and difficult to achieve for human annotators. Also, the parser bias will only affect unclear cases, and still produce good statistics for safe errors, allowing us to flag the most error-prone categories for further inspection.

All in all, ca. 3% of tokens in the test section had errors in primary categories, with 2.6% attachment errors and 1.6% function tag errors. Performance for word tokens alone (ignoring punctuation) is shown in parentheses in table 1. As expected for Esperanto, the extremely regular morphology left almost no room for POS or inflexion errors.

Table 1: Parser performance

	correct attachment	wrong attachment	
correct function	97.04 % (96.53)	1.36 % (1.56)	98.40% (98.09)
wrong function	0.35 % (0.42)	1.25 % (1.49)	1.60 % (1.91)
	97.39 % (96.95)	2.61 % (3.55)	100 % (100)

In a breakdown of individual categories (table 2) pp-attachment problems left their predictable mark, with postnominal pp's (PRP @N<) being attached to the wrong noun, or tagged as adverbial (@ADVL) and attached to a verb. Thus, 19.8% of attachment errors and 26.8% of function errors involved the postnominal category, and 90% of cases were pp's. If predicating appositions are included in this category, it comprises 1/4 - 1/3 of all errors.

Table 2 contains only the major categories, and it lumps all clause functions into only two groups, finite and non-finite, but it clearly shows what is difficult for function tagging and for attachment tagging, respectively. Thus, coordinators (@CO) and, to a lesser degree, adverbials (@ADVL) are more an attachment than a labeling problem,

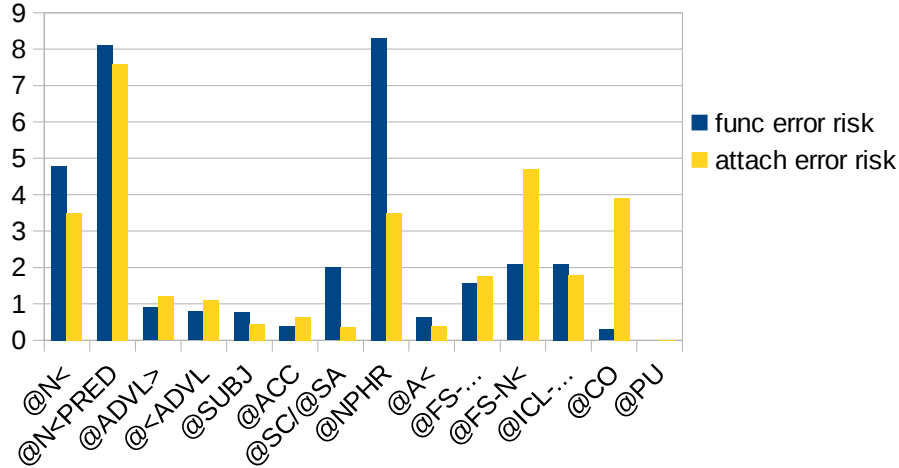
while copula complements (@SC) and subjects (@SUBJ) are more a labeling than an attachment problem. For postnominals (@N<, @N<PRED), a function error will almost always lead to an attachment error, but the latter bears the additional burden of distance errors. That direct objects (@ACC) are so easy to label, is due to the fact that Esperanto has a morphological accusative marker (-n).

Table 2: Error contribution by category (accuracy)

	% of function errors	% of attachment errors	% of all tokens
@N< (postnominal)	26.8	19.8	5.6
@N<PRED (predicat. apposition)	7.3	6.8	0.9
@ADVL> (left adverbial)	5.4	6.8	5.9
@<ADVL (right adverbial)	3.8	5.2	4.8
@SUBJ (subject)	6.9	4.0	8.8
@ACC (direct object)	1.9	3.1	4.8
@SC/@SA (copula complements)	3.8	0.7	1.9
@NPHR (free np, no verb)	5.0	2.1	0.6
@A< (post-adject)	3.4	2.1	5.2
@FS-... (finite clauses)	15.0	16.7	9.5
@FS-N< (relative clause)	2.7	6.1	1.3
@ICL-... (non-finite clauses)	6.1	5.2	2.9
@CO (coordinator)	1.1	12.9	3.3
@PU (punctuation)	0.0	0.1	16.2

While table 2 tells us, where errors occur most in absolute terms, and where added revision and rule-writing should be focused for maximal treebanking efficiency, this is not enough to predict which linguistic information weaned from the corpus is reliable and which is not. For this task, error rates need to be normalized with regard to overall category frequencies. Figure 1 models this category-specific error risk computed as error share divided by token share. The resulting value tells us, how much a category is overrepresented among errors as compared to its share among running tokens. Thus, the most unreliable categories in terms of function labeling are @N<PRED and @NPHR (9 times overrepresented), while all clause-level categories with the exception of complements, i.e. subjects, objects and adverbials are safe (underrepresented).

Fig. 1: Error risk by category



In terms of attachment, the most unreliable categories are coordinators (@CO) and relative clauses (@FS-N<), with a 4x over-representation, and the same np categories that are also unreliable in terms of function (@N<, @N<PRED and @NPHR). All in all, Fig. 1 predicts that an automatically annotated treebank is safer to use for clause-level studies than np-level studies and coordination studies.

5 Linguistic Evaluation

Our first research question was methodological: Does the regular morphology of Esperanto spill over into a more regular syntax in the sense, that parsing will be easier? With our data, the answer to this question appears to be only a little yes. The morphological error rate was indeed very low, but syntactic accuracy (~ 96.5% for word tokens) is only marginally better than what has been reported for CG systems for other languages (e.g. 95-96% for Portuguese [Bick 2014]). Also, recall results for English CG (Prytz 1998) indicate that printed news are probably situated at the high performance end, and that other genres would likely fare worse. Especially the problems with pp attachment and coordination indicate that at the syntactic level, Esperanto is not so different from other languages, and that ambiguity in this area arises from semantics rather than morphology.

The second research question is linguistic - to what degree does Esperanto have free word order? At the clause level, data from Arbobanko indicate a general tendency towards SVO order, but also category-specific deviations. For the statistics in table 3,

relative and interrogative pronouns were excluded because they are always used clause-initially, irrespectively of syntactic category³, in both Esperanto and all etymologically related languages.

Table 3: Clause level constituent placement

	left of V	right of V
subject (@SUBJ)	85.4 %	14.6 %
direct object (@ACC)	10.2 %	89.8 %
copula complement (@SC)	3.6 %	96.4 %
pp/oblique object (@PIV)	2.7 %	97.3 %

As can be seen, subjects and direct objects occur on the "wrong" side of the verb often enough to speak of free word order in the sense that such usage is grammatically acceptable, though not the default, in Esperanto. For oblique and copula arguments, however, left placements (outside relative clauses and questions) is so rare, that it should be considered as marked (e.g. focus-triggered). Object *pronouns* were as (un)likely as object *nouns* to occur left of the verb, so clitic effects in the fashion of Romance languages can be ruled out.

In order to identify complete finite clauses with both subject and object, and count full SVO patterns, we wrote a small mark-up CG mapping %csvo, %vsvo, %sovs etc. tags on the main verbs of these clauses (table 4).

Table 4: SVO variations

	percentage of finite clauses with both subject and direct object
SVO	89.98 %
OVS	2.44 %
SOV	2.69 %
OSV	3.42 %
VSO	-
VOS	1.47 %

³ Yes/no questions with the question particle "ĉu" were not excluded, but were not statistically salient, because only a few contained finite verbs.

The numbers indicate that SVO *is* the default word order for Esperanto outside relative clauses and questions, but that there is no strict rule against other word orders, that together make up 10% of finite S+O clauses. Only VSO did not occur at all. Unexpectedly, the "Yoda" word order OSV is the *most* frequent alternative, in spite of it being the only one that is not documented as a normal word order in natural languages. Non-finite clauses⁴ had a stricter word order than finite clauses, with 98% of objects placed to the right.

At the phrase level, the typologically interesting word-order question is where adjectives are placed in noun phrases. Here, our data did contain some variation, with left placement as the statistical norm, but still 5.9% of adjectives positioned right of their head noun. In addition, heavy modifier material seems to be moved to the right. Thus all modifier clauses, including participle clauses, were placed to the right, as well as half of the coordinated adjectival modifiers.

One conclusion from our np word order data is that Esperanto, despite the fact that the majority of its vocabulary can be traced back to Romance languages, seems to prefer a "Germanic", left placement of adjectives. We therefore also investigated verb phrases, looking for discontinuities, common in Germanic languages. But while we did find about 15.3% discontinuous vp's, almost all interfering material was adverbial, with no sign of post-auxiliary subjects, occurring in many Germanic languages when fronting other constituents.

The last linguistic topic we will present here is the use of complex tense, mode and aspect. For this, Esperanto combines the tense-inflected *esti* ("be") with likewise tense-inflected active and passive participles⁵. Because of the rareness of some combinations, and the low error rate of automatic annotation for this type of auxiliary construction, we have used the entire Monato corpus, with automatic treebank annotation, for the data in table 5.

Table 5:

AUXILIARY:	estas (present)	estis (past)	estos (future)	estus (conditional)
ACTIVE PCP:				
-anta (present, "...-ing")	5	3	-	-

⁴ Non-finite clauses do not take subjects in Esperanto

⁵ These participles carry an adjectival -a ending, and inflect/agree with regard to number and case, allowing them to function as postnominal non-finite clauses, marked @ICL-N< in the treebank, unlike the @ICL-AUX< (argument of auxiliary) we are concerned with here.

-inta (past, "having ...-ed")	7	24	1	13
-onta (future, "going to ...")	1	2	-	1
PASSIVE PCP:				
-ata (present, imperfective, "being ...-ed")	176	79	17	3
-ita (past, perfective, "having been ...-ed")	231	244	30	9
-ota (future, prospective, "to be ...-ed")	2	1	-	-

As can be seen, passives are much more common than actives, probably because the latter cover less linguistic terrain and "only" work as complex tenses, with a "viewer" time marked by the auxiliary, and a relative event time marked as anterior, posterior or simultaneous in the participle. The high-frequency complex passives (*estas/estis/estos* + ...*ata/ita*), on the other hand, are the only way to express finite passives, since only actives have auxiliary-free finite forms. In addition, the participle tense vowel in these forms is used to express aspect (a/present = imperfective, i/past = perfective). The conditional auxiliary form *estus* (last column) is rarest, and mostly used for past conditionals, active or passive. The active present participle is rare, and never used with future and conditional *estos/estus*, implying that no added meaning is achieved compared to the *-os/us* forms of the main on its own.

6 Conclusion and Outlook

We have presented and evaluated a treebank for Esperanto, that we hope will help remedy the lack of NLP resources for the language and trigger further research. By constantly improving the grammar and lexicon of the underlying CG parser, manual revision labour was kept at a minimum. Measured against the revised annotation, the parser achieved a syntactic accuracy (labelling and attachment combined) of 96.5% for non-punctuation tokens, albeit with considerable variation across categories. This relatively high performance should facilitate future work that could include a "raw" (automatic) treebank for the entire *Monato* corpus, as well as new treebank sections for other genres.

On the linguistic side, the treebank has allowed us to establish word (constituent) order statistics classifying Esperanto as an SVO and ADJ-N language with considerable room for word order variation, both at the clause level and for np attributes. What we do not know, and what should be addressed in future research, is to which degree these findings depend on statistical tendencies influenced by the native language of an Esperanto speaker/author, and whether word order variation is

less or more pronounced in the formal written language of a news journal than in spoken or informal written language, as found in social media, e-mail or text messages.

References

1. Bejček, E., Hajičová, E., Hajič, J. et al.: Prague Dependency Treebank 3.0. (Data/Software). Charles University in Prague, MFF, ÚFAL. [<http://ufal.mff.cuni.cz/pdt3.0/>] (2013)
2. Bick, E., Didriksen, T.: CG-3 - Beyond Classical Constraint Grammar. In: Beáta Megyesi: Proceedings of NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania. pp. 31-39. Linköping: LiU Electronic Press (2015)
3. Bick, E.: PALAVRAS, a Constraint Grammar-based Parsing System for Portuguese. In: Tony Berber Sardinha & Thelma de Lurdes São Bento Ferreira (eds.), Working with Portuguese Corpora, pp 279-302. London/New York: Bloomsbury Academic (2014)
4. Bick, E.: A Dependency Constraint Grammar for Esperanto. Constraint Grammar Workshop at NODALIDA 2009, Odense. NEALT Proceedings Series, Vol 8, pp.8-12. Tartu: Tartu University Library. (2009)
5. Bick, E., Tagging and Parsing an Artificial Language: An Annotated Web-Corpus of Esperanto, In: Proceedings of Corpus Linguistics 2007, Birmingham, UK. [<http://ucrel.lancs.ac.uk/publications/CL2007/>] (2007)
6. Bick, E.: Arboretum, a Hybrid Treebank for Danish, in: Joakim Nivre & Erhard Hinrich (eds.), Proceedings of TLT 2003 (2nd Workshop on Treebanks and Linguistic Theory, Växjö, November 14-15, 2003), pp.9-20. Växjö University Press (2003)
7. Böhmová, A., Hajič, J., Panenová, B.H.J., Hajicova, E.: The Prague Dependency Treebank: A 3-Level Annotation Scenario. In: Abeillé, A. (ed.): Treebanks: Building and Using Parsed Corpora. Dordrecht, the Netherlands: Kluwer. pp. 103-126 (2003)
8. Johansson, R., Nugues, P.: Extended Constituent-to-dependency Conversion for English. In Proceedings of NODALIDA 2007. Tartu, Estonia (2007)
9. McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Bertomeu Castelló, N., Lee, J.: Universal dependency annotation for multilingual parsing. In Proceedings of ACL 2013 (2013)
10. Karlsson, F.: Constraint Grammar as a Framework for Parsing Running Text. In: Proceedings of the 13th conference on Computational Linguistics - Vol. 3, pp. 168-173. ACL (1990)
11. Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., Yuret, D.: The CoNLL 2007 Shared Task on Dependency Parsing. In: Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007, pp. 915-932 (2007)
12. Prytz, K.: Evaluation of the Syntactic Parsing Performed by the ENGCG Parser. In: Proceedings of The 11th Nordic Conference on Computational Linguistics, Copenhagen, 28-29 January 1998. ACL web anthology (1998)