FinnTreeBank: Creating a research resource and service for language researchers with Constraint Grammar

Atro Voutilainen
Department of Modern Languages
University of Helsinki
atro.voutilainen@helsinki.fi

Abstract

This paper described ongoing work to develop a large open-source treebank and related Finnish language resources for the R&D community, especially corpus linguistic researchers. Initially, we look at user needs and requirements that these set for corpus annotation. We propose the linguistic Constraint Grammar as a framework to answer the requirements. The second half of the paper describes ongoing work in the FinnTreeBank project to answer these objectives.

1 Needs of corpus linguists

Language researchers need empirical data to help them formulate and test hypotheses e.g. about natural language grammar and meaning. Morphologically annotated (or POS-annotated) text corpora have been available to researchers for many years, and currently such tagged corpora for many languages are accessible. Some of these corpora are very large, even billions of words (e.g. German COSMAS II). Though automatic tagging tends to misanalyse a few words in a hundred, automatically tagged corpora are generally of sufficient quality and quantity for researchers to enable basically word oriented queries and corpus searches in a local context (e.g. "Key Word In Context").

However, corpus linguists are often interested in phenomena that involve more than local character strings: lexically or semantically motivated units in linguistic context (e.g. as part of a syntactic structure). Extraction of such, often nonlocal, linguistic patterns is difficult with string-based corpus searches: queries on POS-tagged corpora to recover clause or sentence level syntactic constructions result in too low accuracy (combination of precision and recall), and the amount of manual postprocessing needed to make the data usable for further analysis is too high to make such searches productive.

1.1 Requirements for syntactic annotation

A corpus with an additional layer of syntactic annotation (e.g. phrase structure or dependency structure) is needed to enable successful queries for clause or sentence level syntactic constructs. To enable successful extraction of desired lexico-syntactic patterns (multiword units with(in) the desired syntactic structure), the syntactically parsed corpora need to have a high correctness rate: most sentences in the parsed corpus (‘treebank’) should have a correct lexical and syntactic analysis.

Further, to enable extraction of patterns containing mid- or low-frequency lexical units in sufficiently high volume for meaningful quantitative analysis, the parsed corpus also should be very large, probably of a size comparable to the largest POS-tagged corpora now available to researchers.

1.2 Limitations with current treebanks

Syntactically parsed corpora, generally referred to as treebanks, are now available for a growing number of languages (cf. Wikipedia entry for "Treebank"), with phrase structure annotations, or, increasingly, with dependency syntactic annotation (to enable analysis of unbounded, or long-distance, dependencies). Most syntactically annotated corpora are very limited in size – typically with thousands, or at most tens of thousands, of sentences (cf. e.g. (Mikulova et al., 2006), (Kromann, 2003) and (Haverinen et al., 2009)).

Assuming corpus linguists are interested in phenomena that involve lexical and syntactic information (involving corpus searches with lexical and syntactic search keys or patterns), a corpus with, say, a 50,000 sentences or a million words, will likely provide far too few ‘hits’ for such complex queries to enable quantitatively meaningful studies. To enable a coverage comparable to local-context lexically oriented searches on POS-tagged corpora, syntactically annotated corpora should be even larger than comparable POS-tagged corpora.

1.3 Limitations with complete sutomatic parsing

Automatic syntactic annotation could be proposed as the obvious solution for providing very large syntactically annotated corpora for researchers. However, automatic syntactic corpus annotation is generally avoided in treebanking efforts, probably because the error rate of automatic syntactic analysis is prohibitively high: even the best statistical dependency parsers (such as Charniak, 2000) assign a correct dependency relation and function to slightly over 90% of tokens (words and punctuation marks). If every tenth word is misanalysed, most text sentences get an incorrect syntactic analysis.

Instead, syntactic corpus annotation is done manually (with some level of supporting automation). At a recent treebank course (organised by CLARA in Prague, December 2010), some of the presenting treebank projects reported manual annotation times at 5-20 minutes per sentence, and there were reports of nearly decade-long treebanking efforts resulting in treebanks of some tens of thousands of sentences.

In the current language technology community, automatic syntactic modelling and analysis are usually carried out with data-driven language models that are based on statistics generated from manually annotated treebanks. Statistical models based on scant or inconsistent data frequently mispredict; even at the lower levels of linguistic analysis with larger quantities of available training data, POS taggers with statistical language models mispredict the category of several words in a hundred (which means that close to or more than half of all sentences are tagged incorrectly). The best statistical dependency parsers reach labelled attachment scores of slightly above 90% at word level in optimal circumstances (training text genre is the same as that of evaluation corpus); for many other languages, the labelled attachment scores reported are substantially lower. With accuracy scores of this magnitude at word level only, incorrectly parsed sentences are likely to constitute the vast majority of all parsed sentences. In short, current statistical models of syntax are probably too inaccurate to provide a complete solution to high-quality automatic treebanking.

To sum, large-scale treebanking efforts seem to be in a deadlock: manual treebanking is too work-intensive (and possibly also too inconsistent) to enable creation of sufficiently large treebanks to support statistically significant corpus linguistic research; statistical parsing efforts so far have failed to provide sufficiently high parsing accuracy to enable automatic creation of high quality research data for corpus linguists.

2 Constraint Grammar as a solution

Constraint Grammar is a reductionistic linguistic paradigm for tagging and surface-syntactic parsing (Karlsson & al, 1995) that has the following properties to make it an attractive environment for treebanking purposes.

- Large-scale work on tagging and parsing has been done in this framework on several languages since late 1980s (cf. Wikipedia entry on Constraint Grammar)

- The most advanced publicly available implementation of the compiler-interpreter (VISL cg3) supports a wide range of functionality, from lexical analysis to disambiguation to dependency syntax

- A grammarian makes and modifies language models (lexicons, parsing grammars), with very competitive accuracy (measured e.g. as precision-recall tradeoff) and modifiability

- CG tagging and parsing can yield full or partial analysis, which enables a necessary control on precision-recall tradeoff for different purposes such as treebanking

As an example case, we consider an early evaluation and comparison on word-class tagging in English (Samuelsson and Voutilainen 1997). In this report, EngCG-2, the second major version of the English Constraint Grammar for word-class disambiguation, was compared with a state-of-the-art statistical ngram tagger (Hidden Markov Model), to answer certain open questions about the original ENGCG by the research community of the late 1990s.

For the experiments, an common tag set and corpora were documented and used, with options for full and partial disambiguation. In EngCG-2, the disambiguation grammar was organised into five increasingly heuristic subgrammars to enable trading recall for precision.

Regarding precision-recall tradeoff of the two taggers in the experiment (cf. Table 1 in the Samuelsson and Voutilainen article), the main observations are:

- With almost fully disambiguated outputs, the ngram tagger discarded a correct analysis 9 times more often than EngCG-2.
- When more ambiguity was permitted in the taggers' analyses, the ngram tagger discarded a correct analysis 28 times more often than EngCG-2.

The possibility to make almost safe predictions in a linguistics-based parsing environment, to control the precision-recall tradeoff, and to achieve a very competitive precision-recall tradeoff is shown in this comparison.

Though we are unaware of similar comparisons at the level of dependency syntax (assignment of dependency functions and dependency relations to words), similar control on the tradeoff between accuracy and partiality of dependency syntactic analysis can be exerted in CG: the rule formalism and development methods when making dependency grammars are highly similar to those used at the (lower) levels of morphological disambiguation and shallow syntactic function assignment.

2.1 Possible solutions for Constraint Grammar based treebanking

Given the CG properties described above, in particular the possibility for partial analysis and for linguistically controlled superior precision-recall tradeoff, several strategies for CG-based treebanking are outlined next.

As a common core to them all is the need to specify the necessary minimal recall needed for the application and to create a language model (lexicon and grammars) to meet this required minimal recall (by permitting some level of ambiguity or partial dependency analysis in analyser output). In the content of treebanking, this could mean something like the following:

- morphology: recall of well over 99%.
- syntactic function tagging: recall of 98% or more.
- correctness of syntactic dependency assignment: over 98% of assigned dependency relations should be corrrect.

The amount of unresolved ambiguity or of unattached words resulting from the minimum recall/correctness requirements depends on several factors, e.g. granularity of the grammatical distinctions that the parser operates with; characteristics of the corpora to be analysed; development time available for the grammarian; development/testing methods and resources available; competence/experience of the constraint grammarian. As an educated guess: 20-30% of input sentences might get a complete unambiguous dependency analysis, which means that about three quarters of the sentences retain some ambiguity or receive a partial dependency analysis.

In any case, an important desirable property of this initial effort is that there is no need to revisit the analytic decisions made by the resulting partial CG parser. The main challenge is what to do with the remaining (morphological and functional) ambiguity and words not attached in the dependency structure. Three solutions are next outlined.

2.1.1 Extraction from a partially parsed treebank

To support search of lexico-syntactic structures from text, the simplest solution is to apply the search key only to dependency trees (representing full sentences or sentence parts). As the analyses provided by the parser are as reliable as specified, the extracted patterns will be of sufficient quality for (minor) postprocessing and quantitative analysis. It is also likely that many search patterns will apply to subsentential constructions (that do not need a complete sentence analysis); this means that a much larger part than the above-estimated 20-30% of sentences will be useful for corpus linguistic searches.

A limitation of this approach is that the corpus accessible for linguistic searches will be skewed, as sentence parts outside the coverage of the parser's language model will not be used.

2.1.2 Resolving remaining ambiguity with a hybrid parser

Data-driven statistical parsers are usually trained on hand-annotated treebanks of limited size (thousands or tens of thousands of sentences), and their accuracies (e.g. Labelled Attachment Scores, LAS), probably fall below the minimum accuracy requirements needed to support linguistic corpus searches (as argued above).

The availability of very large volumes of training data with partial but very dependable morphological and dependency syntactic analyses makes it possible to experiment with training statistical parsing capability to complement (or possibly even replace) partial CG-based parsing, in order to provide a more complete (but still sufficiently accurate) syntactic analysis for text corpora. For instance, it may be the case that lexical information can be used to better advantage in statistical modelling of syntax if the amount of learning data is large (e.g. tens of millions of sentences).

2.1.3 Interactive rule-based dependency parsing

Fully manual syntactic analysis is highly work-intensive. For instance, to provide a dependency analysis and a dependency function to each word in a 20-word sentence, 40 decisions need to be made. This kind of syntactic analysis can easily take several minutes per sentence from a human annotator.

With a high-recall partial dependency parser, probably well over 90% of the analysis decisions are made before there is a need for additional information to support parsing. Given a suitable interface for a human to provide e.g. a part-of-speech disambiguation decision or a dependency analysis to an unattached word in the case of a partially parsed sentence, the language model of the CG parser is usually able to carry on the high-quality syntactic analysis of the sentence, possibly to completion, without further input from the linguist. The reason for this is that the additional analysis provided by the linguist makes the sentence (context) less ambiguous, as a result of which a contextual constraint rule (or a sequence of them) is able to apply, by discarding illegitimate alternative analyses or by adding new dependency relations to the sentence.

The speedup to manual treebanking might be 10-50 fold, which enables cost-effective annotation of much larger treebanks than those available today, but treebanking tens or hundreds of millions of sentences is probably not a practical option even with this semiautomatic method.

3 Ongoing work in FIN-CLARIN

Next we present ongoing work as part of the FIN-CLARIN project (2010-2012) on the creation of a large-scale resource and service for researchers into the Finnish language, focusing on one of its five subprojects, FinnTreeBank. We outline a dependency-syntactic representation for Finnish, and present the first version of the dependency syntactic FinnTreeBank and its use as a "grammar definition corpus" to guide development, testing and evaluation of Constraint Grammar based language models for high-accuracy annotation of large publicly-available Finnish-language corpora, which will be used as empirical data to support linguistic research on Finnish at a large scale.

3.1 Project environment

Our work is done with support from the European CLARIN and METANET consortia, with the following overall aims:

- help researchers discover relevant empirical data and resources more easily with a web service where search is supported e.g. with metadata and persistent identity markers.
- help researchers license and use found resources more easily e.g. with transparent and easy-to-use licensing/access terms and policies.
- help researchers share their own data to support other researchers and to support validation of reported empirical experiments e.g. by means of easy-to-use procedures for data licensing and persistent storage service.
- help researchers use and share existing work by promoting open source.
- help researchers use different resources e.g. by promoting common standards and user-friendly interfaces to data.

At our department, there are several subprojects in the larger META-CLARIN project on different language resources and finite state methods and libraries: Helsinki Finite State Transducer HFST; OMorfi Finnish Open Source Morphology; Finnish WordNet; Finnish Wordbank; FinnTreeBank.

3.2 FinnTreeBank goals and milestones

In addition to th eordinaty academic goal of producing published research with research collaborators, FinnTreeBank has two main goals as a ‘producer’: (i) to provide large high-quality treebanks of Finnish to the research community; (ii) to provide language models of Finnish as open source for use with open-source software, to help researchers analyse additional texts and to help them modify the language models and/or software for an analysis more suitable for their research question.

Recent and near-term FinnTreeBank milestones include the following:

- Evaluation and selection of language resources, technologies and tools for use in FinnTreeBank developments.
- Initial specification of linguistic representation for initial use in treebanking Finnish, with focus on dependency syntax.
- Manual application of dependency syntactic representation on an initial corpus of 19,000 example utterances from a large descriptive grammar of Finnish (including further specification and documentation of the linguistic representation).
- Subcontracting a 3rd party provider to provide a parsing engine (black box) and automatically parsed treebank (EuroParl, JRC-Aquis) for the web service.
- Development of open-source lexicons, parsing grammars and other resources to support high-quality dependency parsing of Finnish by the research community.
- Delivery of new versions of FinnTreeBank with new corpora and higher quality of linguistic analysis.

3.3 Specifying a grammatical representation with a grammar definition corpus

In order to create a high-quality parser and treebank, we need documentation and examples on the linguistic representation and its use in text analysis. In order to approximate also less frequent structures used in a large corpus of text in a comprehensive and systematic way, we need a maximally exhaustive and systematic set of sentences to be analysed and documented e.g. as a guideline for creating a Parsebank. We propose to use a comprehensive descriptive grammar (typically more than a thousand closely-printed book pages) as a source of example sentences to reach a high and systematic coverage of the syntactic structures in the language. A hand-annotated, cross-checked and documented collection of such a systematic set of sentences – in short, a Grammar Definition Corpus – is a workable initial approximation and guideline for annotating or parsing natural language on a large scale. The initial definitional sentence corpus can be extended with new data when ‘leaks’ in the grammar/corpus coverage become evident e.g. on the basis of double-blind annotations (Voutilainen and Purtonen 2011).

A result of this effort is a Grammar definition corpus of Finnish, consisting of about 19,000 example utterances extracted from a comprehensive Finnish grammar (Hakulinen at al, 2004), and manually annotated according to a linguistic representation consisting of a morphological description and a dependency grammar with a basic dependency function palette.

We expect use of the Grammar Definition Corpus to have the following benefits:

- A well-documented Grammar Definition Corpus is useful as a guideline for human annotators, to support consistent and linguistically motivated analysis.
- A Grammar Definition Corpus also is useful for one who writes and tests parsing grammars (e.g. in the CG framework): it helps systematic modelling of target constructions, and it also helps document the scope of the language model (what constructs are covered, and what constructions are left outside the scope of the language model).
- Evaluation and testing of language models, corporan and analysers can be done more objectively if the linguistic representation has been specified in a comprehensive and systematic way.
- When annotating new texts e.g. manually, there is a lower chance to come across unexpected linguistic constructions (given the high coverage of the Grammar Definition Corpus), hence less need to redesign or compromise.

- Encountering constructions not covered by the Grammar Definition Corpus is useful data also for writing a more comprehensive descriptive grammar (compared with the original descriptive grammar from which the example utterances were extracted).

To our knowledge, this effort if the first one based on a comprehensive, well-documented set of sentences. The closest earlier approximation to a Grammar definition corpus we know of is an English corpus, tagged and documented in the early 1990's according to a dependency-oriented representation, and consisting of about 2,000 sentences taken from a comprehensive grammar of English (Quirk et al, 1985). However, the Quirk et al grammar contains much more than the 2,000 sentences (i.e. partial coverage in the corpus), and the annotated corpus itself has not been published, though this early effort is briefly described in (Voutilainen, 1997).

3.4 Dependency representation

Our dependency syntactic representation follows common practice in many ways. For instance, the head of the sentence is the main predicate verb of the main clause, and the main predicate has a number of dependents (clauses or more basic elements such as noun phrases) with a nominal or an adverbial function. More simple elements, such as nominal or adverbial phrases, have their internal dependency structure, where a (usually semantic) head has a number of attributes or other modifiers.

The dependency function palette is fairly ascetic at this stage. The dependency functions for nominals include Subject, Object, Predicative and Vocative; adverbials get the Adverbial function; modifiers get one of two functions, depending on their position relative to the head: premodifying constructions are given an Attributive function tag; postmodifying constructions are given a Modifier function tag. In addition, the function palette includes Auxiliary for auxiliary verbs, Phrasal to cover phrasal verbs, Conjunct for coordination analysis, and Idiom for multiword idioms.

The present surface-syntactic function palette can be extended into a more fine-grained description at a later stage; for instance, the Adverbial function can be divided into functions such as Location, Time, Manner, Recipient and Cause. Such a semantic classification is best done in tandem with a more fine-grained lexical description (entity classification, etc).

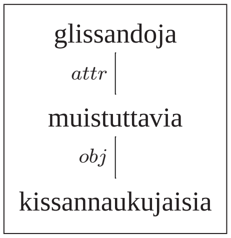
Sometimes, the question arises whether to relate elements to each other on syntactic or on semantic criteria. As an example from English, consider the sentence ”I bought three litres of milk”. On syntactic criteria, the head of the object for the verb ”bought” is ”litres”, but semantically one would prefer ”milk”. Our dependency representation relates elements to each other based on semantic rather than inflectional criteria. Hence our analysis (much as with Prague Tectogrammar and TreeBank) gives a dependent role to categories such as conjunctions, prepositions, postpositions, auxiliaries, determiners, attributes and formal elements (formal subject, formal object, etc.). Sometimes this practice creates a conflict with the accustomed notion that there is a certain correspondence between Finnish cases and syntactic functions (e.g. the genitive or partitive case for the object function): for instance a premodifying quantifier may have the genitive case (for objects), while the semantic object's case may follow from the valency structure of the quantifier. – This feature, like many others, needs to be taken into account in the design of a corpus linguist's search/extraction interface.

3.5 Sample analyses

In this section, some example sentences from the grammar definition corpus are shown in visual form to illustrate the dependency representation outlined above.

3.5.1 Clausal premodifiers

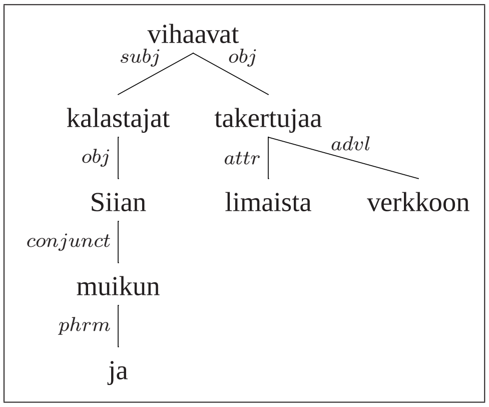
In Finnish, nominals can have clausal modifiers on both sides (premodifying and postmodifying positions). For instance, premodifying participles can have verbal arguments of their own. For instance, the participle "muistuttavia" acts as a premodifier of the noun "kissannaukujaisia" but has also an object, "glissandoja", as its dependent.



kissannaukujaisia [cat-meowings.PartitivePlural] muistuttavia

[resembling.Pcp] glissandoja [glissandos.Plural]

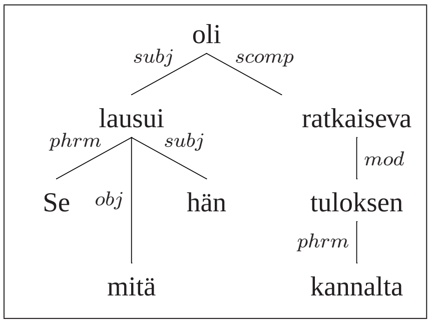
We have also described a restricted class of nouns like this. For instance, agentive nouns like "kalastajat" (fishers) can have objects like "siian" (whitefish) in a premodifying position:



Siian [whitefish.GenSg] ja [and] muikun [vendace.GenSg] kalastajat [fisher.NomPl] vihaavat [hate.VPres] limaista [slimy.PartSg] verkkoon [net.IllatSg] takertujaa [clinger.PartSg].

3.5.2 Phrase markers

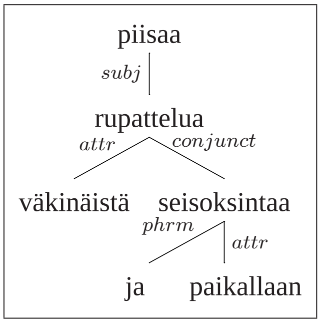
Formal ”se” (’it’) is described as a phrase marker for the subject clause "mitä hän sanoi" (what s/he said); likewise the postposition "kannalta" (regarding) is described as a phrase marker of the noun "tuloksen" (result):



Se [it.NomSg] mitä [what.PartSg] hän (s/he.NomSg) lausui [said] oli [was] tuloksen [result.GenSg] kannalta [regarding.Postposition] ratkaiseva [decisive.NomSg].

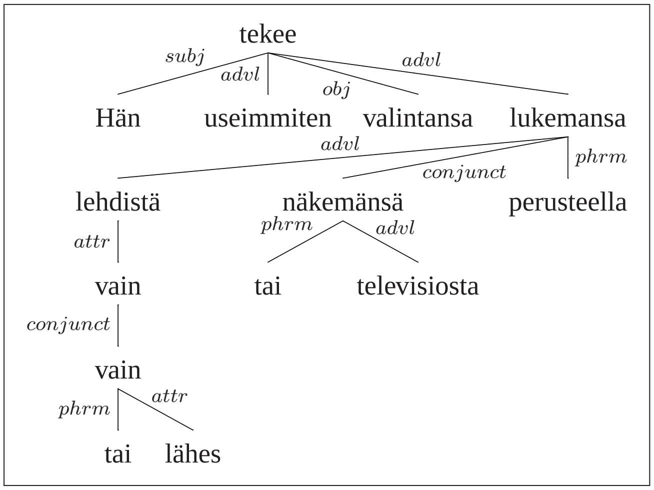
3.5.3 Coordination

The conjunction "ja" (and) is described as a phrase marker of the following conjunct "paikallaan seisoksintaa" (steady standing), which in turn is described as coordinated dependent of the preceding conjunct "väkinäistä rupattelua" (forced chatting):



väkinäistä [forced.PartSg] rupattelua [chatting.PastSg] ja [and] paikallaan [steady.AdessSg] seisoksintaa [standing.PartSg] piisaa [suffices].

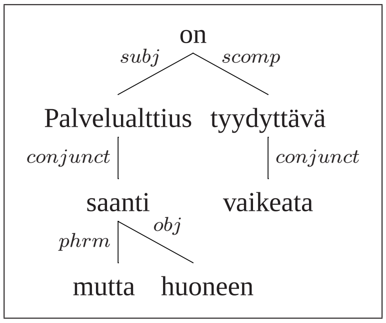
Here is an example with multiple coordinations. The attributes "vain" (only) and "lähes vain" (almost only) are coordinated with "tai" (or); the participles "lukemansa" (read) and "näkemänsä" (seen) are coordinated also with "tai":



Hän [s/he] tekee [makes] useimmiten [usually] valintansa [choice.GenPl] vain [only] tai [or] lähes [almost] vain [only] lehdistä [newspaper.ElatPl] lukemansa [read.PcpPoss] tai [or] televisiosta [television.ElatSg] näkemänsä [see.PcpPoss] perusteella [on-the-basis-of.Postposition].

3.5.4 Ellipsis

Two clauses are coordinated: S-V-C with S-C (verb missing). The subject of the elliptical clause ("huoneen saanti") is described as a conjunct of the subject of the first clause ("palvelualttius"), and the predicative complement ("vaikeata") is described as a conjunct of the predicative complement of the first clause ("tyyydyttävä"):



Palvelualttius [service-readiness.NomSg] on [is] tyydyttävä [satisfactory.NomSg], mutta [but] huoneen [room.GenSg] saanti [getting.NomSg] vaikeata [difficult.NomSg].

4 Ongoing developments

In this final section, we describe some ongoing or near-term developments to meet the objectives of the FinnTreeBank project during the next year and a half.

4.1 Harmonisation of morphology with syntax

The initial dependency syntactic annotation (function and relation assignment by linguists) was mainly done independently of morphological analysis. One motivation for this is savings in labour: a morphological description designed before a syntactic description usually needs to be revised when the detailed decisions on how to model syntax are made (which means that also morphological annotations require substantial revisions). In our solution, the morphological description can be designed "at one go" to agree with the documented syntactic representation. A further advantage of our solution is that resolution of morphological ambiguities can be done with the help of available higher-level (syntactic) analysis.

In practise, the morphological and lexical analysis will be based on the Omorfi open-source lexical and morphological language model (partly derived from publicly available word lists by the Finnish Research Centre of Domestic Languages) and finite-state (HFST) analysis tools. Along with this semiautomatic synchronisation/tagging effort, also consistency checks and corrections to syntactic annotation can be made to improve the quality of the grammar definition corpus treebank.

The morphologically synchronised treebank will be delivered in CONLL-X form with extensive documentation to enable e.g. development of statistical language models for parsing.

4.2 Dependency treebank and parser engine by third-party provider

Another ongoing development is done by a third-party provider (Lingsoft and its collaborators, the Turku BioNLP Group at University of Turku) who is building a statistical language model for dependency parsing on the basis of the initial grammar definition corpus with the dependency syntactic annotation. On the basis of the contract, the provider will deliver automatically parsed language resources (EuroParl corpus and JRC-Aquis, totalling tens of millions of words of Finnish) for distribution via the FIN-CLARIN service.

The provider will also provide a licence to the executable parser engine to enable annotation of additional corpora for FIN-CLARIN users.

4.3 Development of open-source language models for dependency parsing

Alongside the above developments, the FinnTreeBank project develops open-source language models using open-source tools and development environments (e.g. HFST morphology and syntax, VISL cg3) for dependency parsing of Finnish. The FIN-CLARIN users will benefit from the open-source development as it enables them to adapt and apply the language models and resulting parsers to better answer their research questions and to better support development of e.g. Artificial Intellignce solution prototypes. The results of this development can also be used for providing an alternative annotations to existing and new corpora (treebanking).

Also development of commercial or open-sector web services and other solutions should benefit from availability of open-source language technological tools and resources.

4.4 Experiments on treebanking methods

When initial versions of the language models mature, it will be possible to start experimenting with alternative treebanking methods outlined above in section 2.1. This research will likely be carried out in collaboration with other research teams towards (and hopefully after) the end of the ongoing project. The results of the experiments will provide guidance on treebanking efforts in the longer term in Finland, and hopefully in other projects as well.

Acknowledgments

The ongoing project has been funded via CLARIN, FIN-CLARIN, FIN-CLARIN-CONTENT and META-NORD by EU, University of Helsinki and the Academy of Finland. I wish to thank Mikaela Klami, Tanja Purtonen, Satu Leisko-Järvinen, Kristiina Muhonen, Tommi Pirinen and Sam Hardwick, as well as other HFST team members, for their support of this project.

References

Eckhard Bick. 2000. *The parsing system Palavras*. Aarhus: Aarhus University Press.

Christer Samuelsson and Atro Voutilainen. 1997. Comparing a linguistic and a stochastic tagger. *Proc. EACL-ACL'97*.

Pasi Tapanainen and Timo Järvinen. 1997. A non-projective dependency parser. *Proceedings of the 5th Conference on Applied Natural Language Processing*. Washington, D.C.

Auli Hakulinen, Maria Vilkuna, Riitta Korhonen, Vesa Koivisto, Tarja Riitta Heinonen and Irja Alho. 2004. *Iso suomen kielioppi* [Large Finnish Grammar]. Helsinki: Suomalaisen Kirjallisuuden Seura. Online version: http://scripta.kotus.fi/visk URN:ISBN:978-952-5446-35-7.

Katri Haverinen, Filip Ginter, Veronika Laippala, Tapio Viljanen, Tapio Salakoski. 2009. Dependency Annotation of Wikipedia: First Steps towards a Finnish Treebank. *Proceedings of The Eighth International Workshop on Treebanks and Linguistic Theories (TLT8)*.

Matthias Kromann. 2003. The Danish Dependency Treebank and the underlying linguistic theory. *Proc. of the TLT 2003*.

Krister Lindén, Miikka Silfverberg and Tommi Pirinen. 2009. HFST Tools for Morphology – An Efficient Open-Source Package for Construction of Morphological Analyzers. *Proceedings of the Workshop on Systems and Frameworks for Computational Morphology 2009*, Zürich, Switzerland.

Marie Mikulova, Alevtina Bemova, Jan Hajic, Eva Hajicova, Jiri Havelka, Veronika Kolarova, Lucie Kucova, Marketa Lopatkova, Petr Pajas, Jarmila Panevova, Magda Razimova, Petr Sgall, Jan Stepanek, Zdenka Uresova, Katerina Vesela, and Zdenek Zabokrtsky. 2006. Annotation on the Tectogrammatical Level in the Prague Dependency Treebank. Annotation Manual. Technical Report 30, UFAL MFF UK, Prague, Czech Rep.

Joakim Nivre, Jens Nilsson and Johan Hall. 2006. Talbanken05: A Swedish Treebank with Phrase Structure and Dependency Annotation. *Proceedings of the fifth international conference on Language Resources and Evaluation (LREC2006)*.

Ville Oksanen, Krister Lindén and Hanna Westerlund. 2010. Laundry Symbols and License Management: Practical Considerations for the Distribution of LRs based on experiences from CLARIN. *Proceedings of the seventh international conference on Language Resources and Evaluation (LREC2010)*.

Ted Pedersen. 2008. Last Words: Empiricism Is Not a Matter of Faith. *Computational Linguistics, Volume 34, Number 3, September 2008*.

Randolph Quirk, S. Greenbaum, G. Leech, and J. Svartvik. 1995. *A comprehensive grammar of the English language*. London: Longman.

Atro Voutilainen, Krister Lindén and Tanja Purtonen (forthcoming). 2011. Designing a Dependency Representation and Grammar Definition Corpus for Finnish. *Proc. CILC 2011 - III Congreso Internacional de Lingüística de Corpus*.

Atro Voutilainen. 1997. Designing a (Finite State) Parsing Grammar. Roche and Schabes, Eds, *Finite State Language Processing*. The MIT Press.