Syntactic Annotation of Large Corpora in STEVIN

Gertjan van Noord*, Ineke Schuurman†, Vincent Vandeghinste†

*RU Groningen
Netherlands
vannoord@let.rug.nl
† KU Leuven
Belgium
{ineke.schuurman,vincent}@ccl.kuleuven.be

Abstract

The construction of a 500-million-word reference corpus of written Dutch has been identified as one of the priorities in the Dutch/Flemish STEVIN programme. For part of this corpus, manually corrected syntactic annotations will be provided. The paper presents the background of the syntactic annotation efforts, the Alpino parser which is used as an important tool for constructing the syntactic annotations, as well as a number of other annotation tools and guidelines. For the full STEVIN corpus, automatically derived syntactic annotations will be provided in a later phase of the programme. A number of arguments is provided suggesting that such a resource can be very useful for applications in information extraction, ontology building, lexical acquisition, machine translation and corpus linguistics.

1. Background

The Dutch Language Corpus Initiative (D-Coi) is one of the projects funded within the current STEVIN programme.[1] The construction of a 500-million-word reference corpus of written Dutch has been identified as one of the priorities in the programme. In D-Coi, a 50-million-word pilot corpus is being compiled, parts of which will be enriched with (verified) linguistic annotations. In particular, syntactic annotation of a representative sub-corpus of 200.000 words is envisaged. The focus is on written language in order to complement the Spoken Dutch Corpus (CGN).

CGN contains a sub-corpus of 1 million words with syntactic annotations. During the construction of this corpus, no syntactically annotated corpus of Dutch was available to train a statistical parser on, nor an adequate parser for Dutch (requirements: wide-coverage, theory-neutral output, access to both functional and categorial information). This situation has changed considerably since then. Over the last few years, Alpino (van Noord, 2006) was developed at the University of Groningen. Alpino is a computational analyzer of Dutch which aims at full accurate parsing of unrestricted text, and which incorporates both knowledge-based techniques, such as a HPSG-grammar and -lexicon which are both organized as inheritance networks, as well as corpus-based techniques, for instance for training its POS-tagger and its disambiguation component.

2. Alpino parser

The Alpino grammar is a wide-coverage computational HPSG for Dutch. The grammar takes a 'constructional' approach, with rich lexical representations and a large number of detailed, construction specific rules (about 600). Both the lexicon and the rule component are organized in a multiple inheritance hierarchy. By relating rules to each other and to more general structures and principles via inheritance, a rule component can be defined which contains a potentially large number of specific rules, while at the same time the relevant generalizations about these rules are still expressed only once. Beyond considerations of linguistic theory and software engineering an important argument in favor of such an implementation is the fact that parsing on the basis of a grammar with specific rules appears to be more efficient than parsing on the basis of general rule schemata and abstract linguistic principles.

Alpino contains a large lexicon. At the moment, the lexicon contains about 100,000 entries. Flemish (uses of) words are added when necessary. In addition there is a list of about 200,000 named entities. The lexicon is extended with a number of additional lexical rules to recognize dates, temporal expressions and other special named entities. The lexicon is stored as a perfect hash finite automaton, using Jan Daciuk's FSA tools (Daciuk, 2000), providing a very compact representation as well as very efficient access.

For words which are not in the lexicon, the system applies a large variety of unknown word heuristics, which attempt to deal with numbers and number-like expressions, capitalized words, words with missing diacritics, words with 'too many' diacritics, compounds, and proper names. If such heuristics still fail to provide an analysis, then the system attempts to guess a category based on the word's morphological form. If this still does not provide an analysis, then it is assumed that the word is a noun. A crucial component of the Alpino system is the POS-tagger which greatly reduces lexical ambiguity, without an observable decrease in parsing accuracy (Prins and van Noord, 2003).

Based on the categories assigned to words and word sequences, and the set of grammar rules compiled from the HPSG grammar, a left-corner parser finds the set of all parses, and stores this set compactly in a packed parse forest. All parses are rooted by an instance of the top category, which is a category that generalizes over all maximal projections (S, NP, VP, ADVP, AP, PP and some others). If there is no parse covering the complete input, the parser finds all parses for each substring. In such cases, the robust-

[1] `http://taalunieversum.org/taal/technologie/stevin/`

corpus	sents	length	F-sc	CA%
Alpino	7136	20	88.50	87.92
Trouw	1400	17	91.14	90.87

Table 1: Accuracy of Alpino on Alpino treebank, and on Trouw2001 treebank. The table lists the number of sentences, mean sentence length (in tokens), F-score and concept accuracy, both expressed in terms of named dependencies.

ness component will then select the best sequence of non-overlapping parses (i.e., maximal projections) from this set. In order to select the best parse from the compact parse forest, a best-first search algorithm is applied. The algorithm consults a Maximum Entropy disambiguation model to judge the quality of (partial) parses. The disambiguation model and the best-first search algorithm are described in (van Noord and Malouf, 2005).

The output of the parser is evaluated by comparing the generated dependency structure for a corpus sentence to the dependency structure in a treebank containing the correct dependency structure for that sentence. For this comparison, we represent the dependency structure (a directed acyclic graph) as a set of named dependency relations (the edges of the graph). Comparing these sets, we count the number of relations that are identical in the generated parse and the stored structure. This approach is very similar in spirit to the evaluation methodology advocated in (Briscoe et al., 2002), although there are differences with respect to the actual dependencies and the details of the metric.

In table 1, we list the accuracy of the full system. In the first row, the results of the Alpino Treebank is presented using ten-fold cross-validation. The Alpino treebank (van der Beek et al., 2002; Alpino, 2002) contains manually corrected dependency structures of all 7,100 sentences (about 145,000 words) of the newspaper (cdbl) part of the Eindhoven corpus (Uit den Boogaard 1975). In the second row, we list the accuracy of another manually corrected set of dependency structures for 1400 sentences of the Trouw 2001 newspaper (taken from the Twente News corpus[2]).

3. Annotation Guidelines

The original annotation scheme deployed in Alpino was not exactly the same as the one used in CGN (Hoekstra et al., 2004; Schuurman et al., 2003). In order to enhance the possibilities to compare results found in D-Coi on the one hand and CGN on the other, we have adapted the Alpino scheme in such a way that it more closely resembles the CGN annotation scheme. For instance, the treatment of multi-word-units, punctuation tokens, ordinal numbers and *te*-infinitives has been adapted in Alpino and now conforms to the CGN-standard. A few remaining differences are documented exhaustively for the benefit of the users of both corpora (Schuurman et al., 2006). These differences include, for instance, the annotation of subjects of the embedded verb in auxiliary, modal and control structures, and the annotation of the direct object of the embedded verb in passive constructions. In CGN, these are not expressed. In D-Coi we follow the convention of Alpino to encode these subject relations explicitly. An example of such a dependency structure is provided in figure 1, for the sentence:

(1) Enrico Fabris heeft zijn tweede gouden medaille
Enrico Fabris has his second gold medal
veroverd
won
Enrico Fabris won his second gold medal

One of the options Alpino offers and which is currently not being used within the D-Coi project, is its ability to recognize temporal expressions and dates. This might, however, be of interest as soon as semantic annotations, especially temporal ones, will be added to the corpus as well (manual under development, (Monachesi and Schuurman, 2006)).

In D-Coi, we also inherit from Alpino the XML-format in which syntactic annotations are stored. This format directly allows the use of full XPATH and/or Xquery search queries for linguistically interesting queries. Therefore, we can employ standard tools for the exploitation of the syntactic annotations, and there is no need to dedicate resources for the development of specialized query languages. Note that the existing CGN corpus has been translated to the same XML-format, so that the same tools can be used for both corpora.

4. Annotation Tools

For interactive annotation, Alpino provides a variety of tools. These tools include optional interactive assignment and selection of lexical categories. The annotator can pick, if desired, the correct lexical categories for some or all of the words in the input, or add additional lexical categories on the fly. Limiting the parser to the correct lexical categories implies that the parser will find a reduced number of parses (these will generally be closer to the correct parse). In addition, the speed of the parser increases considerably if lexical ambiguity decreases. The initial assignment of lexical categories can be provided by the POS-tagger. We aim to integrate the D-Coi POS-tag annotations (van Eynde, 2005) provided by other project partners in this process.

Another powerful tool is the optional and interactive assignment of syntactic brackets. The annotator can indicate, for instance, that a particular sequence of words must be analyzed as a particular syntactic category, in order to direct the parser to the correct analysis in the case of ambiguities. Both labeled and unlabeled brackets are supported (Wieling et al., 2005). For a typical case of PP-ambiguity, such as:

(2) I saw the man with the telescope

the annotator might edit the input sentence as follows:

(3) I saw [@np the man with the telescope]

The annotations rule out the analysis in which the prepositional phrase is attached to the VP. Using this technique, the right parse can often be constructed with very little manual intervention.

Alpino can be used to obtain the best N or all parses. A parse selection tool is available to select the correct parse

[2] `http://wwwhome.cs.utwente.nl/~druid/TwNC/TwNC-main.html`

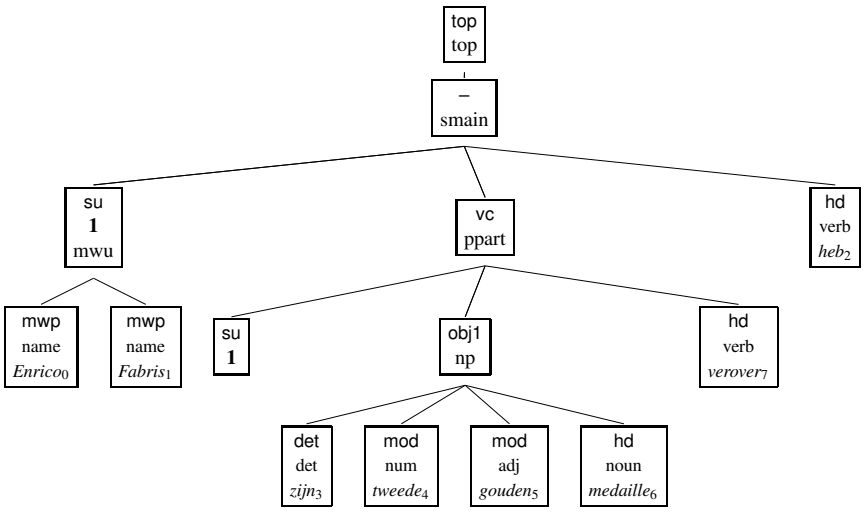

Figure 1: Example Dependency Structure

or the best parse from a potentially large set of parses without the need to consider each of these parses individually (similar to the SRI Treebanker (Carter, 1997)). In this parse selection tool, the annotator makes a number of binary decisions about particular properties of the desired parse. Based on each decision, the tool computes the remaining set of candidate parses, and reduces the number of binary decisions.

The annotator has access to the Thistle editor (Calder, 2000) for intuitive editing CGN-type dependency structures. In addition, a number of XML-based tools is available for automatic consistency checking of the annotations, for browsing the annotations, and for searching the annotations (Bouma and Kloosterman, 2002). The tools are all freely available.[3]

5. Future Directions

One of the ultimate goals of the STEVIN-programme is the construction of a 500-million-word reference corpus of written Dutch. In a future project, called LASSY, we will provide verified syntactic annotations for at least 1 million words. In addition, we intend to provide syntactic annotations (not manually corrected) of this full 500-million-word corpus. Such a large syntactically annotated corpus is useful for a wide variety of applications in information extraction, question answering, corpus linguistics, automated ontology building, lexicography, machine translation etc.

As an initial example, we consider applications in Question Answering (QA). Alpino is used as an important component of a recent Question Answering system for Dutch, called Joost (Bouma et al., 2005b). Alpino is used both to analyze the question, as well as to analyze all potential answers. In order that Joost has access to the full syntactic structure of potential answers (both for on-line and off-line search), the Alpino-system was used to parse the full text collection for the Dutch CLEF2005 Question Answering task. The text collection was tokenized (into 78 million words) and segmented into (4.1 million) sentences. Parsing this amount of text takes well over 500 CPU days.

[3] `http://www.let.rug.nl/~vannoord/alp/`

This CLEF2005 treebank was employed both for on-line question answering, as well as off-line question answering. In the latter case, which is a case of information extraction, answers for typical questions are collected before the question is asked, giving rise to tables consisting of e.g. capitals, causes of deaths, functions of person names, etc. (Bouma et al., 2005a). It was shown that the availability of (automatically constructed) syntactic annotation improves the quality of such tables considerably. The Joost QA-system took part in CLEF2005 (monolingual QA). In this evaluation, the system found the correct answer for 49.5% of the questions, obtaining the best result for Dutch (out of 3 submissions), and the third result overall (out of 42 submissions).

Large, automatically annotated corpora are also useful for applications in corpus linguistics. Bouma, Hendriks and Hoeksema (Bouma et al., to appear) study a.o. the distribution of focus particles in prepositional phrases. Their corpus study on the basis of the CLEF2005 treebank revealed that such focus particles in fact are allowed (and fairly frequent) in Dutch, contradicting claims in theoretical linguistics. Similar techniques have been applied for the study of PP-fronting in Dutch (Bouma, 2004), the order of noun phrases with ditransitives (van der Beek, 2004), the distribution of determiner-less PPs (van der Beek, 2005), the distribution of weak pronouns, the distribution of impersonal pronouns as objects of prepositions, etc.

Very similar techniques have been integrated in other applications in information extraction and ontology building. Van der Plas and Bouma (van der Plas and Bouma, 2005b) apply vector-based methods to compute the semantic similarity of words, based on co-occurrence data extracted from the CLEF2005 treebank. The novel aspect of this work is that they define contexts with respect to the syntactic environment, rather than simple co-occurrence of words. Such syntactic contexts include verb-subject, verb-object, adjective-noun, elements of a coordination, elements in an apposition, and element in a prepositional complement. They show (van der Plas and Bouma, 2005a) that the acquired ontological information correlates with the informa-

tion in Dutch EuroWordNet, and that the performance of question answering improves with such automatically acquired lexico-semantic information.

In (Vandeghinste et al., 2006) it is shown that hybrid Machine Translation not using parallel corpora is a feasible option. At the moment the highest level of analysis used is chunking, as for many languages a (free) parser is not available. Experiments with a fully parsed target-language corpus and a full syntactic analysis of the source language input sentence may extend this approach and enhance translation quality. The accuracy of such an approach can give input for a cost-benefit analysis as to whether it is to be recommended to invest in the construction of a parser in case of machine translation.

6. References

Alpino. 2002. The Alpino treebank 1.0. University of Groningen, November. CDROM; also available via http://www.let.rug.nl/˜vannoord/trees/.

Gosse Bouma and Geert Kloosterman. 2002. Querying dependency treebanks in xml. In *Proceedings of the Third international conference on Language Resources and Evaluation (LREC)*, Gran Canaria.

Gosse Bouma, Jori Mur, and Gertjan van Noord. 2005a. Reasoning over dependency relations for QA. In Farah Benamarah, Marie-Francine Moens, and Patrick Saint-Dizier, editors, *Knowledge and Reasoning for Answering Questions*, pages 15–21. Workshop associated with IJCAI 05.

Gosse Bouma, Jori Mur, Gertjan van Noord, Lonneke van der Plas, and Jörg Tiedemann. 2005b. Question answering for Dutch using dependency relations. In *Proceedings of the CLEF2005 Workshop*.

Gosse Bouma, Petra Hendriks, and Jack Hoeksema. to appear. Focus particles inside prepositional phrases: A comparison between Dutch, English and German. *Journal of Comparative Germanic Linguistics*.

Gosse Bouma. 2004. Treebank evidence for the analysis of PP-fronting. In S. Kubler, J. Nivre, E. Hinrichs, and H. Wunsch, editors, *Third Workshop on Treebanks and Linguistic Theories*, pages 15–26, Seminar für Sprachwissenschaft, Tübingen.

Ted Briscoe, John Carroll, Jonathan Graham, and Ann Copestake. 2002. Relational evaluation schemes. In *Proceedings of the Beyond PARSEVAL Workshop at the 3rd International Conference on Language Resources and Evaluation*, pages 4–8, Las Palmas, Gran Canaria.

Jo Calder. 2000. Thistle and interarbora. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING)*, pages 992–996, Saarbrücken.

D. Carter. 1997. The treebanker: A tool for supervised training of parsed corpora. In *Proceedings of the ACL Workshop on Computational Enviroments For Grammar Development And Linguistic Engineering*, Madrid.

Jan Daciuk. 2000. Finite state tools for natural language processing. In *Using Toolsets and Architectures to Build NLP Systems. Coling 2000 Workshop*, pages 34–37, Luxembourg, August. Centre Universitaire.

P. C. Uit den Boogaart. 1975. *Woordfrequenties in geschreven en gesproken Nederlands*. Oosthoek, Scheltema & Holkema, Utrecht. Werkgroep Frequentieonderzoek van het Nederlands.

H. Hoekstra, M. Moortgat, B. Renmans, M. Schouppe, I. Schuurman, and T. Van der Wouden, 2004. *CGN Syntactische Annotatie*. (http://www.ccl.kuleuven.be/Papers/sa-man_DEF.pdf).

P. Monachesi and I. Schuurman, 2006. *D-Coi Semantische Annotatie*. (Intermediate, project-internal version).

Robbert Prins and Gertjan van Noord. 2003. Reinforcing parser preferences through tagging. *Traitement Automatique des Langues*, 44(3):121–139.

I. Schuurman, M. Schouppe, T. Van der Wouden, and H. Hoekstra. 2003. Cgn, an annotated corpus of Spoken Dutch. In A. Abbeilé, S. Hansen-Schirra, and H. Uszkoreit, editors, *Proceedings of 4th International Workshop on Language Resources and Evaluation*, pages 340–347, Budapest.

I. Schuurman, G. Van Noord, and V. Vandeghinste, 2006. *D-Coi Syntactische Annotatie*. (Intermediate, project-internal version).

Leonoor van der Beek, Gosse Bouma, Robert Malouf, and Gertjan van Noord. 2002. The Alpino dependency treebank. In *Computational Linguistics in the Netherlands*.

Leonoor van der Beek. 2004. Argument order alternations in Dutch. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG'04 Conference*. CSLI Publications.

Leonoor van der Beek. 2005. The extraction of Dutch determinerless PPs. In *Proceedings of the 2nd ACL-SIGSEM Workshop on the Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications*, University of Essex, Colchester.

Lonneke van der Plas and Gosse Bouma. 2005a. Automatic acquisition of lexico-semantic knowledge for QA. In *Proceedings of the IJCNLP workshop on Ontologies and Lexical Resources*.

Lonneke van der Plas and Gosse Bouma. 2005b. Syntactic contexts for finding semantically related words. In Ton van der Wouden, Michaela Poss, Hilke Reckman, and Crit Cremers, editors, *Computational Linguistics in the Netherlands 2004. Selected Papers from the fifteenth CLIN meeting*. LOT, Netherlands Graduate School of Linguistics, Utrecht.

Frank van Eynde. 2005. Part of speech tagging en lemmatisering van het D-COI corpus. Intermediate, project-internal version.

Gertjan van Noord and Robert Malouf. 2005. Wide coverage parsing with stochastic attribute value grammars. Draft available from the authors.

Gertjan van Noord. 2006. **A**t **l**ast **p**arsing **i**s **n**ow **o**perational. In *TALN 2006*, Leuven.

V. Vandeghinste, I. Schuurman, M. Carl, S. Markantonatou, and T. Badia. 2006. METIS-II: Machine Translation for Low Resource Languages. In *Proceedings of LREC 2006*.

Martijn Wieling, Mark-Jan Nederhof, and Gertjan van Noord. 2005. Parsing partially bracketed input. Paper presented at CLIN2005, Amsterdam.