From D-Coi to SoNaR: A reference corpus for Dutch

N. Oostdijk, M. Reynaert, P. Monachesi, G. van Noord,
R. Ordelman, I. Schuurman, V. Vandeghinste

University of Nijmegen, Tilburg University, Utrecht University, University of Groningen,
University of Twente, KULeuven, KULeuven

Abstract

The computational linguistics community in The Netherlands and Belgium has long recognized the dire need for a major reference corpus of written Dutch. In part to answer this need, the STEVIN programme was established. To pave the way for the effective building of a 500-million-word reference corpus of written Dutch, a pilot project was established. The Dutch Corpus Initiative project or D-Coi was highly successful in that it not only realized about 10% of the projected large reference corpus, but also established the best practices and developed all the protocols and the necessary tools for building the larger corpus within the confines of a necessarily limited budget. We outline the steps involved in an endeavour of this kind, including the major highlights and possible pitfalls. Once converted to a suitable XML format, further linguistic annotation based on the state-of-the-art tools developed either before or during the pilot by the consortium partners proved easily and fruitfully applicable. Linguistic enrichment of the corpus includes PoS tagging, syntactic parsing and semantic annotation, involving both semantic role labeling and spatiotemporal annotation. D-Coi is expected to be followed by SoNaR, during which the 500-million-word reference corpus of Dutch should be built.

1. Introduction

With funding of the Dutch and Flemish governments and research foundations the present joint Dutch-Flemish STEVIN programme was put in place in 2004. One of the aims of the STEVIN programme is to realize an appropriate digital language infrastructure for Dutch. The programme also intends to stimulate strategic research in the domains of language and speech technology. The compilation of a reference corpus for Dutch has been identified as one of the priorities in the programme. Such a corpus is considered one of the prerequisites for the development of other resources, various tools, and applications. It is expected that once the corpus is available it will give a significant boost to natural language processing involving the Dutch language.

The reference corpus should be a well-structured, balanced collection of text samples tailored to the uses to which the corpus is going to be put. The contents of the corpus as well as the nature of the annotations to be provided are to be largely determined by the needs of ongoing and projected research and development in the fields of corpus-based natural language processing. Applications such as information extraction, question-answering, document classification, and automatic abstracting that are based on underlying corpus-based techniques will benefit from the large-scale analysis of particular features in the corpus. Apart from supporting corpus-based modeling, the corpus will constitute a test bed for evaluating applications, whether or not these applications are corpus-based.

The construction of a reference corpus requires that motivated decisions be taken for all aspects of its design, encoding, markup, and annotation schemes, while also various protocols and procedures must be in place. Therefore, from June 2005 until December 2006, the STEVIN programme funded the Dutch language Corpus Initiative (D-Coi) project. This project can be characterized as a preparatory project. An outline of D-Coi is given in the next Section and the various phases of text conversion, preprocessing and XML-ization are outlined in Section 3. Section 4 deals with the PoS annotation issues, Section 5 with syntactic parsing and Section 6 with semantic annotation, which comprises both semantic role labeling and spatiotemporal annotation. Section 7 concludes this paper.

2. The pilot project D-Coi

D-Coi aimed to produce a blueprint for the construction of a large (around 500 million words), balanced corpus of contemporary written standard Dutch. This entailed the design of the corpus and the development (or adaptation) of protocols, procedures and tools that are needed for sampling data, cleaning up, converting file formats, marking up, annotating, post-editing, and validating the data. In order to support these developments a 54-million-word pilot corpus was compiled, parts of which were enriched with linguistic annotations. The pilot corpus was intended to demonstrate the feasibility of the approach. It provided the necessary testing ground on the basis of which feedback could be obtained about the adequacy and practicability of the procedures for acquiring material and handling IPR, as well as of various annotation schemes and procedures, and the level of success with which tools can be applied. Moreover, it served to establish the usefulness of this type of resource and annotations for different types of HLT research and the development of applications.

The D-Coi project has been rather successful in what it set out to do, both with respect to the design of the corpus and the development of protocols, procedures and tools. Thus a motivated design was made that should guide the compilation of the reference corpus. The design has profited from the experiences in other large scale projects directed at the compilation of corpora (e.g. BNC, ANC and the Spoken Dutch Corpus (Corpus Gesproken Nederlands, CGN)). In addition, consultation of the user community has contributed to establishing needs and priorities (Oostdijk and Boves, 2006). The design is ambitious as it aims at a 500-million-word reference corpus of contemporary standard written Dutch as encountered in texts (i.e. stretches of running discourse) originating from the Dutch speaking language area in Flanders and the Netherlands as well as Dutch translations published in and targeted at this area. Texts will be included from more conventional genres and text types as well as from new media. Table 1 lists the design of the reference corpus: SoNaR. The corpus will include native speaker language and the language of (professional) translators. It is intended that approximately two-thirds of the texts originate from the Netherlands and one-third from Flanders. Only texts will be included that have appeared from the year 1954 onwards.

text types	SoNaR	D-Coi
Written to be read, published, electronic		
Discussion lists	2.5 MW	
E-books	5 MW	
E-magazines	25 MW	2,289,286
E-mail (spam)	2.5 MW	
Newsletters	2.5 MW	1,917
Press releases	10 MW	332,795
Subtitles	10 MW	
Teletext pages	50 MW	489,128
Websites	50 MW	1,021,922
Wikipedia	20 MW	23,178,848
Written to be read, published, printed		
Abstracts, summaries	10 MW	
Books	75 MW	663,687
Brochures	5 MW	1,232,819
Newsletters	2.5 MW	33,529
Guides, manuals	5 MW	236,099
Legal texts	2.5 MW	10,761,969
Newspapers	50 MW	2,826,465
Periodicals, magazines	10 MW	117,246
Policy documents	5 MW	9,078,771
Proceedings	10 MW	349,102
Reports	5 MW	93,043
Surveys	2.5 MW	
Theses	2.5 MW	
Written to be read, unpublished, electronic		
Chats	25 MW	
E-mail (non-spam)	50 MW	
Minutes	10 MW	
SMS	5 MW	
Written assignments	10 MW	
Written to be read, unpublished, printed		
Theses	10 MW	
Written to be read, unpublished, typed		
Minutes	10 MW	
Written assignments	10 MW	
Written to be spoken, unpublished, electronic		
Autocues	2.5 MW	928,706
Written to be spoken, unpublished, typed		
News scripts	2.5 MW	
Texts for the visually impaired	2.5 MW	676,062

Table 1: Corpus design (MW = million words).

The D-Coi project has proven to be very useful, particularly in gaining experience with acquiring and (pre-)processing language data from a wide range of data sources with different formats. As it turned out, both aspects had been severely underestimated. In the compilation of the D-Coi pilot corpus IPR issues have tended to frustrate the acquisition process so that in order to make sure that sufficient material would be available for testing and evaluation purposes we had to resort to a more opportunistic approach of acquiring data which involved focusing on data that were already in the public domain (e.g. under GPL) or considered low-risk, such as texts found on public websites maintained by the government and public services. Some genres and text types, however, remain underrepresented in the pilot corpus or do not occur in it at all, as can be seen in Table 1, Column 3. Apart from the problems relating to IPR, the conversion of various file formats to the basic XML format that had been defined as target presented some serious problems. Especially the conversion of PDF required manual intervention.

Experiences with the definition of output file formats, the development and adaptation of protocols, procedures and tools can, without exception, be described as positive. Wherever possible we have re-used and built upon previous results from the CGN project. This includes the CGN file format and the annotation scheme for PoS tagging and lemmatization, while also the COREX exploitation software has been adapted so as to accommodate written language data. By re-using and building upon results previously obtained in the CGN project, the D-Coi project has contributed to the development and consolidation of de facto standards for Dutch language corpora and annotations. These standards have been and are being adopted in other initiatives aiming to develop Dutch language resources, such as the JASMIN and the Dutch Parallel Corpus projects.

In summary, the D-Coi project can be judged to have fulfilled its role as preparatory project to the full. It has given us the opportunity to come up with a design for a reference corpus in close consultation with the user community. The compilation of the pilot corpus has given us hands-on experience with the work ahead of us, some facets of which we had underestimated before. With the insights gained we now hold a better view of what realistically can be done and what not.

In February 2007, a consortium of D-Coi project partners made a bid for the Call for Tender directed at the actual compilation of the reference corpus. The proposal was evaluated by an international assessment panel and put up for funding. Recently the consortium received an initial grant to start work on the SoNaR (Stevin Nederlandstalig Referentiecorpus, i.e. Stevin Dutch Reference Corpus) project as from January 2008. Already in 2006, STEVIN funded a follow-up to D-Coi, called LASSY. LASSY focuses on the syntactic annotation of the STEVIN reference corpus (including lemmatization, PoS tagging and dependency structure annotation).

In what follows, we describe the actual steps taken in preprocessing, converting to XML and linguistically annotating the corpus.

3. Text Conversion, Tokenization and Normalization

3.1. Data Storage and Text Conversion

The goal of the work in this stage was to make the incoming data stream suitable for further upstream processing. The text data files that were collected from a variety of sources were gathered centrally and stored along with available metadata (such as content provider, date downloaded, original filename). The often rather exotic original filenames were renamed to unique identifiers that were selected according to the data characteristics. For example, the document group ’Wikipedia’ received identifiers in the form *WR-P-E-I-nnnnnnnnnn*, where *WR* stands for ‘Written to be Read’, *P* for ‘Published’, *E* for ’Electronic’ and *I* identifies the particular text subtype ‘Wikipedia’.

A second step involved the conversion from the different file formats encountered such as PDF, MS-Word, HTML and XML to a uniform D-Coi XML format. This uniform format should allow us to store metadata and the text itself along with linguistic annotations from later processing stages. Moreover, it provides the means to perform XML validation after each processing stage: first after the conversion from original file format to the D-Coi format, and then again whenever new annotations are added. Especially the validation after the first conversion appeared to be a crucial one in order to prevent that the processing chain was jammed due to incorrect conversions.

Putting much effort in the development of conversion tools was regarded outside the scope of the project. However, the conversion from original format to D-Coi XML appeared to be rather problematic in a substantial number of cases. Given the data quantities aimed at in the project, an approach that uses a (semi-)manual format conversion procedure was not regarded a realistic option. Therefore the approach was to use existing conversion tools and repair conversion damage wherever possible. For a large proportion of the data this procedure worked quite well. Sometimes only minor adaptations to the post-processing tools were required in order to fix a validation problem for many files. Some parts of the collected data however had to be temporarily marked as unsuitable for further processing as it would take too much time to adapt the post-processing tools.

Especially the conversion of the PDF formatted files appeared to be difficult. Publicly available tools such as `pdf2html` that allow for the conversion from PDF to some other format often have problems with line-breaks and headers and footers, producing output that is very hard to repair. On the other hand, as moving away from abundantly available content in PDF format would seriously limit the project in finding a balance over text data types, the D-Coi approach was to do PDF conversion semi-automatically for a small part of the collection. A varying amount of effort was required to convert other formats successfully to the D-Coi file format.

Progress of the work in all stages could be monitored by all project partners via a simple PHP web-interface[1] on a MYSQL database containing the relevant information for each file such as the raw word counts, validation status for each level, and total word counts (grand total, counts per document group, validated, etc). The database was synchronised with the information in the D-Coi file system so that project partners could immediately fetch data that became available for their processing stage. The database and web-interface currently serve as intermediate documentation of the work done.

3.2. Text Tokenization and Sentence Splitting

A major aim of the first conversion step to XML is to have titles and paragraphs identified as such. This is because most tokenizers, our own included, may fail to properly recognize titles and because the sentence splitting process expects a paragraph to consist of at least one full sentence. Failure in the first conversion step to recognize that a paragraph in TXT format is split up into *n* lines by newline characters, results in *n* XML paragraphs being defined. This is unrecoverable to the tokenizer. This fact can mostly be detected by the ratio of sentences identified after tokenization in comparison to the number of paragraphs in the non-tokenized version. In such cases both unsuccessful versions were discarded and new ones produced semi-automatically by means of minimal, manual pre-annotation of the raw TXT version of the documents.

The rule-based tokenizer used within D-Coi was developed at the Induction of Linguistic Knowledge research team at Tilburg University prior to D-Coi. It was slightly adapted to the needs of D-Coi on the basis of evaluations conducted by means of TOKEVAL, a tokenizer evaluator developed during the project in order to evaluate the available sentence splitters and tokenizers. These and other tools developed during D-Coi are available from `http://ilk.uvt.nl`, as are our technical reports. A very good alternative to the ILK tokenizer (ILKTOK), is the tokenizer that is available in the Alpino Parser distribution.

As neither of the sentence-splitters/tokenizers available to us handle XML, we developed a wrapper program (WRAPDCOITOK) that deals with the incoming XML stream, sends the actual text to the sentence splitter/tokenizer, receives the outcoming sentences and tokens and wraps them in the appropriate XML. This scheme further allows for collecting sentence and word type statistics and for word type normalization during the tokenization step.

3.3. Text Normalization and Correction

During D-Coi we developed CICCL, which is a set of programs for identifying various types of primarily typographical errors in a large corpus. CICCL stands for ‘Corpus-Induced Corpus Clean-up’ and has in part been described in (Reynaert, 2006). Assumptions underlying this work are: 1) that no resources other than corpus-derived n-gram lists are available, 2) that the task can be performed on the basis of these resources only, to a satisfactory degree, 3) that in order to show that this is so, one needs to measure not only the system’s accuracy in retrieving non-word variations for any given valid word in the language, but also its capabilities of distinguishing between what is most likely a valid word and what is not. CICCL is capable of the above to

[1] `http://hmi.ewi.utwente.nl/searchd-coi`

the extent that for Dutch for every two words retained by the program, one is in actual fact a typographical variant or in short: a typo. The other is nevertheless a false positive: a correct word retrieved as a suspected typo. The conclusion was therefore that it is not yet possible to fully automatically 'clean' a corpus on the basis of nothing but corpus-derived information.

Where diacritics are missing and the word form without diacritics is not a valid word in its own right, fully automatic replacement is mostly possible and has been effected. This was performed for the words requiring diacritics which are listed in (Woordenlijst Nederlandse Taal, 1995), i.e. the official 'Word list of the Dutch Language'. Also we have a list of about 16,500 known typos for Dutch and most of the selections have been screened for these.

In SoNaR, text correction will be performed more thoroughly, i.e. all divergent spelling variants will be automatically lined up with their canonical form by means of TICCL (Text-Induced Corpus Clean-up), which was introduced in (Reynaert, 2008).

4. PoS tagging and Lemmatization

The entire D-Coi pilot corpus was PoS tagged by means of Tadpole, which is available under GPL (online demo: `http://ilk.uvt.nl/cgntagger`, software: `http://ilk.uvt.nl/tadpole`). PoS tagging with Tadpole reaches an accuracy of 96.5% correct tags (98.6% correct on main tag) on unseen text. Tadpole is described in more detail in (van den Bosch et al., 2007).

A more detailed account of how PoS tagging and lemmatization was actually applied in the D-Coi corpus is given in (van den Bosch et al., 2006). Part of the D-Coi corpus (500,000 words) underwent manual correction of the PoS tags.

4.1. Manual Correction of PoS tags: Focusing on suspect tags

The quality of the tagger–lemmatizer makes it hard to find the few mistakes left, when looking through the tags one by one. We are therefore deploying tools that focus on suspect tags only, identified by a low confidence value.

The output of the tagger consists of PoS tagged files, containing all possible tags for each token, together with the probability of that tag. We developed a tool for the manual correction of these automatically generated PoS tagged files. This tool takes a PoS tagged file as input, together with a threshold value. It presents the human annotator only with those cases where more than one possible tag has an above-threshold probability. All other cases where more than one tag is generated by the tagger, or those cases where only one tag is generated, are not presented to the annotator, resulting in a markedly lower workload.

We performed a small experiment to determine at which value we best set the threshold: a threshold value of 0.06 results in a reduction of the number of decisions to be made by the human annotator with 28%, while skipping a mere 1% of errors which are not presented to the annotator.

This shows that, with the benefit of a tagger well-trained on a large volume of manually checked training material, we can manually check much larger amounts of data in the same time, missing hardly any errors.

corpus	sents	length	F-sc%
D-Coi	12390	16	86.72

Table 2: Accuracy of Alpino on the manually corrected syntactically annotated sub-part of D-Coi. The table lists the number of sentences, mean sentence length (in tokens), and F-score in terms of named dependencies.

While following this procedure, we regularly check all manually corrected material against a blacklist of typical errors made by the tagger, particularly on multi-word named entities and high-frequency ambiguous function words such as *dat* (*that*, having the same ambiguity as in English) which the tagger sometimes tags incorrectly but with high confidence.

5. Syntactic Annotation

In D-Coi, part of the corpus has been annotated syntactically. In a follow-up STEVIN-project called LASSY, the syntactically annotated subset of the D-Coi pilot corpus will be extended, so that it contains about 1 million words. These syntactic annotations are manually corrected. The remaining part of the D-Coi corpus will also be assigned syntactic annotations. This, however, will be done fully automatically and the output will not be manually verified. A more detailed description of syntactic annotation in D-Coi and LASSY is given by (van Noord et al., 2006).

The syntactic annotation is based on the annotation guidelines that were developed earlier for the construction of the CGN. The original annotation scheme deployed in D-Coi was not exactly the same as the one used in CGN (Hoekstra et al., 2004; Schuurman et al., 2003). 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 encode these subject relations explicitly.

During the construction of CGN, 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). Over the last years, Alpino was developed at the University of Groningen. Alpino is a computational analyzer of Dutch which provides 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 disambiguation component. An overview of Alpino is given in (van Noord, 2006).

In table 2, we list the accuracy of Alpino on that part of the D-Coi pilot corpus for which manually verified syntactic annotations are now available.

In D-Coi, we also inherited from Alpino the XML format in which syntactic annotations are stored. This format directly allows for the use of full XPath and/or Xquery search queries. Therefore, we can employ standard tools for the exploitation of the syntactic annotations, and there is no need to dedicate resources to 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.

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.

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., 2006). For a typical case of PP ambiguity, such as:

(1) I saw the man with the telescope

the annotator might edit the input sentence as follows:

(2) 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 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 TrEd editor[2] for intuitive editing CGN-type dependency structures. TrEd is a fully customizable and programmable graphical editor and viewer for tree-like structures. TrEd has been extended with new functionality specifically for D-Coi syntactic annotation.

6. Semantic annotation

So far, the creation of semantically annotated corpora has lagged behind dramatically. As a result, the need for such resources has become urgent. Several initiatives have been launched at the international level in the last years, however, they have focussed almost entirely on English and not much attention has been dedicated to the creation of semantically annotated Dutch corpora. Therefore the STEVIN-programme has identified semantic annotation as one of its priorities. Within the D-Coi project, protocols were developed for semantic role assignment and for temporal and spatial semantics.

6.1. Semantic role annotation

During the last few years, corpora enriched with semantic role information have received much attention, since they offer rich data both for empirical investigations in lexical semantics and large-scale lexical acquisition for NLP and Semantic Web applications. Several initiatives are emerging at the international level to develop annotation systems of argument structure. Within the D-Coi project we have exploited existing results whenever possible. In particular, we have evaluated two leading projects in this area, namely FrameNet (Johnson et al. 2002) and PropBank (Kingsbury et al. 2002) in order to assess whether the approach and the methodology they have developed for the annotation of semantic roles could be adopted for our purposes.

FrameNet reaches a level of granularity in the specification of the semantic roles which might be desirable for certain applications (i.e. Question Answering). However, it makes automatic annotation of semantic roles rather problematic and might raise problems with respect to uniformity of role labelling even if human annotators are involved. Furthermore, incompleteness constitutes a serious problem, i.e. several frames and relations among frames are missing mainly because FrameNet is still under development. Adopting the FrameNet lexicon for semantic annotation means contributing to its development with the addition of (language specific) and missing frames.

The other alternative was to employ the PropBank approach which has the advantage of providing clear role labels and thus a transparent annotation for both annotators and users. Furthermore, there are promising results with respect to automatic semantic role labeling for English thus the annotation process could be at least semi-automatic. A disadvantage of this approach is that we would have to give up the classification of frames in an ontology, as is the case in FrameNet, which could be very useful for certain applications, especially those related to the Semantic Web.

However, a third option is also available in which we can reconcile the PropBank approach to role assignment (which is essentially corpus based and syntax driven) with the more semantic driven FrameNet approach (which is based on a network of relations between frames). More generally, we would like to adopt the conceptual structure of FrameNet, but not necessarily the granularity of its role assignment approach. With respect to role assignment, we would like to adopt the annotation approach of PropBank. In order to assess the feasibility of our approach we have carried out two pilot studies. The conclusion was that while it is feasible to adopt the merging approach to annotate D-Coi in the general case, we might encounter problems with language specific phenomena. If we want to pursue this approach further, the first step is to annotate the D-Coi corpus with semantic roles according to the PropBank approach. We refer to (Monachesi and Trapman 2006b) and (Schuurman and Monachesi 2006) for further details.

[2] http://ufal.mff.cuni.cz/˜pajas/tred/

6.2. Revisions to the PropBank guidelines

The PropBank guidelines were originally developed for the semantic annotation of the (English) Penn TreeBank. In order to make the guidelines suitable for use with the D-Coi corpus, they had to be adapted. Notice that both PropBank and D-Coi share the assumption that consistent argument labels should be provided across different realizations of the same verb and that modifiers of the verb should be assigned functional tags. However, they adopt a different approach with respect to the treatment of traces since PropBank creates co-reference chains for empty categories while within D-Coi empty categories are almost nonexistent and in those few cases in which they are attested, a co-indexation has been established already at the syntactic level. Furthermore, D-Coi assumes dependency structures for the syntactic representation of its sentences while PropBank employs phrase structure trees. In addition, Dutch behaves differently from English with respect to certain constructions and these differences should be spelled out.

The revisions we have made have been driven by the annotation process carried out within the D-Coi project and whenever possible, examples form the corpus have been provided. Furthermore a methodology for annotating the D-Coi corpus has been sketched. During the annotation process, some problems have emerged, which we summarize below:

1. linguistic problems: during the annotation we have encountered some phenomena for which linguistic research does not yet provide a standard solution. We have disregarded these cases for the moment but it would be desirable to address them in the future.

2. interaction among levels: we have encountered examples in which the annotation provided by the syntactic parser was not correct as in the case of a PP which was labeled as modifier by the syntactic annotation but which should be labeled as argument according to the PropBank guidelines. Furthermore, we have encountered problems with respect to PP attachment, that is the syntactic representations give us correct and incorrect structures and at the semantic level we are able to disambiguate. We have therefore decided to mark the correct structures with their appropriate tags and to label the incorrect ones with a special tag so that it is possible to identify them. This might be useful also for a learning system. However, more research is necessary to assess whether this is the appropriate strategy.

3. inter-annotator agreement: unfortunately due to lack of resources, it was not possible to have more than one annotator label the corpus. However, we came across several cases in which it would have been good to measure inter-annotator agreement, especially with respect to the labeling of the modifiers.

6.3. Automatic Semantic Role Labeling

One advantage of employing PropBank for the annotation of semantic roles is that it is quite suitable for automatic semantic role labeling. Although there has been an increasing interest in automatic SRL in recent years, previous research has focused mainly on English corpora. Adapting earlier research to the Dutch situation therefore represents an interesting challenge, especially because the machine learning techniques used in previous research cannot be applied to Dutch texts. This is due to the fact that there is no semantically annotated Dutch corpus available that could be used as training data. In order to solve this problem, a novel approach to rule-based tagging based on D-Coi dependency trees has been proposed (Stevens 2006). Intuitively, dependency structures are a great resource for a rule-based semantic tagger, for they directly encode the argument structure of lexical units, e.g. the relation between constituents. Our goal was to make optimal use of this information in an automatic SRL system. In order to achieve this, we first defined a basic mapping between nodes in a dependency graph and PropBank roles.

The approach is implemented in a rule-based semantic argument tagger, called XARA. (XML based Automatic Role-labeler for Alpino-trees) (Stevens 2006).[3] XARA is written in Java, the cornerstone of its rule-based approach is formed by XPath expressions. A rule in XARA consist of an XPath expression that addresses a node in the dependency tree, and a target label for that node, i.e. a rule is a *(path,label)* pair. For example, a rule that selects direct object nodes and labels them with ARG1 can be formulated as:

```
(//node[@rel='obj1'], 1)
```

The evaluation carried out shows that XARA achieves a precision of 65.11%, a recall of 45.83% and an F-score of 53.80%.

After a corpus has been tagged automatically by XARA, manual annotation can be performed relatively fast, since annotators only need to correct XARAs output instead of starting annotation from scratch.

The manually corrected sentences have been used as training and test data for an SRL classification system. For this learning system we have employed a Memory Based Learning (MBL) approach, implemented in the Tilburg Memory based learner (TiMBL) (Daelemans et al., 2004).

From features used in previous systems and some experimentation with TiMBL, we derived the following feature set. The first group of features describes the predicate (verb):

(1) Predicate stem - The verb stem, provided by Alpino.

(2) Predicate voice - A binary feature indicating the voice of the predicate (passive/active).

The second group of features describes the candidate argument:

(3) Argument c-label - The category label (phrasal tag) of the node, e.g. NP or PP.

(4) Argument d-label - The dependency label of the node, e.g. MOD or SU.

[3] The system is available at: `http://www.let.uu.nl/~Paola.Monachesi/personal/dcoi/index.html`

(5) Argument PoS tag - PoS tag of the node if the node is a leaf node, null otherwise.

(6) Argument position - a binary feature which indicates whether the argument is positioned before or after the predicate.

(7) Argument head-word - The head word of the relation if the node is an internal node or the lexical item (word) if it is a leaf.

(8) Head-word PoS tag - The PoS tag of the head word.

(9) c-label pattern of argument - The left to right chain of c-labels of the argument and its siblings.

(10) d-label pattern - The left to right chain of d-labels of the argument and its siblings.

(11) c-label & d-label of argument combined - The c-label of the argument concatenated with its d-label.

In comparison to experiments in earlier work, we had relatively few training data available: our training corpus consisted of 2,395 sentences which comprise 3,066 verbs, 5,271 arguments and 3,810 modifiers.

The classifier obtained a precision of 70.27% a recall of 70.59% and an F-score of 70.43%. We refer to (Stevens 2006) and (Monachesi et al. 2007) for further details.

6.4. Temporal and spatial semantics

Within D-Coi the development of an annotation scheme for temporal and spatial annotation was started. It is called MiniSTEx (Mini SpatioTemporal Expressions)

A first characteristic of this scheme is that it handles temporal and (geo)spatial annotation in one go, using the same approach. It also handles *geotemporal* expressions, i.e. expressions associated with a combination of geospatial and temporal properties (for example in order to express that between the First and the Second World War *Libya*, nowadays an independent country, was a province of Italy.)

A second characteristic is that full advantage is taken of the fact that the origin of the texts to be handled is known as the metadata will contain the date (sometimes even the time) and place of publication, and also the title of the newspaper or the like. From the latter the `background` of the text can be determined, and thus the `intended audience` of the text can be inferred, which to a large extent determines how a spatiotemporal expression is to be interpreted, taking into account Grice's maxims which can be paraphrased as "Don't say too much and don't say too little." This means that the most obvious interpretation of a (spatiotemporal) expression often will not be clarified by the author, whereas other interpretations will. So, in a national Belgian newspaper based in Brussels the use of the notion *summer* without further specification will refer to the months of June, July and August, as Belgium is in the northern hemisphere, whereas the relevant months will be mentioned when a reference is made to summer in countries like Australia or Brazil, i.e. the southern hemisphere. The same holds for toponym resolution: when in the same newspaper no further specifications are given the toponym *Haren* will refer to the village in the Brussels Capital Region (same region), although for example the village with the same name in Germany has a larger population. But also when the much lesser known (and smaller) village *Haren* belonging to the municipality of *Borgloon* in the province of Limburg (Flanders) is meant, this will be mentioned explicitly. However, in a Borgloon based local newspaper, the default interpretation will be that of the nearby *Haren* in Limburg.

The aim of the spatiotemporal annotation scheme is to identify spatiotemporal expressions, and to normalize and disambiguate them in order to facilitate reasoning. Ideally, all eventualities mentioned in a text should be located on a time-axis and the geospatial information contained in these eventualities should be detectable on a map.

The MiniSTEx spatiotemporal annotation scheme reflects the state of the art in geospatial and temporal annotation. With respect to the latter, TimeML (Sauri et al., 2006) and TIDES (Ferro et al., 2005) come to mind. Geospatial annotation as such is far less widespread and standardized.[4] However, the subtask of disambiguation is also a subject in geographic information extraction. Some approaches in this field can be found in (Ding et al., 2000; Leidner, 2006; Volz et al., 2007).

But especially as far as the Netherlands and Belgium are concerned[5] the scheme goes into more detail, explaining (verbatim) where something is located, for example stating that *Haren* is part of the municipality of *Borgloon* which is in the province of *Limburg* etc. etc. This way reasoning is advanced as the annotated corpus should be self-contained, i.e. the user is to be able to use it without the need of having a full spatiotemporal database at his disposal.

Such a database, however, plays a central role in the annotation process, as it contains the common spatiotemporal knowledge of the intended Flemish/Dutch audience. Other spatiotemporal knowledge is expected to be contained in the texts to be annotated, although the data obtained this way will also be used to further populate the database.

Also with respect to the temporal component, the spatiotemporal scheme developed within D-Coi is more detailed than, for example, TimeML in that it makes explicit the dates between which *Easter* may fall on, or when it was *Easter* in a specific year, taking into account the country and religion under consideration.

The syntactically analyzed sentences, i.e. trees, resulting from the syntactic annotation described in Section 5 form the input for the spatiotemporal annotation. Also the results of semantic role assignment are made use of, for example in order to determine whether a spatiotemporal expression is used in a literal way or as a metonym (which affects the annotation to be attached).

More details on MiniSTEx, the spatiotemporal annotation scheme discussed above, are to be found in (Schuurman, 2007b; Schuurman, 2007a; Schuurman, 2008).

7. Final Remarks

At the LREC-2008 conference the results and experiences obtained in the D-Coi project will be presented together

[4] Recently, in the context of the ACE (Automatic Contact Extraction) program, a scheme for geospatial annotation was proposed: SpatialML.

[5] And to a lesser extent also neighbouring countries, and other countries the intended audience is expected to be familiar with.

with the findings of the Center for Sprogteknologi (CST) who carried out an (external) evaluation of the results. In addition, the SoNaR project will be introduced, describing the approach taken in compiling the reference corpus. We still hope then to be able to announce that SoNaR has in fact a bright future.

Acknowledgements

The D-Coi project was funded by the NTU/STEVIN programme (`http://www.taalunieversum.org/stevin`) under grant number STE04008. The SoNaR project received initial funding from the same programme under grant number STE07014. Thanks are due to our collaborators in these projects for their dogged perseverance in the face of uncertain conditions.

8. References

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

W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. 2004. TiMBL: Tilburg Memory Based Learner, version 5.1.0, reference guide. Technical Report ILK 04-02, ILK Research Group, Tilburg University.

J. Ding, L. Gravano, and N. Shivakumar. 2000. Computing geographical scopes of web resources. In *26th International Conference on Very Large Databases, VLDB 2000*, Cairo, Egypt, September 10–14.

L. Ferro, L. Gerber, I. Mani, B. Sundheim, and G. Wilson, 2005. *TIDES 2005 Standard for the Annotation of Temporal Expressions.*

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`).

C. R. Johnson, C. J. Fillmore, M. R. L. Petruck, C. F. Baker, M. J. Ellsworth, J. Ruppenhofer, and E. J. Wood. 2002. FrameNet: Theory and Practice. ICSI Technical Report tr-02-009

P. Kingsbury, M. Palmer and M. Marcus. 2002. Adding Semantic Annotation to the Penn TreeBank. In *Proc. of the Human Language Technology Conference. HLT-2002.* San Diego, California

J. Leidner. 2006. Toponym Resolution: A First Large-Scale Comparative Evaluation. Technical report, School of Informatics, University of Edinburgh, July.

P. Monachesi, and J. Trapman (2006). Merging FrameNet and PropBank in a corpus of written Dutch. In: Proc. of the workshop 'Merging and Layering linguistic information'. *LREC-2006.* Genoa, Italy.

P. Monachesi, G. Stevens and J. Trapman (2007) Adding semantic role annotation to a corpus of written Dutch. In: *Proc. of LAW-07*. ACL 2007 workshop. Prague. Czech Republic.

N. Oostdijk and L. Boves. 2006. 2006. User requirements analysis for the design of a reference corpus of written Dutch. In *Proc. of the Fifth International Conference on Language Resources and Evaluation, LREC-2006*, Genoa, Italy.

M. Reynaert. 2006. Corpus-Induced Corpus Clean-up. In *Proc. of the Fifth International Conference on Language Resources and Evaluation, LREC-2006*, Genoa, Italy.

M. Reynaert. 2008. Non-interactive OCR post-correction for giga-scale digitization projects. In *Proc. of CICLing 2008. Lecture Notes in Computer Science Vol. 4919/2008*, pages 617–630, Berlin / Heidelberg. Springer.

R. Sauri, J. Littman, B. Knippen, R. Gaizauskas, A. Setzer, and J. Pustejovsky, 2006. *TimeML Annotation Guidelines, version 1.2.1.*

I. Schuurman. 2007a. Spatiotemporal Annotation on Top of an Existing Treebank. In *Proc. of the Sixth International Workshop on Treebanks and Linguistic Theories*, pages 151–162, Bergen, Norway.

I. Schuurman. 2007b. Which New York, which Monday? The role of background knowledge and intended audience in automatic disambiguation of spatiotemporal expressions. In *Proc. of CLIN 17.*

I. Schuurman. 2008. Spatiotemporal annotation using MiniSTEx: How to deal with alternative, foreign and obsolete names? In *Proc. of LREC-2008*, Marrakech, Marocco.

I. Schuurman and P. Monachesi (2006) The contours of a semantic annotation scheme for Dutch. In: *Proc. of CLIN-2005.*

I. Schuurman, M. Schouppe, T. Van der Wouden, and H. Hoekstra. 2003. CGN, an annotated corpus of Spoken Dutch. In Proc. of 4th International Workshop on Language Resources and Evaluation, pages 340–347, Budapest.

G. Stevens (2006). Automatic Semantic Role Labeling in a Dutch Corpus. Master thesis. Utrecht University.

A. van den Bosch, I. Schuurman, and V. Vandeghinste. 2006. Transferring PoS-tagging and lemmatization tools from spoken to written Dutch corpus development. In *Proc. of the Fifth International Conference on Language Resources and Evaluation, LREC-2006*, Genoa, Italy.

A. van den Bosch, B. Busser, S. Canisius, and W. Daelemans 2007. An efficient memory-based morphosyntactic tagger and parser for Dutch. In *Proc. of CLIN 17*.

G. van Noord, I. Schuurman, and V. Vandeghinste. 2006. Syntactic annotation of large corpora in STEVIN. In *Proc. of the Fifth International Conference on Language Resources and Evaluation, LREC-2006*, Genoa, Italy.

G. van Noord. 2006. **A**t **L**ast **P**arsing **I**s **N**ow **O**perational. In *TALN 2006 Verbum Ex Machina, Actes De La 13e Conference sur Le Traitement Automatique des Langues naturelles*, pages 20–42, Leuven.

R. Volz, J. Kleb, and W. Mueller. 2007. Towards ontology-based disambiguation of geographical identifiers. In *WWW2007*, Banff, Canada, May 8-12.

M. Wieling, M.J. Nederhof, and G. van Noord. 2006. Parsing partially bracketed input. In Proc. of *CLIN 2005.*

Woordenlijst Nederlandse Taal. 1995. SDU Uitgevers, Den Haag.