Adding semantic role annotation to a corpus of written Dutch

Paola Monachesi, Gerwert Stevens and Jantine Trapman
Utrecht University, Uil-OTS, Trans 10, 3512 JK Utrecht, The Netherlands
{Paola.Monachesi, Gerwert.Stevens, Jantine.Trapman}@let.uu.nl

Abstract

We present an approach to automatic semantic role labeling (SRL) carried out in the context of the Dutch Language Corpus Initiative (D-Coi) project. Adapting earlier research which has mainly focused on English to the Dutch situation poses an interesting challenge especially because there is no semantically annotated Dutch corpus available that can be used as training data. Our automatic SRL approach consists of three steps: bootstrapping from a syntactically annotated corpus by means of a rule-based tagger developed for this purpose, manual correction on the basis of the PropBank guidelines which have been adapted to Dutch and training a machine learning system on the manually corrected data.

1 Introduction

The creation of semantically annotated corpora has lagged dramatically behind. As a result, the need for such resources has now become urgent. Several initiatives have been launched at the international level in the last years, however, they have focused almost entirely on English and not much attention has been dedicated to the creation of semantically annotated Dutch corpora.

Within the *Dutch Language Corpus Initiative* (D-Coi)[1], a recently completed Dutch project, guidelines have been developed for the annotation of a Dutch written corpus. In particular, a pilot corpus has been compiled, parts of which have been enriched with (verified) linguistic annotations.

One of the innovative aspects of the D-Coi project with respect to previous initiatives, such as the Spoken Dutch Corpus (CGN - Corpus Gesproken Nederlands) (Oostdijk, 2002), was the development of a protocol for a semantic annotation layer. In particular, two types of semantic annotation have been addressed, that is semantic role assignment and temporal and spatial semantics (Schuurman and Monachesi, 2006). The reason for this choice lies in the fact that semantic role assignment (i.e. the semantic relationships identified between items in the text such as the agents or patients of particular actions), is one of the most attested and feasible types of semantic annotation within corpora. On the other hand, temporal and spatial annotation was chosen because there is a clear need for such a layer of annotation in applications like information retrieval or question answering.

The focus of this paper is on semantic role annotation. We analyze the choices we have made in selecting an appropriate annotation protocol by taking into consideration existing initiatives such as FrameNet (Johnson et al., 2002) and PropBank (Kingsbury et al., 2002) (cf. also the Chinese and Arabic PropBank). We motivate our choice for the PropBank annotation scheme on the basis of the promising results with respect to automatic semantic role labeling (SRL) which have been obtained for English. Furthermore, we discuss how the SRL research could be adapted to the Dutch situation given that no semantically annotated corpus was available that could be used as training data.

[1] http://lands.let.ru.nl/projects/d-coi/

Proceedings of the Linguistic Annotation Workshop, pages 77–84,
Prague, June 2007.

2 Existing projects

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 our project we have tried to exploit existing results as much as possible and to set the basis for a common standard. We want to profit from earlier experiences and contribute to existing work by making it more complete with our own (language specific) contribution given that most resources have been developed for English.

The PropBank and FrameNet projects have been evaluated 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. Given the results they have achieved, we have taken their insights and experiences as our starting point.

FrameNet reaches a level of granularity in the specification of the semantic roles which might be desirable for certain applications (i.e. Question Answering). Moreover, the predicates are linked to an underlying frame ontology that classifies the verbs within a semantic hierarchy. On the other hand, despite the relevant work of Gildea and Jurafsky (2002), it is still an open issue whether FrameNet classes and frame elements can be obtained and used automatically because of the richness of the semantic structures employed (Dzikovska et al., 2004). Furthermore, the FrameNet approach might raise problems with respect to uniformity of role labeling even if human annotators are involved. Incompleteness, however, constitutes the biggest 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.

In our study, we have assumed that the FrameNet classification even though it is based on English could be applicable to Dutch as well. This assumption is supported by the fact that the German project *Saarbrücken Lexical Semantics Annotation and analysis* (SALSA) (K. Erk and Pinkal, 2003) has adopted FrameNet with good results. Although Dutch and English are quite similar, there are differences on both sides. For example, in the case of the Spanish FrameNet it turned out that frames may differ in their number of elements across languages (cf. (Subirats and Sato, 2004)).

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, in (Monachesi and Trapman, 2006) suggestions are given on how the two approaches could be reconciled.

The prospect of semi-automatic annotation was the decisive factor in the decision to adopt the PropBank approach for the annotation of semantic roles within the D-Coi project.

3 Automatic SRL: bootstrapping a corpus with semantic roles

Ever since the pioneering article of Gildea and Jurafsky (2002), there has been an increasing interest in automatic semantic role labeling (SRL). However, previous research has focused mainly on English. Adapting earlier research to the Dutch situation poses an interesting challenge especially because there is no semantically annotated Dutch corpus available that can be used as training data. Furthermore, no PropBank frame files for Dutch exist.

To solve the problem of the unavailability of training data, we have developed a rule-based tagger to bootstrap a syntactically annotated corpus with semantic roles. After manual correction, this corpus was used as training data for a machine learning SRL system. The input data for our SRL approach consists of Dutch sentences, syntactically annotated by the Dutch dependency parser Alpino (Bouma et al., 2000).

Syntactic annotation of our corpus is based on the Spoken Dutch Corpus (CGN) dependency graphs (Moortgat et al., 2000). A CGN dependency graph is a tree-structured directed acyclic graph in which nodes and edges are labeled with respectively c-labels (category-labels) and d-labels (dependency labels). C-labels of nodes denote phrasal categories, such as NP (noun phrase) and PP, c-labels of leafs denote POS tags. D-Labels describe the grammatical (dependency) relation between the node and its head. Examples of such relations are SU (subject), OBJ (direct object) and MOD (modifier).

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. This mapping forms the basis of our rule-based SRL system (Stevens, 2006).

Mapping subject and object complements to PropBank arguments is straightforward: subjects are mapped to ARG0 (proto-typical agent), direct objects to ARG1 (proto-typical patient) and indirect objects to ARG2. An exception is made for ergatives and passives, for which the subject is labeled with ARG1.

Devising a consistent mapping for higher numbered arguments is more difficult, since their labeling depends in general on the frame entry of the corresponding predicate. Since we could not use frame information, we used a heuristic method. This heuristic strategy entails that after numbering subject/object complements with the rules stated above, other complements are labeled in a left-to-right order, starting with the first available argument number. For example, if the subject is labeled with ARG0 and there are no object complements, the first available argument number is ARG1.

Finally, a mapping for several types of modifiers was defined. We refrained from the disambiguation task, and concentrated on those modifiers that can be mapped consistently. These modifiers are:

- **ArgM-NEG** - Negation markers.
- **ArgM-REC** - Reflexives and reciprocals.
- **ArgM-PRD** - Markers of secondary predication: modifiers with the dependency label PREDM
- **ArgM-PNC** - Purpose clauses: modifiers that start with *om te* . These modifiers are marked by Alpino with the c-label OTI.
- **ArgM-LOC** - Locative modifiers: modifiers with the dependency label LD, the LD label is used by Alpino to mark modifiers that indicate a location of direction.

4 XARA: a rule based SRL tagger

With the help of the mappings discussed above, we developed a rule-based semantic role tagger, which is able to bootstrap an unannotated corpus with semantic roles. We used this rule-based tagger to reduce the manual annotation effort. After all, starting manual annotation from scratch is time consuming and therefore expensive. A possible solution is to start from a (partially) automatically annotated corpus.

The system we developed for this purpose is called XARA (XML-based Automatic Role-labeler for Alpino-trees) (Stevens, 2006). [2] XARA is written in Java, the cornerstone of its rule-based approach is formed by XPath expressions; XPath (Clark and DeRose, 1999) is a powerful query language for the XML format.

The corpus format we used in our experiments is the Alpino XML format. This format is designed to support a range of linguistic queries on the dependency trees in XPath directly (Bouma and Kloosterman, 2002). The structure of Alpino XML documents directly corresponds to the structure of the dependency tree: dependency nodes are represented by NODE elements, attributes of the node elements are the properties of the corresponding dependency node, e.g. c-label, d-label, pos-tag, lemma, etc.

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)
```

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

In this example, a numeric label is used to label a numbered argument. For ARGMs, string value can be used as target label.

After their definition, rules can be applied to local dependency domains, i.e. subtrees of a dependency tree. The local dependency domain to which a rule is applied, is called the rule's context. A context is defined by an XPath expression that selects a group of nodes. Contexts for which we defined rules in XARA are verbal domains, that is, local dependency structures with a verb as head.

Table 1 shows the performance of XARA on our treebank.

Table 1: Results of SRL with XARA

Label	**Precision**	**Recall**	**F**$_{\beta=1}$
Overall	65,11%	45,83%	53,80
Arg0	98.97%	94.95%	96.92
Arg1	70.08%	64.83%	67.35
Arg2	47.41%	36.07%	40.97
Arg3	13.89%	6.85%	9.17
Arg4	1.56%	1.35%	1.45
ArgM-LOC	83.49%	13.75%	23.61
ArgM-NEG	72.79%	58.79%	65.05
ArgM-PNC	91.94%	39.31%	55.07
ArgM-PRD	63.64%	26.25%	37.17
ArgM-REC	85.19%	69.70%	76.67

Notice XARA's performance on highered numbered arguments, especially ARG4. Manual inspection of the manual labeling reveals that ARG4 arguments often occur in propositions without ARG2 and ARG3 arguments. Since our current heuristic labeling method always chooses the first available argument number, this method will have to be modified in order achieve a better score for ARG4 arguments.

5 Manual correction

The annotation by XARA of our tree bank, was manually corrected by one human annotator, however, in order to deal with a Dutch corpus, the PropBank annotation guidelines needed to be revised.

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 non existent and in those few cases in which they are attested, a coindexation 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.

In order to annotate our corpus, the PropBank guidelines needed a revision because they have been developed for English and to add a semantic layer to the Penn TreeBank. Besides the adaption (and extension) of the guidelines to Dutch (Trapman and Monachesi, 2006), we also have to consider a Dutch version of the PropBank frameindex. In PropBank, frame files provide a verb specific description of all possible semantic roles and illustrate these roles by examples. The lack of example sentences makes consistent annotation difficult. Since defining a set of frame files from scratch is very time consuming, we decided to attempt an alternative approach, in which we annotated Dutch verbs with the same argument structure as their English counterparts, thus use English frame files instead of creating Dutch ones. Although this causes some problems, for example, not all Dutch verbs can be translated to a 100% equivalent English counterpart, such problems proved to be relatively rare. In most cases applying the PropBank argument structure to Dutch verbs was straightforward. If translation was not possible, an ad hoc decision was made on how to label the verb.

In order to verify the correctness of the annotation carried out automatically by XARA, we have proceeded in the following way:

1. localize the verb and translate it to English; only the argument structure of *verbs* is considered in our annotation while that of NPs, PPs and other constituents has been neglected for the moment.

2. check the verb's frames file in PropBank; the appropriate roles for each

verb could be identified in PropBank (http://verbs.colorado.edu/framesets/).

3. localize the arguments of the verb; arguments are usually NPs, PPs and sentential complements.

4. localize the modifiers; in addition to the arguments of a verb, modifiers of place, time, manner etc. are marked as well.

An appropriate tool has been selected to carry out the manual correction. We have made an investigation to evaluate three different tools for this purpose: *CLaRK*[3], *Salto*[4] and *TrEd*[5]. On the basis of our main requirements, that is whether the tool is able to handle the xml-structure we have adopted and whether it provides a user-friendly graphical interface and we have come to the conclusion that the *TrEd* tool was the most appropriate for our needs.

During the manual correction process, some problems have emerged, as for example the fact that we have encountered some phenomena, such as the interpretation of modifiers, for which linguistic research doesn't provide a standard solution yet, we have discarded these cases for the moment but it would be desirable to address them in the future.

Furthermore, the interaction among levels should be taken more into consideration. Even though the Alpino parser has an accuracy on our corpus of $81\%-90\%$ (van Noord, 2006) and the syntactic corpus which has been employed for the annotation of the semantic roles had been manually corrected, we have encountered examples in which the annotation provided by the syntactic parser was not appropriate. This is the case of a PP which was labeled as modifier by the syntactic parser but which should be labeled as argument according to the PropBank guidelines. There should thus be an agreement in these cases so that the syntactic structure can be corrected. Furthermore, we have encountered problems with respect to PP attachment, that is the syntactic representation gives us correct and incorrect structures and at the semantic level we are able to disambiguate. More research is necessary about how to deal with the incorrect representations.

[3]http://www.bultreebank.org/clark/index.html

[4]http://www.coli.uni-saarland.de/projects/salsa/

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

6 The TiMBL classification system

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).

TiMBL assigns class labels to training instances on the basis of features and the feature set plays an important role in the performance of a classifier. In choosing the feature set for our system, we mainly looked at previous research, especially systems that participated in the 2004 and 2005 CoNLL shared tasks on semantic role labeling (Carreras and Màrquez, 2005).

It is worth noting that none of the systems in the CoNLL shared tasks used features extracted from dependency structures. However, we encountered one system (Hacioglu, 2004) that did not participate in the CoNLL-shared task but did use the same data and was based on dependency structures. The main difference with our system is that Hacioglu did not use a dependency parser to create the dependency trees, instead existing constituent trees were converted to dependency structures. Furthermore, the system was trained and tested on English sentences.

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.

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

(6) 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.

(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.

The training set consists of predicate/argument pairs encoded in training instances. Each instance contains features of a predicate and its candidate argument. Candidate arguments are nodes (constituents) in the dependency tree. This pair-wise approach is analogous to earlier work by van den Bosch et al. (2004) and Tjong Kim Sang et al. (2005) in which instances were build from verb/phrase pairs from which the phrase parent is an ancestor of the verb. We adopted their approach to dependency trees: only siblings of the verb (predicate) are considered as candidate arguments.

In comparison to experiments in earlier work, we had relatively few training data available: our training corpus consisted of 2,395 sentences which comprise 3066 verbs, 5271 arguments and 3810 modifiers.[6] To overcome our data sparsity problem, we trained the classifier using the leave one out (LOO) method (`-t leave_one_out` option in TiMBL). With this option set, every data item in turn is selected once as a test item, and the classifier is trained on all remaining items. Except for the LOO option, we only used the default TiMBL settings during training, to prevent overfitting because of data sparsity.

7 Results & Evaluation

Table 2 shows the performance of the TiMBL classifier on our annotated dependency treebank. From these sentences, 12,113 instances were extracted. To measure the performance of the systems, the automatically assigned labels were compared to the labels assigned by a human annotator.

Table 2: Results of TiMBL classification

Label	**Precision**	**Recall**	$\mathbf{F}_{\beta=1}$
Overall	70.27%	70.59%	70.43
Arg0	90.44%	86.82%	88.59
Arg1	87.80%	84.63%	86.18
Arg2	63.34%	59.10%	61.15
Arg3	21.21%	19.18%	20.14
Arg4	54.05%	54.05%	54.05
ArgM-ADV	54.98%	51.85%	53.37
ArgM-CAU	47.24%	43.26%	45.16
ArgM-DIR	36.36%	33.33%	34.78
ArgM-DIS	74.27%	70.71%	72.45
ArgM-EXT	29.89%	28.57%	29.21
ArgM-LOC	57.95%	54.53%	56.19
ArgM-MNR	52.07%	47.57%	49.72
ArgM-NEG	68.00%	65.38%	66.67
ArgM-PNC	68.61%	64.83%	66.67
ArgM-PRD	45.45%	40.63%	42.90
ArgM-REC	86.15%	84.85%	85.50
ArgM-TMP	55.95%	53.29%	54.58

It is difficult to compare our results with those obtained with other existing systems, since our system is the first one to be applied to Dutch sentences. Moreover, our data format, data size and evaluation methods (separate test/train/develop sets versus LOO) are different from earlier research. However, to put our results somewhat in perspective, we looked mainly at systems that participated in the CoNLL shared tasks on semantic role labeling.

The best performing system that participated in CoNLL 2005 reached an F_1 of 80. There were seven systems with an F_1 performance in the 75-78 range, seven more with performances in the 70-75 range and five with a performance between 65 and 70 (Carreras and Màrquez, 2005).

A system that did not participate in the CoNLL task, but still provides interesting material for comparison since it is also based on dependency structures, is the system by Hacioglu (2004). This system scored 85,6% precision, 83,6% recall and 84,6 F_1 on the CoNLL data set, which is even higher than the best results published so far on the PropBank data

[6] We refer to (Oostdijk and Boves, 2006) for general information about the domain of the D-Coi corpus and its design.

sets (Pradhan et al., 2005): 84% precision, 75% recall and 79 F_1. These results support our claim that dependency structures can be very useful in the SRL task.

As one would expect, the overall precision and recall scores of the classifier are higher than those of the XARA rule-based system. Yet, we expected a better performance of the classifier on the lower numbered arguments (ARG0 and ARG1). Our hypothesis is that performance on these arguments can be improved by by adding semantic features to our feature set.

Examples of such features are the subcategorization frame of the predicate and the semantic category (e.g. WordNet synset) of the candidate argument. We expect that such semantic features will improve the performance of the classifier for certain types of verbs and arguments, especially the lower numbered arguments ARG0 and ARG1 and temporal and spatial modifiers. For example, the Dutch preposition *over* can either head a phrase indicating a location or a time-span. The semantic category of the neighboring noun phrase might be helpful in such cases to choose the right PropBank label. Thanks to new lexical resources, such as Cornetto (Vossen, 2006), and clustering techniques based on dependency structures (van de Cruys, 2005), we might be able to add lexical semantic information about noun phrases in future research.

Performance of the classifier can also be improved by automatically optimizing the feature set. The optimal set of features for a classifier can be found by employing bi-directional hill climbing (van den Bosch et al., 2004). There is a wrapper script (Paramsearch) available that can be used with TiMBL and several other learning systems that implements this approach[7]. In addition, iterative deepening (ID) can be used as a heuristic way of finding the optimal algorithm parameters for TiMBL.

8 Conclusions & Future work

We have presented an approach to automatic semantic role labeling based on three steps: bootstrapping from a syntactically annotated Dutch corpus with a rule-based tagger developed for this purpose, manual correction and training a machine learning system on the manually corrected data.

The promising results in this area obtained for English on the basis of PropBank role labels was a decisive factor for our choice to adopt the PropBank annotation scheme which has been adapted for the annotation of the Dutch corpus. However, we would like to adopt the conceptual structure of FrameNet, even though not necessarily the granularity of its role assignment approach, to this end we are linking manually the predicates annotated with the PropBank semantic roles to the FrameNet ontology.

Only a small part of the D-Coi corpus has been annotated with semantic information, in order to yield information with respect to its feasibility. We believe that a more substantial annotation task will be carried out in the framework of a follow-up project aiming at the construction of a 500 million word corpus, in which one million words will be annotated with semantic information. Hopefully, in the follow-up project, it will be possible to carry out experiments and measure inter-annotator agreement since due to financial constraints only one annotator has annotated the current corpus.

Finally, it would be interesting to see how the classifier would perform on larger collections and new genres of data. The follow-up of the D-Coi project will provide new semantically annotated data to facilitate research in this area.

References

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

G. Bouma, G. van Noord, and R. Malouf. 2000. Alpino: wide-coverage computational analysis of dutch.

X. Carreras and L. Màrquez. 2005. Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2005)*. Boston, MA, USA.

J. Clark and S. DeRose. 1999. Xml path language (xpath). *W3C Recommendation 16 November 1999*. URL: http://www.w3.org/TR/xpath.

D. Daelemans, D. Zavrel, K. van der Sloot, and A. van den Bosch. 2004. Timbl: Tilburg memory based learner, version 5.1, reference guide. ILK Technical Report Series 04-02, Tilburg University.

[7]URL: http://ilk.uvt.nl/software.html#paramsearch

M. Dzikovska, M. Swift, and J. Allen. 2004. Building a computational lexicon and ontology with framenet. In *Proceedings of the workshop Building Lexical Resources with Semantically Annotated Corpora (LREC) 2004*. Lisbon.

D. Gildea and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Comput. Linguist.*, 28(3):245–288.

K. Hacioglu. 2004. Semantic role labeling using dependency trees. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 1273. August 2004.

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*.

S. Pado K. Erk, A. Kowalski and M. Pinkal. 2003. Towards a resource for lexical semantics: A large german corpus with extensive semantic annotation. In *Proceedings of ACL 2003*. Sapporo.

P. Kingsbury, M. Palmer, and M. Marcus. 2002. Adding semantic annotation to the penn treebank. In *Proceedings of the Human Language Technology Conference (HLT'02)*.

P. Monachesi and J. Trapman. 2006. Merging framenet and propbank in a corpus of written dutch. In *Proceedings of (LREC) 2006*. Genoa.

M. Moortgat, I. Schuurman, and T. van der Wouden. 2000. CGN syntactische annotatie. *Internal report Corpus Gesproken Nederlands*.

N. Oostdijk and L. Boves. 2006. User requirements analysis for the design of a reference corpus of written dutch. In *Proceedings of (LREC) 2006*. Genoa.

N. Oostdijk. 2002. The design of the spoken dutch corpus. In P. Peters, P. Collins, and A. Smith, editors, *New Frontiers of Corpus Research*, pages 105–112. Amsterdam: Rodopi.

S. Pradhan, K., V. Krugler, W. Ward, J. H. Martin, and D. Jurafsky. 2005. Support vector learning for semantic argument classification. *Machine Learning Journal*, 1-3(60):11–39.

E. Tjong Kim Sang, S. Canisius, A. van den Bosch, and T. Bogers. 2005. Applying spelling error correction techniques for improving semantic role labeling. In *Proceedings of the Ninth Conference on Natural Language Learning (CoNLL-2005)*. Ann Arbor, MI, USA.

I. Schuurman and P. Monachesi. 2006. The contours of a semantic annotation scheme for dutch. In *Proceedings of Computational Linguistics in the Netherlands 2005*. University of Amsterdam. Amsterdam.

G. Stevens. 2006. Automatic semantic role labeling in a dutch corpus. Master's thesis, Universiteit Utrecht.

C. Subirats and H. Sato. 2004. Spanish framenet and framesql. In *4th International Conference on Language Resources and Evaluation. Workshop on Building Lexical Resources from Semantically Annotated Corpora*. Lisbon (Portugal), May 2004.

J. Trapman and P. Monachesi. 2006. Manual for the annotation of semantic roles in d-coi. Technical report, University of Utecht.

Tim van de Cruys. 2005. Semantic clustering in dutch. In *Proceedings of CLIN 2005*.

A. van den Bosch, S. Canisius, W. Daelemans, I. Hendrickx, and E. Tjong Kim Sang. 2004. Memory-based semantic role labeling: Optimizing features, algorithm, and output. In H.T. Ng and E. Riloff, editors, *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004)*. Boston, MA, USA.

G. van Noord. 2006. At last parsing is now operational. In *Proceedings of TALN 06*. Leuven.

P. Vossen. 2006. Cornetto: Een lexicaal-semantische database voor taaltechnologie. *Dixit Special Issue*. Stevin.