# A Combined Memory-Based Semantic Role Labeler of English

**Roser Morante, Walter Daelemans, Vincent Van Asch**
CNTS - Language Technology Group
University of Antwerp
Prinsstraat 13, B-2000 Antwerpen, Belgium
{Roser.Morante,Walter.Daelemans,Vincent.VanAsch}@ua.ac.be

## Abstract

We describe the system submitted to the closed challenge of the CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. Syntactic dependencies are processed with the MaltParser 0.4. Semantic dependencies are processed with a combination of memory-based classifiers. The system achieves 78.43 labeled macro F1 for the complete problem, 86.07 labeled attachment score for syntactic dependencies, and 70.51 labeled F1 for semantic dependencies.

## 1 Introduction

In this paper we describe the system submitted to the closed challenge of the CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies (Surdeanu et al., 2008). Compared to the previous shared tasks on semantic role labeling, the innovative feature of this one is that it consists of extracting both syntactic and semantic dependencies. The semantic dependencies task comprises labeling the semantic roles of nouns and verbs and disambiguating the frame of predicates.

The system that we present extracts syntactic and semantic dependencies independently. Syntactic dependencies are processed with the MaltParser 0.4 (Nivre, 2006; Nivre et al., 2007). Semantic dependencies are processed with a combination of memory-based classifiers.

Memory-based language processing (Daelemans and van den Bosch, 2005) is based on the idea that NLP problems can be solved by storing solved examples of the problem in their literal form in memory, and applying similarity-based reasoning on these examples in order to solve new ones. Keeping literal forms in memory has been argued to provide a key advantage over abstracting methods in NLP that ignore exceptions and subregularities (Daelemans et al., 1999).

Memory-based algorithms have been previously applied to semantic role labeling. Van den Bosch et al. (2004) participated in the CoNLL-2004 shared task with a system that extended the basic memory-based learning method with class n-grams, iterative classifier stacking, and automatic output post-processing. Tjong Kim Sang et al. (2005) participated in the CoNLL-2005 shared task with a system that incorporates spelling error correction techniques. Morante and Busser (2007) participated in the SemEval-2007 competition with a semantic role labeler for Spanish based on gold standard constituent syntax. These systems use different types of constituent syntax (shallow parsing, full parsing). We are aware of two systems that perform semantic role labeling based on dependency syntax previous to the CoNLL-2008 shared task. Hacioglu (2004) converts the data from the CoNLL-2004 shared task into dependency trees and uses support vector machines. Morante (2008) describes a memory-based semantic role labeling system for Spanish based on gold standard dependency syntax.

We developed a memory-based system for the CoNLL-2008 shared task in order to evaluate the performance of this methodology in a completely new semantic role labeling setting.

The paper is organised as follows. In Section 2 the system is described, Section 3 contains an analysis of the results, and Section 4 puts forward some





*CoNLL 2008: Proceedings of the 12th Conference on Computational Natural Language Learning,* pages 208–212
Manchester, August 2008

conclusions.

## 2 System description

The system processes syntactic and semantic dependencies independently. The syntactic dependencies are processed with the MaltParser 0.4. The semantic dependencies are processed with a cascade of memory-based classifiers. We use the IB1 classifier as implemented in TiMBL (version 6.1.2) (Daelemans et al., 2007), a supervised inductive algorithm for learning classification tasks based on the $k$-nearest neighbor classification rule (Cover and Hart, 1967). In IB1, similarity is defined by computing (weighted) overlap of the feature values of a test instance and a memorized example. The metric combines a per-feature value distance metric with global feature weights that account for relative differences in discriminative power of the features.

### 2.1 Syntactic dependencies

The MaltParser 0.4[1] (Nivre, 2006; Nivre et al., 2007) is an inductive dependency parser that uses four essential components: a deterministic algorithm for building labeled projective dependency graphs; history-based feature models for predicting the next parser action; support vector machines for mapping histories to parser actions; and graph transformations for recovering non-projective structures.

The learner type used was support vector machines, with the same parameter options reported by (Nivre et al., 2006). The parser algorithm used was Nivre, with the options and model (eng.par) for English as specified on http://w3.msi.vxu.se/users/jha/conll07/. The tagset.pos, tagset.cpos and tagset.dep were extracted from the training corpus.

### 2.2 Semantic dependencies

The semantics task consists of finding the predicates, assigning a PropBank or a NomBank frame to them and extracting their semantic role dependencies. Because of lack of resources, we did not have time to develop a word sense disambiguation system. So, predicates were assigned the frame '.01' by default.

The system handles the semantic role labeling task in three steps: predicate identification, semantic dependency classification, and combination of classifiers.

#### 2.2.1 Predicate identification

In this phase, a classifier predicts if a word is a predicate or not. The IB1 algorithm was parameterised by using overlap as the similarity metric, information gain for feature weighting, using 7 $k$-nearest neighbors, and weighting the class vote of neighbors as a function of their inverse linear distance. The instances represent all nouns and verbs in the corpus and they have the following features:

- Word form, lemma, part of speech (POS), the three last letters of the word, and the lemma and POS of the five previous and five next words. To obtain the previous word we perform a linear left-to-right search. This is how *previous* has to be interpreted further on when features are described.

The accuracy of the classifier on the development test is 0.9599 (4240/4417) for verbs and 0.8981 (9226/10272) for nouns.

#### 2.2.2 Semantic dependency classification

In this phase, three groups of multi-class classifiers predict in one step if there is a dependency between a word and a predicate, and the type of dependency, i.e. semantic role.

**Group 1 (G1)** consists of two classifiers: one for predicates that are nouns and another for predicates that are verbs. The instances represent a predicate-word combination. The predicates are those that have been classified as such in the previous phase. As for the combining words, determiners and certain combinations are excluded based on the fact that they never have a role in the training corpus.

The IB1 algorithm was parameterised by using overlap as the similarity metric, information gain for feature weighting, using 11 $k$-nearest neighbors, and weighting the class vote of neighbors as a function of their inverse linear distance. The features of the noun classifier are:

- About the predicate: word form. About the combining word: word form, POS, dependency type, word form of the two previous and two next words. Chain of POS types between the word and the predicate. Distance between the word and the predicate. Binary feature indicating if the word depends on the predicate. Six chains of POS tags between the word and its three previous and three next predicates in relation to the current predicate.

[1] Web page of MaltParser 0.4: http://w3.msi.vxu.se/∼nivre/research/MaltParser.html.

The features of the verb classifier are:

- The same as for the noun classifier and additionally: POS of the word next to the current combining word, binary feature indicating if the combining word depends on the predicate previous to the current predicate, binary feature indicating if the predicate previous to the combining word is located before or after the current predicate.

The verb classifier achieves an overall accuracy of 0.9244 (80805/87412), and the noun classifier, 0.9173 (69836/76132) in the development set.

**Group 2 (G2)** consists also of two classifiers: one for predicates that are nouns and another for predicates that are verbs. The instances represent combinations of word-predicate, but the test corpus contains only those instances that G1 has classified as having a role.

The IB1 algorithm was parameterised in the same way as for G1, except that it computes 7 $k$-nearest neighbors instead of 11. The two classifiers use the same features:

- About the predicate: word form, chain of lemmas of the syntactic siblings, chain of lemmas of the syntactic children. About the combining word: word form, POS, dependency type, word form of the two previous and the two next words, POS+type of dependency and lemma of the syntactic father, chain of dependency types and chain of lemmas of the syntactic children. Chain of POS types between word and predicate, distance and syntactic dependency type between word and predicate.

The verb classifier achieves an overall accuracy of 0.5656 (4160/7355), and the noun classifier, 0.5017 (2234/4452) in the development set.

**Group 3 (G3)** consists of one classifier. Like G2, instances represent combinations of word-predicate, but the test corpus contains only those instances that G1 has classified as having a role.. The IB1 algorithm was parameterised in the same way as for G2. It uses the following features:

About the predicate: lemma, POS, POS of the 3 previous and 3 next predicates. About the combining word: lemma, POS, and dependency type, POS of the 3 previous and 3 next words. Distance between the predicate and the word. A binary feature indicating if the combining word is located before or after the predicate.

The classifier achieves an overall accuracy of 0.5527 (6526/11807).

#### 2.2.3 Combination of classifiers

In this phase the three groups of classifiers are combined in a simple way: if G2 and G3 agree in classifying a semantic dependency, their solution is chosen, else the solution of G1 is chosen. This system combination choice is explained by the fact that G1 has a higher accuracy than G2 and G3 when the three classifiers are applied to the development set. G2 and G3 are used to eliminate overgeneration of roles by G1.

The performance of the system in the development corpus with only the G1 classifiers is 66.07 labeled F1. The combined system achieves a 10.8% error reduction, with 69.75 labeled F1.

## 3 Results

The results of the system are shown in Table 1. We will focus on commenting on the semantic scores. The system scores 71.88 labeled F1 in the in-domain corpus (WSJ) and 59.23 in the out-of-domain corpus (Brown). Unlabeled F1 in the WSJ corpus is almost 10% higher than labeled F1. Labeled precision is 12.40% higher than labeled recall.

| | WSJ | BROWN |
|---|---|---|
| **SYNTACTIC SCORES** | | |
| Labeled attachment score | 86.88 | 79.58 |
| Unlabeled attachment score | 89.37 | 84.85 |
| Label accuracy score | 91.48 | 86.00 |
| **SEMANTIC SCORES** | | |
| Labeled precision | 78.61 | 65.25 |
| Labeled recall | 66.21 | 54.23 |
| Labeled F1 | 71.88 | 59.23 |
| Unlabeled precision | 89.13 | 83.61 |
| Unlabeled recall | 75.08 | 69.48 |
| Unlabeled F1 | 81.50 | 75.89 |
| **OVERALL MACRO SCORES** | | |
| Labeled macro precision | 82.74 | 72.41 |
| Labeled macro recall | 76.54 | 66.90 |
| Labeled macro F1 | 79.52 | 69.55 |
| Unlabeled macro precision | 89.25 | 84.23 |
| Unlabeled macro recall | 82.22 | 77.16 |
| Unlabeled macro F1 | 85.59 | 80.54 |

Table 1: Results of the system in the WSJ and BROWN corpora expressed in %.

### 3.1 Discussion

The performance of the semantic role labeler is affected considerably by the performance of the first classifier for predicate detection. The system cannot recover from the predicates that are missed in this phase. Experiments without the first classifier and with gold standard predicates (detection and classification) result in 80.89 labeled F1, 9.01 %

higher than the results of the system with predicate detection. We opted for identifying predicates as a first step in order to reduce the number of training instances for the second phase, classification of semantic dependencies. For the same reason, we opted for selecting only nouns and verbs as instances, aware of the fact that we would miss a very low number of predicates with other categories. The results of predicate identification can be improved by setting up a combined system, instead of a single classifier, and by incorporating a system for frame disambiguation.

Equally important would be to find better features for the identification of noun predicates, since the features used generalise better for verbs than for nouns. Table 2 shows that the system is better at identifying verbs than it is at identifying nouns.

| | Total | F1 Pred. Id.&Cl. | F1 Pred. Id. |
|---|---|---|---|
| CC | 3 | - | - |
| CD | 1 | - | - |
| IN | 3 | - | - |
| JJ | 16 | - | - |
| NN | 3635 | 77.57 | 85.59 |
| NNP | 10 | 30.77 | 38.46 |
| NNS | 1648 | 75.47 | 83.65 |
| PDT | 2 | - | - |
| RP | 4 | - | - |
| VB | 1278 | 79.28 | 98.87 |
| VBD | 1320 | 85.44 | 99.24 |
| VBG | 742 | 77.05 | 94.41 |
| VBN | 985 | 76.43 | 92.08 |
| VBP | 343 | 78.60 | 97.81 |
| VBZ | 504 | 80.94 | 97.36 |
| WP | 2 | - | - |
| WRB | 2 | - | - |

Table 2: Predicate (Pred.) identification (Id.) and classification (Cl.) in the WSJ corpus expressed in %.

A characteristic of the semantic role labeler is that recall is considerably lower than precision (12.40 %). This can be further analysed with the data shown in Table 3.

Except for the dependency VB*+AM-NEG, precision is higher than recall for all semantic dependencies. We run the semantic role labeler with gold standard predicates and with gold standard syntax and predicates. The difference between precision and recall is around 10 % in both cases, which confirms that low recall is a characteristic of the semantic role labeler, probably caused by the fact that the features do not generalise good enough. The semantic role labeler with gold standard predicates scores 86.06 % labeled precision and 76.32 % labeled recall. The semantic role labeler with gold standard predicates and syntax scores 89.20 % precision and 79.47 % recall.

| Dependency | Total | Recall | Prec. | F1 |
|---|---|---|---|---|
| NN*+A0 | 2339 | 42.41 | 77.80 | 54.90 |
| NN*+A1 | 3757 | 61.17 | 78.32 | 68.69 |
| NN*+A2 | 1537 | 45.48 | 82.24 | 58.57 |
| NN*+A3 | 349 | 50.14 | 88.38 | 63.98 |
| NN*+AM-ADV | 32 | 9.38 | 37.50 | 15.01 |
| NN*+AM-EXT | 33 | 18.18 | 85.71 | 30.00 |
| NN*+AM-LOC | 232 | 30.60 | 63.96 | 41.40 |
| NN*+AM-MNR | 344 | 34.59 | 79.87 | 48.27 |
| NN*+AM-NEG | 35 | 2.86 | 100.00 | 5.56 |
| NN*+AM-TMP | 492 | 54.88 | 83.33 | 66.18 |
| VB*+A0 | 3509 | 68.99 | 82.63 | 75.20 |
| VB*+A1 | 4844 | 74.24 | 83.28 | 78.50 |
| VB*+A2 | 1085 | 55.94 | 69.21 | 61.87 |
| VB*+A3 | 169 | 41.42 | 79.55 | 54.48 |
| VB*+A4 | 99 | 74.75 | 88.10 | 80.88 |
| VB*+AM-ADV | 488 | 38.93 | 59.19 | 46.97 |
| VB*+AM-CAU | 70 | 50.00 | 70.00 | 58.33 |
| VB*+AM-DIR | 81 | 29.63 | 57.14 | 39.02 |
| VB*+AM-DIS | 315 | 52.70 | 74.11 | 61.60 |
| VB*+AM-EXT | 32 | 50.00 | 59.26 | 54.24 |
| VB*+AM-LOC | 355 | 52.11 | 57.10 | 54.49 |
| VB*+AM-MNR | 335 | 46.57 | 61.18 | 52.88 |
| VB*+AM-MOD | 539 | 92.21 | 95.95 | 94.04 |
| VB*+AM-NEG | 227 | 94.71 | 90.34 | 92.47 |
| VB*+AM-PNC | 113 | 33.63 | 54.29 | 41.53 |
| VB*+AM-TMP | 1068 | 64.51 | 80.40 | 71.58 |
| VB*+C-A1 | 192 | 65.10 | 74.85 | 69.64 |
| VB*+R-A0 | 222 | 65.77 | 87.43 | 75.07 |
| VB*+R-A1 | 155 | 49.68 | 73.33 | 59.23 |
| VB*+R-AM-LOC | 21 | 23.81 | 71.43 | 35.71 |
| VB*+R-AM-TMP | 52 | 46.15 | 66.67 | 54.54 |

Table 3: Semantic dependencies identification and classification in the WSJ corpus for dependencies with more than 20 occurences expressed in %.

Table 3 also shows that the unbalance between precision and recall is higher for dependencies of nouns than for dependencies of verbs, and that both recall and precision are higher for dependencies from verbs. Thus, the system performs better for verbs than for nouns. This is in part caused by the fact that more noun predicates than verb predicates are missed in the predicate identification phase. The scores of the the semantic role labeler with gold standard predicates show lower differences in F1 between verbs and nouns.

The fact that the semantic role labeler performs 3.16 % labeled F1 better with gold standard syntax (compared to the system with gold standard syntax and predicates) confirms that gold standard syntax provides useful information to the system.

Additionally, the difference in performance between the semantic role labeler presented to the

competition and the semantic role labeler with gold standard predicates (9.01 % labeled F1) suggests that, although the results of the system are encouraging, there is room for improvement, and improvement should focus on increasing the recall scores.

## 4 Conclusions

In this paper we have presented a system submitted to the closed challenge of the CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. We have focused on describing the part of the system that extracts semantic dependencies, a combination of memory-based classifiers. The system achieves a semantic score of 71,88 labeled F1. Results show that the system is considerably affected by the first phase of predicate identification, that the system is better at extracting the semantic dependencies of verbs than those of nouns, and that recall is substantially lower than precision. These facts suggest that, although the results are encouraging, there is room for improvement.

## 5 Acknowledgements

This work was made possible through financial support from the University of Antwerp (GOA project BIOGRAPH), and from the Flemish Institute for the Promotion of Innovation by Science and Technology Flanders (IWT) (TETRA project GRAVITAL). The experiments were carried out in the CalcUA computing facilities. We are grateful to Stefan Becuwe for his support.

## References

Cover, T. M. and P. E. Hart. 1967. Nearest neighbor pattern classification. *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, 13:21–27.

Daelemans, W. and A. van den Bosch. 2005. *Memory-based language processing.* Cambridge University Press, Cambridge, UK.

Daelemans, W., A. Van den Bosch, and J. Zavrel. 1999. Forgetting exceptions is harmful in language learning. *Machine Learning, Special issue on Natural Language Learning*, 34:11–41.

Daelemans, W., J. Zavrel, K. Van der Sloot, and A. Van den Bosch. 2007. TiMBL: Tilburg memory based learner, version 6.1, reference guide. Technical Report Series 07-07, ILK, Tilburg, The Netherlands.

Hacioglu, K. 2004. Semantic role labeling using dependency trees. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, Morristown, NJ, USA. ACL.

Morante, R. and B. Busser. 2007. ILK2: Semantic role labelling for Catalan and Spanish using TiMBL. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, pages 183–186.

Morante, R. 2008. Semantic role labeling tools trained on the Cast3LB-CoNLL-SemRol corpus. In *Proceedings of the LREC 2008*, Marrakech, Morocco.

Nivre, J., J. Hall, J. Nilsson, G. Eryigit, and S. Marinov. 2006. Labeled pseudo–projective dependency parsing with support vector machines. In *Proceedings of the Tenth Conference on Computational Natural Language Learning, CoNLL-X*, New York City, NY, June.

Nivre, J., J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi. 2007. MaltParser: a language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Nivre, J. 2006. *Inductive Dependency Parsing*. Springer.

Surdeanu, Mihai, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the 12th Conference on Computational Natural Language Learning (CoNLL-2008)*.

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

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