Semantic and Syntactic Features for Dutch Coreference Resolution

Iris Hendrickx[1], Veronique Hoste[2], and Walter Daelemans[1]

[1] CNTS - Language Technology Group,
University of Antwerp, prinsstraat 13, Antwerp
Belgium
iris.hendrickx@ua.ac.be, walter.daelemans@ua.ac.be
[2] LT3 - Language and Translation Technology Team,
University College Ghent, Groot-Brittaniëlaan 45, Ghent,
Belgium
veronique.hoste@hogent.be

Abstract. We investigate the effect of encoding additional semantic and syntactic information sources in a classification-based machine learning approach to the task of coreference resolution for Dutch. We experiment both with a memory-based learning approach and a maximum entropy modeling method.

As an alternative to using external lexical resources, such as the low-coverage Dutch EuroWordNet, we evaluate the effect of automatically generated semantic clusters as information source. We compare these clusters, which group together semantically similar nouns, to two semantic features based on EuroWordNet encoding synonym and hypernym relations between nouns.

The syntactic function of the anaphor and antecedent in the sentence can be an important clue for resolving coreferential relations. As baseline approach, we encode syntactic information as predicted by a memory-based shallow parser in a set of features. We contrast these shallow parse based features with features encoding richer syntactic information from a dependency parser. We show that using both the additional semantic information and syntactic information lead to small but significant performance improvement of our coreference resolution approach.

1 Introduction

Coreference resolution is the task of resolving different descriptions of the same underlying entity in a given text. Written and spoken texts contain a large number of coreferential relations and a good text understanding largely depends on the correct resolution of these relations. Resolving ambiguous referents in a text can be a helpful preprocessing step for many NLP applications such as text summarization or question answering.

As an alternative to the knowledge-based approaches, in which there has been an evolution from the systems which require an extensive amount of linguistic and non-linguistic information (e.g. [1]) toward more knowledge-poor approaches (e.g. [2]), machine learning approaches have become increasingly popular for this problem. Most of the machine learning approaches (e.g. [3], [4], [5]) are classification-based approaches which use a two-step procedure. This approach requires a corpus annotated with coreferential links between NPs. Next, instances are created between every NP (candidate anaphor) and all of its preceding NPs (candidate antecedents). The first step involves the classification of each pair of NPs as coreferential or not. In a second step, coreferential chains are built on the basis of the positively classified instances. In order to overcome this two-step procedure problem, others such as [6] recently proposed to use features over sets of noun phrases instead of features of pairs of noun phrases.

Most of the current machine learning approaches to coreference resolution use a combination of lexical, positional, syntactic and semantic information sources. Current systems can resolve part of the coreference relations using shallow features, but some cases need deeper linguistic or world knowledge to be resolved, such as for example the referring expressing **House** in the example below.

> **The US House of Representatives** has passed a bill which would fund military operations in Iraq to the end of July. Further funding would be dependent on events in Iraq meeting certain, as yet undefined, benchmarks of progress. President Bush has already vetoed one Iraq funding bill and said he opposed the new proposal, but did say that the idea of benchmarks "made sense". The move came as **the White House** and Democrats struck an accord on standards for bilateral free trade deals. The deal was announced by **House** Speaker Nancy Pelosi, a Democrat, who hailed it as a result of the Democratic triumph in last year's congressional elections.

In this study, we investigate the integration of two semantic sources and a syntactic information source for Dutch coreference resolution. Given the lack of broad-coverage lexical resources for Dutch, we investigate automatically generated semantic clusters [7] to model the semantic classes of NPs. We study the effect of using this information and we compare its effect to the use of two other semantic features based on the Dutch EuroWordNet [8]. Secondly, we investigate the effect of adding features extracted from full parsing in our coreference application for Dutch and we contrast this full-parsing based approach with a shallow parse based approach.

The remainder of this paper is structured as follows. Section 2 gives an overview of the related literature on this topic. Section 3 gives a general overview of the system architecture and in Section 4 and 5 we discuss the construction of the semantic and syntactic features. Section 6 describes the experimental setup, whereas results and conclusions are presented in Sections 7 and 8.

2 Related Work

In the last years, we can observe an increased interest in the use of semantic resources for coreference resolution. Especially WordNet [9] has been and remains a very useful information source for coreference resolution [10–14]. In the last years we observe an increased interest in the integration of additional semantic sources. [15], for example, code semantic information as semantic relations based on the ACE relation ontology relations such as 'membership' and show the beneficial effect on coreference resolution. [13] study the effect of three semantic sources, viz. WordNet, taxonomies extracted from Wikipedia and semantic role labeling and show that these semantic features improve their system. [16] and [17] explore several semantic information sources such as ACE semantic classes and a thesaurus expressing semantic similarity created by [18]. [19] investigate the extraction of automatically discovered patterns which express semantic relatedness information for coreference resolution.

If we consider the use of syntactic features in the existing machine learning systems, we can observe that many systems use some form of shallow syntactic features such as [4, 14]. Some systems also look at deeper syntactic information sources. We will briefly describe three of them. [20] explore syntactic features extracted from dependency parse trees for English, Arabic and Chinese. Part of these features are inspired by the binding theory. They find significant improvements for English and Arabic but not for Chinese. [21] look at predicate-argument structure statistics but found no improvement for the task of pronoun resolution for English. [22] successfully explore the use of parse trees as a structural feature in a kernel-based method for pronoun resolution.

3 Architecture

The first phase of our supervised machine learning approach to coreference resolution is training a classifier on the annotated documents. We start with transforming the annotated documents into training instances. First, the raw texts are preprocessed to determine the noun phrases in the text and to produce information about these nouns. The following preprocessing steps were taken. First, tokenisation was done to split punctuation from adjoining words. For the recognition of names in the text, a memory-based named entity recognition approach [23] was used, which distinguishes between persons, organizations and locations. Part-of-speech tagging and text chunking was performed by the memory-based tagger MBT [24] trained on the Spoken Dutch Corpus (http://lands.let.ru.nl/cgn). Finally, grammatical relation finding was performed to determine grammatical relations between chunks, e.g. subject, object, etc. [25].

On the basis of the preprocessed texts, training instances are created. After the detection of the NPs by the text chunker, every NP is linked to its preceding NPs, with a restriction of 20 sentences backwards. A pair of NPs that belongs to the same coreferential chain, gets a positive label; all other pairs get a negative label. To limit the instance set size we restrict the search scope to 3 sentences for pronominal anaphors and for noun pairs which do not share the same head. For each pair, a feature vector is created to describe the NPs and their relation. These instances are the training set for the classifier.

A combination of different information sources can be used to predict coreferential relations between noun phrases. For our coreference resolution system, we used a combination of positional features (features indicating the number of sentences/NPs between the anaphor and its possible antecedent), morphological and lexical features (such as features which indicate whether a given anaphor, its candidate antecedent or both are pronouns, proper nouns, demonstrative or definite NPs), syntactic features which inform on the syntactic function of the anaphor and its candidate antecedent and check for syntactic parallelism, string-matching features which look for complete and partial matches and finally several semantic features. For the construction of these semantic features, we took into account lists with location names, male and female person names. Furthermore, we looked for female/male pronouns and for gender indicators such as 'Mr.', 'Mrs.' and 'Ms.'. One feature also looked at the named entity type (organization, person, location) of both NPs. Further information was also extracted from the Dutch EuroWordNet synonym and hypernym relations, which we will describe in the following section.

4 Semantic Information Sources

Semantic information can be an important clue to determine whether two referents point to the same entity. For Dutch there are few sources available to obtain semantic knowledge about words. One well-known source is the Dutch part of EuroWordNet [8], a multilingual lexical database. EuroWordNet has approximately 46K entries for Dutch nouns.

We use EuroWordNet to construct two binary features **is_synonym** and **is_hypernym**. These features code for every pair of referents whether their descriptions can be found in EuroWordNet in some synonym or hypernym relation[3]. In case of ambiguous words, we check for all senses of the word.

As a second source we use semantic clusters [7]. These clusters were extracted with unsupervised k-means clustering on the Twente Nieuws Corpus[4], a corpus containing Dutch news paper text. The corpus was first preprocessed by the Alpino parser [26] to extract syntactic relations. The top-10,000 lemmatized nouns including names were clustered into a 1000 groups based on the similarity of their syntactic relations. Table 1 shows four clusters extracted from the Twente Nieuws corpus. These clusters contain both common nouns and names.

For each pair of referents we construct three features as follows. For each referent the lemma of the head word is looked up in the list of clusters. We construct a binary feature marking whether the head words of the referents occur in the same cluster (**same_cluster**) and two features (**cluster1, cluster2**) presenting the cluster number of each referent or zero otherwise. The observation that a potential anaphor is member of a particular cluster may not be informative. However combinations of certain cluster numbers can be informative. For

[3] Two referents with complete string match are also considered as synonyms and hypernyms.

[4] Available from: http://wwwhome.cs.utwente.nl/~druid/TwNC/TwNC-main.html

Table 1. Four semantic clusters extracted with unsupervised k-means clustering. The first column of numbers presents the names of the clusters.

201	{barrière belemmering drempel hindernis hobbel horde knelpunt obstakel struikelblok} (English: barrier impediment threshold hindrance bump hurdle bottleneck obstacle block)
223	{biertje borrel cocktail cola drankje glaasje kopje pilsje} (English: beer booze cocktail cola drink glass cup brew)
320	{Andreotti Berlusconi Bildt Carl_Bildt Craxi Gajdar Jegor_Gajdar Lubbers Martens Margaret_Thatcher Ruud_Lubbers Silvio_Berlusconi Thatcher}
395	{ambtgenoot collega expremier leider minister minister-president opvolger oud-premier partijgenoot premier president vice-premier} (English: fellow colleague ex-premier leader minister Prime_Minister former_premier political_associate premier president vice-president)

example an anaphor "minister-president" is member of cluster 320 in Table 1. A potential antecedent "Margaret Thatcher" is a member of cluster 395. The combination of these two feature values can give a strong clue for a coreferential relation.

To get an insight in the impact of these semantic features, we calculated the percentages of instances in which a particular semantic feature has a non-zero value, shown in Table 2. Only 3.4% of the instances describes a coreferential relation. We computed the percentages on the full set of instances and on the small subset of positive instances[5]. Looking at the full instance set in the first column of the table, the WordNet features are only active in 2% of the instances. But looking at the subset of positive instances, the percentages increase to 36%. This increase implies a clear correlation between the positive class and the active WordNet features.

We also observe an increase for the same_cluster feature. The cluster1 or cluster2 feature are active in 60% of the instances of the full set. On the positive class subset, the percentages drop to 35-37%. This can be explained by the fact that the percentage of pronouns is relatively higher in the subset of positive instances, and pronouns get a zero as cluster value. We also measured to what extent the WordNet feature and the same_cluster feature overlap. In the full instance set 41% of the instances for which the same_cluster is active, has also a positive is_synonym feature. This low percentage of overlap confirm that the two semantic sources cover different parts of the instance space.

5 Syntactic Information

Another important clue for resolving coreferential relations is the syntactic function of the anaphor and antecedent in the sentence. We code syntactic informa-

[5] computed at 90% training part of our data set containing 327,728 instances, and 11,062 positive instances.

Table 2. Percentage of instances in which each semantic feature is active, computed at both the full set of instances and the small subset of the positive class instances.

feature	% inst	% positive inst
is_synonym	2.2	36.4
is_hypernym	2.3	36.1
cluster1	60.1	35.0
cluster2	59.0	37.2
same_cluster	2.3	17.6

tion as predicted by the memory-based shallow parser in our feature set as described in Section 3. We investigate whether the richer syntactic information of a full parser would be a helpful information source for our task. We use the Alpino parser [26], an automatic broad-coverage dependency parser for Dutch to generate the following 11 additional features:

Named Entity label as produced by the Alpino parser, one for the anaphor and one for the antecedent.

Number agreement between the anaphor and antecedent, presented as a four valued feature (values: *sg, pl, both, measurable_nouns*).

Dependency labels as predicted for (the head word of) the anaphor and for the antecedent.

Same dependency label the case that both anaphor and antecedent have the same dependency label is coded as a binary feature.

Dependency path between the governing verb and the anaphor, and between the verb and antecedent.

Clause information is coded as two binary features, is the anaphor / antecedent part of the main clause or not.

Root overlap binary feature that codes overlap between 'roots' or lemmas of the anaphor and antecedent. In the Alpino parser, the root of a noun phrase is the form without inflections. Special cases are compounds and names. Compounds are split and we use the last element in the comparison. For names we take the complete strings.

Next we give an example of these features. The sentence in Example 1 contains a coreferential link between the anaphor "het bedrijf" (the company) and the name "Ford Genk". We list the features as predicted by Alpino. An obvious error is the named entity label of the antecedent, which should have been labeled as 'organization'.

Example 1.
Algemeen directeur Jan Gijsen van Ford Genk maakt bekend dat het bedrijf de volgende twee jaar 1400 banen wil schrappen.
(*English: Head director Jan Gijsen of Ford Genk announces that the company will cut 1400 jobs in the next two years.*)

1. named entity label anaphor: noun
2. named entity label antecedent: person-male
3. number agreement: both (anaphor is singular, antecedent labeled as both)
4. dependency label anaphor: subject
5. dependency label antecedent: object1
6. label match: no
7. dependency path anaphor: [[schrap,hd/su],[wil,hd/su]]
8. dependency path antecedent: [[maak_bekend,hd/su,directeur,hd/mod,van,hd/obj1]]
9. clause anaphor: not in main clause
10. clause antecedent: is in main clause
11. root overlap: no

6 Experimental Setup

We use a Dutch corpus of Flemish news articles, KNACK-2002, annotated with coreference information for NPs [27]. In a first experiment we evaluate the effect of the two semantic sources described in Section 4. We run four experiments with the feature set combinations with and without the WordNet- or cluster-based features. The feature set size varies from 42 features (without WordNet- and cluster-based features) to 47 (with both types of features).

We compare two different machine learning algorithms; memory-based learning [28] and maximum entropy modeling [29]. We use the Timbl software package [30] as our implementation of memory-based learning. For maximum entropy modeling we use the implementation Maxent [31].

In a second experiment we add the features extracted from the Alpino parser output described in Section 5 to the full feature set of 47 features including both types of semantic sources. As the information in these features may largely overlap with the information already presented in the features produced by the memory-based shallow parser, we decided to use genetic algorithms to automatically select an optimal feature selection. Genetic algorithms (GA) have been proposed [32] as an useful method to find an optimal setting in the enormous search space of possible parameter and feature set combinations. We run experiments with a generational genetic algorithm for feature set and algorithm parameter selection of Timbl with 30 generations and a population size of 10. As a comparison we run the GA for both the instance set with vectors of 47 features and for the set with 59 features.

The standard approach to evaluate a coreference resolution system is to compare the predictions of the system to a hand-annotated gold standard test set in cross-validation experiments. The performance of the system can be measured at two levels. One can evaluate the performance of the classifier and determine how well it predicted the presence of a coreference relation for a pair of NPs. In this case, we measure the precision, recall and F-score of the labeled positive NP pairs. We will denote this as evaluation at the instance level. One can also evaluate the construction of the complete coreference chains which can be measured with the MUC scoring software from Vilain et al. [33].

In each experiment we use ten-fold cross validation on 242 documents of KNACK-2002 with both Timbl and Maxent. The GA optimization is done for Timbl and not for Maxent. Timbl is more sensitive to feature redundancy than Maxent as Maxent performs feature weighting internally. The GA is run on the first fold of the ten fold, as running the GA is rather time-consuming. The found optimal setting was also used for the other folds. We also compute a baseline score for the evaluation of the complete coreference chains. The baseline assigns each NP in the test set its most nearby NP as antecedent.

7 Results

The results of the evaluation of the effect of the semantic information sources are shown in Table 3 and 4. Each column presents the results of one of the feature set variations with and without the WordNet features or the cluster-based features. Table 3 presents the micro-averaged F-scores measured at the instance level for Timbl and Maxent. For Timbl adding the WordNet features does not really show any effect, while adding the cluster-based features does show a small improvement. For Maxent adding the WordNet features or the cluster-based features separately gives a small drop in performance. Combining both features has a stronger effect and improves the F-score of Maxent with 1%. The MUC scores presented in Table 4 show the same trends.

Table 3. Micro-averaged F-score computed in 10-fold cross validation experiments for Timbl and Maxent with various feature set variations measured at the instance level.

	−WordNet −cluster	+WordNet −cluster	−WordNet +cluster	+WordNet +cluster
Timbl	46.45	46.43	47.11	47.45
Maxent	49.20	48.71	48.77	49.94

Table 4. Average MUC F-scores computed in 10-fold cross validation experiments for Timbl and Maxent with various feature set variations.

	−WordNet −cluster	+WordNet −cluster	−WordNet +cluster	+WordNet +cluster
Timbl	44.6	44.6	45.6	45.6
Maxent	45.9	45.5	45.7	46.7

The results of our second experiment in which we evaluate the effect of adding features derived from the output of a dependency parser are shown in Table 5 (F-scores at the instance level) and 6 (MUC scores at the chain level).[6]

A first observation is the improvement given by the GA optimization for Timbl. Timbl with 47 features and default algorithmic parameters setting reaches a F-score of 47.45% (Table 3), with optimized settings the F-score of Timbl improves to 54.8% (Table 5).

The differences in F-score at the instance level are small as shown in Table 5. When we look at the score computed at the chain level, we see an improvement of 3% in F-score for Timbl and 1% for Maxent. For Timbl adding the additional features improves the recall at the cost of precision. For Maxent on the other hand both precision and recall are improved by adding the extra features.

Table 5. Micro-averaged F-score and accuracy computed in 10 fold cross validation experiments. Timbl is run with the settings as selected by the genetic algorithm, Maxent with all features.

	recall	precision	F-score	accuracy
TIMBL, GA, 47 features	44.8	70.5	54.8	97.6
TIMBL, GA, 59 features	48.4	64.1	55.1	97.4
MAXENT, 47 features	39.9	66.6	49.9	97.4
MAXENT, 59 features	40.0	68.6	50.5	97.4

Table 6. MUC-scores computed in 10 fold cross validation experiments. Timbl is run with the settings as selected by the genetic algorithm, Maxent with all features.

	recall	precision	F-score
baseline	81.1	24.0	37.0
TIMBL, GA, 47 features	36.8	70.2	48.2
TIMBL, GA, 59 features	44.0	61.4	51.3
MAXENT, 47 features	35.7	67.2	46.7
MAXENT, 59 features	36.8	68.0	47.6

8 Conclusions

We have shown that both the semantic sources and the syntactic information are useful features for our coreference resolution module. We tested these information sources with two different classifiers, memory-based learning and maximum entropy modeling. We evaluated the effect of two types of semantic information

[6] Note that the F-score of Maxent with 47 features shown in the third row of Table 5 is a repetition of F-score the last cell of Table 3.

sources, namely information extracted from WordNet and information extracted from unsupervised learned semantic clusters. Our experiments showed that for Maxent, adding one semantic source can slightly decrease the performance. However, combining the WordNet- and cluster-based features gives a small positive effect for both classifiers. In a second experiment we added features derived from a dependency parser to the feature set. The effect of these additional features is marginal when measured at the instance level, but we do see a small improvement when we evaluate on complete coreference chains.

9 Acknowledgments

This research is funded by the Dutch-Flemish NTU/STEVIN programme (url: http://taalunieversum.org/stevin). We would like to thank Tim Van de Cruys for sharing his data sets of semantic clusters and Gosse Bouma for running the Alpino parser.

References

1. Rich, E., LuperFoy, S.: An architecture for anaphora resolution. In: Proceedings of the Second Conference on Applied Natural Language Processing. (1988) 18–24
2. Mitkov, R.: Robust pronoun resolution with limited knowledge. In: Proceedings of the 17th International Conference on Computational Linguistics (COLING-1998/ACL-1998). (1998) 869–875
3. McCarthy, J.: A Trainable Approach to Coreference Resolution for Information Extraction. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst MA (1996)
4. Soon, W., Ng, H., Lim, D.: A machine learning approach to coreference resolution of noun phrases. Computational Linguistics **27**(4) (2001) 521–544
5. Ng, V., Cardie, C.: Combining sample selection and error-driven pruning for machine learning of coreference rules. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002). (2002) 55–62
6. Culotta, A., Wick, M., Hall, R., McCallum, A.: First-order probabilistic models for coreference resolution. In: Proceedings of HLT/NAACL. (2007) 81–88
7. Van de Cruys, T.: Semantic clustering in dutch. In: Proceedings of the Sixteenth Computational Linguistics in the Netherlands (CLIN). (2005) 17–32
8. Vossen, P., ed.: EuroWordNet: a multilingual database with lexical semantic networks. Kluwer Academic Publishers, Norwell, MA, USA (1998)
9. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998)
10. Poesio, M., Mehta, R., Maroudas, A., Hitzeman, J.: Learning to resolve bridging references. In: Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04). (2004) 143–150
11. Harabagiu, S., Bunescu, R., Maiorano, S.: Text and knowledge mining for coreference resolution. In: Proceedings of the 2nd Meeting of the North American Chapter of the Association of Computational Linguistics (NAACL-2001). (2001) 55–62
12. Markert, K., Nissim, M.: Comparing knowledge sources for nominal anaphora resolution. Computational Linguistics **31**(3) (2005) 367–401

13. Ponzetto, S.P., Strube, M.: Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In: Proceedings of the Human Language Technology Conference of the NAACL, Main Conference. (2006) 192–199
14. Ng, V., Cardie, C.: Improving machine learning approaches to coreference resolution. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002). (2002) 104–111
15. Ji, H., Westbrook, D., Grishman, R.: Using semantic relations to refine coreference decisions. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. (2005) 17–24
16. Ng, V.: Semantic class induction and coreference resolution. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Association for Computational Linguistics (2007) 536–543
17. Ng, V.: Shallow semantics for coreference resolution. In: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07). (2007) 1689–1694
18. Lin, D.: Automatic retrieval and clustering of similar words. In: COLING-ACL. (1998) 768–774
19. Yang, X., Su, J.: Coreference resolution using semantic relatedness information from automatically discovered patterns. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. (2007) 528–535
20. Luo, X., Zitouni, I.: Multi-lingual coreference resolution with syntactic features. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. (2005) 660–667
21. Kehler, A., Appelt, D., Taylor, L., Simma, A.: The (non)utility of predicate-argument frequencies for pronoun interpretation. In: Proceedings of HLT-NAACL. (2004) 289–296
22. Yang, X., Su, J., Tan, C.L.: Kernel-based pronoun resolution with structured syntactic knowledge. In: Proceedings of the 21st International Conference on Computational Linguistics. (2006) 41–48
23. Tjong Kim Sang, E.: Memory-based named entity recognition. In: Proceedings of CoNLL-2002, Taipei, Taiwan (2002) 203–206
24. Daelemans, W., Zavrel, J., van den Bosch, A., van der Sloot, K.: Memory based tagger, version 2.0, reference guide. Technical Report ILK Technical Report - ILK 03-13, Tilburg University (2003)
25. Tjong Kim Sang, E., Daelemans, W., Höthker, A.: Reduction of dutch sentences for automatic subtitling. In: Computational Linguistics in the Netherlands 2003. Selected Papers from the Fourteenth CLIN Meeting. (2004) 109–123
26. Bouma, G., van Noord, G., Malouf, R.: Alpino: Wide-coverage computational analysis of dutch. In: Computational Linguistics in The Netherlands 2000. (2001)
27. Hoste, V., de Pauw, G.: Knack-2002: a richly annotated corpus of dutch written text. In: The fifth international conference on Language Resources and Evaluation (LREC). (2006)
28. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. Institute of Electrical and Electronics Engineers Transactions on Information Theory **13** (1967) 21–27
29. Berger, A., Della Pietra, S., Della Pietra, V.: Maximum Entropy Approach to Natural Language Processing. Computational linguistics **22**(1) (1996)
30. Daelemans, W., Zavrel, J., Van der Sloot, K., Van den Bosch, A.: TiMBL: Tilburg Memory Based Learner, version 5.1, reference manual. Technical Report ILK-0402, ILK, Tilburg University (2004)

31. Le, Z.: Maximum Entropy Modeling Toolkit for Python and C++ (version 20041229). Natural Language Processing Lab, Northeastern University, China. (2004)
32. Daelemans, W., Hoste, V., De Meulder, F., Naudts, B.: Combined optimization of feature selection and algorithm parameter interaction in machine learning of language. In: Proceedings of the ECML. (2003) 84–95
33. Vilain, M., Burger, J., Aberdeen, J., Connolly, D., Hirschman, L.: A model-theoretic coreference scoring scheme. In: Proceedings of the Sixth Message Understanding Conference (MUC-6). (1995) 45–52