ILK: Machine learning of semantic relations with shallow features and almost no data

Iris Hendrickx
CNTS / Language Technology Group
University of Antwerp,
Universiteitsplein 1
2610 Wilrijk, Belgium
iris.hendrickx@ua.ac.be

Roser Morante, Caroline Sporleder,
Antal van den Bosch
ILK / Communication and Information Sciences
Tilburg University, P.O. Box 90153,
5000 LE Tilburg, The Netherlands
{R.Morante,C.Sporleder,
Antal.vdnBosch}@uvt.nl

Abstract

This paper summarizes our approach to the Semeval 2007 shared task on "Classification of Semantic Relations between Nominals". Our overall strategy is to develop machine-learning classifiers making use of a few easily computable and effective features, selected independently for each classifier in wrapper experiments. We train two types of classifiers for each of the seven relations: with and without any WordNet information.

1 Introduction

We interpret the task of determining semantic relations between nominals as a classification problem that can be solved, per relation, by machine learning algorithms. We aim at using straightforward features that are easy to compute and relevant to preferably all of the seven relations central to the task.

The starting conditions of the task provide us with a very small amount of training data, which further stresses the need for robust, generalizable features, that generalize beyond surface words. We therefore hypothesize that generic information on the lexical semantics of the entities involved in the relation is crucial. We developed two systems, based on two sources of semantic information. Since the entities in the provided data were word-sense disambiguated, an obvious way to model their lexical semantics was by utilizing WordNet3.0 (Fellbaum, 1998) (WN). One of the systems followed this route.

We also entered a second system, which did not rely on WN but instead made use of automatically generated semantic clusters (Decadt and Daelemans, 2004) to model the semantic classes of the entities. For both systems we trained seven binary classifiers; one for each relation. From a pool of easily computable features, we selected feature subsets for each classifier in a number of wrapper experiments, i.e. repeated cross-validation experiments on the training set to test out subset selections systematically. Along with feature subsets we also chose the machine-learning method independently for each classifier.

Section 2 presents the system description, Section 3, the results, and Section 4, the conclusions.

2 System Description

The development of the system consists of a preprocessing phase to extract the features, and the classification phase.

2.1 Preprocessing

Each sentence is preprocessed automatically in the following steps. First, the sentence is tokenized with a rule-based tokenizer. Next a part-of-speech tagger and text chunker that use the memory-based tagger MBT (Daelemans et al., 1996) produces part-of-speech tags and NP chunk labels for each token. Then a memory-based shallow parser predicts grammatical relations between verbs and NP chunks such as subject, object or modifier (Buchholz, 2002). The tagger, chunker and parser were all trained on the WSJ Corpus (Marcus et al., 1993). Each token was also lemmatized with a memory-based lemmatizer (Van den Bosch et al., 1996) which is robust to handle unknown or new words such as "labrador".

The features extracted are of three types: semantic, lexical, and morpho-syntactic. The features that apply to the entities in a relation (e1,e2) are extracted for term 1 (t1) and term 2 (t2) of the relation, where t1 is the first term in the relation name, and t2 is the second term. For example, in the relation Cause–Effect, t1 is Cause and t2 is Effect.

The semantic features are the following:

WN semantic class of t1 and t2. The WN semantic class of each entity in the relation. For the WN-based system, we determined the semantic class of the entities on the basis of the lexicographer file numbers (LFN) in WN3.0. The LFN are encoded in the synset number provided in the annotation of the data. For nouns there are 25 file numbers that correspond to suitably abstract semantic classes, namely:

> noun.Tops(top concepts for nouns), act, animal, artifact, attribute, body, cognition, communication event, feeling, food, group, location, motive, object, person, phenomenon, plant, possession, process, quantity, relation, shape, state, substance, time.

Is_container (is_C). Exclusively for the Content–Container relation we furthermore included two binary features that test whether the two entities in the relation are hyponyms of the synset *Container* in WN. For the Part–Whole relation we also experimented with binary features expressing whether the two entities in the relation have some type of meronym and holonym relation, but these features did not prove to be predictive.

Cluster class of t1 and t2. A cluster class identifier for each entity in the relation. This information is drawn from automatically generated clusters of semantically similar nouns (Decadt and Daelemans, 2004) generated on the British National Corpus (Clear, 1993). The corpus was first preprocessed by a lemmatizer and the memory-based shallow parser, and the found verb–object relations were used to cluster nouns in groups. We used the top-5000 lemmatized nouns, that are clustered into 250 groups. This is an example of two of these clusters:

- {can pot basin tray glass container bottle tin pan mug cup jar bowl bucket plate jug vase kettle}
- {booth restaurant bath kitchen hallway toilet bedroom hall suite bathroom interior lounge shower compartment oven lavatory room}

The lexical features are the following:

Lemma of t1 and t2 (lem1, lem2). The lemmas of the entities involved in the relation. In case an entity consisted of multiple words (e.g. *storage room*) we use the lemma of the head noun (i.e. *room*).

Main verb (verb). The main verb of the sentence in which the entities involved in the relation appear, as predicted by the shallow parser.

The morpho-syntactic features are:

GramRel (gr1, gr2). The grammatical relation tags of the entities.

Suffixes of t1 and t2 (suf1, suf2). The suffixes of the entity lemmas. We implemented a rule-based suffix guesser, which determines whether the nouns involved in the relation end in a derivational suffix, such as *-ee*, *-ment* etc. Suffixes often provide cues for semantic properties of the entities. For example, the suffix *-ee* usually indicates animate (and typically human) referents (e.g. *detainee* etc.), whereas (*-ment*) points at abstract entities (e.g. *statement*).

While the features were selected independently for all relations, the seven classifiers in the WN-based system all make use of the WN semantic class features; in the system that did not use WN, the seven classifiers make use of the cluster class features instead.

2.2 Classification

We experimented with several machine learning frameworks and different feature (sub-)sets. For rapid testing of different learners and feature sets, and given the size of the training data (140 examples for each relation), we made use of the Weka machine learning software[1] (Witten and Frank, 1999). We systematically tested the following algorithms: NaiveBayes (NB) (Langley et al., 1992), BayesNet (BN) (Cooper and Herskovits, 1992), J48 (Quinlan, 1993), Jrip (Cohen,), IB1 and IBk (Aha et al., 1991), LWL (Atkeson et al., 1997), and DecisionStumps (DS) (Iba and Langley, 1992), all with default algorithm settings.

The classifiers for all seven relations were optimized independently in a number of 10-fold cross-validation (CV) experiments on the provided train-

[1] `http://www.cs.waikato.ac.nz/ml/weka/`

ing sets. Several feature subsets were tried, varying from all features to only the two 'base' features (WN-or cluster-class). The feature sets and learning algorithms which were found to obtain the highest accuracies for each relation were then used when applying the classifiers to the unseen test data.

The classifiers of the cluster-based system (A) all use the two cluster class features. The other selected features and the chosen algorithms (CL) are displayed in Table 1. Knowledge of the identity of the lemmas was found to be beneficial for all classifiers. With respect to the machine learning framework, Naive Bayes was selected most frequently.

Relation	CL	lem1	lem2	verb	gr1	gr2	suf1	suf2
Cause–Effect	DS	+	+	+	+	+	+	+
Instr–Agency	LWL	+	+	+	+	+		
Product–Producer	NB	+	+	+	+	+	+	+
Origin–Entity	IBk	+	+		+	+	+	+
Theme–Tool	NB	+	+	+			+	+
Part–Whole	NB	+	+		+	+	+	+
Content–Container	NB	+	+		+	+	+	+

Table 1: The final selected algorithms and features for each relation by the cluster-based system (A).

The classifiers of the WN-based system (B) all use at least the WN semantic class features. Table 2 shows the other selected features and algorithm for each relation. None of the classifiers use all the features. For the Part–Whole relation no extra features besides the WN class are selected. Also the classifiers for the relations Cause–Effect and Content–Container only use two additional features. The list of best found algorithms shows that – like with the cluster-based system – a Bayesian approach is favorable, as it is selected in four of seven cases.

Relation	CL	lem1	lem2	verb	gr1	gr2	suf1	suf2	is_C
Cause–Effect	BN						+	+	
Instr–Agency	NB	+	+			+			
Product–Producer	IB1	+	+	+		+			
Origin–Entity	IBk	+	+		+	+	+		
Theme–Tool	NB	+	+	+	+		+	+	
Part–Whole	J48								
Content–Container	BN			+					+

Table 2: The final selected algorithms and features for each relation by the WN-based system (B). (*is_C* is the Content–Container specific feature.)

3 Results

In Table 3 we first present the best results computed on the training set using 10-fold CV for the cluster-based system (A) and the WN-based system (B). These results are generally higher than the official test set results, shown in Tables 4 and 5, possibly showing a certain amount of overfitting on the training sets.

Relation	A	B
Cause–Effect	56.4	72.9
Instrument–Agency	71.4	75.7
Product–Producer	65.0	67.9
Origin–Entity	70.7	78.6
Theme–Tool	75.7	79.3
Part–Whole	65.7	73.6
Content–Container	70.0	75.4
Avg	67.9	74.8

Table 3: Average accuracy on the **training** set computed in 10-fold CV experiments of the cluster-based system (A) and the WN-based system (B).

The official scores on the test set are computed by the task organizers: accuracy, precision, recall and F_1 measure. Table 4 presents the results of the cluster-based system. Table 5 presents the results of the WN-based system. (The column *Total* shows the number of instances in the test set.) Markable is the high accuracy for the Part–Whole relation as the classifier was only trained on two features coding the WN classes.

A4	Pre	Rec	F	Acc	Total
Cause–Effect	53.3	97.6	69.0	55.0	80
Instrument–Agency	56.1	60.5	58.2	57.7	78
Product–Producer	69.1	75.8	72.3	61.3	93
Origin–Entity	60.7	47.2	53.1	63.0	81
Theme–Tool	64.5	69.0	66.7	71.8	71
Part–Whole	48.4	57.7	52.6	62.5	72
Content–Container	71.4	78.9	75.0	73.0	74
Avg	60.5	69.5	63.8	63.5	78.4

Table 4: Test scores for the seven relations of the cluster-based system trained on 140 examples (A4).

The system using all training data with WordNet features, B4 (Table 5), performs better in terms of F-measure on six out of the seven subtasks as compared to the system that does not use the WordNet features but the semantic cluster information instead, A4 (Table 4). This is largely due to a lower precision of the A4 system. The WordNet features

B4	Pre	Rec	F	Acc	Total
Cause–Effect	69.0	70.7	69.9	68.8	80
Instrument–Agency	69.8	78.9	74.1	73.1	78
Product–Producer	79.7	75.8	77.7	71.0	93
Origin–Entity	71.0	61.1	65.7	71.6	81
Theme–Tool	69.0	69.0	69.0	74.6	71
Part–Whole	73.1	73.1	73.1	80.6	72
Content–Container	78.1	65.8	71.4	73.0	74
Avg	72.8	70.6	71.5	73.2	78.4

Table 5: Test scores for the seven relations of the WN-based system trained on 140 examples (B4).

appear to be directly responsible for a relatively higher precision.

In contrast, the semantic cluster features of system A sometimes boost recall. A4's recall on the Cause–Effect relation is 97.6% (the classifier predicts the class 'true' for 75 of the 80 examples), and on Content–Container the system attains 78.9%, markedly better than B4.

4 Conclusion

We have shown that a machine learning approach using shallow and easily computable features performs quite well on this task. The system using WordNet features based on the provided disambiguated word senses outperforms the cluster-based system. It would be interesting to compare both systems to a more realistic WN-based system that uses predicted word senses by a Word Sense Disambiguation system.

However we end by noting that the amount of training and test data in this shared task should be considered too small to base any reliable conclusions on. In a realistic scenario (e.g. when high-precision relation classification would be needed as a component of a question-answering system), more training material would have been gathered, and the examples would not have been seeded by a limited number of queries – especially the negative examples are very artificial now due to their similarity to the positive cases, and the fact that they are down-sampled very unrealistically. Rather, the focus of the task should be on detecting positive instances of the relations in vast amounts of text (i.e. vast amounts of implicit negative examples). Positive training examples should be as randomly sampled from raw text as possible. The seven relations are common enough to warrant a focused effort to annotate a reasonable amount of randomly selected text, gathering several hundreds of positive cases of each relation.

References

D. W. Aha, D. Kibler, M. Albert. 1991. Instance-based learning algorithms. *Machine Learning*, 6:37–66.

C. Atkeson, A. Moore, S. Schaal. 1997. Locally weighted learning. *Artificial Intelligence Review*, 11(1–5):11–73.

S. Buchholz. 2002. *Memory-Based Grammatical Relation Finding*. PhD thesis, University of Tilburg.

J. H. Clear. 1993. *The British national corpus*. MIT Press, Cambridge, MA, USA.

W. Cohen. Fast effective rule induction. In *Proceedings of the 12th International Conference on Machine Learning*. Morgan Kaufmann.

G. F. Cooper, E. Herskovits. 1992. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347.

W. Daelemans, J. Zavrel, P. Berck, S. Gillis. 1996. Mbt: A memory-based part of speech tagger generator. In *Proceedings of the 4th ACL/SIGDAT Workshop on Very Large Corpora*, 14–27.

B. Decadt, W. Daelemans. 2004. Verb classification - machine learning experiments in classifying verbs into semantic classes. In *Proceedings of the LREC 2004 Workshop Beyond Named Entity Recognition: Semantic Labeling for NLP Tasks*, 25–30.

C. Fellbaum, ed. 1998. *WordNet: An Electronic Database*. MIT Press, Cambridge, MA.

W. Iba, P. Langley. 1992. Induction of one-level decision trees. *Proceedings of the Ninth International Conference on Machine Learning*, 233–240.

P. Langley, W. Iba, K. Thompson. 1992. An analysis of Bayesian classifiers. In *Proceedings of the Tenth Annual Conference on Artificial Intelligence*, 223–228.

M. Marcus, S. Santorini, M. Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

J. Quinlan. 1993. C4.5*: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.

A. Van den Bosch, W. Daelemans, A. Weijters. 1996. Morphological analysis as classification: an inductive-learning approach. In *Proceedings of the Second International Conference on New Methods in Natural Language Processing*, 79–89.

I. H. Witten, E. Frank. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufman.