TUD: semantic relatedness for relation classification

György Szarvas* and **Iryna Gurevych**
Ubiquitous Knowledge Processing (UKP) Lab
Computer Science Department
Technische Universität Darmstadt
Hochschulstraße 10., D-64289 Darmstadt, Germany
http://www.ukp.tu-darmstadt.de/

Abstract

In this paper, we describe the system submitted by the team TUD to Task 8 at SemEval 2010. The challenge focused on the identification of semantic relations between pairs of nominals in sentences collected from the web. We applied maximum entropy classification using both lexical and syntactic features to describe the nominals and their context. In addition, we experimented with features describing the semantic relatedness (SR) between the target nominals and a set of clue words characteristic to the relations. Our best submission with SR features achieved 69.23% macro-averaged F-measure, providing 8.73% improvement over our baseline system. Thus, we think SR can serve as a natural way to incorporate external knowledge to relation classification.

1 Introduction

Automatic extraction of typed semantic relations between sentence constituents is an important step towards deep semantic analysis and understanding the semantic content of natural language texts. Identification of relations between a nominal and the main verb, and between pairs of nominals are important steps for the extraction of structured semantic information from text, and can benefit various applications ranging from Information Extraction and Information Retrieval to Machine Translation or Question Answering.

The Multi-Way Classification of Semantic Relations Between Pairs of Nominals challenge (Hendrickx et al., 2010) focused on the identification of specific relation types between nominals (nouns or base noun phrases) in natural language sentences collected from the web. The main task of the challenge was to identify and classify instances of 9 abstract semantic relations between noun phrases, i.e. *Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection, Message-Topic*. That is, given two nominals (*e1* and *e2*) in a sentence, systems had to decide whether *relation(e1,e2)*, *relation(e2,e1)* holds for one of the relation types or the nominals' relation is *other* (falls to a category not listed above or they are unrelated). In this sense, the challenge was an important pilot task towards large scale semantic processing of text.

In this paper, we describe the system we submitted to Semeval 2010, Task 8. We applied maximum entropy classification to the problem using both lexical and contextual features to describe the nominals themselves and their context (i.e. the sentence). In addition, we experimented with features exploiting the strength of association between the target nominals and a predefined set of clue words characteristic to the nine relation types. In order to measure the semantic relatedness (SR) of targets and clues, we used the Explicit Semantic Analysis (Gabrilovich and Markovitch, 2007) SR measure (based on Wikipedia, Wiktionary and WordNet). Our best submission, benefiting from SR features, achieved 69.23% macro-averaged F-measure for the 9 relation types used. Providing 8.73% improvement over our baseline system, we found the SR-based features to be beneficial for the classification of semantic relations.

2 Experimental setup

2.1 Feature set and selection

Feature set In our system, we used both lexical (1-3) and contextual features (4-8) to describe the nominals and their context (i.e. the sentence). Additionally, we experimented with a set of features (9) that exploit the co-occurrence statistics of the

* On leave from the Research Group on Artificial Intelligence of the Hungarian Academy of Sciences

Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pages 210–213,
Uppsala, Sweden, 15-16 July 2010.

nominals and a set of clue words chosen manually, examining the relation definitions and examples provided by the organizers. The clues characterize the relations addressed in the task (e.g. *cargo, goods, content, box, bottle* characterize the *Content-Container* relation)[1]. Each feature type was distinguished from the others using a prefix. All but the semantic relatedness features we used were binary, denoting whether a specific word, lemma, POS tag, etc. is found in the example sentence, or not. SR features were real valued, scaled to $[0, 1]$ for each clue word separately (on train, and the same scaling factores were applied on the test data). The feature types used:

1. *Token:* word unigrams in the sentence in their inflected form. ***2.*** *Lemma:* word uni- and bigrams in the sentence in their lemmatized form. ***3.*** *Target Nouns:* the syntactic head words of the target nouns. ***4.*** *POS:* the part of speech uni- and bi- and trigrams in the sentence. ***5.*** *Between POS:* the part of speech sequence between the target nouns. ***6.*** *Dependency Path:* the dependency path (syntactic relations and directions) between the target nouns. The whole path constituted a single feature. ***7.*** *Target Distance:* the distance between the target nouns (in tokens). ***8.*** *Sentence Length:* the length of the sentence (in tokens). ***9.*** *Semantic Relatedness:* the semantic relatedness scores measuring the strength of association between the target nominals and the set of clue words we collected. In order to measure the semantic relatedness (SR) of targets and clues, we used the Explicit Semantic Analysis (ESA) SR measure.

Feature selection In order to discard uninformative features automatically, we performed feature selection on the binary features. We kept features that satisfied the following three conditions:

$$freq(x) > 3 \quad (1)$$

$$p = argmax_y P(y|x) > t_1 \quad (2)$$

$$p^5 \times freq(x) > t_2 \quad (3)$$

where $freq(x)$ denotes the frequency of feature x observed in the training dataset, y denotes a class label, p denotes the highest posterior probability (for feature x) over the nine relations (undirected) and the *other* class. Finally, t_1, t_2 are filtering thresholds chosen arbitrarily. We used $t_1 = 0.25$ for all features but the dependency path, where we used $t_1 = 0.2$. We set the thresold t_2 to 1.9 for lexical features (i.e. token and lemma features), to 0.3 for dependency path features and to 0.9 for all other features. All parameters for the feature selection process were chosen manually (cross-validating the parameters was omitted due to lack of time during the challenge development period). The higher t_2 value for lexical features was motivated by the aim to avoid overfitting, and the lower thresholds for dependency-based features by the hypothesis that these can be most efficient to determine the direction of relationships (c.f. we disregarded direction during feature selection). As the numeric SR features were all bound to clue words selected specifically for the task, we did not perform any feature selection for that feature type.

[1]The clue list is available at: http://www.ukp.tu-darmstadt.de/research/data/relation-classification/

relation type	size	c4.5	SMO	maxent
cause-effect	1003	75.2%	**78.9%**	78.2%
component-whole	941	46.7%	53.0%	**54.7%**
content-container	540	72.9%	**78.1%**	75.1%
entity-destination	845	77.6%	**82.3%**	82.0%
entity-origin	716	61.8%	65.0%	**68.7%**
instrument-agency	534	40.7%	42.7%	**47.6%**
member-collection	690	68.2%	72.1%	**75.3%**
message-topic	634	41.3%	47.3%	**56.4%**
product-producer	717	43.8%	50.3%	**53.4%**
macro AVG F1	6590	58.7%	63.3%	**65.7%**

Table 1: Performance of different learning methods on train (10-fold).

2.2 Learning models

We compared three learning algorithms, using the baseline feature types (1-8), namely a C4.5 decision tree learner, a support vector classifier (SMO), and a maximum entropy (logistic regression) classifier, all implemented in the Weka package (Hall et al., 2009). We trained the SMO model with polynomial kernel of degree 2, fitting logistic models to the output to get valid probability estimates and the C4.5 model with pruning confidence factor set to 0.33. All other parameters were set to their default values as defined in Weka. We found the maxent model to perform best in 10-fold cross validation on the training set (see Table 1). Thus, we used maxent in our submissions.

3 Results

We submitted 4 runs to the challenge. Table 2 shows the per-class and the macro average F-measures of the 9 relation classes and the accuracy over all classes including *other*, on the train (10-fold) and the test sets (official evaluation):

	Train				Test			
relation type	**Base**	**WP**	**cSR**	**cSR-t**	**Base**	**WP**	**cSR**	**cSR-t**
cause-effect	78.17%	78.25%	**79.42%**	79.10%	80.69%	81.90%	**83.76%**	83.38%
component-whole	54.68%	58.71%	60.18%	**60.79%**	50.52%	57.90%	61.67%	**62.15%**
content-container	75.09%	77.55%	**78.26%**	78.11%	75.27%	**78.96%**	78.33%	78.87%
entity-destination	81.99%	82.97%	**83.12%**	82.90%	77.59%	**82.86%**	81.54%	81.12%
entity-origin	68.74%	70.39%	71.14%	**71.18%**	67.08%	**72.05%**	71.03%	70.36%
instrument-agency	47.59%	56.71%	59.60%	**59.80%**	31.09%	44.06%	46.78%	**46.91%**
member-collection	75.27%	79.43%	80.71%	**80.89%**	66.37%	71.24%	**72.65%**	**72.65%**
message-topic	56.40%	62.68%	64.77%	**65.15%**	49.88%	65.06%	68.15%	**69.83%**
product-producer	53.36%	57.98%	59.97%	**60.70%**	46.04%	**57.94%**	56.00%	**57.85%**
macro AVG F1	65.70%	69.40%	70.80%	**70.96%**	60.50%	68.00%	68.88%	**69.23%**
accuracy (incl. *other*)	62.10%	65.42%	66.83%	**67.12%**	56.13%	63.49%	64.63%	**65.37%**

Table 2: Performance of 4 submissions on train (10-fold) and test.

Baseline (Base) As our baseline system, we used the information extracted from the sentence itself (i.e. lexical and contextual features, types 1-8).

Wikipedia (WP) As a first extension, we added SR features (9) exploiting term co-occurrence information, using the ESA model with Wikipedia.

Combined Semantic Relatedness (cSR) Second, we replaced the ESA measure with a combined measure developed by us, exploiting term co-occurrence not only in Wikipedia, but also in WordNet and Wiktionary glosses. We found this measure to perform better than the Wikipedia-based ESA in earlier experiments.

cSR threshold (cSR-t) We submitted the predictions of the cSR system, with less emphasis on the *other* class: we predicted *other* label only when the following held for the posteriors predicted by cSR: $\frac{argmax_y P(y|x)}{p(other)} < 0.7$. The threshold 0.7 was chosen based on the training dataset.

First, the SR features improved the performance of our system by a wide marging (see Table 2). The difference in performance is even more prominent on the Test dataset, which suggests that these features efficiently incorporated useful external evidence on the relation between the nominals and this not just improved the accuracy of the system, but also helped to avoid overfitting. Thus we conclude that the SR features with the encoded external knowledge helped the maxent model to learn a hypothesis that clearly generalized better.

Second, we notice that the combined SR measure proved to be more useful than the standard ESA measure (Gabrilovich and Markovitch, 2007) improving the performance by approximately 1 percent over ESA, both in terms of macro averaged F-measure and overall accuracy. This confirms our hypothesis that the combined measure is more robust than ESA with just Wikipedia.

prediction category	**cSR**	**cSR-t**
true positive relation (TP)	1555	1612
true positive *other* (TN)	201	164
wrong relation type (FP & FN)	291	341
wrong relation direction (FP & FN)	50	58
relation classified as *other* (FN)	367	252
other classified as relation (FP)	253	290
total	2717	2717

Table 3: Prediction error statistics.

3.1 Error Analysis

Table 3 shows the breakdown of system predictions to different categories, and their contribution to the official ranking as true/false positives and negatives. The submission that manipulated the decision threshold for the *other* class improved the overall performance by a small margin. This fact, and Table 3 confirm that our approach had major difficulties in correctly discriminating the 9 relation categories from *other*. Since this class is an umbrella class for unrelated nominals and the numerous semantic relations not considered in the challenge, it proved to be extremely difficult to accurately characterize this class. On the other hand, the confusion of the 9 specified relations (between each other) and directionality were less prominent error types. The most frequent cross-relation confusion types were the misclassification of *Component-Whole* as *Instrument-Agency* and *Member-Collection*; *Content-Container* as *Component-Whole*; *Instrument-Agency* as *Product-Producer* and vice versa. Interestingly, *Component-Whole* and *Cause-Effect* relations were the most typical sources for wrong direction errors. Lowering the decision threshold for *other* in our system naturally resulted in more true positive relation classifications, but unfortunately not only raised the number of *other* instances falsely classified as being one of the valuable re-

lations, but also introduced several wrong relation classification errors (see Table 3). That is why this step resulted only in marginal improvement.

4 Conclusions & Future Work

In this paper, we presented our system submitted to the Multi-Way Classification of Semantic Relations Between Pairs of Nominals challenge at SemEval 2010. We submitted 4 different system runs. Our first submission was a baseline system (Base) exploiting lexical and contextual information collected solely from the sentence to be classified. A second run (WP) complemented this baseline configuration with a set of features that used Explicit Semantic Analysis (Wikipedia) to model the SR of the nominals to be classified and a set of clue words characteristic of the relations used in the challenge. Our third run (cSR) used a combined semantic relatedness measure that exploits multiple lexical semantic resources (Wikipedia, Wiktionary and WordNet) to provide more reliable relatedness estimates. Our final run (cSR-t) exploited that our system in general was inaccurate in predicting instances of the *other* class. Thus, it used the same predictions as cSR, but favored the prediction of one of the 9 specified classes instead of *other*, when a comparably high posterior for such a class was predicted by the system.

Our approach is fairly simple, in the sense that it used mostly just local information collected from the sentence. It is clear though that encoding as much general world knowledge to the representation as possible is crucial for efficient classification of semantic relations. In the light of the above fact, the results we obtained are reasonable.

As the main goal of our study, we attempted to use semantic relatedness features that exploit texts in an external knowledge source (Wikipedia, Wiktionary or WordNet in our case) to incorporate some world knowledge in the form of term co-occurrence scores. We found that our SR features significantly contribute to system performance. Thus, we think this kind of information is useful in general for relation classification. The experimental results showed that our combined SR measure performed better than the standard ESA using Wikipedia. This confirms our hypothesis that exploiting multiple resources for modeling term relatedness is beneficial in general.

Obviously, our system leaves much space for improvement – the feature selection parameters and the clue word set for the SR features were chosen manually, without any cross-validation (on the training set), due to lack of time. One of the participating teams used an SVM-based system and gained a lot from manipulating the decision thresholds. Thus, despite our preliminary results, it is also an interesting option to use SVMs.

In general, we think that more features are needed to achieve significantly better performance than we reported here. Top performing systems in the challenge typically exploited web frequency information (n-gram data) and manually encoded relations from an ontology (mainly WordNet). Thus, future work is to incorporate such features.

We demonstrated that SR features are helpful to move away from lexicalized systems using token- or lemma-based features. Probably the same holds for web-based and ontology-based features extensively used by top performing systems. This suggests that experimenting with all these to see if their value is complementary is an especially interesting piece of future work.

Acknowledgments

This work was supported by the German Ministry of Education and Research (BMBF) under grant 'Semantics- and Emotion-Based Conversation Management in Customer Support (SIGMUND)', No. 01ISO8042D, and by the Volkswagen Foundation as part of the Lichtenberg-Professorship Program under the grant No. I/82806.

References

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of The 20th International Joint Conference on Artificial Intelligence*, pages 1606–1611.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1):10–18.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th SIGLEX Workshop on Semantic Evaluation*.