UCB: System Description for SemEval Task #4

Preslav I. Nakov
EECS, CS division
University of California at Berkeley
Berkeley, CA 94720
nakov@cs.berkeley.edu

Marti A. Hearst
School of Information
University of California at Berkeley
Berkeley, CA 94720
hearst@ischool.berkeley.edu

Abstract

The UC Berkeley team participated in the SemEval 2007 Task #4, with an approach that leverages the vast size of the Web in order to build lexically-specific features. The idea is to determine which verbs, prepositions, and conjunctions are used in sentences containing a target word pair, and to compare those to features extracted for other word pairs in order to determine which are most similar. By combining these Web features with words from the sentence context, our team was able to achieve the best results for systems of category C and third best for systems of category A.

1 Introduction

Semantic relation classification is an important but understudied language problem arising in many NLP applications, including question answering, information retrieval, machine translation, word sense disambiguation, information extraction, etc. This year's *SemEval* (previously *SensEval*) competition has included a task targeting the important special case of *Classification of Semantic Relations between Nominals*. In the present paper we describe the UCB system which took part in that competition.

The *SemEval* dataset contains a total of 7 semantic relations (not exhaustive and possibly overlapping), with 140 training and about 70 testing sentences per relation. Sentence classes are approximately 50% negative and 50% positive ("near misses"). Table 1 lists the 7 relations together with some examples.

#	Relation Name	Examples
1	Cause-Effect	hormone-growth, laugh-wrinkle
2	Instrument-Agency	laser-printer, ax-murderer
3	Product-Producer	honey-bee, philosopher-theory
4	Origin-Entity	grain-alcohol, desert-storm
5	Theme-Tool	work-force, copyright-law
6	Part-Whole	leg-table, door-car
7	Content-Container	apple-basket, plane-cargo

Table 1: **SemEval dataset**: Relations with examples (context sentences are not shown).

Each example consists of a sentence, two nominals to be judged on whether they are in the target semantic relation, manually annotated WordNet 3.0 sense keys for these nominals, and the Web query used to obtain that example:

```
"Among the contents of the <e1>vessel</e1>
were a set of carpenters <e2>tools</e2>,
several large storage jars, ceramic
utensils, ropes and remnants of food, as
well as a heavy load of ballast stones."
WordNet(e1) = "vessel%1:06:00::",
WordNet(e2) = "tool%1:06:00::",
Content-Container(e2, e1) = "true",
Query = "contents of the * were a"
```

2 Related Work

Lauer (1995) proposes that eight prepositions are enough to characterize the relation between nouns in a noun-noun compound: *of*, *for*, *in*, *at*, *on*, *from*, *with* or *about*. Lapata and Keller (2005) improve on his results by using Web statistics. Rosario et al. (2002) use a "descent of hierarchy", which characterizes the relation based on the semantic category of the two nouns. Girju et al. (2005) apply SVM, decision trees, semantic scattering and iterative seman-

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 366–369,
Prague, June 2007. ©2007 Association for Computational Linguistics

tic specialization, using WordNet, word sense disambiguation, and linguistic features. Barker and Szpakowicz (1998) propose a two-level hierarchy with 5 classes at the upper level and 30 at the lower level. Turney (2005) introduces latent relational analysis, which uses the Web, synonyms, patterns like “X for Y”, “X such as Y”, etc., and singular value decomposition to smooth the frequencies. Turney (2006) induces patterns from the Web, e.g. CAUSE is best characterized by “*Y * causes X*”, and “*Y in * early X*” is the best pattern for TEMPORAL. Kim and Baldwin (2006) propose to use a *predefined* set of seed verbs and multiple resources: WordNet, CoreLex, and Moby’s thesaurus. Finally, in a previous publication (Nakov and Hearst, 2006), we make the claim that the relation between the nouns in a noun-noun compound can be characterized by the set of intervening verbs extracted from the Web.

3 Method

Given an entity-annotated example sentence, we reduce the target entities e_1 and e_2 to single nouns $noun_1$ and $noun_2$, by keeping their last nouns only, which we assume to be the heads. We then mine the Web for sentences containing both $noun_1$ and $noun_2$, from which we extract features, consisting of word(s), part of speech (verb, preposition, verb+preposition, coordinating conjunction), and whether $noun_1$ precedes $noun_2$. Table 2 shows some example features and their frequencies.

We start with a set of exact phrase queries against Google: “$infl_1$ THAT $\star$ $infl_2$”, “$infl_2$ THAT $\star$ $infl_1$”, “$infl_1$ $\star$ $infl_2$”, and “$infl_2$ $\star$ $infl_1$”, where $infl_1$ and $infl_2$ are inflectional variants of $noun_1$ and $noun_2$, generated using WordNet (Fellbaum, 1998); THAT can be *that*, *which*, or *who*; and $\star$ stands for 0 or more (up to 8) stars separated by spaces, representing the Google $\star$ single-word wildcard match operator. For each query, we collect the text snippets from the result set (up to 1000 per query), split them into sentences, assign POS tags using the OpenNLP tagger[1], and extract features:

Verb: If one of the nouns is the subject, and the other one is a direct or indirect object of that verb, we extract it and we lemmatize it using WordNet (Fellbaum, 1998). We ignore modals and auxiliaries, but retain the passive *be*, verb particles and prepositions (in case of indirect object).

Freq.	Feature	POS	Direction
2205	of	P	2 → 1
1923	be	V	1 → 2
771	include	V	1 → 2
382	serve on	V	2 → 1
189	chair	V	2 → 1
189	have	V	1 → 2
169	consist of	V	1 → 2
148	comprise	V	1 → 2
106	sit on	V	2 → 1
81	be chaired by	V	1 → 2
78	appoint	V	1 → 2
77	on	P	2 → 1
66	and	C	1 → 2
...	...	...	...

Table 2: **Most frequent features for *committee member***. V stands for verb, P for preposition, and C for coordinating conjunction.

Preposition: If one of the nouns is the head of an NP which contains a PP, inside which there is an NP headed by the other noun (or an inflectional form thereof), we extract the preposition heading that PP.

Coordination: If the two nouns are the heads of two coordinated NPs, we extract the coordinating conjunction.

In addition, we include some non-Web features[2]:

Sentence word: We use as features the words from the context sentence, after stop words removal and stemming with the Porter stemmer.

Entity word: We also use the lemmas of the words that are part of e_i ($i = 1, 2$).

Query word: Finally, we use the individual words that are part of the query string. This feature is used for category C runs only (see below).

Once extracted, the features are used to calculate the similarity between two noun pairs. Each feature triplet is assigned a weight. We wish to downweight very common features, such as “of” used as a preposition in the $2 \rightarrow 1$ direction, so we apply tf.idf weighting to each feature. We then use the following variant of the Dice coefficient to compare the weight vectors $A = (a_1, \ldots, a_n)$ and $B = (b_1, \ldots, b_n)$:

$$Dice(A, B) = \frac{2 \times \sum_{i=1}^{n} \min(a_i, b_i)}{\sum_{i=1}^{n} a_i + \sum_{i=1}^{n} b_i} \qquad (1)$$

This vector representation is similar to that of

[1] OpenNLP: http://opennlp.sourceforge.net

[2] Features have type prefix to prevent them from mixing.

System	Relation	P	R	F	Acc
UCB-A1	Cause-Effect	58.2	78.0	66.7	60.0
	Instrument-Agency	62.5	78.9	69.8	66.7
	Product-Producer	77.3	54.8	64.2	59.1
	Origin-Entity	67.9	52.8	59.4	67.9
	Theme-Tool	50.0	31.0	38.3	59.2
	Part-Whole	51.9	53.8	52.8	65.3
	Content-Container	62.2	60.5	61.3	60.8
	average	**61.4**	**58.6**	**58.9**	**62.7**
UCB-A2	Cause-Effect	58.0	70.7	63.7	58.8
	Instrument-Agency	65.9	71.1	68.4	67.9
	Product-Producer	80.0	77.4	78.7	72.0
	Origin-Entity	60.6	55.6	58.0	64.2
	Theme-Tool	45.0	31.0	36.7	56.3
	Part-Whole	41.7	38.5	40.0	58.3
	Content-Container	56.4	57.9	57.1	55.4
	average	**58.2**	**57.5**	**57.5**	**61.9**
UCB-A3	Cause-Effect	62.5	73.2	67.4	63.8
	Instrument-Agency	65.9	76.3	70.7	69.2
	Product-Producer	75.0	67.7	71.2	63.4
	Origin-Entity	48.4	41.7	44.8	54.3
	Theme-Tool	62.5	51.7	56.6	67.6
	Part-Whole	50.0	46.2	48.0	63.9
	Content-Container	64.9	63.2	64.0	63.5
	average	**61.3**	**60.0**	**60.4**	**63.7**
UCB-A4	Cause-Effect	63.5	80.5	71.0	66.2
	Instrument-Agency	70.0	73.7	71.8	71.8
	Product-Producer	76.3	72.6	74.4	66.7
	Origin-Entity	50.0	47.2	48.6	55.6
	Theme-Tool	61.5	55.2	58.2	67.6
	Part-Whole	52.2	46.2	49.0	65.3
	Content-Container	65.8	65.8	65.8	64.9
	average	**62.7**	**63.0**	**62.7**	**65.4**
	Baseline (majority)	81.3	42.9	30.8	57.0

Table 3: **Task 4 results.** UCB systems $A1$-$A4$.

System	Relation	P	R	F	Acc
UCB-C1	Cause-Effect	58.5	75.6	66.0	60.0
	Instrument-Agency	65.2	78.9	71.4	69.2
	Product-Producer	81.4	56.5	66.7	62.4
	Origin-Entity	67.9	52.8	59.4	67.9
	Theme-Tool	50.0	31.0	38.3	59.2
	Part-Whole	51.9	53.8	52.8	65.3
	Content-Container	62.2	60.5	61.3	60.8
	Average	**62.4**	**58.5**	**59.4**	**63.5**
UCB-C2	Cause-Effect	58.0	70.7	63.7	58.8
	Instrument-Agency	67.5	71.1	69.2	69.2
	Product-Producer	80.3	79.0	79.7	73.1
	Origin-Entity	60.6	55.6	58.0	64.2
	Theme-Tool	50.0	37.9	43.1	59.2
	Part-Whole	43.5	38.5	40.8	59.7
	Content-Container	56.4	57.9	57.1	55.4
	Average	**59.5**	**58.7**	**58.8**	**62.8**
UCB-C3	Cause-Effect	62.5	73.2	67.4	63.8
	Instrument-Agency	68.2	78.9	73.2	71.8
	Product-Producer	74.1	69.4	71.7	63.4
	Origin-Entity	56.8	58.3	57.5	61.7
	Theme-Tool	62.5	51.7	56.6	67.6
	Part-Whole	50.0	42.3	45.8	63.9
	Content-Container	64.9	63.2	64.0	63.5
	Average	**62.7**	**62.4**	**62.3**	**65.1**
UCB-C4	Cause-Effect	63.5	80.5	71.0	66.2
	Instrument-Agency	70.7	76.3	73.4	73.1
	Product-Producer	76.7	74.2	75.4	67.7
	Origin-Entity	59.0	63.9	61.3	64.2
	Theme-Tool	63.0	58.6	60.7	69.0
	Part-Whole	52.2	46.2	49.0	65.3
	Content-Container	64.1	65.8	64.9	63.5
	Average	**64.2**	**66.5**	**65.1**	**67.0**
	Baseline (majority)	81.3	42.9	30.8	57.0

Table 4: **Task 4 results.** UCB systems $C1$-$C4$.

Lin (1998), who measures word similarity by using triples extracted from a dependency parser. In particular, given a noun, he finds all verbs that have it as a subject or object, and all adjectives that modify it, together with the corresponding frequencies.

4 Experiments and Results

Participants were asked to classify their systems into categories depending on whether they used the WordNet sense (WN) and/or the Google query (GC). Our team submitted runs for categories A (WN=no, QC=no) and C (WN=no, QC=yes) only, since we believe that having the target entities annotated with the correct WordNet senses is an unrealistic assumption for a real-world application.

Following Turney and Littman (2005) and Barker and Szpakowicz (1998), we used a 1-nearest-neighbor classifier. Given a test example, we calculated the Dice coefficient between its feature vector and the vector of each of the training examples. If there was a single highest-scoring training example, we predicted its class for that test example. Otherwise, if there were ties for first, we assumed the class predicted by the majority of the tied examples. If there was no majority, we predicted the class that was most likely on the training data. Regardless of the classifier's prediction, if the head words of the two entities e_1 and e_2 had the same lemma, we classified that example as negative.

Table 3 and 4 show the results for our A and C runs for different amounts of training data: 45 ($A1$, $C1$), 90 ($A2$, $C2$), 105 ($A3$, $C3$) and 140 ($A4$, $C4$). All results are above the baseline: always propose the majority label ("true"/"false") in the test set. In fact, our category C system is the best-performing (in terms of F and Acc) among the participating systems, and we achieved the third best results for category A. Our category C results are slightly but consistently better than for A for all measures (P, R, F, Acc), which suggests that knowing the query is helpful. Interestingly, systems UCB-$A2$ and UCB-$C2$ performed worse than UCB-$A1$ and UCB-$C1$, which means that having more training data does not necessarily help with a 1NN classifier.

Table 5 shows additional analysis for $A4$ and $C4$. We study the effect of adding extra Google contexts (using up to 10 stars, rather than 8), and using different subsets of features. We show the results for: (a) leave-one-out cross-validation on the training data, (b) on the test data, and (c) our official UCB runs.

Acknowledgements: This work is supported in part by NSF DBI-0317510.

References

Ken Barker and Stan Szpakowicz. 1998. Semi-automatic recognition of noun modifier relationships. In *Proceedings of COLING-ACL'98*, pages 96–102.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Roxana Girju, Dan Moldovan, Marta Tatu, and Daniel Antohe. 2005. On the semantics of noun compounds. *Computer Speech and Language*, 19(4):479–496.

Su Nam Kim and Timothy Baldwin. 2006. Interpreting semantic relations in noun compounds via verb semantics. In *Proceedings of COLING/ACL 2006. (poster)*, pages 491–498.

Mirella Lapata and Frank Keller. 2005. Web-based models for natural language processing. *ACM Transactions on Speech and Language Processing*, 2:1–31.

Mark Lauer. 1995. *Designing Statistical Language Learners: Experiments on Noun Compounds*. Ph.D. thesis, Department of Computing Macquarie University NSW 2109 Australia.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of International Conference on Machine Learning*, pages 296–304.

Preslav Nakov and Marti Hearst. 2006. Using verbs to characterize noun-noun relations. In *Proceedings of AIMSA*, pages 233–244.

Barbara Rosario, Marti Hearst, and Charles Fillmore. 2002. The descent of hierarchy, and selection in relational semantics. In *ACL*, pages 247–254.

Peter Turney and Michael Littman. 2005. Corpus-based learning of analogies and semantic relations. *Machine Learning Journal*, 60(1-3):251–278.

Peter Turney. 2005. Measuring semantic similarity by latent relational analysis. In *Proceedings IJCAI*, pages 1136–1141.

Peter Turney. 2006. Expressing implicit semantic relations without supervision. In *Proceedings of COLING-ACL*, pages 313–320.

Features Used	**Leave-1-out**	**Test**	**UCB**
Cause-Effect			
$sent$	45.7	50.0	
p	55.0	53.8	
v	59.3	68.8	
$v+p$	57.1	63.7	
$v+p+c$	70.5	67.5	
$v+p+c+sent$	58.5	66.2	66.2
$v+p+c+sent+query$	59.3	66.2	66.2
Instrument-Agency			
$sent$	63.6	59.0	
p	62.1	70.5	
v	71.4	69.2	
$v+p$	70.7	70.5	
$v+p+c$	70.0	70.5	
$v+p+c+sent$	68.6	71.8	71.8
$v+p+c+sent+query$	70.0	73.1	73.1
Product-Producer			
$sent$	47.9	59.1	
p	55.7	58.1	
v	70.0	61.3	
$v+p$	66.4	65.6	
$v+p+c$	67.1	65.6	
$v+p+c+sent$	66.4	69.9	66.7
$v+p+c+sent+query$	67.9	69.9	67.7
Origin-Entity			
$sent$	64.3	72.8	
p	63.6	56.8	
v	69.3	71.6	
$v+p$	67.9	69.1	
$v+p+c$	66.4	70.4	
$v+p+c+sent$	68.6	72.8	55.6
$v+p+c+sent+query$	67.9	72.8	64.2
Theme-Tool			
$sent$	66.4	69.0	
p	56.4	56.3	
v	61.4	70.4	
$v+p$	56.4	67.6	
$v+p+c$	57.1	69.0	
$v+p+c+sent$	52.1	62.0	67.6
$v+p+c+sent+query$	52.9	62.0	69.0
Part-Whole			
$sent$	47.1	51.4	
p	57.1	54.1	
v	60.0	66.7	
$v+p$	62.1	63.9	
$v+p+c$	61.4	63.9	
$v+p+c+sent$	60.0	61.1	65.3
$v+p+c+sent+query$	60.0	61.1	65.3
Content-Container			
$sent$	56.4	54.1	
p	57.9	59.5	
v	71.4	67.6	
$v+p$	72.1	67.6	
$v+p+c$	72.9	67.6	
$v+p+c+sent$	69.3	67.6	64.9
$v+p+c+sent+query$	71.4	71.6	63.5
Average A4		**67.3**	**65.4**
Average C4		**68.1**	**67.0**

Table 5: **Accuracy for different features and extra Web contexts:** on leave-one-out cross-validation, on testing data, and in the official UCB runs.