SENSEVAL-3: Third International Workshop on the Evaluation of Systems
for the Semantic Analysis of Text, Barcelona, Spain, July 2004
Association for Computational Linguistics

Class-based Collocations for Word-Sense Disambiguation

Tom O'Hara
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003-8001
tomohara@cs.nmsu.edu

Rebecca Bruce
Department of Computer Science
University of North Carolina at Asheville
Asheville, NC 28804-3299
bruce@cs.unca.edu

Jeff Donner
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003-8001
jdonner@cs.nmsu.edu

Janyce Wiebe
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260-4034
wiebe@cs.pitt.edu

Abstract

This paper describes the NMSU-Pitt-UNCA word-sense disambiguation system participating in the Senseval-3 English lexical sample task. The focus of the work is on using semantic class-based collocations to augment traditional word-based collocations. Three separate sources of word relatedness are used for these collocations: 1) WordNet hypernym relations; 2) cluster-based word similarity classes; and 3) dictionary definition analysis.

1 Introduction

Supervised systems for word-sense disambiguation (WSD) often rely upon word *collocations* (i.e., sense-specific keywords) to provide clues on the most likely sense for a word given the context. In the second Senseval competition, these features figured predominantly among the feature sets for the leading systems (Mihalcea, 2002; Yarowsky et al., 2001; Seo et al., 2001). A limitation of such features is that the words selected must occur in the test data in order for the features to apply. To alleviate this problem, class-based approaches augment word-level features with category-level ones (Ide and Véronis, 1998; Jurafsky and Martin, 2000). When applied to collocational features, this approach effectively uses class labels rather than wordforms in deriving the collocational features.

This research focuses on the determination of class-based collocations to improve word-sense disambiguation. We do not address refinement of existing algorithms for machine learning. Therefore, a commonly used decision tree algorithm is employed to combine the various features when performing classification.

This paper describes the NMSU-Pitt-UNCA system we developed for the third Senseval competition. Section 2 presents an overview of the feature set used in the system. Section 3 describes how the class-based collocations are derived. Section 4 shows the results over the Senseval-3 data and includes detailed analysis of the performance of the various collocational features.

2 System Overview

We use a decision tree algorithm for word-sense disambiguation that combines features from the local context of the target word with other lexical features representing the broader context. Figure 1 presents the features that are used in this application. In the first Senseval competition, we used the first two groups of features, *Local-context features* and *Collocational features*, with competitive results (O'Hara et al., 2000).

Five of the local-context features represent the part of speech (POS) of words immediately surrounding the target word. These five features are $POS\pm i$ for i from *-2* to *+2*), where *POS+1*, for example, represents the POS of the word immediately following the target word.

Five other local-context features represent the word tokens immediately surrounding the target word ($Word\pm i$ for i from -2 to $+2$). Each $Word\pm i$ feature is multi-valued; its values correspond to all possible word tokens.

There is a collocation feature $WordColl_s$ defined for each sense s of the target word. It is a binary feature, representing the absence or presence of any word in a set specifically chosen for s. A word w that occurs more than once in the training data is included in the collocation set for sense s if the relative percent gain in the conditional probability over the prior probabil-

	Local-context features
POS:	part-of-speech of target word
POS±i:	part-of-speech of word at offset i
WordForm:	target wordform
Word±i:	stem of word at offset i
	Collocational features
WordColl_s:	word collocation for sense s
WordColl_*	wordform of non-sense-specific collocation (enumerated)
	Class-based collocational features
HyperColl_s:	hypernym collocation for s
$\text{HyperColl}_{*,i}$:	non-sense-specific hypernym collocation
SimilarColl_s:	similarity collocation for s
DictColl_s:	dictionary collocation for s

Figure 1: *Features for word-sense disambiguation.* All collocational features are binary indicators for sense s, except for WordColl_*.

ity is 20% or higher:

$$\frac{(P(s|w) - P(s))}{P(s)} \geq 0.20.$$

This threshold was determined to be effective via an optimization search over the Senseval-2 data. $\textit{WordColl}_*$ represents a set of non-sense-specific collocations (i.e., not necessarily indicative of any one sense), chosen via the G^2 criteria (Wiebe et al., 1998). In contrast to $\textit{WordColl}_s$, each of which is a separate binary feature, the words contained in the set $\textit{WordColl}_*$ serve as values in a single enumerated feature.

These features are augmented with *class-based* collocational features that represent information about word relationships derived from three separate sources: 1) WordNet (Miller, 1990) hypernym relations (*HyperColl*); 2) cluster-based word similarity classes (*SimilarColl*); and 3) relatedness inferred from dictionary definition analysis (*DictColl*). The information inherent in the sources from which these class-based features are derived allows words that do not occur in the training data context to be considered as collocations during classification.

3 Class-based Collocations

The *HyperColl* features are intended to capture a portion of the information in the WordNet hypernyms links (i.e., *is-a* relations). Hypernym-based collocations are formulated by replacing each word in the context of the target word (e.g., in the same sentence as the target word) with its complete hypernym ancestry from WordNet. Since context words are not sense-tagged, each synset representing a different sense of a context word is included in the set of hypernyms replacing that word. Likewise, in the case of multiple inheritance, each parent synset is included.

The collocation variable $\textit{HyperColl}_s$ for each sense s is binary, corresponding to the absence or presence of any hypernym in the set chosen for s. This set of hypernyms is chosen using the ratio of conditional probability to prior probability as described for the $\textit{WordColl}_s$ feature above. In contrast, $\textit{HyperColl}_{*,i}$ selects non-sense-specific hypernym collocations: 10 separate binary features are used based on the G^2 selection criteria. (More of these features could be used, but they are limited for tractability.) For more details on hypernym collocations, see (O'Hara, forthcoming).

Word-similarity classes (Lin, 1998) derived from clustering are also used to expand the pool of potential collocations; this type of semantic relatedness among words is expressed in the *SimilarColl* feature. For the *DictColl* features, definition analysis (O'Hara, forthcoming) is used to determine the semantic relatedness of the defining words. Differences between these two sources of word relations are illustrated by looking at the information they provide for 'ballerina':

word-clusters:
dancer:0.115 baryshnikov:0.072
pianist:0.056 choreographer:0.049
... *[18 other words]*
nicole:0.041 wrestler:0.040
tibetans:0.040 clown:0.040

definition words:
dancer:0.0013 female:0.0013 ballet:0.0004

This shows that word clusters capture a wider range of relatedness than the dictionary definitions at the expense of incidental associations (e.g., 'nicole'). Again, because context words are not disambiguated, the relations for all senses of a context word are conflated. For details on the extraction of word clusters, see (Lin, 1998); and, for details on the definition analysis, see (O'Hara, forthcoming).

When formulating the features *SimilarColl* and *DictColl*, the words related to each context word are considered as *potential collocations* (Wiebe et al., 1998). Co-occurrence fre-

Sense Distinctions	Precision	Recall
Fine-grained	.566	.565
Course-grained	.660	.658

Table 1: *Results for Senseval-3 test data.* 99.72% of the answers were attempted. All features from Figure 1 were used.

quencies $f(s, w)$ are used in estimating the conditional probability $P(s|w)$ required by the relative conditional probability selection scheme noted earlier. However, instead of using a unit weight for each co-occurrence, the relatedness weight is used (e.g., 0.056 for 'pianist'); and, because a given related-word might occur with more than one context word for the same target-word sense, the relatedness weights are added. The conditional probability of the sense given the relatedness collocation is estimated by dividing the weighted frequency by the sum of all such weighted co-occurrence frequencies for the word:

$$P(s|w) \approx \frac{wf(s, w)}{\sum_{s'} wf(s', w)}$$

Here *wf(s, w)* stands for the weighted co-occurrence frequency of the related-word collocation w and target sense s.

The relatedness collocations are less reliable than word collocations given the level of indirection involved in their extraction. Therefore, tighter constraints are used in order to filter out extraneous potential collocations. In particular, the relative percent gain in the conditional versus prior probability must be 80% or higher, a threshold again determined via an optimization search over the Senseval-2 data. In addition, the context words that they are related to must occur more than four times in the training data.

4 Results and Discussion

Disambiguation is performed via a decision tree formulated using Weka's J4.8 classifier (Witten and Frank, 1999). For the system used in the competition, the decision tree was learned over the entire Senseval-3 training data and then applied to the test data. Table 1 shows the results of our system in the Senseval-3 competition.

Table 2 shows the results of 10-fold cross-validation just over the Senseval-3 training data (using Naive Bayes rather than decision trees.) To illustrate the contribution of the three types of class-based collocations, the table shows results separately for systems developed using a single feature type, as well as for all features in combination. In addition, the performance of these systems are shown with and without the use of the local features (*Local*), as well as with and without the use of standard word collocations (*WordColl*). As can be seen, the related-word and definition collocations perform better than hypernym collocations when used alone. However, hypernym collocations perform better when combined with other features. Future work will investigate ways of ameliorating such interactions. The best overall system ($HyperColl + WordColl + Local$) uses the combination of local-context features, word collocations, and hypernym collocations. The performance of this system compared to a more typical system for WSD ($WordColl + Local$) is statistically significant at $p < .05$, using a paired t-test.

Experiment	Precision	
	−Local	+Local
Local	-	.593
WordColl	.490	.599
HyperColl	.525	.590
DictColl	.532	.570
SimilarColl	.534	.586
HyperColl+WordColl	.525	.611
DictColl+WordColl	.501	.606
SimilarColl+WordColl	.518	.596
All Collocations	.543	.608

#Words:	57	Avg. Entropy:	1.641
Avg. #Senses:	5.3	Baseline:	0.544

Table 2: *Results for Senseval-3 training data.* All values are averages, except *#Words*, which is the number of distinct word types classified. *Baseline* always uses the most-frequent sense.

We analyzed the contributions of the various collocation types to determine their effectiveness. Table 3 shows performance statistics for each collocation type taken individually over the training data. Precision is based on the number of correct positive indicators versus the total number of positive indicators, whereas recall is the number correct over the total number of training instances (7706). This shows that hypernym collocations are nearly as effective as word collocations. We also analyzed the occurrence of unique positive indicators provided by the collocation types over the training data. Ta-

Feature	Total #Corr.	Total #Pos.	Recall	Prec.
DictColl	273	592	.035	.461
HyperColl	2932	6479	.380	.453
SimilarColl	528	1535	.069	.344
WordColl	3707	7718	.481	.480

Table 3: *Collocation performance statistics. Total #Pos.* is number of positive indicators for the collocation in the training data, and *Total #Corr.* is the number of these that are correct.

Feature	Unique #Corr.	Unique #Pos.	Prec.
DictColl	110	181	.608
HyperColl	992	1795	.553
SimilarColl	198	464	.427
DictColl	1244	2085	.597

Table 4: *Analysis of unique positive indicators. Unique #Pos.* is number of training instances with the feature as the only positive indicator, and *Unique #Corr.* is number of these correct.

ble 4 shows how often each feature type is positive for a particular sense when all other features for the sense are negative. This occurs fairly often, suggesting that the different types of collocations are complementary and thus generally useful when combined for word-sense disambiguation. Both tables illustrate coverage problems for the definition and related word collocations, which will be addressed in future work.

References

Nancy Ide and Jean Véronis. 1998. Introduction to the special issue on word sense disambiguation: the state of the art. *Computational Linguistics*, 24(1):1–40.

Daniel Jurafsky and James H. Martin. 2000. *Speech and Language Processing.* Prentice Hall, Upper Saddle River, New Jersey.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proc. COLING-ACL 98*, pages 768–764, Montreal. August 10-14.

Rada Mihalcea. 2002. Instance based learning with automatic feature selection applied to word sense disambiguation. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002)*, Taiwan. August 26-30.

George Miller. 1990. Introduction. *International Journal of Lexicography*, 3(4): Special Issue on WordNet.

Tom O'Hara, Janyce Wiebe, and Rebecca F. Bruce. 2000. Selecting decomposable models for word-sense disambiguation: The GRLING-SDM system. *Computers and the Humanities*, 34(1-2):159–164.

Thomas P. O'Hara. forthcoming. *Empirical acquisition of conceptual distinctions via dictionary definitions.* Ph.D. thesis, Department of Computer Science, New Mexico State University.

Hee-Cheol Seo, Sang-Zoo Lee, Hae-Chang Rim, and Ho Lee. 2001. KUNLP system using classification information model at SENSEVAL-2. In *Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL-2)*, pages 147–150, Toulouse. July 5-6.

Janyce Wiebe, Kenneth McKeever, and Rebecca F. Bruce. 1998. Mapping collocational properties into machine learning features. In *Proc. 6th Workshop on Very Large Corpora (WVLC-98)*, pages 225–233, Montreal, Quebec, Canada. Association for Computational Linguistics. SIGDAT.

Ian H. Witten and Eibe Frank. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.* Morgan Kaufmann, San Francisco, CA.

David Yarowsky, Silviu Cucerzan, Radu Florian, Charles Schafer, and Richard Wicentowski. 2001. The Johns Hopkins SENSEVAL2 system descriptions. In *Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL-2)*, pages 163–166, Toulouse. July 5-6.