UNT-Yahoo: SuperSenseLearner: Combining SenseLearner with SuperSense and other Coarse Semantic Features

Rada Mihalcea and **Andras Csomai**
University of North Texas
rada@cs.unt.edu,csomaia@unt.edu

Massimiliano Ciaramita
Yahoo! Research Barcelona
massi@yahoo-inc.com

Abstract

We describe the SUPERSENSELEARNER system that participated in the English all-words disambiguation task. The system relies on automatically-learned semantic models using collocational features coupled with features extracted from the annotations of coarse-grained semantic categories generated by an HMM tagger.

1 Introduction

The task of word sense disambiguation consists of assigning the most appropriate meaning to a polysemous word within a given context. Applications such as machine translation, knowledge acquisition, common sense reasoning, and others, require knowledge about word meanings, and word sense disambiguation is considered essential for all these tasks.

Most of the efforts in solving this problem were concentrated so far toward targeted supervised learning, where each sense tagged occurrence of a particular word is transformed into a feature vector, which is then used in an automatic learning process. The applicability of such supervised algorithms is however limited only to those few words for which sense tagged data is available, and their accuracy is strongly connected to the amount of labeled data available at hand.

Instead, methods that address all words in unrestricted text have received significantly less attention. While the performance of such methods is usually exceeded by their supervised lexical-sample alternatives, they have however the advantage of providing larger coverage.

In this paper, we describe SUPERSENSELEARNER – a system for solving the semantic ambiguity of all words in unrestricted text. SUPERSENSELEARNER brings together under one system the features previously used in the SENSELEARNER (Mihalcea and Csomai, 2005) and the SUPERSENSE (Ciaramita and Altun, 2006) all-words word sense disambiguation systems. The system is using a relatively small pre-existing sense-annotated data set for training purposes, and it learns global semantic models for general word categories.

2 Learning for All-Words Word Sense Disambiguation

Our goal is to use as little annotated data as possible, and at the same time make the algorithm *general* enough to be able to disambiguate as many content words as possible in a text, and *efficient* enough so that large amounts of text can be annotated in real time. SUPERSENSELEARNER is attempting to learn general semantic models for various word categories, starting with a relatively small sense-annotated corpus. We base our experiments on SemCor (Miller et al., 1993), a balanced, semantically annotated dataset, with all content words manually tagged by trained lexicographers.

The input to the disambiguation algorithm consists of raw text. The output is a text with word meaning annotations for all open-class words.

The algorithm starts with a preprocessing stage, where the text is tokenized and annotated with part-

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 406–409,
Prague, June 2007. ©2007 Association for Computational Linguistics

of-speech tags; collocations are identified using a sliding window approach, where a collocation is defined as a sequence of words that forms a compound concept defined in WordNet (Miller, 1995).

Next, a semantic model is learned for all predefined word categories, where a word category is defined as a group of words that share some common syntactic or semantic properties. Word categories can be of various granularities. For instance, a model can be defined and trained to handle all the *nouns* in the test corpus. Similarly, using the same mechanism, a finer-grained model can be defined to handle all the verbs for which at least one of the meanings is of type e.g., "<move>". Finally, small coverage models that address one word at a time, for example a model for the adjective "small," can be also defined within the same framework. Once defined and trained, the models are used to annotate the ambiguous words in the test corpus with their corresponding meaning. Sections 3 and 4 below provide details on the features implemented by the various models.

Note that the semantic models are applicable only to: (1) words that are covered by the word category defined in the models; and (2) words that appeared at least once in the training corpus. The words that are not covered by these models (typically about 10-15% of the words in the test corpus) are assigned the most frequent sense in WordNet.

3 SenseLearner Semantic Models

Different semantic models can be defined and trained for the disambiguation of different word categories. Although more general than models that are built individually for each word in a test corpus (Decadt et al., 2004), the applicability of the semantic models built as part of SENSELEARNER is still limited to those words previously seen in the training corpus, and therefore their overall coverage is not 100%.

Starting with an annotated corpus consisting of all the annotated files in SemCor, augmented with the SENSEVAL-2 and SENSEVAL-3 all-words data sets, a separate training data set is built for each model. There are seven models provided with the current SENSELEARNER distribution, implementing the following features:

3.1 Noun Models

modelNN1: A contextual model that relies on the first noun, verb, or adjective before the target noun, and their corresponding part-of-speech tags.
modelNNColl: A collocation model that implements collocation-like features based on the first word to the left and the first word to the right of the target noun.

3.2 Verb Models

modelVB1 A contextual model that relies on the first word before and the first word after the target verb, and their part-of-speech tags.
modelVBColl A collocation model that implements collocation-like features based on the first word to the left and the first word to the right of the target verb.

3.3 Adjective Models

modelJJ1 A contextual model that relies on the first noun after the target adjective.
modelJJ2 A contextual model that relies on the first word before and the first word after the target adjective, and their part-of-speech tags.
modelJJColl A collocation model that implements collocation-like features using the first word to the left and the first word to the right of the target adjective.

Based on previous performance in the SENSEVAL-2 and SENSEVAL-3 evaluations, we selected the noun and verb collocational models for inclusion in the SUPERSENSELEARNER system participating in the SEMEVAL all-words task.

4 SuperSenses and other Coarse-Grained Semantic Features

A great deal of work has focused in recent years on shallow semantic annotation tasks such as named entity recognition and semantic role labeling. In the former task, systems analyze text to detect mentions of instances of coarse-grained semantic categories such as "person", "organization" and "location". It seems natural to ask if this type of shallow semantic information can be leveraged to improve lexical disambiguation. Particularly, since the best performing taggers typically implement sequential decoding schemes, e.g., Viterbi decoding, which have linear

complexity and can be performed quite efficiently. In practice thus, this type of pre-processing resembles POS-tagging and could provide the WSD system with useful additional evidence.

4.1 Tagsets

We use three different tagsets. The first is the set of WordNet supersenses (Ciaramita and Altun, 2006): a mapping of WordNet's synsets to 45 broad lexicographers categories, 26 for nouns, 15 for verbs, 3 for adjectives and 1 for adverbs. The second tagset is based on the ACE 2007 English data for entity mention detection (EMD) (ACE, 2007). This tagset defines seven entity types: Facility, Geo-Political Entity, Location, Organization, Person, Vehicle, Weapon; further subdivided in 44 subtypes. The third tagset is derived from the BBN Entity Corpus (BBN, 2005) which complements the Wall Street Journal Penn Treebank with annotations of a large set of entities: 12 named entity types (Person, Facility, Organization, GPE, Location, Nationality, Product, Event, Work of Art, Law, Language, and Contact-Info), nine nominal entity types (Person, Facility, Organization, GPE, Product, Plant, Animal, Substance, Disease and Game), and seven numeric types (Date, Time, Percent, Money, Quantity, Ordinal and Cardinal). Several of these types are further divided into subtypes, for a total of 105 classes.[1]

4.2 Taggers

We annotate the training and evaluation data using three sequential taggers, one for each tagset. The tagger is a Hidden Markov Model trained with the perceptron algorithm introduced in (Collins, 2002), which applies Viterbi decoding and is regularized using averaging. Label to label dependencies are limited to the previous tag (first order HMM). We use a generic feature set for NER based on words, lemmas, POS tags, and word shape features, in addition we use as a feature of each token the supersense of a first (super)sense baseline. A detailed description of the features used and the tagger can be found in (Ciaramita and Altun, 2006). The supersense tagger is trained on the Brown sections one and two of SemCor. The BBN tagger is trained on sections 2-21 of the BBN corpus. The ACE tagger is trained on the 599 ACE 2007 training files. The accuracy of the tagger is, approximately, 78% F-score for supersenses and ACE, and 87% F-score for the BBN corpus.

[1] BBN Corpus documentation.

4.3 Features

The taggers disregard the lemmatization of the evaluation data. In practice, this means that multiword lemmas such as "take off", are split into their basic components. In fact, the goal of the tagger is to guess the elements of the instances of semantic categories by means of the usual BIO encoding. In other words, the tagger predicts a labeled bracketing of the tokens in each sentence. As an example, the supersense tagger annotates the tokens in the phrase "substance abuse" as "substance$_{B-noun.act}$" and "abuse$_{I-noun.act}$", although the gold standard segmentation of the data does not identify the phrase as one lemma. We use the labels generated in this way as features of each token to disambiguate.

5 Feature Combination

For the final system we create a combined feature set for each target word, consisting of the lemma, the part of speech, the collocational SENSELEARNER features, and the three coarse grained semantic tags of the target word. Note that the semantic features are represented as *lemma_TAG* to avoid overgeneralization.

In the training stage, a feature vector is constructed for each sense-annotated word covered by a semantic model. The features are model-specific, and feature vectors are added to the training set pertaining to the corresponding model. The label of each such feature vector consists of the target word and the corresponding sense, represented as *word#sense*. Table 1 shows the number of feature vectors constructed in this learning stage for each semantic model. To annotate new text, similar vectors are created for all the content-words in the raw text. Similar to the training stage, feature vectors are created and stored separately for each semantic model.

Next, word sense predictions are made for all the test examples, with a separate learning process run for each semantic model. For learning, we are using the Timbl memory based learning algorithm (Daele-

mode	Training size	RESULTS	
		Precision	Recall
noun	89052	0.658	0.228
verb	48936	0.539	0.353
all	137988	0.583	0.583

Table 1: Precision and recall for the SUPERSENSE-LEARNER semantic models.

mode	Training size	RESULTS	
		Precision	Recall
noun	89052	0.666	0.233
verb	48936	0.554	0.360
all	137988	0.593	0.593

Table 2: Precision and recall for the SUPERSENSE-LEARNER semantic models - without U labels.

mans et al., 2001), which was previously found useful for the task of word sense disambiguation (Hoste et al., 2002; Mihalcea, 2002).

Following the learning stage, each vector in the test data set is labeled with a *predicted* word and sense. If the word predicted by the learning algorithm coincides with the target word in the test feature vector, then the predicted sense is used to annotate the test instance. Otherwise, if the predicted word is different from the target word, no annotation is produced, and the word is left for annotation in a later stage (e.g., using the most frequent sense back-off method).

6 Results

The SUPERSENSELEARNER system participated in the SEMEVAL all-words word sense disambiguation task. Table 1 shows the results obtained for each part-of-speech (nouns and verbs), as well as the overall results. We have also ran a separate evaluation excluding the U (unknown) tag, which is shown in Table 2. SUPERSENSELEARNER was ranked the third among the fourteen participating systems, proving the validity of the approach.

Acknowledgments

We would like to thank Mihai Surdeanu for providing a pre-processed version of the ACE data.

References

2007. Automatic content extraction workshop. http://www.nist.gov/speech/tests/ace/ace07/index.htm.

2005. BBN pronoun coreference and entity type corpus. Linguistic Data Consortium (LDC) catalog number LDC2005T33.

M. Ciaramita and Y. Altun. 2006. Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

M. Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Philadelphia, July. Association for Computational Linguistics.

W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. 2001. Timbl: Tilburg memory based learner, version 4.0, reference guide. Technical report, University of Antwerp.

B. Decadt, V. Hoste, W. Daelemans, and A. Van den Bosch. 2004. Gambl, genetic algorithm optimization of memory-based wsd. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, Spain, July.

V. Hoste, W. Daelemans, I. Hendrickx, and A. van den Bosch. 2002. Evaluating the results of a memory-based word-expert approach to unrestricted word sense disambiguation. In *Proceedings of the ACL Workshop on "Word Sense Disambiguatuion: Recent Successes and Future Directions"*, Philadelphia, July.

R. Mihalcea and A. Csomai. 2005. Senselearner: Word sense disambiguation for all words in unrestricted text. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics*, Ann Arbor, MI.

R. 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)*, Taipei, Taiwan, August.

G. Miller, C. Leacock, T. Randee, and R. Bunker. 1993. A semantic concordance. In *Proceedings of the 3rd DARPA Workshop on Human Language Technology*, Plainsboro, New Jersey.

G. Miller. 1995. Wordnet: A lexical database. *Communication of the ACM*, 38(11):39–41.