WordNet::SenseRelate::AllWords - A Broad Coverage Word Sense Tagger that Maximizes Semantic Relatedness

Ted Pedersen and **Varada Kolhatkar**
Department of Computer Science
University of Minnesota
Duluth, MN 55812 USA
{tpederse,kolha002}@d.umn.edu
http://senserelate.sourceforge.net

Abstract

WordNet::SenseRelate::AllWords is a freely available open source Perl package that assigns a sense to every content word (known to WordNet) in a text. It finds the sense of each word that is most related to the senses of surrounding words, based on measures found in WordNet::Similarity. This method is shown to be competitive with results from recent evaluations including SENSEVAL-2 and SENSEVAL-3.

1 Introduction

Word sense disambiguation is the task of assigning a sense to a word based on the context in which it occurs. This is one of the central problems in Natural Language Processing, and has a long history of research. A great deal of progress has been made in using supervised learning to build models of disambiguation that assign a sense to a single target word in context. This is sometimes referred to as the lexical sample or target word formulation of the task.

However, to be effective, supervised learning requires many manually disambiguated examples of a single target word in different contexts to serve as training data to learn a classifier for that word. While the resulting models are often quite accurate, manually creating training data in sufficient volume to cover even a few words is very time consuming and error prone. Worse yet, creating sufficient training data to cover all the different words in a text is essentially impossible, and has never even been attempted.

Despite these difficulties, word sense disambiguation is often a necessary step in NLP and can't simply be ignored. The question arises as to how to develop broad coverage sense disambiguation modules that can be deployed in a practical setting without investing huge sums in manual annotation efforts. Our answer is WordNet::SenseRelate::AllWords (SR-AW), a method that uses knowledge already available in the lexical database WordNet to assign senses to every content word in text, and as such offers broad coverage and requires no manual annotation of training data.

SR-AW finds the sense of each word that is most related or most similar to those of its neighbors in the sentence, according to any of the ten measures available in WordNet::Similarity (Pedersen et al., 2004).

It extends WordNet::SenseRelate::TargetWord, a lexical sample word sense disambiguation algorithm that finds the maximum semantic relatedness between a target word and its neighbors (Patwardhan et al., 2003). SR-AW was originally developed by (Michelizzi, 2005) (through version 0.06) and is now being significantly enhanced.

2 Methodology

SR-AW processes a text sentence by sentence. It proceeds through each sentence word by word from left to right, centering each content word in a balanced window of context whose size is determined by the user. Note that content words at the start or end of a sentence will have unbalanced windows associated with them, since the algorithm does not cross sentence boundaries and treats each sentence independently.

Proceedings of NAACL HLT 2009: Demonstrations, pages 17–20,
Boulder, Colorado, June 2009.

All of the possible senses of the word in the center of the window are measured for similarity relative to the possible senses of each of the surrounding words in the window in a pairwise fashion. The sense of the center word that has the highest total when those pairwise scores are summed is considered to be the sense of that word. SR-AW then moves the center of the window to the next content word to the right. The user has the option of fixing the senses of the words that precede it to those that were discovered by SR-AW, or allowing all their senses to be considered in subsequent steps.

WordNet::Similarity[1] offers six similarity measures and four measures of relatedness. Measures of similarity are limited to making noun to noun and verb to verb comparisons, and are based on using the hierarchical information available for nouns and verbs in WordNet. These measures may be based on path lengths (path, wup, lch) or on path lengths augmented with Information Content derived from corpora (res, lin, jcn). The measures of relatedness may make comparisons between words in any part of speech, and are based on finding paths between concepts that are not limited to hierarchical relations (hso), or on using gloss overlaps either for string matching (lesk) or for creating a vector space model (vector and vector-pairs) that are used for measuring relatedness.

The availability of ten different measures that can be used with SR-AW leads to an incredible richness and variety in this approach. In general word sense disambiguation is based on the presumption that words that occur together will have similar or related meanings, so SR-AW allows for a wide range of options in deciding how to assess similarity and relatedness. SR-AW can be viewed as a graph based approach when using the path based measures, where words are assigned the senses that are located most closely together in WordNet. These path based methods can be easily augmented with Information Content in order to allow for finer grained distinctions to be made. It is also possible to lessen the impact of the physical structure of WordNet by using the content of the glosses as the primary source of information.

[1] http://wn-similarity.sourceforge.net

3 WordNet::SenseRelate::AllWords Usage

Input : The input to SR-AW can either be plain untagged text (raw), or it may be tagged with Penn Treebank part of speech tags (tagged : 47 tags; e.g., run/VBD), or with WordNet part of speech tags (wntagged: 4 tags for noun, verb, adjective, adverb; e.g., run#v). Penn Treebank tags are mapped to WordNet POS tags prior to SR-AW processing, so even though this tag set is very rich, it is used simply to distinguish between the four parts of speech WordNet knows, and identify function words (which are ignored as WordNet only includes open class words). In all cases simple morphological processing as provided by WordNet is utilized to identify the root form of a word in the input text.

Examples of each input format are shown below:

- (raw) : The astronomer married a movie star.
- (tagged) : The/DT astronomer/NN married/VBD a/DT movie_star/NN
- (wntagged) : The astronomer#n married#v a movie_star#n

If the format is raw, SR-AW will identify WordNet compounds before processing. These are multiword terms that are usually nouns with just one sense, so their successful identification can significantly improve overall accuracy. If a compound is not identified, then it often becomes impossible to disambiguate. For example, if White House is treated as two separate words, there is no combination of senses that will equal the residence of the US president, where that is the only sense of the compound White_House. To illustrate the scope of compounds, of the 155,287 unique strings in WordNet 3.0, more than 40% (64,331) of them are compounds. If the input is tagged or wntagged, it is assumed that the user has identified compounds by connecting the words that make up a compound with _ (e.g., white_house, movie_star).

In the tagged and wntagged formats, the user must identify compounds and also remove punctuation. In the raw format SR-AW will simply ignore punctuation unless it happens to be part of a compound (e.g., adam's_apple, john_f._kennedy). In all formats the upper/lower case distinction is ignored, and it is

assumed that the input is already formatted one line per sentence, one sentence per line.

SR-AW will then check to see if a stoplist has been provided by the user, or if the user would like to use the default stoplist. In general a stoplist is highly recommended, since there are quite a few words in WordNet that have unexpected senses and might be problematic unless they are excluded. For example, *who* has a noun sense of World Health Organization. *A* has seven senses, including angstrom, vitamin A, a nucleotide, a purine, an ampere, the letter, and the blood type. Many numbers have noun senses that define them as cardinal numbers, and some have adjective senses as well.

In the raw format, the stoplist check is done after compounding, because certain compounds include stop words (e.g., us_house_of_representatives). In the wntagged and tagged formats the stoplist check is still performed, but the stoplist must take into account the form of the part of speech tags. However, stoplists are expressed using regular expressions, making it quite convenient to deal with part of speech tags, and also to specify entire classes of terms to be ignored, such as numbers or single character words.

Disambiguation Options : The user has a number of options to control the direction of the SR-AW algorithm. These include the very powerful choices regarding the measure of similarity or relatedness that is to be used. There are ten such measures as has been described previously. As was also already mentioned, the user also can choose to fix the senses of words that have already been processed.

In addition to these options, the user can control the size of the window used to determine which words are involved in measuring relatedness or similarity. A window size of N includes the center word, and then extends out to the left and right of the center for $N/2$ content words, unless it encounters the sentence boundaries. If N is odd then the number of words to the left and right $(N-1)/2$, and if N is even there are $N/2$ words to the left, and $(N/2)-1$ words to the right.

When using a measure of similarity and tagged or wntagged text, it may be desirable to coerce the part of speech of surrounding words to that of the word in the center of the window of context. If this is not done, then any word with a part of speech other than that of the center word will not be included in the calculation of semantic similarity. Coercion is performed by first checking for forms of the word in a different part of speech, and then checking if there are any derivational relations from the word to the part of speech of the center word. Note that in the raw format part of speech coercion is not necessary, since the algorithm will consider all possible parts of speech for each word. If the sense of previous words has already been fixed, then part of speech coercion does not override those fixed assignments.

Finally, the user is able to control several scoring thresholds in the algorithm. The user may specify a context score which indicates a minimum threshold that a sense of the center word should achieve with all the words in the context in order to be selected. If this threshold is not met, no sense is assigned and it may be that the window should be increased.

The pair score is a finer grained threshold that indicates the minimum values that a relatedness score between a sense of the center word and a sense of one of the neighbors must achieve in order to be counted in the overall score of the center word. If this threshold is not met then the pair will contribute 0 to that score. This can be useful for filtering out noise from the scores when set to modest values.

Output : The output of SR-AW is the original text with WordNet sense tags assigned. WordNet sense tags are given in WPS form, which means word, part of speech, and sense number. In addition, glosses are displayed for each of the selected senses.

There are also numerous trace options available, which can be combined in order to provide more detailed diagnostic output. This includes displaying the window of context with the center word designated (1), the winning score for each context window (2), the non-zero scores for each sense of the center word (4), the non-zero pairwise scores (8), the zero values for any of the previous trace levels (16), and the traces from the semantic relatedness measures from WordNet::Similarity (32).

4 Experimental Results

We have evaluated SR-AW using three corpora that have been manually annotated with senses from WordNet. These include the SemCor corpus, and

Table 1: SR-AW Results (%)

	2			5			15		
SC	P	R	F	P	R	F	P	R	F
lch	56	13	21	54	29	36	52	35	42
jcn	65	15	24	64	31	42	62	41	49
lesk	58	49	53	62	60	61	62	61	61
S2	P	R	F	P	R	F	P	R	F
lch	48	10	16	50	24	32	48	31	38
jcn	55	9	15	55	21	31	55	31	39
lesk	54	44	48	58	56	57	59	59	59
S3	P	R	F	P	R	F	P	R	F
lch	48	13	20	49	29	37	48	35	41
jcn	55	14	22	55	31	40	53	38	46
lesk	51	43	47	54	52	53	54	53	54

the SENSEVAL-2 and SENSEVAL-3 corpora. SemCor is made up of more than 200,000 words of running text from news articles found in the Brown Corpus. The SENSEVAL data sets are each approximately 4,000 words of running text from Wall Street Journal news articles from the Penn Treebank. Note that only the words known to WordNet in these corpora have been sense tagged. As a result, there are 185,273 sense tagged words in SemCor, 2,260 in SENSEVAL-2, and 1,937 in SENSEVAL-3. We have used versions of these corpora where the WordNet senses have been mapped to WordNet 3.0[2].

In Table 4 we report results using Precision (P), Recall (R), and F-Measure (F). We use three window sizes in these experiments (2, 5, and 15), three WordNet::Similarity measures (lch, jcn, and lesk),and three different corpora : SemCor (*SC*), SENSEVAL-2 (*S2*), SENSEVAL-3 (*S3*). These experiments were carried out with version 0.17 of SR-AW.

For all corpora we observe the same patterns. The lesk measure tends to result in much higher recall with smaller window sizes, since it is able to measure similarity between words with any parts of speech, whereas lch and jcn are limited to making noun-noun and verb-verb measurements. But, as the window size increases so does recall. Precision continues to increase for lesk as the window size increases. Our best results come from using the lesk measure with a window size of 15. For SemCor this results in an F-measure of 61%. For SENSEVAL-2 it results in an F-measure of 59%, and for SENSEVAL-3 it results in an F-measure of 54%. These results would have ranked 4th of 22 teams and 15th of 26 in the respective SENSEVAL events.

A well known baseline for all words disambiguation is to assign the first WordNet sense to each ambiguous word. This results in an F-measure of 76% for SemCor, 69% for SENSEVAL-2, and 68% for SENSEVAL-3. A lower bound can be established by randomly assigning senses to words. This results in an F-Measure of 41% for SemCor, 41% for SENSEVAL-2, and 37% for SENSEVAL-3. This is relatively high due to the large number of words that have just one possible sense (so randomly selecting will result in a correct assignment). For example, in SemCor approximately 20% of the ambiguous words have just one sense. From these results we can see that SR-AW lags behind the sense one baseline (which is common among all words systems), but significantly outperforms the random baseline.

5 Conclusions

WordNet::SenseRelate::AllWords is a highly flexible method of word sense disambiguation that offers broad coverage and does not require training of any kind. It uses WordNet and measures of semantic similarity and relatedness to identify the senses of words that are most related to each other in a sentence. It is implemented in Perl and is freely available from the URL on the title page both as source code and via a Web interface.

References

J. Michelizzi. 2005. Semantic relatedness applied to all words sense disambiguation. Master's thesis, University of Minnesota, Duluth, July.

S. Patwardhan, S. Banerjee, and T. Pedersen. 2003. Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257, Mexico City, February.

T. Pedersen, S. Patwardhan, and J. Michelizzi. 2004. Wordnet::Similarity - Measuring the relatedness of concepts. In *Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 38–41, Boston, MA.

[2]http://www.cse.unt.edu/~rada/downloads.html