Statistical Corpus-Based Word Sense Disambiguation: Pseudowords vs. Real Ambiguous Words

Tanja Gaustad
Alfa-Informatica
Rijksuniversiteit Groningen
Postbus 716
NL-9700 AS Groningen, The Netherlands
T.Gaustad@let.rug.nl

Abstract

In this paper we investigate whether the task of disambiguating pseudowords (artificial ambiguous words) is comparable to the disambiguation of real ambiguous words. Since the two methods are inherently different, a direct comparison is not possible. An indirect approach is taken where the setup for both systems is as similar as possible, i.e. using the same corpus and settings. The results obtained clearly indicate that the tasks are quite different. We conclude that the current practice of using pseudowords cannot be taken as a substitute for testing with real ambiguous words.

1 Introduction

Word sense disambiguation is a difficult task. Also, it is difficult to get sense-tagged material for certain languages. One technique to experiment with sense disambiguation algorithms is to introduce artificially ambiguous words, pseudowords. In this paper we will investigate in how far pseudowords are a good alternative or simulation of real word sense disambiguation.

First, we will briefly sketch the outline of the problem discussed in this paper, then we will explain the mode of comparison as well as the settings used. Finally, we will compare and evaluate our results and draw conclusions.

2 Word Sense Disambiguation

A major problem in natural language processing is that of lexical ambiguity, be it syntactic or semantic. A word's *syntactic* ambiguity can be resolved by applying part-of-speech taggers which predict the syntactic category of a word in texts with high levels of accuracy (see for example (Brill, 1995) or (Brants, 2000)). The problem of resolving *semantic* ambiguity, which is generally known as word sense disambiguation (WSD), has proved to be more difficult than syntactic disambiguation.

The only way to determine the meaning of a word in a particular usage is to examine its context. For instance, the English word *bank*—an extensively cited example of lexical ambiguity—can refer to the bank of a river or to the pecuniary institution. For this reason, a computer program analyzing the sentence "The boy leapt from the bank into the cold water" will need to decide which reading of 'bank' was intended, in order to be able to come up with the correct meaning for the sentence. The overall goal of word sense disambiguation systems is to attribute the correct sense(s) to words in a text.

There are three ways to approach this problem: a *knowledge-based approach*, which uses an explicit lexicon, *corpus-based disambiguation*, where the relevant information about word senses is gathered from training on a large corpus, or, third alternative, a *hybrid approach* combining aspects of the aforementioned methodologies (see (Ide and Véronis, 1998) for a more thorough discussion).

If one chooses to work with a corpus-based approach, the possible means used to attribute senses to ambiguous words are *distributional information* and *context words*. Distributional information about an ambiguous word is the frequency distribution of its senses. Context words are the words found to the right and/or to the left of a certain word, thus collocational information.

Supervised WSD tends to use a machine learning algorithm. During training on a disambiguated corpus probabilistic information about context words as well as distributional information about the different senses of an ambiguous word are collected. In the testing phase, the sense with the highest probability computed on the basis of the training data (context words) is chosen.

Training and evaluating such a supervised algorithm presupposes the existence of disambiguated material, i.e. sense-tagged corpora. When one is working on a language where no such material exists, several problems arise. For training, a possible solution is the use of an unsupervised approach (see e.g. (Yarowsky, 1995)), but for evaluation purposes sense-tagged material is still needed. To be able to evaluate the performance of a particular algorithm, a possibility is to artificially create such data using pseudowords.

3 Pseudowords

The technique of pseudowords consists of introducing a form of artificial ambiguity in (untagged) corpora. First of all, two or more words, *sense words*, are chosen. Training then takes place on the original, 'disambiguated' corpus, collecting probabilities for the chosen sense words (see section 6.2 for a complete description of the algorithm used).

For testing, all occurrences of the sense words are replaced by a non-existing word, a *pseudoword*. The goal is then to recover the correct sense word for every pseudoword introduced in the corpus.

Gale *et al.* (1992) used pseudowords to overcome the "testing material bottleneck", as well as Schütze (1992), who tried to escape the need for hand-labeling using artificial ambiguous words for evaluation purposes.

4 Outline of the Problem

The idea to compare the task of disambiguating real ambiguous words to disambiguating artificially ambiguous words arose from our work on supervised WSD for Dutch[1]. Since there are no sense-tagged corpora available for Dutch, another means of testing algorithms has to be used. An obvious solution is the use of pseudowords: they are easily created, only raw text material is needed and any supervised algorithm can be tested. The one question that remained unanswered was whether using pseudowords would yield results comparable to real WSD and whether the seemingly 'easy way out' could really be seen as equivalent to the disambiguation of real ambiguous words.

Unfortunately, there has not been a lot of work on pseudowords and, to the best of our knowledge, no work at all on their usefulness in testing word sense disambiguation systems. The major problem involved in this comparison is to find a valid setting for a comparison: the elements to be compared—pseudowords and real ambiguous words —are too different from each other to be compared directly. Schütze (1998) explains it in the following way: "[The better performance on pseudowords] can be explained by the fact that pseudowords have two focused senses—the two word pairs they are composed of." Real ambiguous words, on the other hand, consist of subsenses that are difficult to identify for humans as well as for computers.

5 Way of Proceeding

A direct comparison of the task of WSD and the task of disambiguating pseudowords is not possible. The only way to compare these two tasks is to *indirectly* compare their results on the same corpus, using the same algorithm and general settings.

The attempted comparison does have its limitations: Although we do use the same settings for both tasks, the difference between them lies in the actual words (or pseudowords) to be disambiguated. There is no measure to express their

[1]The interest in Dutch lies grounded in the fact that we are working in the context of a project concerned with developing NLP tools for Dutch (see `http://www.let.rug.nl/~vannoord/alp`).

differences or similarities. This is precisely why there is no possibility of a direct comparison.

We decided to proceed in two steps. First, real ambiguous words were chosen from the SENSEVAL1 corpus making use of the dictionary entries as well as the training and testing material provided. Only nouns which were not ambiguous regarding part of speech and for which there was training data were taken into account (see section 6.3 for further details).

In a second step, we chose the sense words of a pseudoword according to the frequency distribution of the senses of the real ambiguous words we tested. Among the possible sense words that exhibited the same frequency distributions as the real ambiguous words, i.e. which fulfilled the constraint of having approximately the same baseline, an arbitrary selection was made.

If the results of this second task are significantly different from the results of the first task on the same corpus, this will show that the results involving pseudowords depend entirely on the choice of sense words. This means that the disambiguation of pseudowords is *not* identical to the real WSD task.

6 Settings

We will briefly talk about the corpus used, explain the classification algorithm, and specify the ambiguous words and pseudowords that were tested.

6.1 Corpus

The corpus we used in the described experiments are the English SENSEVAL1 resources[2]. The advantage of using this material is that it is (lexically) sense-tagged for a number of real ambiguous words which means that the evaluation data for real ambiguous words is at hand.

Furthermore, there have been numerous publications on the construction of the material, on choices made regarding annotation, on inter-annotator agreement, etc. (Kilgarriff, 1998; Kilgarriff and Rosenzweig, 2000), which allow for a thorough understanding of the real world disambiguation task. This is an important precondition to being able to extensively compare this task to nearly the same task using pseudowords.

6.2 Classification Algorithm

In the case of this experiment, the choice of classification algorithm does not have any influence considering the goal of the experiment. We chose to work with a *naive Bayes classifier* (Duda and Hart, 1973) because it is easy to implement, is rather fast, used fairly often, and performs relatively well in general.

In addition to that, a Bayes classifier uses only distributional information and context words to compute probabilities which corresponds to only using information which is available from the corpus itself without the need of any additional material, such as a dictionary or the like. The context words are assumed to be independent of position and of each other—they constitute a *bag of words*—which corresponds to the Bayes independence assumption.

First, the disambiguation algorithm is trained on part of the unambiguous corpus, attributing probabilities to the context words found to the right and the left of the sense word(s) for various context window sizes. This is done using Bayes rule

$$P(s_k|c) = \frac{P(c|s_k)}{P(c)} P(s_k)$$

where s_k is sense k of ambiguous word w in context $c = \{c_1, \ldots, c_n\}$, the context words within the specified context window. In the reported experiments, the context window was chosen to be 3 words to the left and the right of the ambiguous word/pseudoword.[3]

Testing takes place on the ambiguous text where the algorithm selects the most probable sense word for each pseudoword according to Bayes decision rule

$$\text{Decide } s' \text{ if } P(s'|c) > P(s_k|c) \text{ for } s_k \neq s'$$

Finally, the computed sense words are compared to the original sense words in the disambiguated corpus and the percentage of correctly

[2] Publicly available from `http://www.itri.brighton.ac.uk/events/senseval/ARCHIVE/resources.html`.

[3] Comparable results can be observed when different context sizes are used. We take an approach similar to (Chodorow et al., 2000) choosing a fixed context window size of ± 3.

disambiguated instances of pseudowords is calculated.

Despite its relatively 'naive' approach, the Naive Bayes classifier performs relatively well, especially in comparison with other, more sophisticated approaches (see (Mooney, 1996; Escudero et al., 2000) for comparative results).

6.3 Tested words/pseudowords

The perhaps most important factor in this comparison is the choice of elements of comparison, in this case the ambiguous words and the sense words chosen to constitute the different pseudowords.

The choice of ambiguous words depended, on the one hand, on the available SENSEVAL1 material (evaluation data). On the other hand, we only selected nouns which were not part-of-speech ambiguous.[4] No stemming was used. The ambiguous words and their senses[5] chosen for the experiments can be seen in table 1.[6]

The main criteria for choosing the sense words constituting the pseudowords were their frequency in the corpus as well as their part of speech. For the comparison with each ambiguous word, five arbitrary pseudowords were made up. The distribution of these pseudowords' sense words was chosen to be as similar as possible to the distribution of the different senses of the ambiguous words. An overview (including frequencies) of the pseudowords and the corresponding ambiguous word is given in table 1.

7 Results and Evaluation

The results as shown in table 2 clearly show that the performance of the naive Bayes classification algorithm used is significantly better on pseudowords than on real ambiguous words. A possible reason for this is the relatedness of sense distinctions in real ambiguous words whereas the sense words that constitute pseudowords have two very clearly distinct senses.

	Basel.	Results	Diff. [Stand. error]
accident	92.88	84.45	**- 8.43**
timwe	91.90	91.56	- 0.34
yeatra	92.47	91.77	- 0.70
peolang	93.10	91.88	- 0.59
woan	92.12	93.44	+ 0.97
goveq	92.35	91.33	- 1.14
mean			- 0.40 [± 0.89]
behaviour	95.70	84.95	**-10.75**
peostan	93.40	92.99	- 0.41
tima	95.33	95.64	+ 0.31
yeagro	95.34	94.04	- 1.30
wodat	94.92	93.79	- 1.13
gopay	95.26	96.36	+ 1.10
mean			- 0.29 [± 1.24]
excess	58.06	50.35	**- 7.71**
womuconba	58.62	71.86	+13.24
gopoemch	57.64	72.92	+15.28
dacipapro	57.71	73.98	+16.27
pemanora	58.64	73.00	+14.36
heterite	58.37	74.39	+16.02
mean			+15.03 [± 1.55]
shirt	58.98	57.50	**- 1.48**
schoclu	59.02	72.79	+13.77
mastre	58.55	74.83	+16.28
cimon	58.04	78.69	+20.65
coufam	57.96	63.91	+ 5.95
wogia	58.33	72.22	+13.89
mean			+14.1 [± 6.6]

Table 2: Results (in %)

Note that the baseline of most ambiguous nouns in the SENSEVAL corpus is relatively high which means that one sense accounts for most occurrences of the ambiguous word. This makes the disambiguation task comparatively harder and might be a possible explanation for the bad performance on real ambiguous words.

We conclude from our results that the task of disambiguating pseudowords is only comparable in a limited way to the task of disambiguating real ambiguous words. The results on pseudowords will usually be better which might lead to false assumptions about the performance of a given algorithm on the real problem.

8 Conclusion

To evaluate WSD systems, the evaluation procedure uses sense-tagged corpora. If no sense-tagged corpora are available, this standard evaluation is not possible. To circumvent this problem,

[4] A number of ambiguous words in the SENSEVAL1 material had to be simultaneously part-of-speech and lexically disambiguated, e.g. *bet*, *giant*, *promise*. There were also cases with no training material provided (*disability*, *hurdle*, *rabbit*, *steering*) which were not taken into account given that we worked with a supervised algorithm.

[5] The senses were taken from the SENSEVAL1 dictionary entries. Only the coarse-grained distinctions were taken into account.

[6] Since the sense *hairsh* does not occur in the testing data, we decided to only consider two senses for *shirt* and, consequently, for the pseudowords.

	Ambiguous/Pseudoword	Senses/Sense Words	Freq. train	Freq. test	Baseline
Ambiguous word	accident	crash	1058	248	92.88%
		chance	178	19	
Pseudowords	timwe	time	722	306	91.90%
		weekend	73	27	
	yeatra	year	708	307	92.47%
		traffic	86	25	
	peolang	people	673	268	92.10%
		language	54	23	
	woan	world	422	187	92.12%
		animal	39	16	
	goveq	government	396	184	92.35%
		equipment	31	15	
Ambiguous word	behavior	social	969	267	95.70%
		ofthing	29	12	
Pseudowords	peostan	people	673	268	93.40%
		standards	41	19	
	tima	time	722	306	95.33%
		machine	49	15	
	yeagro	year	708	307	95.34%
		growth	58	15	
	wodat	world	422	187	94.92%
		data	36	10	
	gopay	government	396	181	95.26%
		payment	30	9	
Ambiguous word	excess	aglut	103	108	58.06%
		ott	65	67	
		surplus	10	9	
		toomuch	73	2	
Pseudowords	womuconba	world	422	187	58.62%
		music	231	97	
		concert	43	16	
		battle	42	19	
	gopoemch	government	396	184	57.64%
		police	218	98	
		empire	37	16	
		champion	45	19	
	dacipapro	day	373	161	57.71%
		city	211	83	
		palace	37	16	
		protection	45	19	
	pemanora	people	673	268	58.64%
		man	377	154	
		noise	33	16	
		railway	33	19	
	heterite	head	349	150	58.37%
		team	162	72	
		river	42	16	
		technology	34	19	
Ambiguous word	shirt	teesh	132	73	57.06%
		garment	336	105	
Pseudowords	schoclu	school	178	87	59.02%
		club	140	72	
	mastre	market	190	89	58.55%
		street	158	63	
	cimon	city	211	83	58.04%
		month	130	60	
	coufam	country	201	91	57.96%
		family	117	66	
	wogia	women	189	91	58.33%
		giants	140	65	

Table 1: Overview ambiguous words and corresponding pseudowords

artificially ambiguous words (pseudowords) can be introduced in an (untagged) corpus.

In this paper, we have compared the disambiguation of real ambiguous words with that of pseudowords. We explain that pseudowords and real ambiguous words cannot be compared directly. We have chosen an indirect comparison, i.e. using the same corpus and settings.

The results obtained from disambiguating artifical ambiguous words differ greatly from the results of real ambiguous words. This indicates that pseudowords cannot be taken as a substitute for testing with real ambiguous words.

Testing of WSD algorithms is very difficult without evaluation data. The assumption that artificially created ambiguous words are a good substitute for real ambiguous words is *not* valid. Thus the initial problem—wanting to test algorithms for languages without sense tagged corpora—remains.

Acknowledgments

This research was carried out within the framework of the PIONIER Project *Algorithms for Linguistic Processing*. This PIONIER Project is funded by NWO (Dutch Organization for Scientific Research) and the University of Groningen. We are grateful to Gertjan van Noord, Menno van Zaanen, and the PIONIER Group for comments and discussions.

References

Thorsten Brants. 2000. TnT—a statistical part-of-speech tagger. In *Proceedings of the 6th Applied Natural Language Processing Conference ANLP-2000*, Seattle, WA, April 29 -May 3.

Eric Brill. 1995. Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics*, 21(4):543–565.

Martin Chodorow, Claudia Leacock, and George Miller. 2000. A topical/local classifier for word sense identification. *Computers and the humanities*, 34(1-2):115–120.

R. O. Duda and P. E. Hart. 1973. *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York.

Gerard Escudero, Lluís Màrquez, and German Rigau. 2000. A comparison between supervised learning algorithms for word sense disambiguation. In *Proceedings of the 4th Conference on Computational Natural Language Learning, CoNLL'2000*, pages 31–36, Lissabon.

Bill Gale, Kenneth Church, and David Yarowsky. 1992. Work on statistical methods for word sense disambiguation. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 54–60, Cambridge, MA.

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.

Adam Kilgarriff and Joseph Rosenzweig. 2000. Framework and results for English SENSEVAL. *Computers and the humanities*, 34(1-2):15–48.

Adam Kilgarriff. 1998. SENSEVAL: An exercise in evaluating word sense disambiguation programs. In *Proceedings LREC*, pages 581–588, Granada, May.

Raymond Mooney. 1996. Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 82–91, University of Pennsylvania.

Hinrich Schütze. 1992. Context space. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 113–120, Cambridge, MA.

Hinrich Schütze. 1998. Automatic word sense disambiguation. *Computational Linguistics*, 24(1):97–123.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, MA, June.