Domain Specific Sense Disambiguation with Unsupervised Methods

Diana Steffen[♠], Bogdan Sacaleanu[♦], Paul Buitelaar[♦]

[♠]Consultants for Language Technology
Saarbrücken, Germany
steffen@clt-st.de

[♦]DFKI GmbH
Saarbrücken, Germany
{sacaleanu,paulb}@dfki.de

Abstract

Most approaches in sense disambiguation have been restricted to supervised training over manually annotated, non-technical, English corpora. Application to a new language or technical domain requires extensive manual annotation of appropriate training corpora. As this is both expensive and inefficient, unsupervised methods are to be preferred, specifically in technical domains such as medicine. In the context of a project in the medical domain, we developed and evaluated two unsupervised methods for sense disambiguation.

1 Introduction

Although a lot of work on sense disambiguation has been reported in recent years (for an overview, see: Ide and Veronis, 1998; Kilgarriff and Palmer, 2000; Preiss and Yarowsky, 2001), most of these approaches are restricted to supervised training over manually annotated, non-technical, English corpora like SEMCOR (Fellbaum, 1997) and DSO (Ng and Lee 1996). Application of such systems to a new language or technical domain requires extensive manual annotation of appropriate training corpora. As this is both expensive and inefficient, unsupervised methods are to be preferred, specifically in technical domains such as medicine.

In the context of a project on cross-language information retrieval (CLIR) in the medical domain, we developed two unsupervised methods for sense disambiguation. The project is concerned with a systematic comparison of concept-based and corpus-based methods in medical CLIR. Primary goals of the project are: 1. to develop and evaluate methods for the effective use of multilingual semantic resources in the semantic annotation of English and German medical texts; 2. to subsequently evaluate and compare the impact of semantic information on the retrieval of these annotated texts.

The semantic resources used are UMLS[1] (Unified Medical Language System), a multilingual database of medical terms, and EuroWordNet (Vossen, 1997), which interconnects a number of wordnets for several European languages. However, given that the size of the German part in EuroWordNet is rather small, all our experiments reported here on development of a sense disambiguation system use the considerably larger GermaNet (Hamp and Feldweg, 1997) database instead.

For our experiments we used a corpus of German medical scientific abstracts, obtained from the Springer Link web site[2]. The corpus consists approximately of 1 million tokens. Abstracts are from 41 medical journals, each of which constitutes a relatively homogeneous medical sub-domain (e.g. Neurology, Radiology, etc.).

[1] http://umls.nlm.nih.gov

[2] http://link.springer.de/

The two unsupervised methods are described in the next section, followed by a detailed overview of experiments and results in Section 3., and an outlook on future work in Section 4.

2 Methods

2.1 Domain Specific Sense

Within the context of a specific technical domain, one of the senses of an ambiguous word may be more appropriate in the context of this domain than the other senses (Cucchiarelli and Velardi, 1998; Magnini and Strapparava, 2000; Magnini et al., 2001; Buitelaar, 2001; Buitelaar and Sacaleanu, 2001). Here, we describe a method that automatically determines such a domain specific sense on the basis of its statistical relevance across several domain specific corpora.

For this purpose, we first compute the domain relevance of each term and use this information to compute the cumulative relevance of each sense. As senses in GermaNet correspond to sets of similar terms (i.e. *synsets*), we may compute the relevance of each synset in which domain specific terms occur. This allows for a ranking of synsets (senses) according to domain relevance.

The relevance measure used in this process is a slightly adapted version of standard *tf.idf*, as used in vector-space models for information retrieval (Salton and Buckley, 1988):

$$rlv(t \mid d) = (1 + \log(tf_{t,d})) * \log(\frac{N}{df_t})$$

where t represents the term, d the domain (corpus), N is the total number of domain (corpora) taken into account. Term frequency has been scaled logarithmically because more occurrences of a word indicate higher importance, but not as much importance as the count solely would suggest. By scaling domain frequency as well, this formula gives full weight to terms that occur in just one domain and a weight of zero to those occurring in all domains.

Given term relevance, we are now able to compute the relevance of each synset. This is simply the sum of the relevance of each term in the synset, which may be defined as follows:

$$rlv(c \mid d) = \sum_{t \in c} rlv(t \mid d)$$

However, suppose we want to compute the relevance for the following senses (i.e. the synsets in which this term occurs) of `Zelle`:

```
[Zelle,Gefängniszelle] prison cell
[Zelle] living cell
```

Although `Zelle` will have a high relevance in the medical domain, the occurrence of `Gefängniszelle` in this domain is very unlikely and therefore the relevance value of both concepts will be equal. Although the latter concept is more relevant to the medical domain, we would not be able to automatically determine this by merely adding up the relevance of the terms in each of the synsets. Therefore we reconsidered the concept relevance definition to take into account the number of terms in the synset that actually occur in the domain corpus:

$$rlv(c \mid d) = \frac{T}{|c|} \sum_{t \in c} rlv(t \mid d)$$

where T represents the lexical coverage, and $|c|$ is the length of synset c. This relevance measure reflects the intuition that if many terms in the synset occur in the domain, then the more likely it is that the synset is relevant for that domain.

To increase the number of terms to be found within a domain corpus, we considered adding hyponyms to a given synset as these are often directly related. For example, the two synsets for `Zelle` can be extended with hyponyms as follows:

```
[Zelle,Gefängniszelle,Todeszelle]
[Zelle,Körperzelle,Pflanzenzelle]
```

Adding hyponyms changes the relevance formula accordingly:

$$rlv(c+ \mid d) = \frac{T}{|c|} \sum_{t \in c+} rlv(t \mid d)$$

where c+ is the extended synset. Note that *T* (number of terms in the concept that occur in the domain) and |*c*| (number of terms in the synset) have not changed. That is, hyponyms do not affect lexical coverage, but only add to the summed weight of the synset.

2.2 Instance-Based Learning

2.2.1 Introduction

The second method we used in our experiments implements a k-nearest neighbor instance-based learning algorithm using the WEKA[3] suit of machine-learning tools (Witten and Eibe, 2000). In this method, sense disambiguation is seen as a classification task, in which an ambiguous word needs to be classified to the appropriate class given a particular context.

There have been several reports on the use of instance-based learning in sense disambiguation (Ng and Lee, 1996; Mihalcea, 2002). However, all of these approaches were supervised, based on a manually annotated training corpus. Here we report on the use of instance-based learning in an unsupervised manner by generalizing over Resnik`s work on selection restrictions (Resnik, 1997).

The basic idea is as follows. Consider these (ambiguous and non-ambiguous) instances of the verb `drink` in the context of the semantic classes (i.e. senses) `FOOD` and `LIQUID`:

```
He drank coffee <LIQUID>
He drank tea <LIQUID>
He drank chocolate <FOOD,LIQUID>
```

From these examples we may infer that the verb drink has a preference for the semantic class `LIQUID`. We can apply this in the disambiguation of the following ambiguous instance (and select `LIQUID`):

```
He drank Java <GEOGAPHICAL,LIQUID>
```

2.2.2 Algorithm

An instance-based learning algorithm consists of a training step and an application step:

Training: Collecting classified instances from a training corpus (as our method is unsupervised, this corpus has not been previously annotated)

An instance is a set of attribute-value pairs, one of which identifies the class attribute. Classifying an instance then means finding the missing value for this class attribute.

Constructing an instance involves the following. Let *w* be a word in the training corpus. We can build instances for *w*, where the values of the attributes are always its left and right neighbor words in a context of size n, and the value of the class attribute varies over its senses.

Collecting classified instances from the training corpus may now be defined as follows. Given a training corpus annotated with part-of-speech and morphology, for any ambiguous word *w* and its set of senses *S*:

- determine all contexts, in which *w* occurs organized according to part-of-speech pattern
- for every part-of-speech pattern *p* collect all instances corresponding to contexts of *w* of pattern *p* in the training corpus, under the constraint that the value of the class attribute belongs to *S*

To illustrate the construction of particular instances, consider the following sentence from our corpus:

In dem Fall, sind korrigierende Eingriffe nur eingeschränkt möglich.
(In this case the possibility of corrective surgery is limited.)

The ambiguous word *Eingriff* has the following two senses (identified by their GermaNet synset ids):

460326: Operation, Eingriff
388935: Eingriff, Intervention, Eingreifen

From the sentence we may now derive the following instances for *Eingriff* with context size 5 (2 words on the left, 2 words on the right) of the part-of-speech pattern

[3] http://www.cs.waikato.ac.nz/~ml/weka/

-, ADJ, NOUN, -, VERB

where *Eingriff* takes the position of the *NOUN* and '-' stands for other parts-of-speech:

[sein, korrigieren, nur, einschränken,388935]
[sein, korrigieren, nur, einschränken,460326]

Application: Classifying an occurrence of an ambiguous word *w* by finding the *k* most similar training instances:

- determine its part-of-speech pattern *p*
- extract the corresponding set of instances *I(w,p)* as found in the training step

For instance, the set:

[und,therapeutisch,werden,vorstellen,388935]
[und,therapeutisch,werden,vorstellen, 460326]
[ein,chirurgisch,nicht,profitieren ,388935]
[sein,korrigieren,stets,ermoeglichen , 460326]
[oder,offen,zu,erfassen, 460326]
[sein, korrigieren,nur,einschraenken,388935]
[sein, korrigieren,nur,einschraenken,460326]

- delete all instances corresponding to the occurrence (i.e. instances for the occurrence that correspond to each sense – the last two instances in the example set), resulting in the set of instances *I´(w,p)*
- create an instance for the occurrence, with the class attribute missing:

[sein, korrigieren,nur,einschränken,?]

- classify the instance using *I´(w,p)*

The algorithm does not return a specific sense, but a probability distribution over all senses of the ambiguous word. We assign the sense with highest probability to the corresponding word occurrence. If such a sense does not exist, no decision is made.

3 Evaluation

3.1 Evaluation Corpus

An important aspect in the development of a word sense disambiguation system is the evaluation of different methods and parameters. Unfortunately, there is a lack of test sets for evaluation, specifically for languages other than English and even more so for specific domains like medicine. Given that our work focuses on German text in the medical domain, we needed to construct an evaluation corpus specifically for this purpose.

Selection of ambiguous GermaNet terms to be included in the evaluation corpus proceeded in several steps. First, we calculated relevance values regarding the medical domain for all GermaNet noun synsets occurring in the medical corpus, using the method described in Section 2.1. Given these relevance values, we compiled a list of terms with high relevance, at least 100 occurrences in the medical corpus and with more than one synset in GermaNet. This produced a list of 40 terms, for each of which we then automatically extracted 100 occurrences at random.

Three annotators (a medical expert and two linguistics students) annotated the occurrences of the 40 ambiguous terms. They were allowed to annotate an occurrence with more than one sense if needed or with *undef*, if GermaNet did not contain any appropriate sense. With a further arbitration step to settle any disagreement cases they then produced together a gold standard. Removing the occurrences annotated with *undef* from the gold standard gave us the final evaluation corpus, which we used in our experiments.

3.2 Experiments

The evaluation corpus was used to experiment with the previously mentioned methods and a combination thereof. For each experiment we computed *recall* (number of correctly disambiguated occurrences divided by the number of occurrences to be disambiguated) and *precision* (number of correctly[4] disambiguated occurrences divided by the number of disambiguated occurrences). A theoretical baseline for the evaluation corpus was computed as follows, where GS means gold standard and GN means GermaNet:

[4] If an occurrence was assigned several senses by the human annotators and the system delivered one of them, we counted the occurrence as *correct*.

$$prec_{random} = \frac{1}{|GS|} \sum_{o \in GS} \frac{|GS_{sense(s)}|}{|GN_{senses}|}$$

For every occurrence in the gold standard, the probability of assigning it the correct sense is computed by dividing the number of senses in the gold standard by the number of corresponding GermaNet senses. The average precision is the sum of all probabilities divided by the number of all occurrences. For our evaluation corpus the precision (= recall) is 36%, by a coverage of 100% (F-measure F1: 0.36).

3.2.1 Domain Specific Sense

The identification of domain specific senses has been evaluated as an individual component in (Buitelaar and Sacaleanu, 2001). Here we evaluated this method as part of a broader sense disambiguation system. For all GermaNet senses in the training corpus we computed a domain relevance score, according to the method described in Section 2.1 We experimented with different sets of domain specific corpora and with different corpora sizes. The corpora used are:

sp	Springer (medical abstracts)
dp	Deutsche Presse Agentur (news)
fb	Fussball (soccer game reports)
wr	Wirtschaftswoche (economic news)
rd	Radiology (examination reports)

In disambiguation, the sense with the highest domain relevance was selected. If no sense had a relevance value, no decision was made. Table 1. shows the evaluation results for different corpora sets and sizes:

Corpora	Size	Rec	Prec	F1
sp-dp-fb-wr	2Mb	4%	77%	0.08
sp-dp	2Mb	6%	99%	0.11
sp-dp	10Mb	4%	26%	0.07
rd-dp-fb-wr	2Mb	17%	44%	0.24
rd-dp	2Mb	9%	50%	0.15
rd-dp	10Mb	3%	34%	0.05

Table 1: Domain Specific Sense

Unfortunately, F-measure results show that none of the experiments actually improve on the baseline mentioned above. However, as will be discussed in the next section, a combination of this method with the instance-based learning method does result in an improvement if compared to the use of instance-based learning by itself.

In terms of recall and precision we can observe the following. Precision reaches highest values when the domain specific corpora are small (i.e. 2 Mb). Large corpora have a correspondingly large set of common terms, for which the relevance score will be zero[5] – see Section 2.1.

3.2.2 Instance-Based Learning (IBL)

In the training and application steps we experimented with four parameters.

- **Training Corpus**: Springer (S) vs. Radiology (R)

We were interested to see how well our system performs when training and application use the same corpus compared to when the training corpus (Radiology reports) is different from the test corpus (Springer medical abstracts), but still belonging to the same domain.

- **Context Size**: 3 vs. 5 words

We were interested to measure the effect of larger vs. smaller contaxt sizes. Larger contexts give a higher precision, but will have less instances – with correspondingly fewer occurrences that can be disambiguated .

- **Part-of-Speech Selection**: all PoS (all) vs. only nouns, verbs or adjectives (N/V/A)

We wanted to find out if words with little content have any influence on the disambiguation result. In order to discard them in some experiments we gave all

[5] In further experiments we intend to adjust the term relevance measure so as to assign a non-zero weight even to those terms occurring in all domains.

attributes corresponding to parts-of-speech other than N/V/A the value *null*.

- **Attribute-Values**: lemmas vs. lemmas and synsets

It is hard to classify instances with attribute-values (i.e. particular lemmas), which do not occur in the training corpus. We introduced synsets (i.e. senses) as values for these attributes. This in effect maps a particular lemma to a set of lemmas, thereby reducing this sparse data problem.

Corpus; Context Size, PoS		Lemma			Lemma + Synsets		
		Rec	**Prec**	**F1**	**Rec**	**Prec**	**F1**
S	**3-all**	**30%**	**49%**	**0.37**	29%	48%	0.36
	3-N/V/A	17%	43%	0.24	17%	43%	0.24
	5-all	18%	54%	0.27	17%	53%	0.26
	5-N/V/A	21%	47%	0.29	21%	48%	0.29
R	3-all	21%	43%	0.28	21%	43%	0.28
	3-N/V/A	13%	44%	0.20	13%	44%	0.20
	5-all	13%	42%	0.20	13%	43%	0.20
	5-N/V/A	16%	48%	0.24	16%	46%	0.24

Table 2: Instance-Based Learning

Training Corpus: As we expected, precision and recall are better when the training corpus is the same with the test corpus.

Context Size: We cannot say much about recall if we only consider the context size. This is only relevant together with the part-of-speech selection. Best recall values are reached with context size 3 and all parts-of-speech, followed by context size 5 with nouns, verbs and adjectives. On the other hand, precision will be highest when using larger contexts (5), as these will contain more words that contribute to the selection of a particular sense.

Part-of-Speech Selection: With contexts of size 3 precision values are better when using all parts-of-speech. This makes sense, because very often in small contexts no noun, verb or adjective occurs and therefore we can not build any useful training instances. With context size 5, different training corpora produce different results. For Springer, the results are better when using all parts-of-speech (54% vs. 47%), while for Radiology the use of only nouns, verbs and adjectives will do a better job (48% vs. 42%).

Attribute-Values: Using synsets as values in addition to lemmas does not bring any improvement, but rather some slight degradation of results.

3.2.3 Combination of Methods

Here we used the domain relevance values which led to the best results in the first set of experiments and the sets of training instances generated for the second set of experiments. For every occurrence of an ambiguous word we applied the two methods disjunctively, that is, if the first method could not make any decision, the second one was applied. The order in which the methods were applied is an extra parameter. Table 3 shows the results:

Corpus; Context Size, PoS		IBL→ DOMSpecSense			DOMSpecSense → IBL		
		Rec	**Prec**	**F1**	**Rec**	**Prec**	**F1**
S	**3-all**	**35%**	**52%**	**0.42**	**35%**	**53%**	**0.42**
	3-N/V/A	23%	50%	0.31	24%	51%	0.33
	5-all	25%	60%	0.35	25%	60%	0.35
	5-N/V/A	27%	53%	0.36	27%	53%	0.36
R	3-all	29%	45%	0.35	28%	44%	0.34
	3-N/V/A	24%	44%	0.31	24%	43%	0.31
	5-all	24%	46%	0.31	24%	44%	0.31
	5-N/V/A	27%	49%	0.35	26%	46%	0.33

Table 3: IBL and Domain Specific Sense

It is interesting to note that instance-based learning produces better results (precision as well as recall) in combination with the domain specific sense method. In these experiments we used the domain specific senses that were computed with the corpus set *sp-dp-fb-wr* and a corpus size of *2Mb*.

In comparison to the domain specific sense method, recall is much better in combination with instance-based learning, which was of course to be expected. However, precision (highest at 60%) does not reach the highest

result (77%) that we saw when using the domain specific sense method by itself.

Finally, we note that the order of applying the two methods has some significance. Applying the instance-based learning method first produces slightly better results than applying the domain specific sense method first. This may result from the fact that the domain specific sense method always selects the same sense for every occurrence whereas the instance-based learning method selects a sense depending on a particular context.

3.3 Related Work

Unfortunately, a straightforward comparison of our work with other related work in sense disambiguation is not possible, as German medical language has not been studied widely in this respect. Nevertheless, some work has been done on sense disambiguation in other languages, primarily for English (Rindflesch et al., 1994; Weeber et al., 2001; Liu et al., 2001), but also for instance for French (Bouillon et al., 2000). The work most similar to our work is that of (Liu et al., 2001), who also report on an unsupervised method for sense disambiguation in medical text The object of this study, however, is the ambiguity of medical terms as specified in the medical semantic resource UMLS whereas we report on the disambiguation of more general terms as used in medical text.

4 Future Work

Sense disambiguation is concerned with the selection of the appropriate interpretation of a word in respect of a given semantic lexicon. Obviously this implies that the word at hand is represented in the lexicon, which is often not the case. However, in semantic tagging, the task of mapping words to semantic classes, we would like to tag each word and not only those occurring in the given semantic lexicon.

Therefore, in addition to sense disambiguation of *known* words we need to classify *unknown* words. As senses may be simply viewed as semantic classes, these tasks can also be combined. In this respect we intend to treat classification of unknown words as sense disambiguation between a dynamically selected set of domain specific senses (i.e. semantic classes).

5 Conclusions

In this paper we describe two unsupervised methods to sense disambiguation of terms in medical text. The first method automatically determines a domain specific sense on the basis of its statistical relevance across several domain specific corpora. The second approach implements a k-nearest neighbor instance-based learning algorithm. Experiments with an evaluation corpus built specifically for this task show that a combination of the two methods produces the best results.

Acknowledgements

This research has in part been supported by EC/NSF grant *IST-1999-11438* for the MUCHMORE project.

References

Bouillon P., Baud R., Robert G., Ruch P. 2000. Indexing by Statistical Tagging. In: Proceedings of JADT 2000.

Buitelaar, P. & Sacaleanu, B. 2001. *Ranking and Selecting Synsets by Domain Relevance*. In: Proceedings NAACL WordNet Workshop.

Buitelaar, P. 2001. *The SENSEVAL-II Panel on Domains, Topics and Senses* In: Proceedings of SENSEVAL-II , Toulouse.

Cucchiarelli, A. & Velardi, P. 1998. *Finding a Domain-Appropriate Sense Inventory for Semantically Tagging a Corpus.* In: Journal of Natural language Engineering.

Fellbaum Chr. 1997. *Analysis of a hand-tagging task.* Proceedings of ANLP-97 Workshop on Tagging Text with Lexical Semantics: Why, What, and How? Washington D.C., USA.

Hamp, B. & Feldweg, H. 1997. *GermaNet: a Lexical-Semantic Net for German.* In: Proceedings of the ACL/EACL97 workshop on Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications, Madrid

Ide, N. & Véronis, J. (Eds.). 1998. *Word Sense Disambiguation* Introduction to a

Special Issue of Computational Linguistics, 24(1).

Kilgarriff, A. & M. Palmer. 2000. *Introduction to the special issue on SENSEVAL*. Computers and the Humanities 34(1/2):1-13.

Liu H., Lussier Y., Friedman C. 2001. Disambiguating Ambiguous Biomedical Terms in Biomedical Narrative Text: An Unsupervised Method. In: Journal of Biomedical Informatics, Vol. 34, No. 4.

Magnini, B. & Strapparava, C. 2000. *Experiments in Word Domain Disambiguation for Parallel Texts*. In: Proc. of SIGLEX Workshop on Word Senses and Multi-linguality, Hong-Kong.

Magnini B., Strapparava, C., Pezzulo G., Gliozzo A. 2001. *Using Domain Information for Word Sense Disambiguation*. In: Proceedings of SENSEVAL-2, ACL, Toulouse, France.

Mihalcea R. 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.

Miller G., Chodorow M., Landes S., Leacock C., Thomas R. 1994 *Using a Semantic Concordance for Sense Identification*. In: ARPA Workshop on Human Language Technology, Plainsboro NJ.

Miller, G.A. 1995. *WordNet: A Lexical Database for English*. Communications of the ACM 11.

Ng, H.T. & Lee, H.B. 1996. *Integrating multiple knowledge sources to disambiguate word sense: An examplar--based approach*. In: Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL), Pages 40--47.

Preiss J. & Yarowsky D. 2001. Proceedings of SENSEVAL-2, ACL, Toulouse, France.

Resnik, P. 1997. *Selectional preference and sense disambiguation*. In: Proceedings of ANLP-97 Workshop on Tagging Text with Lexical Semantics: Why, What. and How?, Washington D.C., USA.

Rindflesch, T. C. & Aronson A. R. 1994. *Ambiguity Resolution while Mapping Free Text to the UMLS Metathesaurus*. In: Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care.

Salton, G. and Buckley, C. 1988. *Term-Weighting Approaches In Automatic Text Retrieval*. In: Information Processing & Management. 24, 5, pp.515-523.

Vossen P. 1997. *EuroWordNet: a multilingual database for information retrieval*. In: Proc. of the DELOS workshop on Cross-language Information Retrieval, March 5-7, Zürich, Switzerland.

Witten, I. H. & Eibe F. 2000. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Series in Data Management Systems.

Weeber M., Mork J., Aronson A.R. 2001. *Developing a Test Collection for Biomedical Word Sense Disambiguation*. In: Proceedings of AMIA 2001.