Unknown word sense detection as outlier detection

Katrin Erk
Computational Linguistics
Saarland University
Saarbrücken, Germany
erk@coli.uni-sb.de

Abstract

We address the problem of *unknown word sense detection*: the identification of corpus occurrences that are not covered by a given sense inventory. We model this as an instance of *outlier detection*, using a simple nearest neighbor-based approach to measuring the resemblance of a new item to a training set. In combination with a method that alleviates data sparseness by sharing training data across lemmas, the approach achieves a precision of 0.77 and recall of 0.82.

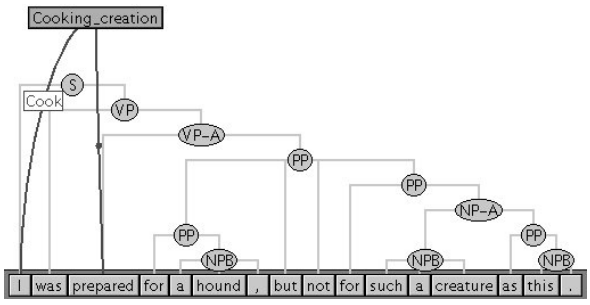


Figure 1: Wrong assignment due to missing sense: from the Hound of the Baskervilles, Ch. 14

1 Introduction

If a system has seen only positive examples, how does it recognize a negative example?[1] This is the problem addressed by *outlier detection*, also called *novelty detection*[2] (Markou and Singh, 2003a; Markou and Singh, 2003b; Marsland, 2003): to detect novel or unknown items that differ from all the seen training data. Outlier detection approaches typically derive some model of "normal" objects from the training set and use a distance measure and a threshold to detect abnormal items.

In this paper, we apply outlier detection techniques to the task of *unknown sense detection*: the identification of corpus occurrences that are not covered by a given sense inventory. The training set against which new occurrences are compared will consist of sense-annotated text.

Unknown sense detection is related to word sense disambiguation (WSD) and to word sense discrimination (Schütze, 1998), but differs from both. In WSD all senses are assumed known, and the task is to select one of them, while in unknown sense detection the task is to decide whether a given occurrence matches any of the known senses or none of them, and all training instances, regardless of the sense to which they belong, are modeled as *one* group of *known* data. Unknown sense detection also differs from word sense discrimination, where no sense inventory is given and the task is to group occurrences into senses. In unknown sense detection the model respects the given word senses.

The main motivation for this study comes from

[1]We acknowledge the funding of the German Science Foundation DFG for the SALSA project (Title PI 154/9-2).

[2]The term *novelty detection* is also used for the distinction of novel and repeated information in information retrieval, a different if related topic.

shallow semantic parsing, by which we mean a combination of WSD and the automatic assignment of semantic roles to free text. In cases where a sense is missing from the inventory, WSD will wrongly assign one of the existing senses. Figure 1 shows an example, a sentence from the *Hound of the Baskervilles*, analyzed by the SHALMANESER (Erk and Pado, 2006) shallow semantic parser. The analysis is based on FrameNet (Baker et al., 1998), a resource that lists senses and semantic roles for English expressions. FrameNet is lacking a sense of "expectation" or "being mentally prepared" for the verb *prepare*, so *prepared* has been assigned the sense COOKING_CREATION, a possible but improbable analysis[3]. Such erroneous labels can be fatal when further processing builds on the results of shallow semantic parsing, e.g. for drawing inferences. Unknown sense detection can prevent such mistakes.

All sense inventories face the problem of missing senses, either because of their small overall size (as is the case for some non-English WordNets) or when they encounter domain-specific senses. Our study will be evaluated on FrameNet because of our main aim of improving shallow semantic parsing, but the method we propose is applicable to any sense inventory that has annotated data; in particular, it is also applicable to WordNet.

In this paper we model unknown sense detection as outlier detection, using a simple Nearest Neighbor-based method (Tax and Duin, 2000) that compares the local probability density at each test item with that of its nearest training item.

To our knowledge, there exists no other approach to date to the problem of detecting unknown senses. There are, however, approaches to the complementary problem of determining the closest known sense for unknown words (Widdows, 2003; Curran, 2005; Burchardt et al., 2005), which can be viewed as the logical next step after unknown sense detection.

Plan of the paper. After a brief sketch of FrameNet in Section 2, we describe the experimental setup used throughout this paper in Section 3. Section 4 tests whether a very simple model suffices for detecting unknown senses: a threshold on confidence scores returned by the SHALMANESER WSD system. The result is that recall is much too low. Section 5 introduces the NN-based outlier detection approach that we use in section 6 for unknown sense detection, with better results than in the first experiment but still low recall. Section 7 repeats the experiment of section 6 with added training data, making use of the fact that one semantic class in FrameNet typically pertains to several lemmas and achieving a marked improvement in results.

[3]Unfortunately, the semantic roles have been mis-assigned by the system. The word *I* should fill the FOOD role, while *for a hound* could be assigned the optional RECEIVER role.

2 FrameNet

Frame Semantics (Fillmore, 1982) models the meanings of a word or expression by reference to *frames* which describe the background and situational knowledge necessary for understanding what the predicate is "about". Each frame provides its specific set of semantic roles.

The Berkeley FrameNet project (Baker et al., 1998) is building a semantic lexicon for English describing the frames and linking them to the words and expressions that can *evoke* them. These can be verbs as well as nouns, adjectives, prepositions, adverbs, and multiword expressions. Frames are linked by IS-A and other relations. Currently, FrameNet contains 609 frames with 8,755 lemma-frame pairs, of which 5,308 are exemplified in annotated sentences from the British National Corpus. The annotation comprises 133,846 sentences.

As FrameNet is a growing resource, many lemmas are still lacking senses, and many senses are still lacking annotation. This is problematic for the use of FrameNet analyses as a basis for inferences over text, as e.g. in Tatu and Moldovan (2005).

For example, the verb *prepare* from Figure 1 is associated with the frames

COOKING_CREATION: prepare food
ACTIVITY_PREPARE: get ready for an activity
ACTIVITY_READY_STATE: be ready for an activity
WILLINGNESS: be willing

of which only the COOKING_CREATION sense has been annotated. The sense in Figure 1 is not covered yet: ACTIVITY_READY_STATE would be more appropriate than COOKING_CREATION, but still not optimal, since the sentence refers to a mental state rather than the preparation of an activity.

3 Experimental setup and data

Experimental setup. To evaluate an unknown sense detection system, we need occurrences that are guaranteed not to belong to any of the seen senses. To that end we use sense-annotated data, in our case the FrameNet annotated sentences, simulating unknown senses by designating one sense of each ambiguous lemma as unknown. All occurrences of that sense are placed in the test set, while occurrences of all other senses are split randomly between training and test set, using 5-fold cross-validation. We repeat the experiment with each of the senses of an ambiguous lemma playing the part of the unknown sense once. Viewing each cross-validation run for each unknown sense as a separate experiment, we then report precision and recall averaged over unknown senses and cross-validation runs.

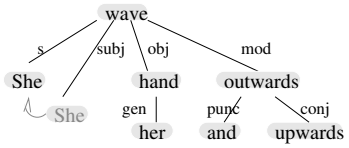
It may seem questionable that in this experimental setup, the *unknown sense* occurrences of each lemma all belong to the same sense. However, this does not bias the experiment since none of the models we study take advantage of the shape of the test set in any way. Rather, each test item is classified individually, without recourse to the other test items.

Data. All experiments in this paper were performed on the FrameNet 1.2 annotated data pertaining to ambiguous lemmas. After removal of instances that were annotated with more than one sense, we obtain 26,496 annotated sentences for the 1,031 ambiguous lemmas. They were parsed with Minipar (Lin, 1993); named entities were computed using Heart of Gold (Callmeier et al., 2004).

4 Experiment 1: WSD confidence scores for unknown sense detection

In this section we test a very simple model of unknown sense detection: Classifiers often return a confidence score along with the assigned label. We will try to detect unknown senses by a threshold on confidence scores, declaring anything below the threshold as unknown. Note that this method can only be applied to lemmas that have more than one sense, since for single-sense lemmas the system will always return the maximum confidence score.

Data. While the approach that we follow in this section is applicable to all lemmas with at least two senses, we need lemmas with at least three senses to evaluate it: One of the senses of each lemma is treated as *unknown*, which for lemmas with three or more senses leaves at least two senses for the training set. This reduces our data set to 125 lemmas with 7,435 annotated sentences.



(1): `subj`, `obj`, `mod` (since `s` and `subj` corefer, we use only one of them)
(2): `she`, `hand`, `outwards`
(3): `subj-she`, `obj-hand`, `mod-outwards`
(4): `mod-obj-subj`

Figure 2: Sample Minipar parse and extracted grammatical function features

Modeling. We test whether the WSD system built into SHALMANESER (Erk, 2005) can distinguish *known sense* items from *unknown sense* items reliably by its confidence scores. The system extracts a rich feature set, which forms the basis of all three experiments in this paper:

- a bag-of-words context, with a window size of one sentence;
- bi- and trigrams centered on the target word;
- grammatical function information: for each dependent of the target, (1) its function label, (2) its headword, and (3) a combination of both are used as features. (4) The concatenation of all function labels constitutes another feature. For PPs, function labels are extended by the preposition. As an example, Figure 2 shows a BNC sentence and its grammatical function features.
- for verb targets, the target voice.

The feature set is based on Florian et al. (2002) but contains additional syntax-related features. Each word-related feature is represented as four features for word, lemma, part of speech, and named entity.

SHALMANESER trains one Naive Bayes classifier per lemma to be disambiguated. For this experiment,

θ	Precision		Recall	
0.5	0.6524	(σ 0.115)	0.0011	(σ 0.0004)
0.75	0.7855	(σ 0.0086)	0.0527	(σ 0.0013)
0.9	0.7855	(σ 0.0093)	0.1006	(σ 0.0021)
0.98	0.7847	(σ 0.0073)	0.1744	(σ 0.0025)

Table 1: Experiment 1: Results for label *unknown sense*, WSD confidence level approach. θ: confidence threshold. σ: std. dev.

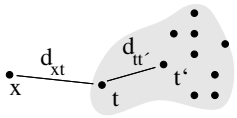


Figure 3: Outlier detection by comparing distances between nearest neighbors

all system parameters were set to their default settings. To detect unknown senses building on this WSD system, we use a fixed confidence threshold and label all items below the threshold as *unknown*.

Results and discussion. Table 1 shows precision and recall for labeling instances as *unknown* using different confidence thresholds θ, averaged over unknown senses and 5-fold cross-validation[4]. We see that while the precision of this method is acceptable at 0.74 to 0.765, recall is extremely low, i.e. almost no items were labeled *unknown*, even at a threshold of 0.98. However, SHALMANESER has very high confidence values overall: Only 14.5% of all instances in this study had a confidence value of 0.98 or below (7,697 of 53,206).

We conclude that with the given WSD system and (rather standard) features, this simple method cannot detect items with an unknown sense reliably. This may be due to the indiscriminately high confidence scores; or it could indicate that classifiers, which are geared at *distinguishing* between known classes rather than *detecting* objects that differ from all seen data, are not optimally suited to the task. However, one further disadvantage of this approach is that, as mentioned above, it can only be applied to lemmas with more than one annotated sense. For FrameNet 1.2, this comprises only 19% of the lemmas.

5 A nearest neighbor-based method for outlier detection

In the previous section we have tested a simple approach to unknown sense detection using WSD confidence scores. Our conclusion was that it was not a viable approach, given its low recall and given that it is only applicable to lemmas with more than one known sense. In this section we introduce an alternative approach, which uses distances to nearest neighbors to detect outliers.

In general, the task of outlier detection is to decide whether a new object belongs to a given training set or not. Typically, outlier detection approaches derive some boundary around the training set, or they derive from the set some model of "normality" to which new objects are compared (Markou and Singh, 2003a; Markou and Singh, 2003b; Marsland, 2003). Applications of outlier detection include fault detection (Hickinbotham and Austin, 2000), hand writing deciphering (Tax and Duin, 1998; Schölkopf et al., 2000), and network intrusion detection (Yeung and Chow, 2002; Dasgupta and Forrest, 1999). One standard approach to outlier detection estimates the probability density of the training set, such that a test object can be classified as an outlier or non-outlier according to its probability of belonging to the set.

Rather than estimating the complete density function, Tax and Duin (2000) approximate local density at the test object by comparing distances between nearest neighbors. Given a test object x, the approach considers the training object t nearest to x and compares the distance d_{xt} between x and t to the distance $d_{tt'}$ between t and its own nearest training data neighbor t'. Then the quotient between the distances is used as an indicator of the (ab-)normality of the test object x:

$$p_{NN}(x) = \frac{d_{xt}}{d_{tt'}}$$

When the distance d_{xt} is much larger than $d_{tt'}$, x is considered an outlier. Figure 3 illustrates the idea.

The normality or abnormality of test objects is decided by a fixed threshold θ on p_{NN}. The lowest

[4]Note that the minimum confidence score is 0.5 if 2 senses are present in the training set, 0.33 for 3 present senses etc.

threshold that makes sense is 1.0, which rejects any x that is further apart from its nearest training neighbor t than t is from its neighbor. Tax and Duin use Euclidean distance, i.e.

$$d_{xt} = \sqrt{\sum_i (x_i - t_i)^2}$$

Applied to feature vectors with entries either 0 or 1, this corresponds to the size of the symmetric difference of the two feature sets.

6 Experiment 2: NN-based outlier detection

In this section we use the NN-based outlier detection approach of the previous section for an experiment in unknown sense detection. Experimental setup and data are as described in Section 3.

Modeling. We model unknown sense detection as an outlier detection task, using Tax and Duin's outlier detection approach that we have outlined in the previous section. Nearest neighbors (by Euclidean distance) were computed using the ANN tool (Mount and Arya, 2005). We compute one outlier detection model per lemma. With training and test sets constructed as described in Section 3, the average training set comprises 22.5 sentences.

We use the same features as in Section 4, with feature vector entries of 1 for present and 0 for absent features. For a more detailed analysis of the contribution of different feature types, we test on reduced as well as full feature vectors:

`All`: full feature vectors

`Cx`: only bag-of-word context features (words, lemmas, POS, NE)

`Syn`: function labels of dependents

`Syn-hw`: `Syn` plus headwords of dependents

We compare the NN-based model to that of Experiment 1, but not to any simpler baseline. While for WSD it is possible to formulate simple frequency-based methods that can serve as a baseline, this is not so in unknown sense detection because the frequency of unknown senses is, by definition, unknown. Furthermore, the number of annotated sentences per sense in FrameNet depends on the number of subcategorization frames of the lemma rather than the frequency of the sense, which makes frequency calculations meaningless.

Features	Precision		Recall	
`All`	0.7072	(σ 0.0088)	0.2683	(σ 0.0043)
`Cx`	0.7016	(σ 0.0041)	0.3511	(σ 0.0035)
`Syn`	0.8333	(σ 0.0085)	0.2099	(σ 0.0042)
`Syn-hw`	0.7784	(σ 0.0029)	0.2368	(σ 0.0022)

Table 2: Experiment 2: Results for label *unknown sense*, NN-based outlier detection, $\theta = 1.0$. σ: standard deviation

	Precision			Recall		
Features	all	≥ 10	≥ 20	all	≥ 10	≥ 20
`All`	0.71	0.70	0.67	0.27	0.35	0.45
`Cx`	0.70	0.70	0.67	0.35	0.47	0.58
`Syn`	0.83	0.81	0.77	0.21	0.22	0.21
`Syn-hw`	0.78	0.76	0.73	0.24	0.28	0.31

Table 3: Experiment 2: Results by training set size, $\theta = 1.0$

Results. Table 2 shows precision and recall for labeling instances as *unknown* using a distance quotient threshold of θ=1.0, averaged over unknown senses and over 5-fold cross-validation. We see that recall is markedly higher than in Experiment 1, especially for the two conditions that include context words, `All` and `Cx`. The syntax-based conditions `Syn` and `Syn-hw` show a higher precision, with a less pronounced increase in recall.

Raising the distance quotient threshold results in little change in precision, but a large drop in recall. For example, `All` vectors with a threshold of $\theta = 1.1$ achieve a recall of 0.14 in comparison to 0.27 for $\theta = 1.0$.

Training set size is an important factor in system results. Table 3 lists precision and recall for all training sets, for training sets of size ≥ 10, and for training sets of size ≥ 20. Especially in conditions `All` and `Cx`, recall rises steeply when we only consider cases with larger training sets. However note that precision does not rise with larger training sets, rather it shows a slight decline.

Another important factor is the number of senses that a lemma has, as the upper part of Table 7 shows. For lemmas with a higher number of senses, preci-

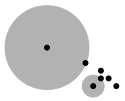


Figure 4: "Acceptance radius" of an outlier within the training set (left) and a more "normal" training set object (right)

sion is much lower, while recall is much higher.

Discussion. While results in this experiment are better than in Experiment 1 – in particular recall has risen by 19 points for `Cx` –, system performance is still not high enough to be usable in practice.

The uniformity of the training set has a large influence on performance, as Table 7 shows. The more senses a lemma has, the harder it seems to be for the model to identify *known sense* occurrences. Precision for the assignment of the *unknown* label drops, while recall rises. We see a tradeoff between precision and recall, in this table as well as in Table 3. There, we see that many more *unknown* test objects are identified when training sets are larger, but a larger training set does not translate into universally higher results.

One possible explanation for this lies in a property of Tax and Duin's approach. If a training item t is situated at distance d from its nearest neighbor in the training set, then any test item within a radius of d around t will be considered *known*. Thus we could term d the "acceptance radius" of t. Now if t is an outlier *within* the training set, then d will be large, as illustrated in Figure 4. The sparser the training set is, the more training outliers we are likely to find, with large acceptance radii that assign a label of *known* even to more distanced test items. Thus a sparse training set could lead to lower recall of *unknown sense* assignment and at the same time higher precision, as the items labeled *unknown* would be the ones at great distance from any items on the training set – conforming to the pattern in Tables 3 and 7.

7 Experiment 3: NN-based outlier detection with added training data

While the NN-based outlier detection model we used in the previous experiment showed better results than the WSD confidence model, its recall is still low. We have suggested that data sparseness may be responsible for the low performance. Consequently, we repeat the experiment of the previous section with more, but less specific, training data.

Target lemma: put
Senses: ENCODING, PLACING
Sense currently treated as unknown: PLACING
Extend training set by: all annotated sentences for lemmas other than *put* in the sense ENCODING: couch.v, expression.n, formulate.v, formulation.n, frame.v, phrase.v, word.v, wording.n

Table 4: Extending training sets: an example

Features	Precision		Recall	
`All`	0.7709	(σ 0.001)	0.7243	(σ 0.0018)
`Cx`	0.7727	(σ 0.0027)	0.8172	(σ 0.0035)
`Syn`	0.8571	(σ 0.0045)	0.1694	(σ 0.0012)
`Syn-hw`	0.8025	(σ 0.0041)	0.3383	(σ 0.0025)

Table 5: Experiment 3: Results for label *unknown sense*, NN-based outlier detection, $\theta = 1.0$. σ: standard deviation

Like WordNet synsets, FrameNet frames are semantic classes that typically comprise several lemmas or expressions. So, assuming that words with similar meaning occur in similar contexts, the context features for lemmas in the same frame should be similar. Following this idea, we supplement the training data for a lemma by all the *other* annotated data for the senses that are present in the training set, where by "other data" we mean data with other target lemmas. Table 4 shows an example[5].

Modeling. Again, we use Tax and Duin's outlier detection approach for unknown sense detection. The experimental design and evaluation are the same as in Experiment 2, the only difference being the training set extension. Training set extension raises the average training set size from 22.5 to 374.

Results. Table 5 shows precision and recall for labeling instances as *unknown*, with a distance quotient threshold of 1.0, averaged over unknown senses and 5-fold cross-validation. In comparison to Exper-

[5]Conditions `Syn` and `Syn-hw` were also tested using only other target lemmas with the same part of speech. Results were virtually unchanged.

Features	Precision all	Precision ≥ 50	Precision ≥ 200	Recall all	Recall ≥ 50	Recall ≥ 200
All	0.77	0.77	0.73	0.72	0.80	0.87
Cx	0.77	0.77	0.73	0.82	0.89	0.94
Syn	0.86	0.85	0.82	0.17	0.16	0.13
Syn-hw	0.80	0.79	0.76	0.38	0.36	0.38
Syn	0.86	0.85	0.82	0.17	0.17	0.14
Syn-hw	0.81	0.80	0.76	0.35	0.37	0.38

Table 6: Experiment 3: Results by training set size, $\theta = 1.0$

		Number of senses 2	3	4	5
Exp. 2	Prec.	0.78	0.68	0.59	0.55
	Rec.	0.21	0.38	0.47	0.59
Exp. 3	Prec.	0.83	0.71	0.63	0.56
	Rec.	0.68	0.81	0.89	0.88

Table 7: Experiments 2 and 3: Results by the number of senses of a lemma, condition All, $\theta = 1.0$

iment 2, precision has risen slightly, and for conditions All, Cx and Syn-hw, recall has risen steeply; the maximum recall is achieved by Cx at 0.82.

As before, increasing the distance quotient threshold leads to little change in precision but a sharp drop in recall. For All vectors, recall is 0.72 for threshold 1.0, 0.56 for $\theta = 1.1$, and 0.41 for $\theta = 1.2$.

Table 6 shows system performance by training set size. As the average training set in this experiment is much larger than in Experiment 2, we are now inspecting sets of minimum size 50 and 200 rather than 10 and 20. We find the same effect as in Experiment 2, with noticeably higher recall for lemmas with larger training sets, but slightly lower precision.

Table 7 breaks down system performance by the degree of ambiguity of a lemma. Here, too, we see the same effect as in Experiment 2: the more senses a lemma has, the lower the precision and the higher the recall of *unknown* label assignment.

Discussion. In comparison to Experiment 2, Experiment 3 shows a dramatic increase in recall, and even some increase in precision. Precision and recall for conditions All and Cx are good enough for the system to be usable in practice.

Of the four conditions, the three that involve context words, All, Cx and Syn-hw, show considerably higher recall than Syn. Furthermore, the two conditions that do not involve syntactic features, All and Cx, have markedly higher results than Syn-hw. This could mean that syntactic features are not as helpful as context features in detecting unknown senses; however in Experiment 2 the performance difference between Syn and the other conditions was not by far as large as in this experiment. It could also mean that frames are not as uniform in their syntactic structure as they are in their context words. This seems plausible as FrameNet frames are constructed mostly on semantic grounds, without recourse to similarity in syntactic structure.

Table 6 points to a sparse data problem, even with training sets extended by additional items. It also shows that the more a test condition relies on context word information, the more it profits from additional data. So it may be worthwhile to explore methods for a further alleviation of data sparseness, e.g. by generalizing over context words.

Table 7 underscores the large influence of training set uniformity: the more senses a lemma has, the more likely the model is to classify a test instance as *unknown*. This is the case even for extended training sets. One possible way of addressing this problem would be to take into account more than a single nearest neighbor in NN-based outlier detection in order to compute more precise boundaries between known and unknown instances.

8 Conclusion and outlook

We have defined and addressed the problem of *unknown word sense detection*: the identification of corpus occurrences that are not covered by a given sense inventory, using a training set of sense-annotated data as a basis. We have modeled this problem as an instance of *outlier detection*, using the simple nearest neighbor-based approach of Tax and Duin to measure the resemblance of a new occurrence to the training data. In combination with a method that alleviates data sparseness by sharing training data across lemmas, the approach achieves good results that make it usable in practice: With items represented as vectors of context words (including lemma, POS and NE), the system achieves 0.77 precision and 0.82 recall in an evaluation on FrameNet 1.2. The training set extension method, which proved crucial to our approach, relies solely

on a grouping of annotated data by semantic similarity. As such, the method is applicable to any resource that groups words into semantic classes, for example WordNet.

For this first study on unknown sense detection, we have chosen a maximally simple outlier detection method; many extensions are possible. One obvious possibility is the extension of Tax and Duin's method to more than one nearest training neighbor for a more accurate estimate of local density. Furthermore, more sophisticated feature vectors can be employed to generalize over context words, and other outlier detection approaches (Markou and Singh, 2003a; Markou and Singh, 2003b; Marsland, 2003) can be tested on this task.

Our immediate goal is to use unknown sense detection in combination with WSD, to filter out items that the WSD system cannot handle due to missing senses. Once items have been identified as *unknown*, they are available for further processing: If possible one would like to assign some measure of sense information even to these items. Possibilities include associating items with similar existing senses (Widdows, 2003; Curran, 2005; Burchardt et al., 2005) or clustering them into approximate senses.

References

C. Baker, C. Fillmore, and J. Lowe. 1998. The Berkeley FrameNet Project. In *Proc. ACL-98*, Montreal.

A. Burchardt, K. Erk, and A. Frank. 2005. A WordNet detour to FrameNet. In *Proc. GLDV 2005 Workshop GermaNet II*, Bonn.

U. Callmeier, A. Eisele, U. Schäfer, and M. Siegel. 2004. The DeepThought core architecture framework. In *Proc. LREC-04*, Lisbon.

James Curran. 2005. Supersense tagging of unknown nouns using semantic similarity. In *Proc. ACL-05*, Ann Arbor.

D. Dasgupta and S. Forrest. 1999. Novelty detection in time series data using ideas from immunology. In *Proc. International Conference on Intelligent Systems*.

Katrin Erk and Sebastian Pado. 2006. Shalmaneser - a toolchain for shallow semantic parsing. In *Proc. LREC-06*, Genoa.

K. Erk. 2005. Frame assignment as word sense disambiguation. In *Proc. IWCS 2005*, Tilburg.

C. Fillmore. 1982. Frame Semantics. *Linguistics in the Morning Calm*.

R. Florian, S. Cucerzan, C. Schafer, and D. Yarowsky. 2002. Combining classifiers for word sense disambiguation. *Journal of Natural Language Engineering*, 8(4):327–431.

S. Hickinbotham and J. Austin. 2000. Neural networks for novelty detection in airframe strain data. In *Proc. International Joint Conference on Neural Networks*.

D. Lin. 1993. Principle-based parsing without overgeneration. In *Proc. ACL-93*, Columbus, OH.

M. Markou and S. Singh. 2003a. Novelty detection: A review. part 1: Statistical approaches. *ACM Signal Processing*, 83(12):2481 – 2497.

M. Markou and S. Singh. 2003b. Novelty detection: A review. part 2: Neural network based approaches. *ACM Signal Processing*, 83(12):2499 – 2521.

S. Marsland. 2003. Novelty detection in learning systems. *Neural computing surveys*, 3:157–195.

D. Mount and S. Arya. 2005. ANN: A library for approximate nearest neighbor searching. Download from `http://www.cs.umd.edu/~mount/ANN/`.

B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt. 2000. Support vector method for novelty detection. *Advances in neural information processing systems*, 12.

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

M. Tatu and D. Moldovan. 2005. A semantic approach to recognizing textual entailment. In *Proc. HLT/EMNLP 2005*, Vancouver.

D. Tax and R. Duin. 1998. Outlier detection using classifier instability. In *Advances in Pattern Recognition: the Joint IAPR International Workshops*.

D. Tax and R. Duin. 2000. Data description in subspaces. In *International Conference on Pattern recognition*, volume 2, Barcelona.

Dominic Widdows. 2003. Unsupervised methods for developing taxonomies by combining syntactic and statistical information. In *Proc. HLT/NAACL-03*, Edmonton.

D. Yeung and C. Chow. 2002. Parzen-window network intrusion detectors. In *Proc. International Conference on Pattern Recognition*.