

# *A Detailed Comparison of WSD Systems*

## An Analysis of the System Answers for the SENSEVAL-2 English All Words Task

Judita Preiss

*Computer Laboratory  
JJ Thomson Avenue  
Cambridge CB3 0FD  
United Kingdom  
Judita.Preiss@cl.cam.ac.uk*

(Received)

---

### Abstract

We compare the word sense disambiguation systems submitted for the English-all-words task in SENSEVAL-2. We give several performance measures for the systems, and analyze correlations between system performance and word features. A decision tree learning algorithm is employed to discover the situations in which systems perform particularly well, and the resulting decision tree is examined. We investigate using a decision tree based on the SENSEVAL systems to (i) filter out senses unlikely to be correct, and to (ii) combine WSD systems. Some combinations created in this way outperform the best SENSEVAL system.

---

### 1 Introduction

In this work we investigate the hypothesis that word sense disambiguation (WSD) systems based on different principles disambiguate different words correctly. The results of our investigation not only allow us to analyze the strengths and weaknesses of the individual WSD systems, but also to produce a combined system that outperforms them.

The novelty in our work lies in the number of systems we investigate: our comparison is based on the 21 systems submitted to the English-all-words task of SENSEVAL-2 (Palmer et al., 2002). There is also no need for us to re-implement any of the systems (and thus potentially introduce differences), rather we use the answers each system submitted in carrying out the SENSEVAL-2 WSD evaluation exercise.<sup>1</sup> This task consisted of automatically annotating all instances of nouns, verbs, adjectives and adverbs in three given corpora, for which a gold standard annotation was created. We therefore have the 21 systems' sense assignments on an identical corpus, and so can investigate the performance of these systems with respect to a

<sup>1</sup> These are available from: <http://www.sle.sharp.co.uk/senseval2/>

number of different word features. The features we investigate were selected either because they are fundamental inherent properties of a word (such as polysemy), or because they are related to how the WSD systems work. Because of the close link with the systems, we expect a correlation to exist between the feature value and precision of at least some of the systems. For example, if a system selects those senses which maximize the number of overlapping words in their WordNet definitions (Lesk (1986) overlap), we would expect WordNet definition length to correlate with the system’s performance.

We use decision tree learning to discover types of words which are usually disambiguated correctly and types of words which WSD systems consistently disambiguate wrongly. Word types are defined by particular settings of the features we investigate, an example being words with polysemy one. We find that for the majority of words, the decision is not based on the answer supplied by a particular WSD system; the system is used purely to find the word’s type and in most cases a decision can then be made without knowing the system’s suggested answer. Therefore for majority of words, all systems lead to the same decision.

The acquired decision tree shows that for certain word types a system’s decision is always wrong – thus we can use the tree to build a ‘filtering’ system, which removes the sense suggested by the system from further consideration. This system is only expected to reduce the polysemy of ambiguous words (without removing the correct sense), leaving the final decision to a subsequent WSD system. The SENSEVAL systems vary in their suitability for this task, as some rule out the correct sense more often than others. We build decision tree WSD systems from a combination of the SENSEVAL systems. Some of the systems created in this way outperform the best SENSEVAL system, and we conclude that only judicious combinations of a small number of genuinely complementary WSD systems are likely to yield significant improvement on the WSD task.

Section 2 introduces the SENSEVAL-2 evaluation exercise: we describe the English all-words task, summarize the algorithms employed by each participating system, and present some performance figures. We select a number of features and investigate their correlation with system performance in Section 3. Decision tree learning is used to find out system specializations in Section 4. In Section 5 we investigate which systems could be used as a filter in a combined WSD system. In this section, we also present the results of combining a number of SENSEVAL systems using decision tree learning. Conclusions are presented in Section 6.

## **2 The Task and the Systems**

### ***2.1 The SENSEVAL Task***

The SENSEVAL task was created to allow a performance comparison of various WSD systems. For the English-all-words task in SENSEVAL-2, nouns, verbs, adjectives and adverbs in three given texts (taken from the Wall Street Journal) were manually annotated to create a gold standard. This resulted in 1082 distinct words in the

Information	Nouns	Verbs	Adj	Adv	Correct PoS
Polysemy:	3.99	7.20	3.62	2.59	5.24
Entropy:	1.54	2.14	1.40	0.94	1.79

Table 1. Average polysemy and entropy of all words in the English all-words task

texts, corresponding to 2473 instances to be automatically annotated<sup>2</sup> with senses by participating systems. Twenty one systems sense-annotated the given texts, and submitted their answers for a calculation of precision and recall against the gold standard. These answers have been made publicly available since the workshop.

## 2.2 Corpus Information

Table 1 shows the average polysemy and entropy over all 1082 words, broken down into parts of speech (PoS). A priori, we would expect that words with higher polysemy are harder for WSD systems to correctly disambiguate, and we investigate this hypothesis in Section 3. Average polysemy within a part of speech  $pos$  is computed as follows:

$$\text{polysemy}_{pos} = \frac{\sum_{w \in \text{text}} p_{pos}(w)}{|\{w \in \text{text} : 0 < p_{pos}(w)\}|}$$

where  $p_{pos}(w)$  is the polysemy of word  $w$  within the part of speech  $pos$ . Entropy is a measure of precision weighted by polysemy, and is defined to be:

$$\text{entropy}_{pos} = \frac{\sum_{w \in \text{text}} \log_2(p_{pos}(w))}{|\{w \in \text{text} : 0 < p_{pos}(w)\}|}$$

In both cases we divide by the number of words which have at least one sense in the given PoS, rather than the total number of words: a word without senses in the given PoS will not contribute to the numerator. The last column of the table refers to the average polysemy/entropy of each instance within its correct part of speech. For example, since *dog* has 6 noun senses and 1 verb sense within WordNet, an instance of *dog* will contribute 6 to the noun column and 1 to the verb column (and zero to the adjective and adverb columns). If the instance actually is a verb, then 1 is contributed to the correct PoS column. The polysemy within the correct PoS reflects the number of senses which most WSD systems are disambiguating between, since they usually use a PoS tagger to filter out all senses with the wrong PoS.

## 2.3 System Information

<sup>2</sup> This includes 86 instances annotated with the ‘‘U’’ tag. This tag was assigned when the correct sense of a word did not appear in WordNet 1.7.

---



---

BCU-ehu-dlist-all	<b>Supervised:</b> SEMCOR 1.6 Based on Yarowsky's decision lists, learns bigrams and trigrams for lemmas, wordforms and PoS from training data. Also acquires context words from a $\pm 4$ word window.
CNTS-Antwerp	<b>Supervised:</b> SEMCOR 1.6 A number of machine-learning word experts are trained, and the best one (based on training data) is individually selected for each word-PoS combination.
DIMAP	<b>Unsupervised</b> Uses WordNet to exploit contextual clues: focuses on collocation patterns, contextual overlap with definitions and examples, and topical area matches.
IIT	<b>Unsupervised</b> IIT1 uses WordNet examples and synsets to choose the sense which has the largest overlap with the instance's context. <i>IIT2</i> reduces contributions to score from distant context words. <i>IIT3</i> optimizes using the chosen senses for words preceding the current instance.
IRST	<b>Unsupervised</b> Uses WordNet semantic domains to initially determine a word's domain. The similarity between a domain vector and training data domain vectors for the same lemma is used to select a sense.
SMUaw	<b>Supervised:</b> SEMCOR 1.6, examples from WordNet 1.7, own heuristically-created sense-tagged corpus Uses training data to learn a number of patterns from local context.
Sheffield	<b>Unsupervised</b> Restricted to nouns, where it chooses senses by minimizing WordNet distance using simulated annealing. Extra information is provided by an anaphora resolution preprocessing step.
Sinequa-LIA-HMM	<b>Supervised:</b> SEMCOR 1.6 Uses WordNet semantic classes as well as short range context to acquire semantic classification trees from training data. Combined with a long-range similarity measure.
Sussex-sel	<b>Unsupervised</b> Identifies subject-verb and verb-direct object relationships and disambiguate these words using class-based selectional preferences. <i>Sussex-sel-ospd</i> and <i>Sussex-sel-ospd-ana</i> implement the one sense per discourse heuristic. <i>Sussex-sel-ospd-ana</i> also uses anaphora resolution to increase coverage.
UCLA-gchao	<b>Supervised:</b> SEMCOR 1.6 Creates a probabilistic network for each sentence, modeling dependencies between words. The parameters are obtained automatically and the systems differ in the way these are smoothed. The senses for the sentence are chosen by performing a query over the network.
UNED	<b>Unsupervised</b> Uses mutual information and co-occurrence information along with some frequency heuristics to select a sense.
USM	<b>Unsupervised</b> Partially disambiguated definitions in WordNet and used a semantic distance matrix between senses to make a choice. The three systems are different in the number of words they use from the definitions.

---



---

Table 2. *System Descriptions*

In this section, we describe the scoring method and the performance figures for all participating WSD systems. Table 2 summarizes the main algorithm employed by each of the systems (more details on each of the systems can be found in the SENSEVAL-2 proceedings (Preiss and Yarowsky, 2002)).

Systems were allowed to submit multiple answers for a word. Such answers could

Gold standard	System answers	Score
art <sub>1</sub>	art <sub>1</sub>	1
bell_ringing <sub>1</sub>	–	–
change <sub>1</sub>	change <sub>1</sub> ( $\frac{1}{2}$ )    change <sub>2</sub> ( $\frac{1}{2}$ )	$\frac{1}{2}$

Table 3. *Toy corpus: precision* =  $\frac{1+\frac{1}{2}}{2} = \frac{3}{4}$  *and recall* =  $\frac{1+0+\frac{1}{2}}{3} = \frac{1}{2}$

System	S	Precision	Recall	Coverage	Entropy	Diff
BCU-ehu-dlist-all	✓	57.21%	29.05%	50.79%	0.47	3.12
CNTS-Antwerp	✓	63.53%	63.53%	100.00%	0.47	2.82
DIMAP	×	45.09%	45.09%	100.00%	0.36	1.67
IIT1	×	28.70%	3.34%	11.65%	0.32	1.48
IIT2	×	32.81%	3.82%	11.65%	0.35	1.87
IIT3	×	29.40%	3.42%	11.65%	0.33	1.53
IRST	✓	74.68%	35.67%	47.76%	0.34	2.19
SMUaw	✓	68.94%	68.94%	100.00%	0.65	3.37
Sheffield	×	44.54%	19.98%	44.84%	0.34	1.82
Sinequa-LIA-HMM	✓	61.75%	61.75%	100.00%	0.43	2.64
Sussex-sel	×	59.79%	13.95%	23.33%	0.32	1.64
Sussex-sel-ospd	×	56.61%	16.92%	29.88%	0.30	1.76
Sussex-sel-ospd-ana	×	54.47%	16.92%	31.06%	0.29	1.70
UCLA-gchao	✓	50.83%	44.44%	87.42%	0.36	1.72
UCLA-gchao2	✓	48.26%	45.29%	93.85%	0.38	1.66
UCLA-gchao3	✓	48.08%	45.13%	93.85%	0.38	1.73
UNED-AW-T	×	57.44%	56.81%	98.91%	0.43	2.33
UNED-AW-U	×	55.52%	54.91%	98.91%	0.43	2.25
USM1	×	34.18%	31.58%	92.40%	0.28	1.29
USM2	×	36.02%	35.99%	99.92%	0.28	1.35
USM3	×	33.59%	33.56%	99.92%	0.27	1.28
correct	–	100.00%	100.00%	100.00%	1.56	6.49
frequency	–	66.90%	64.58%	96.52%	0.44	2.96
random	–	35.48%	34.24%	96.52%	0.27	1.48

Table 4. *Precision, recall, coverage and entropy for all systems*

be weighted by a probability distribution<sup>3</sup> and would contribute to the score of the system in proportion to the weight assigned (see Table 3 for a tiny example corpus). As part of the exercise, it was also necessary for systems to identify multi-word constructions and words manually annotated with the “U” tag.

The precision (number of correct answers divided by the number of words attempted) and recall (number of correct answers divided by the total number of test instances) values for each system, along with a value for coverage (number of instances answered divided by the total number of test instances), are displayed

<sup>3</sup> If such a weighting was not present, the uniform distribution was assumed.

in Table 4.<sup>4</sup> The table also includes a value for the entropy of each system, which emphasizes the cases where a system assigns a correct sense to a word with high polysemy. If a system correctly disambiguates a word with polysemy  $n$ , then the entropy score is increased by  $\log_2(n)$ . The entropy score of a system is the sum of the entropy values of the instances that it correctly disambiguated, divided by the number of instances the system attempted (so it should be regarded as ‘entropy precision’).

The systems can also be scored according to how ‘difficult’ were the words they disambiguated correctly (the Diff column). The difficulty rating of an instance is simply the number of systems that got the instance wrong. The difficulty score of a system is the sum of the difficulty ratings of the instances that it correctly disambiguated, divided by the number of instances the system attempted. Easy words (i.e., words with low difficulty ratings over all instances) include words which only have one sense (e.g. *chromosome*) and words which are only ambiguous between part of speech (e.g. *parent*). Examining the corpus shows that difficult words are usually high polysemy words (correlation 0.41), for example *test* (polysemy 15 over three different parts of speech) or *believe* (polysemy 5).

For each scoring method, we also include results for the correct file (which is a copy of the gold standard) and for two baselines (most frequent sense and random sense).<sup>5</sup> For each system a tick in the S column reminds us that the system is supervised, meaning that it employs a training corpus.

By statistically analyzing the results of the systems which attempted instances in all PoSs, we can use the precision on each PoS to rank the difficulty of disambiguation of individual PoS. The (one-tailed paired)  $t$ -test implies (with confidence  $\geq 99\%$ ) that adverbs were most often correctly disambiguated, followed by nouns, adjectives and finally verbs. This does not quite mirror the polysemy of the PoS (c.f. Table 1), which would imply that if system performance decreases as polysemy increases, the order of nouns and adjectives should be interchanged. We will investigate this further in the following section.

### 3 Analysis of System Results

#### 3.1 Feature Correlations

Systems rely on various resources to perform disambiguation, such as training examples in SEMCOR or definitions in WordNet. This section illustrates most clearly the novelty in our work: WSD comparison studies such as Yarowsky and Florian (2002) have re-implemented a small number of algorithms and focused mainly on

<sup>4</sup> These values are slightly different to the official SENSEVAL-2 results. Some systems submitted answers for the same test instance more than once and this was not discarded in the official version of the scorer. The scoring of the systems presented in this paper was carried out with our own scorer which is available at <http://www.cl.cam.ac.uk/users/jp233/software/>.

<sup>5</sup> Precision for the two baselines was only evaluated on words not annotated with “U”. If these were included, the precision of the most frequent sense would be 64.62% and that of the random choice would be 34.26%.

the effect of varying internal parameters (e.g. the window size used by the algorithm). This study uses the answers of 21 existing WSD systems to investigate the relationship between system precision and the most frequently used features (see system descriptions in Table 2). Positive correlations could reveal how to combine systems in a future WSD system to optimize performance.

As there was no training data available for this task, a number of the supervised systems used SEMCOR as their training corpus. Table 4 suggests that the most frequent sense baseline performs better than all but two systems.<sup>6</sup> It uses the WordNet 1.7 `cntlist` (frequency information file) to provide the ranking of senses, and this frequency information was obtained from SEMCOR.<sup>7</sup> Indeed, a correlation graph shows a relation exists for supervised systems between precision on a word and number of occurrences in SEMCOR (correlation values around 0.1 for supervised systems).

However, the distribution of senses in training data (and indeed occurrences in text) tends to be skewed towards the most frequent sense (e.g. Kilgarriff and Rosenzweig (2000)). We therefore investigate the precision of the systems on words where the most frequent sense is not correct; these words form 32% of the corpus (790 instances). The precision of the systems on these words varies between 11% (DIMAP) and 31% (SMUaw) with an average of 18%, and random achieving 17%. The SMU system (Mihalcea, 2002) for the English all words task uses “patterns” (local contexts) learnt from SEMCOR, GENCOR and WordNet definitions. We note that the training corpus employed by the SMUaw system is larger than that used by other systems. This suggests that the size of their corpus may be beginning to balance out the skewed distribution of occurrences of word’s senses in texts to provide enough training examples to learn patterns from even the less frequent senses.

The SMU system is not the only one to use the WordNet definitions; these are also used for disambiguation by the UNED systems and by the DIMAP system. Therefore, we next investigate the correlation between definition length and precision: if a definition is used for overlap and the words in the definition are relevant, then the longer the definition, the greater the chance of an overlap occurring. Thus, we would expect a positive correlation between the precision of systems using the definitions and definition length. WordNet 1.7 includes a gloss (a definition and possibly some example sentences) for every word in the dictionary.<sup>8</sup> The correlation graph can be seen in Figure 1.

The system with the highest correlation is Sussex-sel, which is based on selec-

<sup>6</sup> However, this baseline has access to perfect part of speech information about the word, and also has perfect knowledge of multi-words in the text.

<sup>7</sup> As an official WordNet 1.6 to WordNet 1.7 mapping has not yet been released, we have used SMU’s mapping from WordNet 1.6 to WordNet 1.7 for SEMCOR to find out whether most of the training examples in SEMCOR could be used by a supervised system. The number of wrongly tagged instances was very small and we therefore concluded that most supervised systems could have used SEMCOR in full.

<sup>8</sup> For this experiment, we remove the example sentences (since they frequently use only unrelated words), then lemmatize and stoplist the remaining definition. Definition length is then the number of remaining words.

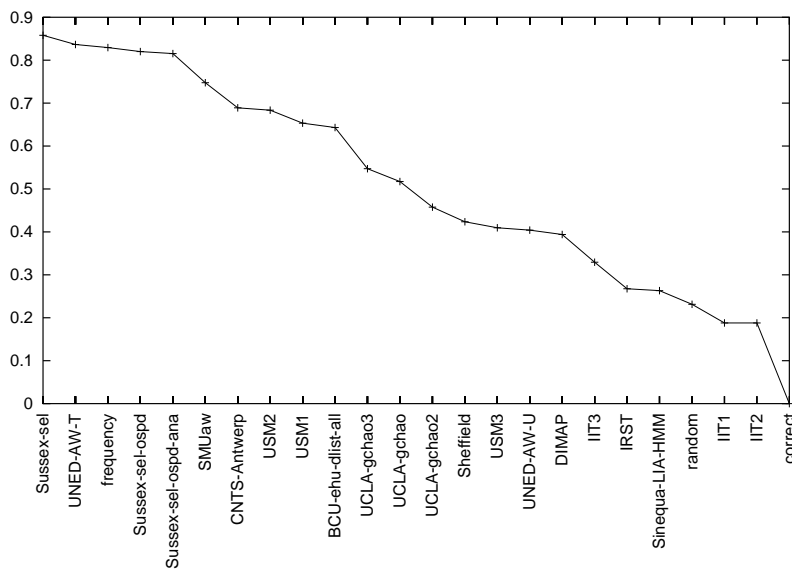


Fig. 1. Correlation of WordNet definition length with system precision

tional preferences and does not make any use of the WordNet definitions! However, these results must be placed in their proper context. The correlation of definition length with polysemy is  $-0.78$ , which means that the longer the definition, the less polysemous the words are. Therefore, if the Sussex-sel system attempted words which were below average in polysemy, it would be expected to have a higher correlation with definition length. This is true: the average polysemy within the correct part of speech of the words attempted by the Sussex-sel system is 3.36 (which is much lower than the WordNet average given in Table 1). This also explains why the most frequent sense baseline is the third most correlated.

This leads us to investigate the correlation of polysemy with precision. We would naively expect polysemy to be inversely correlated to precision – the more senses a word has, the more difficult it may be to disambiguate.<sup>9</sup> For each system we compute the precision on words of all polysemies separately. These are then correlated to produce a correlation value for each system, presented in Figure 2. Remarkably, the IIT systems correlate positively with polysemy (the low recall of the IIT systems is not due to their choice of words to disambiguate but a bug which meant that only answers for the initial 20% of the corpus were submitted). These systems use WordNet examples to score the similarity of the word’s local context in the text to the example. The relative success of IIT on more polysemous words therefore supports the need to take some context into account in disambiguation. The remaining systems do not appear to have a high correlation, supporting the sug-

<sup>9</sup> Consider the difference between a multiple choice test with three answers per question, versus one with fifteen answers per question.

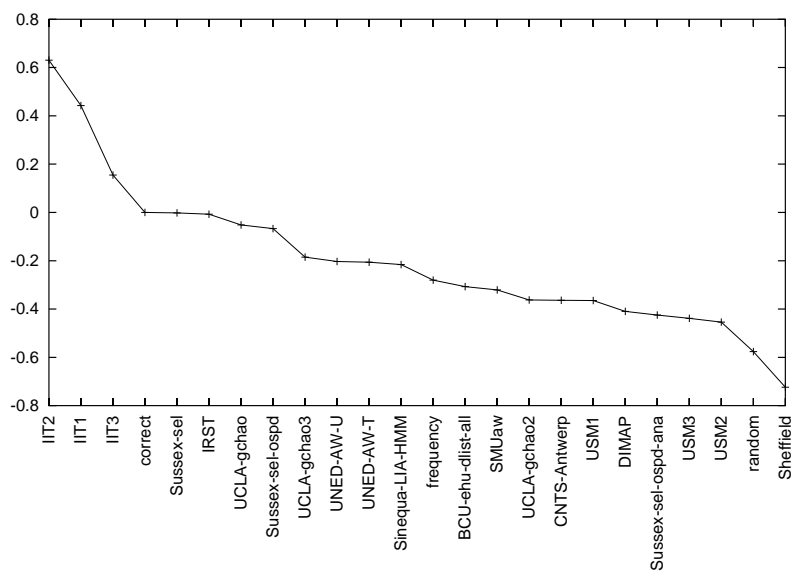


Fig. 2. Correlation of polysemy with system precision

Polysemy	Better distribution	Worse distribution	Confidence
2	WordNet class of	WordNet class of	99.5%
3	the word <b>does</b>	the word	99.5%
4	<b>not contain</b>	<b>contains</b>	95.0%
5	<b>any other</b>	<b>multiple senses</b>	99.5%
6	<b>sense of the</b>	of the word.	99.5%
8	word.		90.0%

Table 5. *t*-test comparing precision on words with multiple senses in WN classes

gestion of Kilgarriff and Rosenzweig (2000) that polysemy is not an ideal measure of difficulty.

We also examined the precision of systems on words where the WordNet class of the correct sense did not contain any other senses of the word. All words in WordNet belong to one of 44 classes: adverbs are not split into classes; there is a class for descriptive adjectives and relational adjectives; and nouns and verbs are classed according to their semantic fields (Miller et al., 1990). We found the precision of each system on words where the correct sense was in a WordNet class not shared with other senses (isolated), and on words where both the correct sense and an incorrect sense were in the same WordNet class (shared). These were further split up according to the degree of polysemy. The results in Table 5 show the results of *t*-tests

of precision variation between the two distributions for each level of polysemy.<sup>10</sup> These clearly indicate improved precision where the correct WordNet class did not contain other candidate senses regardless of the degree of polysemy.

### 3.2 Linguistic Analysis

This section presents a short analysis of the words which were mislabelled by at least 20 (out of the 21) systems, subject to the constraint that the words occurred in the corpus at least three times.<sup>11</sup>

The most frequently mislabelled nouns were *form*, *one*, *change*, *bang*, *service*, *loss*, and *development*. These words were often mislabelled with a ‘related’ sense; for example, the word *change* occurs three times in the corpus in the change%1:19:00:: sense: “*the result of alteration or modification*”. Many systems chose instead change%1:11:00, which is defined as “*an event that occurs when something passes from one state or phase to another*”. However, the synonyms of change%1:11:00 are *alteration* and *modification*, and thus for a definition and synonym overlap based system the two senses are quite likely to have similar overlap scores.

In the case of verbs, the most frequently mislabelled ones were *lead*, *turn*, *make*, *find*, *miss*, *say*, *involve*, *think*, *show*, and *develop*. These verbs have a number of closely semantically related senses, which systems often confused.<sup>12</sup> For example, lead%2:42:12:: (“*tend to or result in*”) was often confused by systems with lead%2:42:04::, which is defined as “*result in*”.

The most frequently mislabelled adjectives were *common*, *rare*, *defective*, and *simple*. All of these were wrongly annotated when used in the adjectival satellite sense – the systems preferred to select a non-satellite adjective sense. This is not surprising as the satellite synset for a satellite is designed to represent a similar concept to the head adjective synset. This finding indicates that a promising approach to improving the performance of systems on adjectives is to properly deal with satellites.

## 4 Specializations

### 4.1 Decision Tree Learning

The correlations observed in Section 3.1 support the hypothesis that different systems may be better at disambiguating different words. The idea of combining knowledge sources is not new: Stevenson and Wilks (1999) combine a number of weak knowledge sources for WSD, Pedersen (2000) creates an ensemble of Naive Bayes classifiers, Florian and Yarowsky (2002) investigate combinations of six different

<sup>10</sup> The *t*-test is only performed for degrees of polysemy containing at least 5 significant systems, where a significant system is one that attempts at least 20 words in both categories.

<sup>11</sup> We are considering each word only within one part of speech, i.e., we are assuming a perfect part of speech tagger.

<sup>12</sup> Senses which are in the same WordNet class are thought to be semantically related.

No.	Feature	Values
$F_1$	position_in_sentence	[0, 1]
$F_2$	words_attempted_in_sentence	real
$F_3$	polysemy_overall	real
$F_4$	polysemy_within_attempted_pos	real
$F_5$	answer_pos	{Noun, Verb, Adj, Adv}
$F_6$	definition_length	real
$F_7$	word_difficulty [within class]	real
$F_8$	system	system names
$F_9$	num_training_egs	real
$F_{10}$	frequency_rank	real

Table 6. Features in decision tree

$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$	$F_8$	$F_9$	$F_{10}$	class
0.00	4	4	1	4	6	0	BCU-ehu-dlist-all	50	1	no
0.60	11	4	1	4	6	0	frequency	50	1	no
0.50	11	4	1	4	6	0	UCLA-gchao	50	1	no
0.90	11	4	1	4	6	0	UCLA-gchao2	50	1	no
0.90	11	4	1	4	6	0	UCLA-gchao3	50	1	no
0.00	11	4	1	4	6	0	UNED-AW-T	50	1	no
0.90	11	4	1	4	6	0	UNED-AW-U	50	1	no
0.40	11	4	1	4	6	0	USM1	50	1	no
0.40	11	4	1	4	4	0	USM2	16	2	no
0.40	11	4	1	4	6	0	USM3	8	3	yes
0.00	11	4	1	4	6	0	IIT1	50	1	no
0.00	11	4	1	4	6	0	IIT1	8	3	yes
0.00	11	4	1	4	6	0	IIT2	50	1	no
0.00	11	4	1	4	6	0	IIT3	50	1	no
0.00	11	4	1	4	6	0	IIT3	8	3	yes
0.40	11	4	1	4	6	0	CNTS-Antwerp	50	1	no
0.70	11	4	1	4	6	0	DIMAP	50	1	no
0.40	11	4	1	4	4	0	Sinequa-LIA-HMM	16	2	no
0.20	6	4	1	4	4	0	IRST	16	2	no
0.40	11	4	1	4	6	0	SMUaw	50	1	no
0.25	5	4	1	4	6	0	Sheffield	50	1	no

Table 7. Example of training input to decision tree learning algorithm

classifiers. However, in this section, our goal is not to create a WSD system, instead we aim to analyze the types of words for which each SENSEVAL system is suited. Types of words are defined by our choice of features which we correlate, e.g. words with high frequency, words with polysemy equal to two. To carry out the analysis, we chose ten features  $F_1, \dots, F_{10}$  (see Table 6) to use in decision tree learning. We now briefly justify our choice of features:

$F_1$  – The **position in sentence** will affect bigram or trigram information (poten-

- tially none present as no preceding words). A word at position  $i$  in a sentence with  $n$  words will have `position_in_sentence` =  $\frac{i-1}{n-1}$ . E.g. BCU-ehu-dlist-all.
- $F_2$  – For example, IIT3 uses already chosen senses (for preceding words) to disambiguate the target word. In this case, the **number of words attempted in a sentence** may help indicate how confident the system can be about its answer.
- $F_3$  and  $F_4$  – **Overall polysemy** and **polysemy within the given part of speech** affects all systems. In the case where the target word is monosemous in the correct PoS (but is polysemous over all),  $F_4$  will tell us about the precision of the tagger used by the systems.
- $F_5$  – A system which relies on the WordNet hierarchy (which is said to be more detailed for nouns than for other PoS), may perform worse on particular **parts of speech**. E.g. Sussex-sel.
- $F_6$  – The performance of a system which uses definition overlap (e.g. DIMAP) may be affected by the **length of the definitions**.
- $F_7$  – IRST and Sinequa-LIA-HMM systems make use of the WordNet semantic classes. It may therefore be useful to know **how many other senses of the target word are in the class of the chosen sense**.
- $F_8$  – The **system name**.
- $F_9$  – The **number of training examples for the chosen sense** may influence the performance of supervised systems.
- $F_{10}$  – The **frequency rank of the chosen sense** may affect our confidence in the system's answer, if the system backs off to the most frequent sense, e.g. Sheffield.

Decision tree learning thus appears well suited to our task: our target function is discrete valued (the annotation is either correct or incorrect) and the instances can be described as attribute-value pairs in terms of our features. We are also expecting the training data to be noisy and decision tree learning is usually able to cope with this. For an introduction to decision tree learning, see e.g. Mitchell (1997).

We use the WEKA (Witten and Frank, 2000)<sup>13</sup> implementation of the C4.5 decision tree learning algorithm (Quinlan, 1993), to create a decision tree of system results and word features. The training instances consist of the features in Table 6 relative to the system answer. Training instances are classified as positive or negative depending on whether the system disambiguated them correctly. See Table 7 for an example of the input to the decision tree learner of the test instance d00.s00.t01. Note that a system may appear more than once per test instance, if it submitted more than one answer.

## 4.2 Systems Specializations

The C4.5 algorithm constructs a decision tree top-down, by selecting the attribute that best classifies the local training examples at each node. When faced with a

<sup>13</sup> Available from <http://www.cs.waikato.ac.nz/~ml/weka/>.



as any other system which submitted an answer. Thus for these types of words, we don't need to employ twenty two different systems, but we only need one which will always provide an answer. This criterion is satisfied by the most frequent baseline system. Note that the proportion of classification made without using a particular system is 79%, this supports the 80/20 rule of time management (Pedersen, 2001), which suggests that 20% of effort accounts for 80% of results.

The two subtrees beneath the splits on the SENSEVAL systems account for 7039 classifications. The initial decisions leading up to the 'system' split are:

1.  $(\text{frequency\_rank} \leq 1) \wedge (2 < \text{polysemy\_overall} \leq 4) \wedge$   
 $(2 < \text{definition\_length} \leq 9) \wedge (\text{polysemy\_within\_attempted\_pos} \leq 2) \wedge$   
 $(\text{num\_training\_egs} \leq 3)$
2.  $(1 < \text{frequency\_rank} \leq 3) \wedge (\text{polysemy\_overall} > 1) \wedge$   
 $(\text{definition\_length} \leq 15)$

The first case deals with the situation when the suggested sense is the highest frequency sense (of a word which is not highly polysemous). In this case, the systems can be trusted in their positive or negative answers and quickly lead to a final decision (there is at most one further split in the decision tree). But this subtree only accounts for 156 classifications (out of 7039).

A decision is not reached quite as quickly in the second case. However, it can be seen that classifications are independent of SENSEVAL systems whenever the frequency rank of a suggested sense exceeds 3.

It is interesting to note that the number of further splits varies from system to system, as do the features involved. This is where we hoped to gain useful insight into the situations in which some or all of the systems perform well. For example, looking again at the excerpt in Figure 3, beneath the USM1 system the first split is whether the overall polysemy of the word is less than or equal to 3, suggesting that performance might be significantly different in each of these cases.<sup>15</sup> However, there are unfortunate correlations between the features that make the decision tree very hard to interpret. Note in Figure 3, beneath USM1 there is a split on the number of training examples in SEMCOR. However, the USM1 system is unsupervised and does not perform training, so this split must represent some hidden dependence between the features that happens to make this split the most informative. Many other examples of this phenomenon occur underneath the main system split, and so unfortunately we cannot draw many conclusions about the system specializations.

But there are still two striking and surprising features of the resulting decision tree:

1. For the majority of words, the individual systems are not split upon to make an optimal decision.
2. The systems are split upon at just two different points in the decision tree.

<sup>15</sup> In fact, the performance of USM1 is 43% when the polysemy is less than or equal to 3 (but greater than 1), and 16% otherwise.

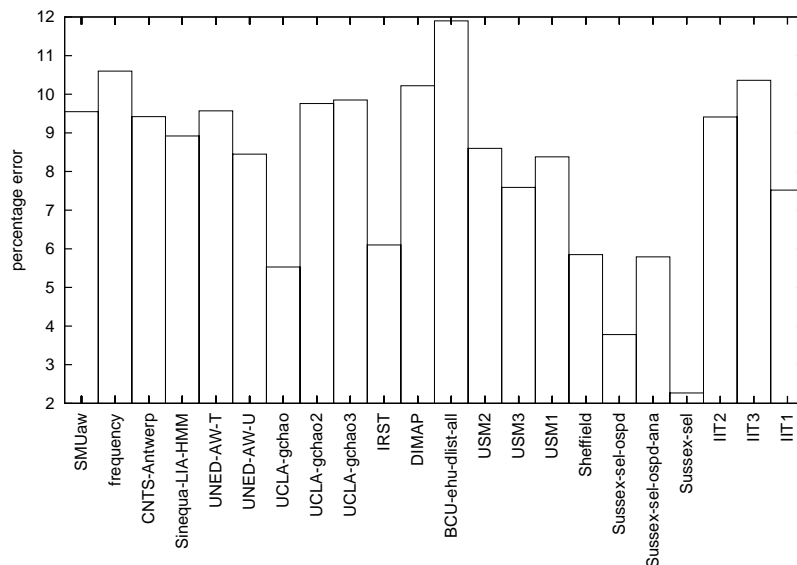


Fig. 4. Misclassifications in Filtering

Therefore we can conclude that in building a future WSD system we would do well to focus mainly on identifying the words which cannot be decided using a baseline system and optimizing the new system for these difficult words. These words can be identified by examining the created decision tree and following the paths which include a system split. Focussing on words which cannot be decided using a baseline system may seem an obvious point, however it is important to realize that many systems have a lower performance than the most frequent sense baseline (even when this does not have access to perfect information), and so this common sense point is not being implemented in many systems.

## 5 Systems

A decision tree based on the inputs of a number of WSD systems, as described in Section 4, could be employed in new WSD systems. We present two example applications:

1. A decision tree used as a component of a WSD system, which rules out certain senses from consideration.
2. A full system which, based on the output of some of the systems, chooses a sense for each word.

In the following two subsections we investigate these applications.

### 5.1 Filter System

Instead of using the SENSEVAL systems to produce a sense for each word in a

corpus, we investigate their suitability as filters. For each system, we learn a decision tree from 90% of the system’s answers and find out how frequently the decision tree classifies the correct sense as incorrect (i.e. the correct sense is filtered out) on the remaining 10% of data. This is motivated by the assumption that although a word may have a number of senses, a system may only be deciding between a few of them. In that case, it would be useful to be able to reliably rule out some of these. For example, our WSD system may immediately rule out the two inanimate senses of the word *dog* (6 senses in WordNet). If we can reliably use the SENSEVAL systems to say that *dog* cannot have the “You lucky dog” sense, our WSD system is only deciding between 3 remaining senses.

For example, consider the decision tree created for the USM1 system using the first 90% of the system’s answers, which contains the following branch:

$$(\text{polysemy\_overall} > 1) \wedge (\text{frequency\_rank} > 2) \Rightarrow \text{no}$$

This means that if the word which the USM1 system is disambiguating is polysemous, and the sense suggested by the WSD system is of a higher rank than 2, the decision tree predicts that this is an incorrect annotation, so this sense could be removed. The percentage of misclassifications (filtering out the correct sense) for each system is presented in Figure 4 (systems are ordered according to their F-measure in this figure).<sup>16</sup> We can see that some of the SENSEVAL systems appear to be better suited to the filtering task than others: the Sussex-sel system (unsupervised, based on selectional preference information) only rules out 2.3% of the correct senses, whereas the BCU-ehu-dlist-all system (supervised, using decision lists) rules out 11.9%. However, the BCU-ehu-dlist-all system also attempted many harder words (both in terms of polysemy and in terms of our difficulty, defined in Section 3.1) than the Sussex-sel system, which may explain some of the difference.

## 5.2 Combined Systems

For each word, we group together the answers given by a number of the SENSEVAL systems. The decision tree is created from the answers to 90% of all words. For each word to be classified, we pass a number of system answers to the decision tree. As part of its classification, the decision tree produces a confidence in a “yes” or a “no” answer. The confidences in the answers are then ranked and the sense with the highest confidence is chosen.<sup>17</sup>

We ranked the systems according to their F-measure (producing the list: SMUaw, frequency, CNTS-Antwerp, Sinequa-LIA-HMM, UNED-AW-T, UNED-AW-U, UCLA-gchao, UCLA-gchao2, UCLA-gchao3, IRST, DIMAP, BCU-ehu-dlist-all, USM2, USM3, USM1, Sheffield, Sussex-sel-ospd, Sussex-sel-ospd-ana, Sussex-sel, IIT2, IIT3, IIT1). A number (between 1 and 22) of top-ranked systems was then combined to

<sup>16</sup> Note that in most cases when the systems rule out a sense, it is not the correct one and thus the percentage of misclassifications is not equal to precision.

<sup>17</sup> In order to be able to truly choose the highest confidence, we convert 95% confidence in “no” into a 5% confidence in “yes”, so all confidences are presented in terms of “yes”.

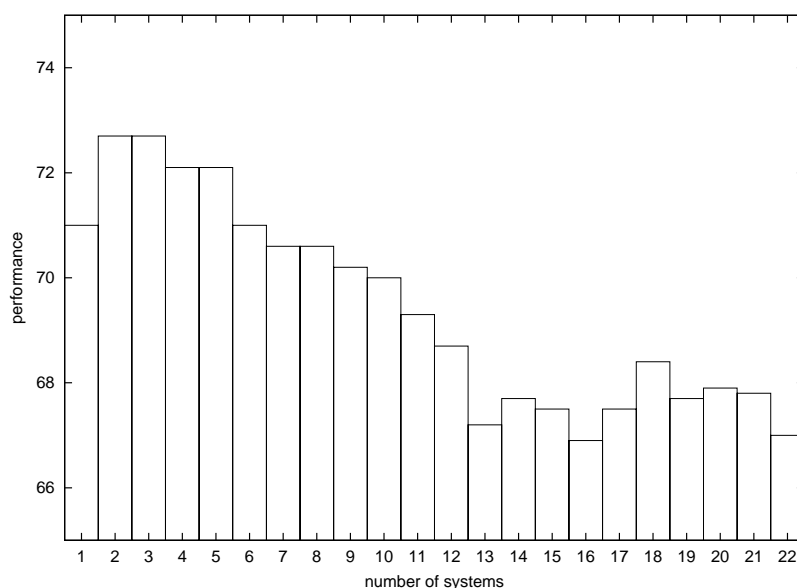


Fig. 5. Combined Systems

	1	2	3	4	5
1	*	—	—	—	—
2	99.0	*	—	80.0	70.0
3	99.5	50.0	*	85.0	75.0
4	99.5	—	—	*	—
5	97.5	—	—	50.0	*
6	—	—	—	—	—

Table 8. Excerpt from the Performance Comparison of Combination Systems

create a WSD system, which was tested using the method above. The results of the 10 fold cross validation are presented in Figure 5. In this table, “number of systems = 1” represents the performance of the decision tree when it is only trained and tested on the SMUaw system, “number of systems = 2” represents the performance when trained and tested on both SMUaw and frequency, etc.

An excerpt from a (one-tailed)  $t$ -test comparison between all pairs of combined systems is presented in Table 8. The full matrix tells us which combined systems are significantly better than others. Due to space constraints, we only present the section of the table which shows that the combined systems significantly outperform the best SENSEVAL-2 system. This table is to be interpreted as follows:

	1	2	3	...
2	99.0	*	—	...

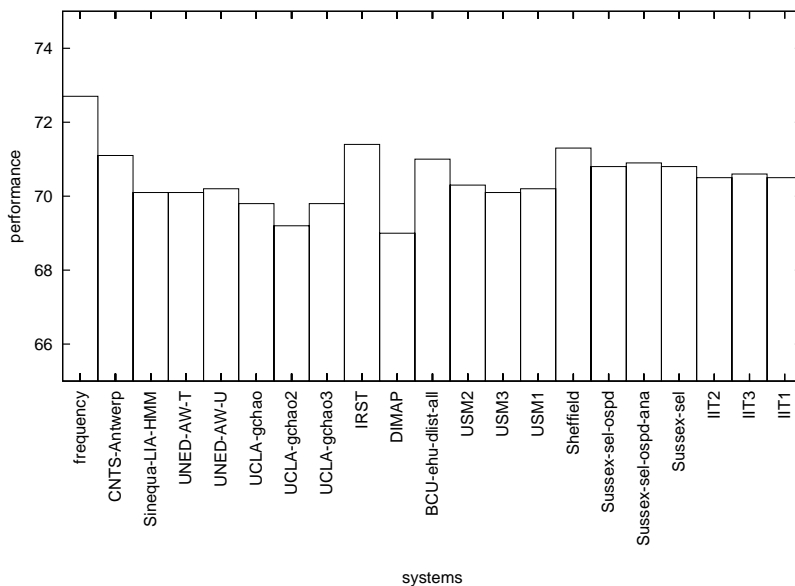


Fig. 6. Paired System Results

shows that a combination of two systems outperforms one system with a confidence of 99%. However, the combination of two systems does not outperform three systems (indicated by “-”). Table 8 shows that combining the top 2, 3, 4, or 5 outperforms the current best system with varying degrees of confidence. Combining more than the top five systems does not lead to a more accurate system. From our investigation, it seems that the best system results from combining the SMU, the most frequent and the CNTS-Antwerp systems. Although both SMUaw and CNTS-Antwerp back-off to the most frequent sense, they can be seen to be potentially complementary. The SMUaw system learns rules from local contexts. The CNTS-Antwerp system also includes a word expert which considers keywords from a context of 3 sentences.

It is interesting to note that combining the SMU system with a small number of other systems yields best performance. This indicates that the decision tree learner cannot compensate for the noise introduced by many systems in combination. It is possible that one of the systems further down in the F-measure ranking is a good complement for the SMU system, but this is being hidden by the noise. We therefore evaluate the performance of all 21 systems paired with the SMU system; these results are presented in Figure 6. In this case, the “frequency” column corresponds to a combination of the SMU and frequency systems, the “CNTS-Antwerp” column corresponds to a combination of the SMU and CNTS-Antwerp systems, and so on. No combination outperforms the SMU-frequency combination, although SMU-BCU and SMU-Sheffield perform better than is indicated by Figure 5. The high performance of the SMU-IRST combination is not quite as surprising, since the IRST system has a very high precision, but has a low F-measure due to its low

recall. In the combination, the assignments not in the coverage of IRST will be made by the SMU system.

We summarize the main findings of this section:

1. Combining a large number of systems leads to too much noise.
2. Combining the first three (F-measure) ranked SENSEVAL systems yields better performance than the current best system (precision 72.69%, recall 72.69%).

We conclude that although decision tree learning seems very well suited to automatically finding complementary systems, this is not necessarily true for WSD when the combined systems are very intertwined. For future WSD systems, effort may best be spent by creating systems which are independent. As we remarked in section 4.2, there are hidden dependencies between features, which complicates the task of creating independent systems. For example, the frequency rank feature is not independent of the number of training examples for a word, as a more frequent sense will mean more training examples exist. Thus a supervised system which does not directly use frequency rank will still not be independent from the most frequent sense baseline. However, it is possible that techniques from statistics could be used to create several artificial systems that would perform well when combined together in this way.

## 6 Conclusion

We have used the word sense disambiguation systems submitted for the English-all-words task in SENSEVAL-2 to test a number of hypotheses, which will be useful for building a better WSD system in the future. Primarily, we investigated whether WSD systems based on different principles disambiguate different words correctly. We focused on a number of features which are directly or indirectly used by the SENSEVAL-2 systems, and hypothesised that each will correlate with precision. We did not find a conclusive correlation between WordNet definition length and precision, probably because words with fewer senses tend to have longer definitions.

However, the number of training examples in Semcor did correlate with precision for supervised systems. Systems also performed significantly better when the WordNet class of the correct sense did not contain any other senses. Surprisingly, there wasn't a high correlation between polysemy and precision, supporting Kilgarriff and Rosenzweig's (2000) hypothesis that polysemy is not the best measure of how difficult a word is to disambiguate. In fact, our analysis refines that of Kilgarriff and Rosenzweig, since they correlate system performance with task polysemy while we correlate performance with polysemy of individual words. Precision also told us that adverbs are easier words to disambiguate, followed by nouns, adjectives and verbs. Our findings also support the hypothesis of Pedersen (2002), namely that a significant number of instances will be very difficult for any method to resolve.

Next, we hypothesised that the features which we investigated could be used in conjunction with decision tree learning to find which systems are complementary. We found that there is no need to use a particular system for the majority of words – all systems which annotated these words were found to be equally good. Thus the

resulting decision tree did not uncover significant system variation. In this paper we have covered many inherent features of words, but it is possible that there are other useful features which would lead to more pronounced correlations. Finding these features is a possible avenue for future work.

Instead of using the systems to directly disambiguate words, we investigated whether it would be possible to use them as a filter, i.e. having the ability to rule out unlikely senses to reduce the polysemy of words. Having built a decision tree from the training data, we found that many systems' decision trees rule out the correct sense about 10% of the time on the test data. However, some systems rule out the correct sense very infrequently and so would be suitable to use as a filter. An example of such a system is Sussex-sel.

Lastly, we hypothesised that decision tree learning could be used to create a combined WSD system. By combining the top (F-measure) 2, 3, 4 or 5 systems, we obtained significantly better performance than the best system on its own. The performance decreased (below that of the best system) when more systems were combined, probably due to the fact that the best combined system is obtained when complementary systems are combined (Pedersen, 2000). We suggest that only combinations of a small number of genuinely complementary systems are likely to yield significant improvements for the WSD task.

### Acknowledgements

This work was supported by UK EPSRC project GR/N36462/93: 'Robust Accurate Statistical Parsing'. I would like to thank Rada Mihalcea and the SMU group for their copy of SEMCOR 1.7. My thanks also go to Ted Briscoe and Joe Hurd for reading previous drafts of this paper.

### References

- R. Florian and D. Yarowsky. 2002. Modeling consensus: classifier combination for word sense disambiguation. *Proceedings of EMNLP'02*, pages 25–32.
- A. Kilgarriff and J. Rosenzweig. 2000. Framework and results for English SENSEVAL. *Computers and the Humanities*, 34(1–2):15–48.
- M. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. *Proceedings of AGM SIGDOC Conference*, pages 24–26.
- R. Mihalcea. 2002. Word Sense Disambiguation Using Pattern Learning and Automatic Feature Selection. *Journal of Natural Language and Engineering*, 8(4):343–358.
- G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. 1990. Introduction to WordNet: An on-line lexical database. *Journal of Lexicography*, 3(4):235–244.
- T. M. Mitchell. 1997. *Machine Learning*. McGraw-Hill International Editions.
- M. Palmer, C. Fellbaum, S. Cotton, L. Delfs, and H. T. Dang. 2002. English Tasks: All-Words and Verb Lexical Sample. *Proceedings of the Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pages 40–46.
- T. Pedersen. 2002. Assessing system agreement and instance difficulty in the lexical sample tasks of SENSEVAL-2. *Proceedings of the Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pages 40–46.

- T. Pedersen. 2001. Machine learning with lexical features: the Duluth approach to SENSEVAL-2. *Proceedings of SENSEVAL-2*, pages 139–142.
- T. Pedersen, 2000. A simple approach for building ensembles of naive Bayesian learners for word sense disambiguation. *Proceedings of ANLP-NAACL 2000*, pages 63-69.
- J. Preiss and D. Yarowsky, editors. 2002. *Proceedings of SENSEVAL-2: Second International Workshop on Evaluating Word Sense Disambiguating Systems*.
- J. R. Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufman Publishers.
- M. Stevenson and Y. Wilks. 1999. Combining weak knowledge sources for sense disambiguation. In *Proceedings of the International Joint Conference for Artificial Intelligence (IJCAI-99)*.
- I. H. Witten and E. Frank, 2000. *Data mining: Practical Machine Learning Tools and Techniques with Java Implementations*, chapter 8. Morgan Kaufmann Publishers.
- D. Yarowsky and R. Florian, 2002. Evaluating Sense Disambiguation Across Diverse Parameter Spaces. *Journal of Natural Language Engineering*, 8(4):293–310.