Domain-Specific Semantic Class Disambiguation Using WordNet

Li Shiuan Peh
DSO National Laboratories
20 Science Park Drive
Singapore 118230
plishiua@dso.org.sg

Hwee Tou Ng
DSO National Laboratories
20 Science Park Drive
Singapore 118230
nhweetou@dso.org.sg

Abstract

This paper presents an approach which exploits general-purpose algorithms and resources for domain-specific semantic class disambiguation, thus facilitating the generalization of semantic patterns from word-based to class-based representations. Through the mapping of the domain-specific semantic hierarchy onto WordNet and the application of general-purpose word sense disambiguation and semantic distance metrics, the approach proposes a portable, wide-coverage method for disambiguating semantic classes. Unlike existing methods, the approach does not require annotated corpora. When tested on the MUC-4 terrorism domain, the approach is shown to outperform the most frequent heuristic substantially and achieve comparable accuracy with human judges. Its performance also compares favourably with two supervised learning algorithms.

1 Introduction

The semantic classification of words refers to the abstraction of ambiguous (surface) words to unambiguous concepts. These concepts may be explicitly expressed in a pre-defined taxonomy of classes, or implicitly derived through the clustering of semantically-related words. Semantic classification has proved useful in a range of application areas, such as information extraction (Soderland et al., 1995), acquisition of domain knowledge (Mikheev and Finch, 1995) and improvement of parsing accuracy through the specification of selectional restrictions (Grishman and Sterling, 1994; Grishman and Sterling, 1992).

In this paper, we address the problem of semantic class disambiguation, with a view towards applying it to information extraction. The disambiguation of the semantic class of words in a particular context facilitates the generalization of semantic extraction patterns used in information extraction from word-based to class-based forms. This abstraction is effectively tapped by CRYSTAL (Soderland et al., 1995), one of the first few approaches to the automatic induction of extraction patterns.

Many existing information extraction systems (MUC-6, 1996) rely on tedious knowledge engineering approaches to hard-code semantic classes of words in a semantic lexicon, thus hampering the portability of their systems to different domains. A notable exception is the approach taken by the University of Massachusetts. Its knowledge acquisition framework, Kenmore, uses a case-based learning mechanism to learn domain knowledge automatically (Cardie, 1993). Kenmore, being a supervised algorithm, relies on an annotated corpus of domain-specific classes. Grishman et al. (1992) too ventured towards automatic semantic acquisition for information extraction. However, they expressed reservations regarding the use of WordNet to augment their semantic hierarchy automatically, citing examples of unintended senses of words resulting in erroneous semantic classification.

To circumvent the annotation bottleneck faced by Kenmore, our approach exploits general algorithms and resources for the disambiguation of domain-specific semantic classes. Unlike Grishman et al.'s approach, our application of general word sense disambiguation algorithms and semantic distance metrics allows for an effective use of the fine sense granularity of WordNet. Experiments carried out on the MUC-4 (1992) terrorism domain saw our approach outperforming supervised algorithms and matching human judgements.

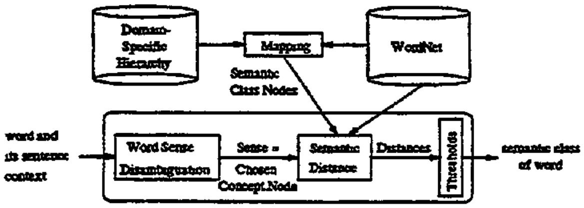

Figure 1 : Semantic Class Disambiguation.

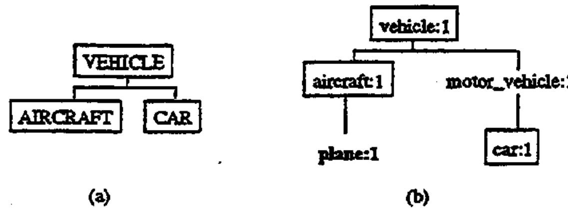

Figure 2 : (a) A simple domain-specific hierarchy (b) The classes of the domain-specific hierarchy as mapped onto WordNet, together with the word to be disambiguated, "plane".

2 Our Approach

As opposed to proponents of "domain-specific information for domain-specific applications", our approach ventures towards the application of general-purpose algorithms and resources to our domain-specific semantic class disambiguation problem.

Our information source is the extensive semantic hierarchy WordNet (Miller, 1990) which was designed to capture the semantics of general nuances and uses of the English language. Our approach reconciles the domain-specific hierarchy with this vast network and exploits WordNet to uncover semantic classes, without the need of an annotated corpus.

Firstly, the domain-specific hierarchy is mapped onto the semantic network of WordNet, by manually assigning corresponding WordNet node(s) to the classes in the domain-specific hierarchy. To disambiguate a word, the sentence context of the word is first streamed through a general word sense disambiguation module which assigns the appropriate sense of the word. The word sense disambiguation module hence effectively pinpoints a particular node in WordNet that corresponds to the current sense of the word. Thereafter, this chosen concept node is piped through a semantic distance module which determines the semantic distances between this concept node and all the semantic class nodes in the domain-specific hierarchy. If the distance between the concept node and a semantic class node is below some threshold, the semantic class node becomes a candidate class node. The nearest candidate class node is then chosen as the semantic class of the word. If no such candidates exist, the word does not belong to any of the semantic classes in the hierarchy, and is usually labelled as the "entity" class. The flow of our approach is illustrated in Figure 1.

A walkthrough of the approach with a simple example will better illustrate it. Consider a domain-specific hierarchy with just 3 classes :- VEHICLE, AIRCRAFT and CAR, as shown in Figure 2(a).

Mapping this domain-specific hierarchy to WordNet simply involves finding the specific sense(s) of the classes. In this case, all three classes correspond to their first sense in WordNet.

Then, given a sentence, say, "The plane will be taking off in 5 minutes time.", to disambiguate the semantic class of the word "plane", the sentence is fed to the word sense disambiguation module. The module will determine the sense of this word. In this example, the correct sense of "plane" is sense 1, i.e. the sense of an aeroplane. Having identified the particular concept node in WordNet that "plane" corresponds to, the distances between this concept node and the three semantic class nodes are then calculated by the semantic distance module. Based on WordNet, the module will conclude that the concept node "plane:1" is nearer to the semantic class node "aircraft:1" and should hence be classified as AIRCRAFT. Figure 2(b) shows the relative positions of the concept node "plane:1" and the three semantic class nodes in WordNet.

2.1 Word Sense Disambiguation

Word sense disambiguation is an active research area in natural language processing, with a great number of novel methods proposed. Methods can typically be delineated along two dimensions, corpus-based vs. dictionary-based approaches.

Corpus-based word sense disambiguation algorithms such as (Ng and Lee, 1996; Bruce and Wiebe, 1994; Yarowsky, 1994) relied on supervised learning from annotated corpora. The main drawback of these approaches is their requirement of a sizable sense-tagged corpus. Attempts to alleviate this tagging bottleneck include bootstrapping (Teo et al., 1996; Hearst, 1991) and unsupervised algorithms (Yarowsky, 1995).

Dictionary-based approaches rely on linguistic knowledge sources such as machine-readable dictionaries (Luk, 1995; Veronis and Ide, 1990) and WordNet (Agirre and Rigau, 1996; Resnik, 1995) and exploit these for word sense disambiguation.

Thus far, two notable sense-tagged corpora, the semantic concordance of WordNet 1.5 (Miller et al. , 1994) and the DSO corpus of 192,800 sense-tagged occurrences of 191 words used by (Ng and Lee, 1996) are still insufficient in scale for supervised algorithms to perform well on a wide range of texts.

Unsupervised algorithms such as (Yarowsky, 1995) have reported good accuracy that rivals that of supervised algorithms. However, the algorithm was only tested on coarse-level senses and not on the refined sense distinctions of WordNet, which is the required sense granularity of our approach.

We hence turn to dictionary-based approaches, focusing on WordNet-based algorithms since they fit in snugly with our WordNet-based semantic class disambiguation task.

Information Content

Resnik (1995) proposed a word sense disambiguation algorithm which determines the senses of words in noun groupings. The sense of a word is disambiguated by choosing the sense which is most highly supported by the other nouns of the noun group. The extent of support depends on the information content of the subsumers of the nouns in WordNet, whereby information content is defined as negative log likelihood $-log\,p(c)$, and $p(c)$ is the probability of encountering an instance of concept c.

As mentioned in his paper, although his approach was only reported on the disambiguation of words in related noun groupings, it can potentially be applied to word sense disambiguation of nouns in running text.

In our implementation of his approach, we applied the method to general word sense disambiguation. We used the surrounding nouns of a word in free running text as the "noun grouping" and followed his algorithm without modifications[1].

Conceptual Density

Agirre and Rigau's (1996) approach has a similar motivation as Resnik's. Both approaches hinge on the belief that surrounding nouns[2] provide strong clues as to the sense of a word.

The main difference lies in how they determine the extent of support offered by the surrounding nouns. Agirre and Rigau uses the conceptual density of the ancestors of the nouns in WordNet as their metric.

Our implementation follows the pseudo-code presented in (Agirre and Rigau, 1996)[3]. For words which the algorithm failed to disambiguate (when no senses or more than one sense is returned), we relied on the most frequent heuristic.

2.2 Semantic Distance

The task of the semantic distance module is to reflect accurately the notion of "closeness" between the chosen concept node of the word and the semantic class nodes. It thus requires a metric which can effectively represent the semantic distance between two nodes in a taxonomy such as WordNet.

Conceptual Distance

Rada et. al (1989) proposed such a metric termed as conceptual distance. Conceptual distance between two nodes is defined as the minimum number of edges separating the two nodes. Take the example in Figure 2(b), the conceptual distance between "plane:1" and "aircraft:1" is 1, that between "plane:1" and "vehicle:1" is 2, and that between "plane:1" and "car:1" is 4[4].

Link Probability

The link probability metric is our variant of the conceptual distance metric. Instead of considering all edges as equi-distance, the probability of the link (or edge) is used to bias its distance. This metric is motivated by Resnik's use of the probability of instance occurrences of concepts, $p(c)$ (Resnik, 1995). Link probability is defined as the difference between the probability of instance occurrences of the parent and child of the link. Formally,

$$LinkPr(a, b) = p(a) - p(b),$$

[1]The pseudo-code of his algorithm is detailed in (Resnik, 1995).

[2]Surrounding nouns in the original Resnik's approach refers to the other nouns in the noun grouping.

[3]We clarified with the authors certain parts of the algorithm which we find unclear. These are the points worth noting :-
(1) *compute_conceptual_density* of Step 2 only computes the conceptual density of concepts which are not marked invalid;
(2) *exitloop* of Step 3 occurs when all senses subsumed by *concept* were already previously disambiguated or when one or more senses of the word to be disambiguated are subsumed by *concept*;
(3) *mark_disambiguated_senses* of Step 4 marks senses subsumed by *concept* as disambiguated, marks *concept* and its children as invalid, and discards other senses of the words with sense(s) disambiguated by *concept*;
(4) disambiguated senses of words which form the context are not brought forward to the next window.

[4]In WordNet, there are 25 unique beginners of the taxonomy, instead of a common root. Hence, in our implementation, we assign a large conceptual distance of 999 to the virtual edges between two unique beginners.

$$
\begin{aligned}
\textit{where } p(c) &= \frac{\sum_{n \in words(c)} count(n)}{N} \\
&\quad \textit{where } words(c) \textit{ is the set of nouns in the} \\
&\quad \textit{corpus which are subsumed by the concept } c, \\
&\quad \textit{and } N \textit{ is the total number of noun} \\
&\quad \textit{occurrences in the corpus, (Resnik, 1995)} \\
a &= \textit{parent of the link}, \\
b &= \textit{child of the link}.
\end{aligned}
$$

The intuition behind this metric is that the distance between the parent and the child should be "closer" if the probability of the parent is close to that of the child, since that implies that whenever an instance of the parent occurs in the corpus, it is usually an instance of the child.

Descendant Coverage

In the same spirit, the descendant coverage metric attempts to tweak the constant edge distance assumption of the conceptual distance metric. Instead of relying on corpus statistics, static information from WordNet is exploited. Descendant coverage of a link is defined as the difference in the percentage of descendants subsumed by the parent and that subsumed by the child :-

$$
\begin{aligned}
DescCov(a,b) &= d(a) - d(b), \\
\textit{where } d(c) &= \frac{\textit{Number of descendants of } c}{\textit{Total number of descendants in WordNet}}, \\
a &= \textit{parent of the link}, \\
b &= \textit{child of the link}.
\end{aligned}
$$

The same intuition underlies this metric; that the distance between the parent and the child should be "nearer" if the percentage of descendants subsumed by the parent is close to that of the child, since it indicates that most descendants of the parent are also descendants of the child.

Taxonomic Link (IS-A)

All the metrics detailed above were designed to capture semantic similarity or closeness. The semantic class disambiguation problem, however, is essentially to identify membership of the chosen concept node in the semantic class nodes.

A simple implementation of the semantic distance module can thus be just a traversal of the taxonomic links (IS-A) of WordNet. If the chosen concept node is a descendant of a semantic class node, it should be classified as that semantic class.

3 Evaluation

The domain we worked on is the MUC-4 (1992) terrorism domain. Nouns are extracted from the first 18 passages (dev-muc4-0001 to dev-muc4-0018) of the corpus of news wire articles to form our test corpus. The nouns extracted are the head nouns within noun phrases which are recognised by WordNet, including proper nouns such as "United States". These 1023 nouns are hand-tagged with their sense and semantic class in the particular context to form the answer keys for subsequent experiments.

3.1 Mapping domain-specific hierarchy onto WordNet

The domain-specific hierarchy used in our work is that crafted by researchers from the University of Massachusetts for their information extraction system, which was one of the participants at MUC-4 (Riloff, 1994).

Mapping from the domain-specific hierarchy to WordNet typically requires only the assignment of senses to the classes. For instance, the semantic class "human" is mapped onto its sense 1 node in WordNet, the "human:1" concept node. Classes can also be mapped onto more than one concept node in WordNet. The semantic class "attack", for example, is mapped onto both senses 1 and 5.

There are cases where the exact wording of a semantic class in the domain-specific hierarchy is not present in WordNet. Take for instance the semantic class "government_official" in the domain-specific hierarchy. Since the collocation is not in WordNet, we mapped it to the concept node "government_agent:1" which we felt is closest in meaning.

The set of mapped semantic classes in WordNet is shown in Figure 3[5].

3.2 Word Sense Disambiguation

We ran our two implementations of word sense disambiguation algorithms, the information content algorithm and the conceptual density method, on our domain-specific test set. For the information content algorithm, a window size of 10, i.e. 5 nouns to the left and right, was found to yield the best results; whilst for the conceptual density algorithm, the optimum window size was found to be 30. For both algorithms, only the nouns of the same passage are incorporated into the context window. If the noun to be disambiguated is the first noun of the passage, the window will include the subsequent N nouns of the same passage.

The probability statistics required for Resnik's information content algorithm were collected over

[5] As this hierarchy is adopted, and not created by us, occasionally, we can only furnish guesses as to the exact meaning of the semantic classes.

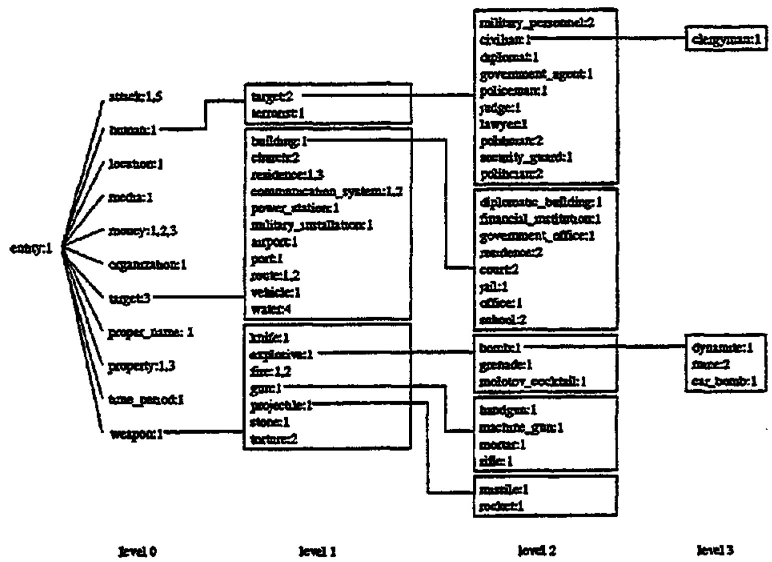

Figure 3 : MUC-4 semantic class hierarchy as mapped onto WordNet.

777,857 noun occurrences of the entire Brown corpus and Wall Street Journal corpus.

The results are shown in Table 1. The most frequent baseline is obtained by following the strategy of always picking sense 1 of WordNet, since WordNet orders its senses such that sense 1 is the most likely sense.

As both algorithms performed below the most frequent baseline, it prompted us to evaluate the indicativeness of surrounding nouns for word sense disambiguation. We hence provided 2 human judges with a randomly selected sample of 80 examples from the 734 polysemic nouns of our test corpus of 1023 examples. The human judges are provided with the 10 nouns surrounding the word to be disambiguated. Based only on these clues, they have to select a single sense of the word in the particular sentence context. Their responses are then tallied with the sense-tagged test corpus.

Table 2 shows the accuracies attained by the human judges. Both judges are able to perform substantially better than the most frequent heuristic baseline, despite the seemingly impoverished knowledge source. Feedback from the judges reveal possible leverage for future improvements. Firstly, judges reflect that frequently, just one indicative surrounding noun is enough to provide clear evidence for sense disambiguation. The other nouns will just be glossed over and do not contribute to the decision. Also, indicative nouns may not just hold is-a relationships, which are the only relationships exploited by both algorithms. Rather, they are simply related in some manner to the noun to be disambiguated. For instance, a surrounding context including the word "church" will indicate a strong support for the "pastor" sense of "minister" as opposed to its other senses. These reflections of the human judges seem to point towards the need for an effective method for selecting only particular nouns in the surrounding context as evidence. Use of other relationships besides is-a may also help in disambiguation, as is already expounded by (Sussna, 1993).

3.3 Semantic Distance Metrics

To evaluate the semantic distance metrics, we feed the semantic distance module with the correct senses of the entire test corpus and observe the resultant semantic class disambiguation accuracy.

The conceptual distance, link probability and descendant coverage metrics all require traversal of links from one node to another. However, all of the metrics are commutative, i.e. distance from concept a to b is the same as that from b to a. In semantic class disambiguation, a distinction is necessary since the taxonomic links indicate membership relationships which are not commutative ("aircraft:1" is a "vehicle:1" but "vehicle:1" need not be an "aircraft:1"). We hence associate different weights to the upwards and downwards traversal of links, with the 25 unique beginners of WordNet being the topmost nodes. Upward traversal of links towards the unique beginners are weighted consistently at 0.3 whilst downward traversal of links towards the leaves

	#Examples	#Disambiguated	#Correct	Accuracy
Information content (polysemic)	734	734	292	39.78 %
Conceptual density (polysemic)	734	275	68	24.73 %
Conceptual density + Most frequent heuristic (polysemic)	734	734	385	52.45 %
Most frequent heuristic (polysemic)	734	734	464	63.22 %
Information content (overall)	1023	1023	581	56.79 %
Conceptual density (overall)	1023	564	357	63.30 %
Conceptual density + Most frequent heuristic (overall)	1023	1023	674	65.88 %
Most frequent heuristic (overall)	1023	1023	753	73.61 %

Table 1: Word sense disambiguation results.

	#Examples	#Correct	Accuracy
Human A	80	57	71.25 %
Human B	80	59	73.75 %
Most frequent heuristic	80	45	56.25 %

Table 2: Word sense disambiguation using surrounding nouns.

are weighted at 1.7[6].

Also, different thresholds are used for different levels of the domain-specific hierarchy. Since higher level classes, such as the level 0 "human" class, encompasses a wider range of words, it is evident that the thresholds for higher level classes cannot be stricter than that of lower level classes. For fair comparison of each metric, the best thresholds are arrived through exhaustive searching of a reasonable space[7]. The results are detailed in Table 3.

Accuracy on specific semantic classes refers to an exact match of the program's response with the corpus answer. The general semantic class disambiguation accuracy, on the other hand, considers a response correct as long as the response class is in the sub-hierarchy which originated from the same level 0 class as the answer. For example, if the program's response is class "politician", whilst the answer is class "lawyer", since both classes originated from the same level 0 class "human", this response is considered correct when calculating the general semantic class accuracy. The specific semantic class disambiguation accuracy is hence the stricter measure.

It may seem puzzling that semantic class disambiguation does not achieve 100% accuracy even when supplied with the correct senses, i.e. even when the word sense disambiguation module is able to attain 100% accuracy, the overall semantic class disambiguation accuracy still lags behind the ideal. Since the taxonomic links in WordNet are designed to capture membership of words in classes, it may seem odd that the correct identification of the word sense coupled with the IS-A taxonomic links still do not guarantee correct semantic class disambiguation.

The reason for this paradox is perceptive differences; that between the designers of the MUC-4 domain-specific hierarchy we adopted and the WordNet hierarchy, and that between the annotator of the answer corpus and the WordNet designers.

Take for example the monosemic word "kidnapping". Its correct semantic class is "attack:5[8]". However, it is not a descendant of "attack:5" in WordNet. The hypernyms of "kidnapping" are [capture → felony → crime → evil-doing → wrong-doing → activity → act] and that of "attack:5" are [battery → crime → evil-doing → wrong-doing → activity → act]. Both perceptions of "kidnapping" are correct. "kidnapping" can be viewed as a form of "attack:5" and similarly, it can be viewed as a form of "capture".

An effective semantic distance metric is hence needed here. The semantic distance module should infer the close distance between the two concept nodes "kidnapping" and "attack:5" and thus correctly classify "kidnapping".

3.4 Semantic Class Disambiguation

After evaluation of the separate phases, we combined the best algorithms of the two phases and evaluated the performance of our semantic class disambiguation approach. Hence, the most frequent

[6]These weights are found to be optimum for all three metrics.

[7]Integral thresholds are searched for the conceptual distance metric, whilst the thresholds of the other metrics are searched in steps of 0.01.

[8]"attack:5" refers to an assault on someone whilst "attack:1" refers to the beginning of an offensive.

	Disambiguation Accuracy		Thresholds[a]
	Specific Classes	General Classes	
Conceptual Distance	81.52 %	87.10 %	(3,2,2,1)
Link Probability	80.16 %	85.24 %	(0.1,0.01,0.01,0.01)
Descendant Coverage	77.81 %	83.87 %	(0.02,0.01,0.01,0.01)
Taxonomic	79.67 %	85.14 %	Not applicable

Table 3: Effect of different semantic distance metrics on semantic class disambiguation. (Assuming perfect word sense disambiguation)

[a]Format :- $(t_{l0}, t_{l1}, t_{l2}, t_{l3})$, where t_{li} is the threshold that is applied to the ith level of the hierarchy.

sense heuristic is used for the word sense disambiguation module and the conceptual distance metric is adopted for the semantic distance module.

It should be emphasized, however, that our approach to semantic class disambiguation need not be coupled with any specific word sense disambiguation algorithm. The most frequent WordNet sense is chosen simply because current word sense disambiguation algorithms still cannot beat the most frequent baseline consistently for all words. Our approach, in effect, allows domain-specific semantic class disambiguation to latch onto the improvements in the active research area of word sense disambiguation.

As a baseline, we again sought the most frequent heuristic, which is the occurrence probability of the most frequent semantic class "entity".[9]

We compared our approach with supervised methods to contrast their reliance on annotated corpora with our reliance on WordNet. One of the foremost semantic class disambiguation system which employs machine learning is the Kenmore framework (Cardie, 1993). However, as we are unable to report comparative tests with Kenmore[10], we adapted two other supervised algorithms, both successfully applied to general word sense disambiguation, to the task of semantic class disambiguation.

The first is the LEXAS algorithm which uses an exemplar-based learning framework similar to the case-based reasoning foundation of Kenmore (Ng, 1997; Ng and Lee, 1996). LEXAS was shown to achieve high accuracy as compared to other word sense disambiguation algorithms.

We also applied Teo et al's Bayesian word sense disambiguation algorithm to the task (Teo et al., 1996). The approach compares favourably with other methods in word sense disambiguation when tested on a common data set of the word "interest".

The features used for both supervised algorithms are the local collocations of the surrounding 4 words[11]. Local collocation was shown to be the most indicative knowledge source for LEXAS and these 7 features are the common features used in both LEXAS and Teo et al's Bayesian algorithm. Both algorithms are used for learning the specific semantic class of words.

For both algorithms, the 1023-sentence test set is randomly partitioned into a 90% training set and a 10% testing set, in proportion with the overall class distribution. The algorithms are trained on the training set and then used to disambiguate the distinct testing set. This was averaged over 10 runs. As with Kenmore, the training set contains features of all the words in the training sentences, and the algorithms are to pick one semantic class for each word in the testing set. A word in the testing set need not have occurred in the training set. This is unlike word sense disambiguation, whereby the training set contains features of one word, and the algorithm picks one sense for each occurence of this word in the testing set.

To obtain a gauge of human performance on this task, we sourced two independent human judgements. Two human judges are presented with a set of 80 sentences randomly selected from the 1023-example test corpus, each with a noun to be disambiguated. Based on their understanding of the sentence, each noun is assigned a specific semantic class of the domain-specific hierarchy. Their responses are then compared against the tagged answers of the test corpus.

The semantic class disambiguation results are compiled and tabulated in Table 4. The definitions of general and specific semantic class disambiguation accuracy are detailed in Section 3.3.

As is evident, our approach outperforms the most frequent heuristic substantially. Also, the perfor-

[9]This baseline is also used to evaluate the performance of Kenmore (Cardie, 1993).

[10]As work on one of the important input sources, the conceptual parser, is underway, performance results of Kenmore on semantic class disambiguation cannot yet be reported.

[11]Given a word w in the following sentence segment :- l_2 l_1 w r_1 r_2, the 7 features used are $l_2\_l_1$, $l_1\_r_1$, $r_1\_r_2$, l_2, l_1, r_1 and r_2, whereby the first 3 features are concatenations of the words.

	Disambiguation Accuracy	
	Specific Classes	General Classes
Our Approach (1023 examples)	**73.90 %**	**80.16 %**
Most frequent heuristic (1023 examples)	46.92 %	46.92 %
Supervised (LEXAS)	57.30 %	57.30 %
Supervised (Bayes)	57.13 %	58.88 %
Our Approach (80 examples)	71.15 %	75.00 %
Human C (80 examples)	77.50 %	82.50 %
Human D (80 examples)	70.00 %	75.00 %
Most frequent heuristic (80 examples)	51.25 %	51.25 %

Table 4: Semantic class disambiguation results.

mance of both supervised algorithms lag behind that of our approach. Comparable performance with the two human judges is also achieved.

It should be noted, though, that the amount of training data available to the supervised algorithms may not be sufficient. Ng and Lee (1996) found that training sets of 1000-1500 examples per word are necessary for sense disambiguation of one highly ambiguous word. The amount of training data needed for a supervised learning algorithm to achieve good performance on semantic class disambiguation may be larger than what we have used. Cardie (1993), for instance, used a larger 2056-instance case base in the evaluation of Kenmore.

4 Conclusion

We have presented a portable, wide-coverage approach to domain-specific semantic class disambiguation which performs comparably with human judges. Our approach harnesses WordNet effectively to outperform supervised methods which rely on annotated corpora. Unlike existing methods which require hand-crafting of lexicon or manual annotation, the only human effort involved in our approach is the mapping of the domain-specific semantic classes onto WordNet. Through the use of general word sense disambiguation algorithms and semantic distance metrics, our approach correlates the performance of semantic class disambiguation with the improvements in these actively researched fields.

References

Eneko Agirre and German Rigau. 1996. Word Sense Disambiguation Using Conceptual Density. In *Proceedings of COLING-96*.

Rebecca Bruce and Janyce Wiebe. 1994. Word Sense Disambiguation using Decomposable Models. In *Proceedings of ACL-94*.

Claire Cardie. 1993. A Case-Based Approach to Knowledge Acquisition for Domain-Specific Sentence Analysis. In *Proceedings of AAAI-93*.

Ralph Grishman, John Sterling and Catherine Macleod. 1992. New York University Proteus System: MUC-4 Test Results and Analysis. In *Proceedings of MUC-4*.

Ralph Grishman and John Sterling. 1992. Acquisition of Selectional Patterns. In *Proceedings of COLING-92*.

Ralph Grishman and John Sterling. 1994. Generalizing Automatically Generated Selectional Patterns. In *Proceedings of COLING-94*.

Marti A. Hearst. 1991. Noun Homograph Disambiguation Using Local Context in Large Text Corpora. In *Using Corpora*, University of Waterloo, Waterloo, Ontario.

Alpha K. Luk. 1995. Statistical Sense Disambiguation with Relatively Small Corpora Using Dictionary Definitions. In *Proceedings of ACL-95*.

Andrei Mikheev and Steven Finch. 1995. Towards a Wordbench for Acquisition of Domain Knowledge from Natural Language. In *Proceedings of EACL-95*.

George A. Miller. 1990. An On-Line Lexical Database. In *International Journal of Lexicography*, 3(4):235-312, 1990.

George A. Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G. Thomas. 1994. Using a Semantic Concordance for Sense Identification. In *Proceedings of the ARPA Human Language Technology Workshop*.

MUC-4 Proceedings. 1992. *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, San Mateo, CA: Morgan Kaufmann.

MUC-6 Proceedings. 1996. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, San Mateo, CA: Morgan Kaufmann.

Hwee Tou Ng and Hian Beng Lee. 1996. Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-Based Approach. In *Proceedings of ACL-96.*

Hwee Tou Ng. 1997. Exemplar-Based Word Sense Disambiguation: Some Recent Improvements. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing.*

Roy Rada, Hafedh Mili, Ellen Bicknell and Maria Blettner. 1989. Development and Application of a Metric on Semantic Nets. In *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 1, Jan/Feb.

Philip Resnik. 1995. Disambiguating Noun Groupings with Respect to WordNet Senses. In *Proceedings of the Third Workshop on Very Large Corpora.*

Ellen M. Riloff. 1994. Information Extraction As a Basis for Portable Text Classification Systems. *PhD thesis*, University of Massachusetts, September 1994.

Stephen Soderland, David Fisher, Jonathan Aseltine and Wendy Lehnert. 1995. CRYSTAL: Inducing a Conceptual Dictionary. In *Proceedings of IJCAI-95.*

Michael Sussna. 1993. Word Sense Disambiguation for Free-Text Indexing using a massive Semantic Network. In *Proceedings of the Second International Conference on Information and Knowledge Management (CIKM-93).*

Edward Teo, Christopher Ting, Li-Shiuan Peh and Hian-Beng Lee. 1996. Probabilistic Word-Sense Disambiguation and Bootstrapping with Unsupervised Gradient Descent. manuscript.

Jean Veronis and Nancy Ide. 1990. Word Sense Disambiguation with Very Large Neural Networks extracted from Machine Readable Dictionaries. In *Proceedings of COLING-90.*

David Yarowsky. 1994. Decision Lists for Lexical Ambiguity Resolution: Application to Accent Restoration in Spanish and French. In *Proceedings of ACL-94.*

David Yarowsky. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *Proceedings of ACL-95.*