Using Syntactic Dependency as Local Context to Resolve Word Sense Ambiguity

Dekang Lin
Department of Computer Science
University of Manitoba
Winnipeg, Manitoba, Canada R3T 2N2
lindek@cs.umanitoba.ca

Abstract

Most previous corpus-based algorithms disambiguate a word with a classifier trained from previous usages of the same word. Separate classifiers have to be trained for different words. We present an algorithm that uses the same knowledge sources to disambiguate different words. The algorithm does not require a sense-tagged corpus and exploits the fact that two different words are likely to have similar meanings if they occur in identical local contexts.

1 Introduction

Given a word, its context and its possible meanings, the problem of word sense disambiguation (WSD) is to determine the meaning of the word in that context. WSD is useful in many natural language tasks, such as choosing the correct word in machine translation and coreference resolution.

In several recent proposals (Hearst, 1991; Bruce and Wiebe, 1994; Leacock, Towwell, and Voorhees, 1996; Ng and Lee, 1996; Yarowsky, 1992; Yarowsky, 1994), statistical and machine learning techniques were used to extract classifiers from hand-tagged corpus. Yarowsky (Yarowsky, 1995) proposed an unsupervised method that used heuristics to obtain seed classifications and expanded the results to the other parts of the corpus, thus avoided the need to hand-annotate any examples.

Most previous corpus-based WSD algorithms determine the meanings of polysemous words by exploiting their **local contexts**. A basic intuition that underlies those algorithms is the following:

(1) Two occurrences of the *same* word have *identical* meanings if they have *similar* local contexts.

In other words, most previous corpus-based WSD algorithms learn to disambiguate a polysemous word from previous usages of the same word. This has several undesirable consequences. Firstly, a word must occur thousands of times before a good classifier can be learned. In Yarowsky's experiment (Yarowsky, 1995), an average of 3936 examples were used to disambiguate between two senses. In Ng and Lee's experiment, 192,800 occurrences of 191 words were used as training examples. There are thousands of polysemous words, e.g., there are 11,562 polysemous nouns in WordNet. For every polysemous word to occur thousands of times each, the corpus must contain billions of words. Secondly, learning to disambiguate a word from the previous usages of the *same* word means that whatever was learned for one word is not used on other words, which obviously missed generality in natural languages. Thirdly, these algorithms cannot deal with words for which classifiers have not been learned.

In this paper, we present a WSD algorithm that relies on a different intuition:

(2) Two *different* words are likely to have *similar* meanings if they occur in *identical* local contexts.

Consider the sentence:

(3) The new facility will employ 500 of the existing 600 employees

The word "facility" has 5 possible meanings in WordNet 1.5 (Miller, 1990): (a) installation, (b) proficiency/technique, (c) adeptness, (d) readiness, (e) toilet/bathroom. To disambiguate the word, we consider other words that appeared in an identical local context as "facility" in (3). Table 1 is a list of words that have also been used as the subject of "employ" in a 25-million-word Wall Street Journal corpus. The "freq" column are the number of times these words were used as the subject of "employ".

Table 1: Subjects of "employ" with highest likelihood ratio

word	freq	$log\lambda$	word	freq	$log\lambda$
ORG	64	50.4	machine	3	6.56
plant	14	31.0	corporation	3	6.47
company	27	28.6	manufacturer	3	6.21
operation	8	23.0	insurance company	2	6.06
industry	9	14.6	aerospace	2	5.81
firm	8	13.5	memory device	1	5.79
pirate	2	12.1	department	3	5.55
unit	9	9.32	foreign office	1	5.41
shift	3	8.48	enterprise	2	5.39
postal service	2	7.73	pilot	2	5.37

*ORG includes all proper names recognized as organizations

The $log\lambda$ column are their likelihood ratios (Dunning, 1993). The meaning of "facility" in (3) can be determined by choosing one of its 5 senses that is most similar[1] to the meanings of words in Table 1. This way, a polysemous word is disambiguated with past usages of other words. Whether or not it appears in the corpus is irrelevant.

Our approach offers several advantages:

- The same knowledge sources are used for all words, as opposed to using a separate classifier for each individual word.
- It requires a much smaller corpus that needs not be sense-tagged.
- It is able to deal with words that are infrequent or do not even appear in the corpus.
- The same mechanism can also be used to infer the semantic categories of unknown words.

The required resources of the algorithm include the following: (a) an untagged text corpus, (b) a broad-coverage parser, (c) a concept hierarchy, such as the WordNet (Miller, 1990) or Roget's Thesaurus, and (d) a similarity measure between concepts.

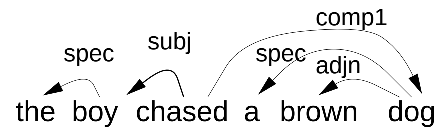
In the next section, we introduce our definition of local contexts and the database of local contexts. A description of the disambiguation algorithm is presented in Section 3. Section 4 discusses the evaluation results.

2 Local Context

Psychological experiments show that humans are able to resolve word sense ambiguities given a narrow window of surrounding words (Choueka and Lusignan, 1985). Most WSD algorithms take as input a polysemous word and its local context. Different systems have different definitions of local contexts. In (Leacock, Towwell, and Voorhees, 1996), the local context of a word is an unordered set of words in the sentence containing the word and the preceding sentence. In (Ng and Lee, 1996), a local context of a word consists of an ordered sequence of 6 surrounding part-of-speech tags, its morphological features, and a set of collocations.

In our approach, a local context of a word is defined in terms of the syntactic dependencies between the word and other words in the same sentence.

A dependency relationship (Hudson, 1984; Mel'čuk, 1987) is an asymmetric binary relationship between a word called **head** (or governor, parent), and another word called **modifier** (or dependent, daughter). Dependency grammars represent sentence structures as a set of dependency relationships. Normally the dependency relationships form a tree that connects all the words in a sentence. An example dependency structure is shown in (4).

(4)


The local context of a word W is a triple that corresponds to a dependency relationship in which W is the head or the modifier:

```
(type word position)
```

where `type` is the type of the dependency relationship, such as `subj` (subject), `adjn` (adjunct), `comp1` (first complement), *etc.*; `word` is the word related to W via the dependency relationship; and `position` can either be `head` or `mod`. The `position` indicates whether `word` is the head or the modifier in depen-

[1] to be defined in Section 3.1

dency relation. Since a word may be involved in several dependency relationships, each occurrence of a word may have multiple local contexts.

The local contexts of the two nouns "boy" and "dog" in (4) are as follows (the dependency relations between nouns and their determiners are ignored):

(5)

Word	Local Contexts
boy	`(subj chase head)`
dog	`(adjn brown mod) (comp1 chase head)`

Using a broad coverage parser to parse a corpus, we construct a **Local Context Database**. An entry in the database is a pair:

(6) $(lc, C(lc))$

where lc is a local context and $C(lc)$ is a set of `(word frequency likelihood)`-triples. Each triple specifies how often `word` occurred in lc and the likelihood ratio of lc and `word`. The likelihood ratio is obtained by treating `word` and lc as a bigram and computed with the formula in (Dunning, 1993). The database entry corresponding to Table 1 is as follows:

$$\begin{bmatrix} lc & = & \texttt{(subj employ head)} \\ C(lc) & = & \texttt{((ORG 64 50.4) (plant 14 31.0)} \\ & & \texttt{...... (pilot 2 5.37))} \end{bmatrix}$$

3 The Approach

The polysemous words in the input text are disambiguated in the following steps:

Step A. Parse the input text and extract local contexts of each word. Let LC_w denote the set of local contexts of all occurrences of w in the input text.

Step B. Search the local context database and find words that appeared in an identical local context as w. They are called selectors of w: $\text{Selectors}_w = (\bigcup_{lc \in LC_w} C(lc)) - \{w\}$.

Step C. Select a sense s of w that maximizes the similarity between w and Selectors_w.

Step D. The sense s is assigned to all occurrences of w in the input text. This implements the "one sense per discourse" heuristic advocated in (Gale, Church, and Yarowsky, 1992).

Step C. needs further explanation. In the next subsection, we define the similarity between two word senses (or concepts). We then explain how the similarity between a word and its selectors is maximized.

3.1 Similarity between Two Concepts

There have been several proposed measures for similarity between two concepts (Lee, Kim, and Lee, 1989; Rada et al., 1989; Resnik, 1995b; Wu and Palmer, 1994). All of those similarity measures are defined directly by a formula. We use instead an information-theoretic definition of similarity that can be derived from the following assumptions:

Assumption 1: The commonality between A and B is measured by

$$I(common(A, B))$$

where $common(A, B)$ is a proposition that states the commonalities between A and B; $I(s)$ is the amount of information contained in the proposition s.

Assumption 2: The differences between A and B is measured by

$$I(describe(A, B)) - I(common(A, B))$$

where $describe(A, B)$ is a proposition that describes what A and B are.

Assumption 3: The similarity between A and B, $sim(A, B)$, is a function of their commonality and differences. That is,

$$sim(A, B) = f(I(common(A, B)), I(describe(A, B)))$$

The domain of $f(x, y)$ is $\{(x, y)|x \geq 0, y > 0, y \geq x\}$.

Assumption 4: Similarity is independent of the unit used in the information measure.

According to Information Theory (Cover and Thomas, 1991), $I(s) = -log_b P(s)$, where $P(s)$ is the probability of s and b is the unit. When $b = 2$, $I(s)$ is the number of bits needed to encode s. Since $log_b x = \frac{log_{b'} x}{log_{b'} b}$, Assumption 4 means that the function f must satisfy the following condition:

$$\forall c > 0, f(x, y) = f(cx, cy)$$

Assumption 5: Similarity is additive with respect to commonality.

If $common(A, B)$ consists of two independent parts, then the $sim(A, B)$ is the sum of the similarities computed when each part of the commonality is considered. In other words: $f(x_1 + x_2, y) = f(x_1, y) + f(x_2, y)$.

A corollary of Assumption 5 is that $\forall y, f(0, y) = f(x + 0, y) - f(x, y) = 0$, which means that when there is no commonality between A and B, their similarity is 0, no matter how different they are. For example, the similarity between "depth-first search" and "leather sofa" is neither higher nor lower than the similarity between "rectangle" and "interest rate".

Assumption 6: The similarity between a pair of identical objects is 1.

When A and B are identical, knowning their commonalities means knowing what they are, i.e., $I(common(A,B)) = I(describe(A,B))$. Therefore, the function f must have the following property: $\forall x, f(x,x) = 1$.

Assumption 7: The function $f(x,y)$ is continuous.

Similarity Theorem: The similarity between A and B is measured by the ratio between the amount of information needed to state the commonality of A and B and the information needed to fully describe what A and B are:

$$sim(A,B) = \frac{logP(common(A,B))}{logP(describe(A,B))}$$

Proof: To prove the theorem, we need to show $f(x,y) = \frac{x}{y}$. Since $f(x,y) = f(\frac{x}{y}, 1)$ (due to Assumption 4), we only need to show that when $\frac{x}{y}$ is a rational number, $f(x,y) = \frac{x}{y}$. The result can be generalized to all real numbers because f is continuous and for any real number, there are rational numbers that are infinitely close to it.

Suppose m and n are positive integers.

$$f(nx,y) = f((n-1)x,y) + f(x,y) = nf(x,y)$$

(due to Assumption 5). Thus, $f(x,y) = \frac{1}{n}f(nx,y)$. Substituting $\frac{x}{n}$ for x in this equation:

$$f(x,y) = nf(\frac{x}{n},y) = nf(x,ny) = n(\frac{1}{m}f(mx,ny))$$

Since $\frac{x}{y}$ is rational, there exist m and n such that $\frac{x}{y} = \frac{n}{m}$. Therefore,

$$f(x,y) = \frac{n}{m}f(\frac{mx}{ny},1) = \frac{n}{m}f(1,1) = \frac{n}{m} = \frac{x}{y}$$

Q.E.D.

For example, Figure 1 is a fragment of the WordNet. The nodes are concepts (or synsets as they are called in the WordNet). The links represent IS-A relationships. The number attached to a node C is the probability $P(C)$ that a randomly selected noun refers to an instance of C. The probabilities are estimated by the frequency of concepts in SemCor (Miller et al., 1994), a sense-tagged subset of the Brown corpus.

If x is a Hill and y is a Coast, the commonality between x and y is that "x is a GeoForm and y is a GeoForm". The information contained in this

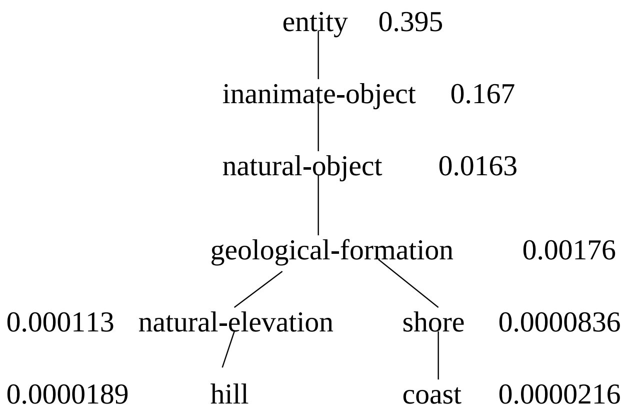


Figure 1: A fragment of WordNet

statement is $-2 \times logP(GeoForm)$. The similarity between the concepts Hill and Coast is:

$$sim(Hill, Coast) = \frac{2 \times logP(GeoForm)}{logP(Hill) + logP(Coast)} = 0.59$$

Generally speaking,

$$sim(C, C') = \frac{2 \times logP(\bigcap_i C_i)}{logP(C) + logP(C')} \qquad (7)$$

where $P(\bigcap_i C_i)$ is the probability of that an object belongs to all the maximally specific super classes (C_is) of both C and C'.

3.2 Disambiguation by Maximizing Similarity

We now provide the details of Step C in our algorithm. The input to this step consists of a polysemous word W_0 and its selectors $\{W_1, W_2, \ldots, W_k\}$. The word W_i has n_i senses: $\{s_{i1}, \ldots, s_{in_i}\}$.

Step C.1: Construct a similarity matrix (8). The rows and columns represent word senses. The matrix is divided into $(k+1) \times (k+1)$ blocks. The blocks on the diagonal are all 0s. The elements in block S_{ij} are the similarity measures between the senses of W_i and the senses of W_j. Similarity measures lower than a threshold θ are considered to be noise and are ignored. In our experiments, $\theta = 0.2$ was used.

$$S_{ij}(l,m) = \begin{cases} sim(s_{il}, s_{jm}) & \text{if } i \neq j \text{ and} \\ & sim(s_{il}, s_{jm}) \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

$$
(8) \quad \begin{array}{c|c|c|c|}
 & s_{01} \ldots s_{0n_0} & \ldots & s_{k1} \ldots s_{kn_k} \\
\hline
\begin{array}{c} s_{01} \\ \vdots \\ s_{0n_0} \end{array} & 0 & & S_{0k} \\
\hline
\begin{array}{c} s_{11} \\ \vdots \\ s_{1n_1} \end{array} & S_{10} & & S_{1k} \\
\hline
\vdots & & & \\
\hline
\begin{array}{c} s_{k1} \\ \vdots \\ s_{kn_k} \end{array} & S_{k0} & & 0 \\
\hline
\end{array}
$$

Step C.2: Let A be the set of polysemous words in $\{W_0, \ldots, W_k\}$:

$$A = \{W_i | n_i > 1\}$$

Step C.3: Find a sense of words in A that gets the highest total support from other words. Call this sense $s_{i_{max} l_{max}}$:

$$s_{i_{max} l_{max}} = argmax_{s_{il}} \sum_{j=0}^{k} support(s_{il}, W_j)$$

where s_{il} is a word sense such that $W_i \in A$ and $l \in [1, n_i]$ and $support(s_{il}, W_j)$ is the support s_{il} gets from W_j:

$$support(s_{il}, W_j) = \max_{m \in [1, n_j]} S_{ij}(l, m)$$

Step C.4: The sense of $W_{i_{max}}$ is chosen to be $s_{i_{max} l_{max}}$. Remove $W_{i_{max}}$ from A.

$$A \longleftarrow A - \{W_{i_{max}}\}$$

Step C.5: Modify the similarity matrix to remove the similarity values between other senses of $W_{i_{max}}$ and senses of other words. For all l, j, m, such that $l \in [1, n_{i_{max}}]$ and $l \neq l_{max}$ and $j \neq i_{max}$ and $m \in [1, n_j]$:

$$S_{i_{max} j}(l, m) \longleftarrow 0$$

Step C.6: Repeat from **Step C.3** unless $i_{max} = 0$.

3.3 Walk Through Examples

Let's consider again the word "facility" in (3). It has two local contexts: subject of "employ" `(subj employ head)` and modifiee of "new" `(adjn new mod)`. Table 1 lists words that appeared in the first local context. Table 2 lists words that appeared in the second local context. Only words with top-20 likelihood ratio were used in our experiments.

The two groups of words are merged and used as the selectors of "facility". The words "facility" has 5 senses in the WordNet.

Table 2: Modifiees of "new" with the highest likelihood ratios

word	freq	$log\lambda$	word	freq	$log\lambda$
post	432	952.9	bonds	223	245.4
issue	805	902.8	capital	178	241.8
product	675	888.6	order	228	236.5
rule	459	875.8	version	158	223.7
law	356	541.5	position	236	207.3
technology	237	382.7	high	152	201.2
generation	150	323.2	contract	279	198.1
model	207	319.3	bill	208	194.9
job	260	269.2	venture	123	193.7
system	318	251.8	program	283	183.8

1. something created to provide a particular service;
2. proficiency, technique;
3. adeptness, deftness, quickness;
4. readiness, effortlessness;
5. toilet, lavatory.

Senses 1 and 5 are subclasses of artifact. Senses 2 and 3 are kinds of state. Sense 4 is a kind of abstraction. Many of the selectors in Tables 1 and Table 2 have artifact senses, such as "post", "product", "system", "unit", "memory device", "machine", "plant", "model", "program", *etc.* Therefore, Senses 1 and 5 of "facility" received much more support, 5.37 and 2.42 respectively, than other senses. Sense 1 is selected.

Consider another example that involves an unknown proper name:

(9) DreamLand employed 20 programmers.

We treat unknown proper nouns as a polysemous word which could refer to a person, an organization, or a location. Since "DreamLand" is the subject of "employed", its meaning is determined by maximizing the similarity between one of {person, organization, locaton} and the words in Table 1. Since Table 1 contains many "organization" words, the support for the "organization" sense is much higher than the others.

4 Evaluation

We used a subset of the SemCor (Miller et al., 1994) to evaluate our algorithm.

4.1 Evaluation Criteria

General-purpose lexical resources, such as WordNet, Longman Dictionary of Contemporary English (LDOCE), and Roget's Thesaurus, strive to achieve completeness. They often make subtle distinctions between word senses. As a result, when the WSD task is defined as choosing a sense out of a list of senses in a general-purpose lexical resource, even humans may frequently disagree with one another on what the correct sense should be.

The subtle distinctions between different word senses are often unnecessary. Therefore, we relaxed the correctness criterion. A selected sense s_{answer} is correct if it is "similar enough" to the sense tag s_{key} in SemCor. We experimented with three interpretations of "similar enough". The strictest interpretation is $sim(s_{answer}, s_{key})$=1, which is true only when s_{answer}=s_{key}. The most relaxed interpretation is $sim(s_{answer}, s_{key})$ >0, which is true if s_{answer} and s_{key} are the descendents of the same top-level concepts in WordNet (e.g., entity, group, location, *etc.*). A compromise between these two is $sim(s_{answer}, s_{key}) \geq 0.27$, where 0.27 is the average similarity of 50,000 randomly generated pairs (w, w') in which w and w' belong to the same Roget's category.

We use three words "duty", "interest" and "line" as examples to provide a rough idea about what $sim(s_{answer}, s_{key}) \geq 0.27$ means.

The word "duty" has three senses in WordNet 1.5. The similarity between the three senses are all below 0.27, although the similarity between Senses 1 (responsibility) and 2 (assignment, chore) is very close (0.26) to the threshold.

The word "interest" has 8 senses. Senses 1 (sake, benefit) and 7 (interestingness) are merged.[2] Senses 3 (fixed charge for borrowing money), 4 (a right or legal share of something), and 5 (financial interest in something) are merged. The word "interest" is reduced to a 5-way ambiguous word. The other three senses are 2 (curiosity), 6 (interest group) and 8 (pastime, hobby).

The word "line" has 27 senses. The similarity threshold 0.27 reduces the number of senses to 14. The reduced senses are

- Senses 1, 5, 17 and 24: something that is communicated between people or groups.
  - 1: a mark that is long relative to its width
  - 5: a linear string of words expressing some idea
  - 17: a mark indicating positions or bounds of the playing area
  - 24: as in "drop me a line when you get there"
- Senses 2, 3, 9, 14, 18: group
  - 2: a formation of people or things beside one another
  - 3: a formation of people or things one after another
  - 9: a connected series of events or actions or developments
  - 14: the descendants of one individual
  - 18: common carrier
- Sense 4: a single frequency (or very narrow band) of radiation in a spectrum
- Senses 6 and 25: cognitive process
  - 6: line of reasoning
  - 25: a conceptual separation or demarcation
- Senses 7, 15, and 26: instrumentation
  - 7: electrical cable
  - 15: telephone line
  - 26: assembly line
- Senses 8 and 10: shape
  - 8: a length (straight or curved) without breadth or thickness
  - 10: wrinkle, furrow, crease, crinkle, seam, line
- Senses 11 and 16: any road or path affording passage from one place to another;
  - 11: pipeline
  - 16: railway
- Sense 12: location, a spatial location defined by a real or imaginary unidimensional extent;
- Senses 13 and 27: human action
  - 13: acting in conformity
  - 27: occupation, line of work;
- Sense 19: something long and thin and flexible
- Sense 20: product line, line of products
- Sense 21: space for one line of print (one column wide and 1/14 inch deep) used to measure advertising
- Sense 22: credit line, line of credit
- Sense 23: a succession of notes forming a distinctived sequence

where each group is a reduced sense and the numbers are original WordNet sense numbers.

[2]The similarities between senses of the same word are computed during scoring. We do not actually change the WordNet hierarchy

4.2 Results

We used a 25-million-word Wall Street Journal corpus (part of LDC/DCI[3] CDROM) to construct the local context database. The text was parsed in 126 hours on a SPARC-Ultra 1/140 with 96MB of memory. We then extracted from the parse trees 8,665,362 dependency relationships in which the head or the modifier is a noun. We then filtered out (lc, `word`) pairs with a likelihood ratio lower than 5 (an arbitrary threshold). The resulting database contains 354,670 local contexts with a total of 1,067,451 words in them (Table 1 is counted as one local context with 20 words in it).

Since the local context database is constructed from WSJ corpus which are mostly business news, we only used the "press reportage" part of SemCor which consists of 7 files with about 2000 words each. Furthermore, we only applied our algorithm to nouns. Table 3 shows the results on 2,832 polysemous nouns in SemCor. This number also includes proper nouns that do not contain simple markers (e.g., Mr., Inc.) to indicate its category. Such a proper noun is treated as a 3-way ambiguous word: person, organization, or location. We also showed as a baseline the performance of the simple strategy of always choosing the first sense of a word in the WordNet. Since the WordNet senses are ordered according to their frequency in SemCor, choosing the first sense is roughly the same as choosing the sense with highest prior probability, except that we are not using all the files in SemCor.

It can be seen from Table 3 that our algorithm performed slightly worse than the baseline when the strictest correctness criterion is used. However, when the condition is relaxed, its performance gain is much lager than the baseline. This means that when the algorithm makes mistakes, the mistakes tend to be close to the correct answer.

5 Discussion

5.1 Related Work

The Step C in Section 3.2 is similar to Resnik's noun group disambiguation (Resnik, 1995a), although he did not address the question of the creation of noun groups.

The earlier work on WSD that is most similar to ours is (Li, Szpakowicz, and Matwin, 1995). They proposed a set of heuristic rules that are based on the idea that objects of the same or similar verbs are similar.

[3] http://www.ldc.upenn.edu/

5.2 Weak Contexts

Our algorithm treats all local contexts equally in its decision-making. However, some local contexts hardly provide any constraint on the meaning of a word. For example, the object of "get" can practically be anything. This type of contexts should be filtered out or discounted in decision-making.

5.3 Idiomatic Usages

Our assumption that similar words appear in identical context does not always hold. For example,

(10) ... the condition in which the **heart** beats between 150 and 200 beats a minute

The most frequent subjects of "beat" (according to our local context database) are the following:

(11) PER, badge, bidder, bunch, challenger, democrat, Dewey, grass, mummification, pimp, police, return, semi, and soldier.

where PER refers to proper names recognized as persons. None of these is similar to the "body part" meaning of "heart". In fact, "heart" is the only body part that beats.

6 Conclusion

We have presented a new algorithm for word sense disambiguation. Unlike most previous corpus-based WSD algorithm where separate classifiers are trained for different words, we use the same local context database and a concept hierarchy as the knowledge sources for disambiguating all words. This allows our algorithm to deal with infrequent words or unknown proper nouns.

Unnecessarily subtle distinction between word senses is a well-known problem for evaluating WSD algorithms with general-purpose lexical resources. Our use of similarity measure to relax the correctness criterion provides a possible solution to this problem.

Acknowledgement

This research has also been partially supported by NSERC Research Grant OGP121338 and by the Institute for Robotics and Intelligent Systems.

References

Bruce, Rebecca and Janyce Wiebe. 1994. Word-sense disambiguation using decomposable models. In *Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics*, pages 139–145, Las Cruces, New Mexico.

Table 3: Performance on polysemous nouns in 7 SemCor files

correctness criterion	our algorithm	first sense in WordNet
$sim(s_{answer}, s_{key}) > 0$	73.6%	67.2%
$sim(s_{answer}, s_{key}) \geq 0.27$	68.5%	64.2%
$sim(s_{answer}, s_{key}) = 1$	56.1%	58.9%

Choueka, Y. and S. Lusignan. 1985. Disambiguation by short contexts. *Computer and the Humanities*, 19:147–157.

Cover, Thomas M. and Joy A. Thomas. 1991. *Elements of information theory*. Wiley series in telecommunications. Wiley, New York.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, March.

Gale, W., K. Church, and D. Yarowsky. 1992. A method for disambiguating word senses in a large corpus. *Computers and the Humannities*, 26:415–439.

Hearst, Marti. 1991. noun homograph disambiguation using local context in large text corpora. In *Conference on Research and Development in Information Retrieval ACM/SIGIR*, pages 36–47, Pittsburgh, PA.

Hudson, Richard. 1984. *Word Grammar*. Basil Blackwell Publishers Limited., Oxford, England.

Leacock, Claudia, Goeffrey Towwell, and Ellen M. Voorhees. 1996. Towards building contextual representations of word senses using statistical models. In *Corpus Processing for Lexical Acquisition*. The MIT Press, chapter 6, pages 97–113.

Lee, Joon Ho, Myoung Ho Kim, and Yoon Joon Lee. 1989. Information retrieval based on conceptual distance in is-a hierarchies. *Journal of Documentation*, 49(2):188–207, June.

Li, Xiaobin, Stan Szpakowicz, and Stan Matwin. 1995. A wordnet-based algorithm for word sense disambiguation. In *Proceedings of IJCAI-95*, pages 1368–1374, Montreal, Canada, August.

Mel'čuk, Igor A. 1987. *Dependency syntax: theory and practice*. State University of New York Press, Albany.

Miller, George A. 1990. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–312.

Miller, George A., 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*.

Ng, Hwee Tow and Hian Beng Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An examplar-based approach. In *Proceedings of 34th Annual Meeting of the Association for Computational Linguistics*, pages 40–47, Santa Cruz, California.

Rada, Roy, Hafedh Mili, Ellen Bicknell, and Maria Blettner. 1989. Development and application ofa metric on semantic nets. *IEEE Transaction on Systems, Man, and Cybernetics*, 19(1):17–30, February.

Resnik, Philip. 1995a. Disambiguating noun groupings with respect to wordnet senses. In *Third Workshop on Very Large Corpora*. Association for Computational Linguistics.

Resnik, Philip. 1995b. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of IJCAI-95*, pages 448–453, Montreal, Canada, August.

Wu, Zhibiao and Martha Palmer. 1994. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics*, pages 133–138, Las Cruces, New Mexico.

Yarowsky, David. 1992. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of COLING-92*, Nantes, France.

Yarowsky, David. 1994. Decision lists for lexical ambiguity resolution: Application to accent restoration in spanish and french. In *Proceedings of 32nd Annual Meeting of the Association for Computational Linguistics*, pages 88–95, Las Cruces, NM, June.

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