UNIVERSITÄT
DES
SAARLANDES

Integrating Syntax and Semantics for Word Sense Disambiguation

by

Will Roberts

A thesis submitted in partial fulfillment for the
degree of M.Sc. Language Science & Technology

in the
Computational Linguistics Department
Fachrichtung 4.7 Allgemeine Linguistik
Philosophische Fakultät II

under the supervision of
PD Dr. Valia Kordoni
Prof. Dr. Hans Uszkoreit

Tuesday, 26 April, 2011

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Declaration

I hereby confirm that the thesis presented here is my own work, with all assistance acknowledged.

Signed: ________________________

Date: Saarbrücken, __________________

UNIVERSITÄT DES SAARLANDES

Abstract

Philosophische Fakultät II
Fachrichtung 4.7 Allgemeine Linguistik
Computational Linguistics Department

Master of Science: Language Science & Technology

by Will Roberts

This thesis explores a number of topics related by the theme of using syntax to predict semantics and vice-versa. We conduct a set of experiments to map the lexicon of a deep parser to WordNet and so assign syntactic types to WordNet entries. We implement a wide-coverage word sense disambiguation algorithm and extend it with a novel paradigm to integrate word sense frequency information. We develop a statistical model of verb subcategorization, showing that it is effective for sense disambiguation; finally, we show how to integrate this model into our word sense disambiguation system and evaluate the final system on several commonly used competition tasks.

Acknowledgements

I thank Sara Hadji Moradlou for patiently listening to my worries during various stages of this project, and for feeding me countless times when I was writing. I am grateful to my mother, and to my flatmate Daniel Schneider for giving me their thoughts, comments and questions on the text of this thesis. Dr. Yi Zhang provided the idea of, and help with creating the mapping of the ERG to WordNet. Finally, my thanks to my project supervisor Dr. Valia Kordoni, who offered advice and help when it was most needed.

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

BNC	British National Corpus
ERG	English Resource Grammar
HPSG	Head-driven Phrase Structure Grammar
MFS	Most Frequent Sense baseline
MLE	Maximum Likelihood Estimate
NLP	Natural Language Processing
ODE	Oxford Dictionary of English
PCFG	Probabilistic Context-Free Grammar
POS	Part of Speech
SCF	Subcategorization Frame
SSI	Structural Semantic Interconnections
WSD	Word Sense Disambiguation
XWN	eXtended WordNet

Chapter 1

Introduction

This thesis examines the integration of syntactic and semantic information for enabling Natural Language Processing tasks, in particular for the problem of Word Sense Disambiguation. This kind of synthesis is important to the continuing research in deep syntactic processing and computational semantics, technologies which will be needed to support future applications such as the Semantic Web. Furthermore, integrating existing linguistic resources which encode different kinds of language knowledge is very valuable, considering the cost of developing these resources.

To this end, this work sets out to quantify the potential of semantics to predict syntax (in the context of extending a lexicon for a highly lexicalized deep grammar) and vice-versa (using syntactic features to improve the performance of a Word Sense Disambiguation system). The content of this thesis paper is organized as follows:

- Chapter 2 presents a theoretical background for the work done in this thesis.
- Chapter 3 summarizes an experiment to determine the degree to which lexical type and lexical semantics predict each other.
- Chapter 4 describes the implementation of a recent unsupervised Word Sense Disambiguation algorithm.
- Chapter 5 presents a novel extension integrating word sense information into the algorithm introduced in Chapter 4.
- Chapter 6 introduces a joint model of verb subcategorization and integrates it into the Word Sense Disambiguation algorithm. The original algorithm, as well as the modified versions developed in Chapters 5 and 6, are evaluated on several standard data sets.

- Chapter 7 summarizes the major points of the thesis.

Chapter 2

Background

This chapter provides an overview of previous research relevant to this thesis. In particular, Section 2.1 introduces the WordNet database, used in the experiments discussed in Chapter 3 as well as for the Word Sense Disambiguation tasks. Section 2.2 gives an overview of Word Sense Disambiguation. Section 2.3 introduces several data sets commonly used to evaluate Word Sense Disambiguation systems.

2.1 WordNet

WordNet (Miller et al., 1990; Fellbaum, 1998)[1] is a structured lexical database organized on psycholinguistic principles. It is freely available and has good coverage of the English language (on par with most college-level dictionaries), which has made it one of the most frequently used machine-readable dictionaries. In particular, it is widely accepted in Natural Language Processing research, and there are a number of freely available standard data sets using the WordNet inventory. The WordNet model is now being extended to include other languages, for example, by EuroWordNet (Vossen, 1998)[2], and MultiWordNet (Pianta et al., 2002)[3].

A concept in WordNet is represented by a *synset*, which is a collection of synonymous word senses[4] . Every word in the lexicon is represented by one or more word senses, and each word sense belongs to one and only one synset. As shown in Table 2.1, WordNet lists eight word senses for the lemma *table*; the first noun sense belongs to the synset $\{table_n^1,$

[1] http://wordnet.princeton.edu/

[2] http://www.illc.uva.nl/EuroWordNet/

[3] http://multiwordnet.itc.it

[4] This thesis uses the term *lemma* to refer to the base form of a word, and *word sense* to refer to a specific meaning of a lemma.

Synset	POS	Sense #	Definition
{table, tabular array}	noun	1	a set of data arranged in rows and columns: *"see table 1"*
{table}	noun	2	a piece of furniture having a smooth flat top that is usually supported by one or more vertical legs: *"it was a sturdy table"*
{table}	noun	3	a piece of furniture with tableware for a meal laid out on it: *"I reserved a table at my favorite restaurant"*
{mesa, table}	noun	4	flat tableland with steep edges: *"the tribe was relatively safe on the mesa but they had to descend into the valley for water"*
{table}	noun	5	a company of people assembled at a table for a meal or game: *"he entertained the whole table with his witty remarks"*
{board, table}	noun	6	food or meals in general: *"she sets a fine table"*; *"room and board"*
{postpone, prorogue, hold over, put over, table, shelve, set back, defer, remit, put off}	verb	1	hold back to a later time: *"let's postpone the exam"*
{table, tabularize, tabularise, tabulate}	verb	2	arrange or enter in tabular form

Table 2.1: The WordNet senses for the lemma *table*

tabular array$_n^1$}[5]. Note that WordNet also contains fixed *multi-word expressions*, such as *tabular array* or *put off*. Each synset has an associated dictionary definition (*gloss*), and possibly several usage examples. The latest version, WordNet 3.0, contains 155,287 words organized into 117,659 synsets.

In addition to listing word senses, WordNet also encodes a rich set of semantic relations between synsets or word senses:

antonymy (e.g., *rich* and *poor*);

hypernymy is also termed the *kind-of* or *is-a* relationship (e.g., *carnivore* is a hypernym of *canine*, which, in turn, is a hypernym of *dog*); the inverse relationship is termed *hyponymy*. Most synsets in WordNet have one hypernym, and thus hypernym-hyponym links organize nouns and verbs in the WordNet database into a semantic hierarchy or *ontology*;

meronymy is also called the *part-of* or *has-a* relationship (e.g., *hair* is a meronym of *mammal*); the inverse relationship is called *holonymy*;

pertainymy relating adjectives and nouns (e.g., *dental* pertains to *tooth*);

entailment for verbs (e.g., *snore* entails *sleep*);

[5] The convention w_p^s or `w#p#s` is used to identify WordNet word senses, where w is the lemma, p is the part of speech (n for nouns, v for verbs, a for adjectives, and r for adverbs), and s is the sense number of the given word in the WordNet dictionary, with 1 being the most frequently used sense, and higher sense numbers indicating less frequent senses.

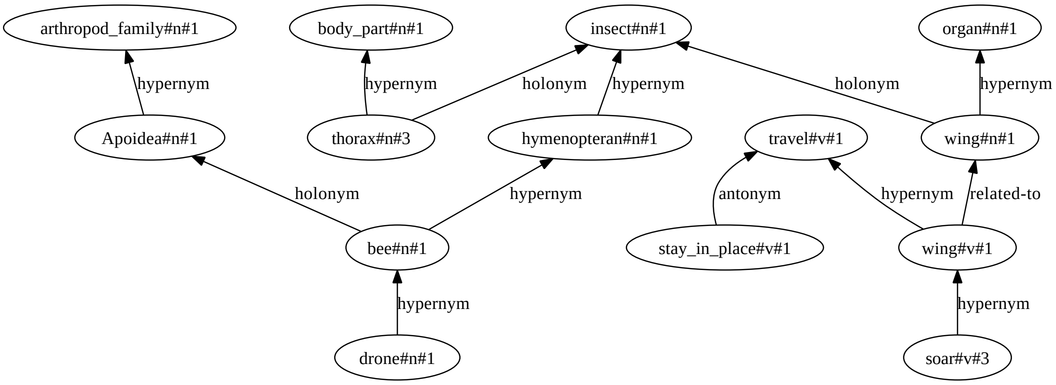


Figure 2.1: A fragment of the WordNet lexicon

> *Early*$_r^1$ in *November*$_n^1$ the *clouds*$_n^2$ *lifted*$_v^4$ *enough*$_r^1$ to carry out the *assigned*$_a^1$ *mission*$_n^2$. And Sweeney Squadron *put*$_v^1$ its *first*$_a^3$ *marks*$_n^2$ on the *combat*$_n^1$ *record*$_n^5$. Every *plane*$_n^1$ that could *fly*$_v^1$ was *sent*$_v^2$ into the *air*$_n^3$. Cricket *took*$_v^{11}$ *eight*$_a^1$ *ships*$_n^1$ and *went*$_v^1$ *south*$_r^1$ across the Straits and along the *north*$_a^1$ *coast*$_n^1$ of Mindanao to Cagayan.

Figure 2.2: Excerpt of the SemCor sense-tagged corpus

causation for verbs (e.g., *ignite* causes *burn*);

similarity for strong similarity between adjectives (e.g., *beautiful* is similar to *pretty*);

attribute relating nouns and adjectives (e.g., *temperature* is an attribute for *hot*);

see also for adjectives (e.g., *beautiful* is related to *attractive*)

Figure 2.1 illustrates some of these relations, and gives a sense of how the network is organized into a hierarchy through the hypernymy relation. Examples of the semantic relations listed in WordNet can also be seen in Table 4.1.1 (page 31).

2.1.1 SemCor

SemCor (Miller et al., 1993) is a subset of the Brown Corpus (Kučera and Francis, 1967) where all open-class content words (nouns, verbs, adjectives, and adverbs) have been manually tagged by trained lexicographers with word senses from WordNet. It was produced during the development of WordNet, and played a role in expanding WordNet's coverage. SemCor comprises 352 texts and around 234,000 sense-tagged words. Although it is the largest sense-tagged corpus available, it is a fairly small corpus: for instance, it contains only 83 words for which there are more than 100 tagged instances. Figure 2.2 shows an excerpt of SemCor.

2.2 Word Sense Disambiguation

Most common words have more than one meaning; Word Sense Disambiguation (WSD) refers to the task of determining which meaning is intended by a word in a particular context[6]. For example, in the sentences "she cashed a cheque at the *bank*" and "he sat on the *bank* of the river," the word *bank* is used with two different senses. While this distinction may be so obvious to a human as to escape notice, it is a prerequisite for Natural Language Processing systems which must determine the underlying meaning of texts; for example, a translation of these two sentences into French would use the word *banque* in the first sentence, but *rive* in the second.

WSD is an old problem in the field of Natural Language Processing, and was originally raised in the context of machine translation; Bar-Hillel (1960) discusses the problem of resolving lexical ambiguity as a serious challenge to high-quality automatic translation. NLP researchers have long assumed that high-quality sense disambiguation could be useful for other applications such as question answering or the development of the ontologies needed for the Semantic Web (Berners-Lee et al., 2001). Some results suggest that WSD can already improve the quality of statistical machine translation (Vickrey et al., 2005) and information retrieval (Stokoe, 2005) systems. To date, however, the state of the art in WSD is not considered reliable enough to make significant contributions to these fields.

2.2.1 Methods for WSD

The task of interpreting the meaning of a polysemous[7] word in context is classed as a "hard" Artificial Intelligence problem, and the difficulties it poses for automatic systems stem from it being a fundamentally knowledge-intensive undertaking (Cuadros and Rigau, 2008; Navigli, 2009; Ponzetto and Navigli, 2010).

Syntax, semantics, pragmatics, corpora, and dictionaries are all useful information sources for accurate WSD; Stevenson and Wilks (2001) give the following example:

1. "John did not feel *well*."

2. "John tripped near the *well*."

[6] For more background information, the reader is referred to Navigli (2009) for a recent survey of WSD, and to Ide and Véronis (1998) for an overview of the earlier history of the field. (Agirre and Edmonds, 2006) is a comprehensive collection discussing the major issues facing WSD.

[7] Some words such as *river* have only one sense, and are called *monosemous*. Words with multiple senses are called *polysemous*. Note that polysemous words may have monosemous synonyms; for example, *coinage* can mean a newly invented word, or a collective noun for coins—a synonym for this latter sense is the unambiguous *specie*.

3. "The *bat* slept."
4. "He bought a *bat* from the sports shop."

In distinguishing the senses of the word *well* in sentences (1) and (2), we note that the word is used as an adverb in the first sentence, but as a noun in the second. Indeed, part of speech (i.e., a type of *syntactic* information) is a very simple and effective cue for reducing semantic ambiguity. In sentence (3), we can use the selectional preferences (*semantic* knowledge) of the verb *sleep* to deduce that the subject should be animate, and choose the mammal sense of *bat.* These methods do not help with sentence (4), which needs a *pragmatic* interpretation of what items might be reasonably bought at a sports shop.

The tokens in the immediate context of an ambiguous word (*collocational* evidence) can be helpful, for instance, in telling "*bow* wave" apart from "*bow* and arrow". Knowledge of a text's *domain* can also be crucial: for example, consider the specialized meanings of *phone* in linguistics, *slice* in golf, and *capital* in architecture.

WSD implementations may make use of simple heuristics to reduce the complexity of the problem. Two of best known are:

one sense per discourse (Gale et al., 1992b): in a given text, all occurrences of one particular word will share the same sense; and

one sense per collocation (Yarowsky, 1993): a word in a particular local context will always have the same sense.

WSD is often described as a classification problem, whereby each word should be assigned a sense from a list of possible senses, as defined externally by some resource. Under this formulation, WSD requires at the very least a lexicon of words in the language, and a listing of their possible senses.

An early and conceptually simple WSD algorithm, which still performs quite well on current WSD exercises was presented by Lesk (1986). The algorithm only makes use of a machine-readable dictionary. A word is disambiguated by comparing the dictionary definitions of its various senses to the other words in the same context (e.g., in the same sentence); the sense whose definition has the highest overlap with the context words is chosen. For example, assume that the word *pine* can mean either *"a kind of evergreen tree with needle-shaped leaves"* or *"to waste away through sorrow or illness,"* and that *cone* can mean *"a solid body which narrows to a point," "something of this shape whether solid or hollow,"* or *"the fruit of certain evergreen trees."* Then, *pine* *cone* can be automatically disambiguated by finding the senses of the two words whose glosses overlap to the greatest degree; here, a Lesk implementation will choose sense 1 of *pine* and sense 3 of *cone*, due to the overlap of the terms *tree* and *evergreen*.

Since then, systems have appeared that use a greater variety of linguistic tools and resources, including syntactic features and semantic knowledge, coupled with sophisticated methods such as neural networks, machine learning algorithms, and combined classifiers. Most recently, WSD conferences have been dominated by supervised learning systems which are trained on word co-occurrence data. However, improving WSD quality using this approach is difficult due to the lack of high-quality annotated data for training these systems (this is termed the *knowledge acquisition bottleneck* (Gale et al., 1992a)).

2.2.2 What are Word Senses?

Intuitively, some words have multiple meanings. Fundamental to WSD is the question of how to represent those word meanings. The traditional approach taken by lexicographers is to collect these various senses from textual evidence and list them (Ayto, 1983); thus, a dictionary comprises one kind of *sense inventory*. Here, sense inventory means a resource which lists, for each word in the language, the set of possible senses that that word can have.

Any sense inventory must adopt a policy for dealing with the overlap of word senses. For instance, a population of language users will exhibit variation on sense judgements: some people will consider two uses of a word to represent the same sense, while others will disagree. Some psycholinguistic research suggests that senses of a word may be related asymmetrically, so that, for example, the sense of the word *firm* meaning *strict* seems to be associated with the sense meaning *solid*, but not vice-versa (Williams, 1992). Further, Stock (1983) argues that some words such as *culture* seem to derive their usefulness precisely from their *lack* of clear sense divisions.

Sense inventories can be either explicit or implicit: either senses are deliberately enumerated in predefined lists, or else they are inferred in some way according to the demands of the application. For the purposes of WSD, implicit sense inventories can be built using several different approaches. In sense clustering, or *word sense discrimination* (Schütze, 1998), uses of a word in context are clustered according to some metric, and the clusters are taken to be distinct word senses[8]. In machine translation or in other cross-lingual contexts, the appropriate translation of a word can be used as its sense (Resnik and Yarowsky, 2000). McCarthy (2002) lately proposed *lexical substitution* as

[8] This approach follows from the *distributional hypothesis*; as characterized by Firth (1957), *"you shall know a word by the company that it keeps."*

a way to study word meanings: here, words in context are paraphrased using another word or multi-word expression.

The most common paradigm used in WSD, however, is to employ a dictionary such as WordNet as an explicit list of senses[9]. A problem with this approach are the idiosyncratic and unpredictable phenomena which make it difficult for an explicit sense inventory to be complete and adequately expressive:

regular polysemy the predictable alternations of sense in some word classes. An example is *chicken* as an animal as opposed to *chicken* as food; this particular alternation extends to similar words such as *duck*, *lamb*, or *fish*, although not to *cow/beef* or *pig/pork*.

metaphor the use of one word to represent another by analogy, often found in poetry. John Keats' *Ode on a Grecian Urn* begins: *"Thou still unravish'd bride of quietness, / thou foster-child of silence and slow time, / sylvan historian, who canst thus express / a flowery tale more sweetly than our rhyme ... "*

slang and jargon For example, *pig* can be a slang term for a police officer, a tool for cleaning and inspecting pipelines, a block of metal from a smelting furnace, or, in chemistry, an apparatus used in distillation. In Cockney Rhyming Slang, a *pig* can even mean a beer (*pig's ear*).

semantic drift the tendency of words to change their meaning over time. For instance, the word *cartoon* originally meant a drawing, made on heavy paper, for producing frescoes (cognate with the Italian *cartone* and Dutch *karton*); the common sense of the word as it is used today was only introduced by *Punch* magazine in 1843.

coinages new words continuously created in a language. The March 2011 update to the Oxford English Dictionary added the words *couch surfer*, *LOL*, *OMG* and *wassup* to the largest catalogue of the English language.

A second issue with explicit sense inventories is that of *sense granularity*, meaning how fine the distinctions made between senses are. Sense granularity is illustrated clearly by dictionaries, which organize lexical knowledge by part of speech and etymology into hierarchical lists, sorted by head words. Typically, dictionaries carry two levels of sense distinction:

***homonyms* or *homographs*,** words that are written the same way but have unrelated meanings, such as the *bark* of a tree vs. a dog's *bark*[10], and

[9] Some recent work has even examined the possibility that pages in Wikipedia can be taken as word senses (Bunescu and Paşca, 2006; Mihalcea, 2007). See also section 4.1.5.

[10] Example taken from Krovetz (1997).

finer *word senses* or *polysemes*, with related meanings, such as a violin *bow* and a *bow* for shooting arrows. Several of these polysemes may be grouped together under one homograph.

Sense distinctions are sometimes presented as a "tree", with a few homonym distinctions at the top, followed by increasingly fine distinctions; in the limit, every separate use of the word has a slightly different meaning[11]. This makes sense granularity into an important parameter for WSD problems.

While Stevenson and Wilks (2001), among others, have argued that the homonym level (i.e., relatively coarse sense distinctions) is the proper resolution for WSD[12], it seems that sense granularity is generally agreed to be application-dependent. In machine translation, for instance, there are cases where semantic ambiguity is preserved across languages (e.g., *interest* in English and French); however, in applications for supporting language learning, very fine-grained sense distinctions may be helpful. Issues of sense inventory granularity will be discussed below in greater detail in Section 2.2.5.

2.2.3 WSD Evaluation

Following the model of DARPA competitive evaluations in speech recognition and information retrieval, the first SENSEVAL workshop was organized in 1998 to evaluate automatic WSD systems for the English, French and Italian languages (Kilgarriff, 1998). The workshop set out to establish benchmarks for WSD performance and demonstrate the validity of WSD as an NLP task.

SENSEVAL popularized the model of evaluating WSD in an *in-vitro* context[13], meaning that WSD systems are evaluated in a task-independent manner. Typically, evaluation uses predefined lists of senses from a dictionary, and assumes that a word in context can be tagged with a single best sense. The first SENSEVAL competition used the HECTOR dictionary (Atkins, 1993) as a sense inventory; subsequent workshops have used WordNet.

[11] This issue is well-known to lexicographers, who create dictionaries according to an editorial policy that advocates either "lumping" or "splitting" (Kilgarriff, 1997).

[12] Krovetz (1997) even argues that senses in different parts of speech could be grouped together for some applications (e.g., *review* as a noun and as a verb). Ide and Wilks (2006), however, make the point that some polysemes, though etymologically related, are regularly used in ways just as distinct as homonyms. They give as examples "a sheet of *paper*" and "a daily *paper*", or a "finger *nail*" and a "picture *nail*". Indeed, these usages would have different translations, for example, in French.

[13] *In-vivo* evaluation (evaluation conducted in the context of some other task), or methodologies without explicit predefined sense inventories are also possible and valid. Examples include the automatic subcategorization acquisition task (Preiss and Korhonen, 2004) on SENSEVAL-3, and the word sense discrimination and lexical substitution tasks (McCarthy and Navigli, 2007), which were evaluated in the SemEval-2007 workshop. The reader is referred to Edmonds and Kilgarriff (2002) for a detailed discussion of alternative evaluation paradigms.

Participating systems generate sense annotations either for a small selection of polysemous words (*lexical sample* tasks) or all open-class words (*all-words* tasks); responses are compared to a human-tagged gold standard (i.e., a manually sense-annotated corpus), and performance is measured with precision and recall statistics, as in information retrieval tasks.

For reference, these statistics are computed as

$$\begin{aligned} P \text{ (precision)} &= \frac{\text{tp}}{\text{tp} + \text{fp}} \\ R \text{ (recall)} &= \frac{\text{tp}}{\text{tp} + \text{fn}} \end{aligned}$$

where tp is the count of the *true positives*, fp is *false positives*, fn is *false negatives* (and tn is *true negatives*). For easier comparison of systems, the F_1 *score*, or simply *F-score* is the harmonic mean of these two statistics:

$$F_1 = \frac{2PR}{P + R}$$

Since 1998, SENSEVAL (now renamed to SemEval) workshops continue to be organized every three years to study and evaluate the field of WSD. The effort hopes to generate consensus on what tasks should be evaluated, what metrics should be used to measure performance, and how to compare different systems, and thereby stimulate research progress. Several commonly used SENSEVAL tasks are discussed below in Section 2.3.

2.2.4 Bounds on Performance

The standard evaluation methodology established by the SENSEVAL competitions raises questions about reasonable bounds on system performance. A lower bound on performance is usually defined by some baseline score, which should represent a naïve implementation. The *random baseline* is such an implementation, and picks word senses from a uniform probability distribution; its scores tend to be very low, and it is easily outperformed by most systems.

2.2.4.1 Most Frequent Sense Baseline

Another common baseline is the *most frequent sense* baseline (MFS), or *first sense* baseline. This baseline relies on the fact that WordNet senses are ordered in the database by their frequency in the SemCor sense-tagged corpus: sense number one of a given word is its most frequent sense in SemCor, and increasing sense numbers indicate less frequent senses. Thus, the MFS baseline simply selects the first listed sense for a word in WordNet.

In contrast to the random baseline, however, the use of the MFS baseline as a lower bound for performance is problematic: the MFS baseline commonly achieves F-scores between 60–80% (Snyder and Palmer, 2004; Navigli et al., 2007), which surpass results from many WSD algorithms.

The reasons for this are examined by Kilgarriff (2004), who notes that word senses tend to be highly skewed, with a few very frequent senses, and many very infrequent senses. The more common a word is, the more senses it tends to have, and the more skewed is the distribution of these senses. According to this model, the first sense for a very common word might easily account for as many as 90–95% of that word's occurrences.

It may be tempting to conclude that, given its good performance, the MFS baseline is satisfactory for disambiguating words. There are, however, shortcomings to the first sense baseline. The MFS baseline uses sense rankings in WordNet, which in turn are determined by word sense counts from SemCor; since the SemCor corpus is relatively small, sense distributions for infrequent words can be estimated incorrectly (e.g., the most common sense of *tiger* in SemCor is *"a fierce or audacious person"*). Furthermore, SemCor is a subset of the Brown corpus, and so represents textual content balanced for topic and style. However, words can have very different dominant senses depending on the domain of the text in which they are used. The sense rankings in WordNet are thus not valid in domain-specific text that is very different to the Brown corpus (e.g., *star* in SemCor most often means a *"celestial body"*; in popular news, it would more likely refer to a celebrity). As such, it is not surprising that researchers have observed its performance to be worse on domain-specific texts (e.g., (Navigli et al., 2007)).

However, if the principle underlying the MFS baseline could be adapted to text from other domains, this might provide an answer to the lack of annotated data that supervised WSD systems suffer from. McCarthy et al. (2004) present such a system. For any given corpus of unstructured text (e.g., the British National Corpus), they automatically construct a thesaurus from a distributional similarity metric. This produces for each word, a list of its nearest neighbour words. The sense distribution for each word can then be estimated: a semantic similarity metric such as the Lesk algorithm (see Section 2.2.1) is used to compute the contribution of these neighbours to each sense of the word. The result allows the prediction of the most frequent sense of any word found in the corpus used. In evaluation on the SENSEVAL-2 English all-words task, the first senses determined by this method performed almost as well as the SemCor first sense heuristic, despite not requiring any sense-tagged data.

2.2.4.2 Inter-annotator Agreement

The upper bound on WSD performance is usually taken to be the human inter-annotator agreement, since it is difficult to see how a WSD system could reasonably outperform a human on sense disambiguation, and, if it did, how such a result should be interpreted.

Senseval-1 judged high inter-annotator agreement to be a requirement for establishing the validity of the WSD task as a whole, since, as a ceiling on performance, it needs to be high enough to enable practical applications. The workshop, which used the Hector dictionary as a sense inventory for a lexical sample task, employed professional lexicographers to produce the sense-tagged gold standard, with arbitration in cases where the resulting sense tags did not agree. The results were high inter-annotator agreement and good replicability (both statistics around 95%) measured for the sense tagging (Kilgarriff and Rosenzweig, 2000).

Sense-tagging is a labor-intensive and expensive enterprise[14], however, and subsequent evaluations have invested less effort in making the gold standard than the first Senseval. Annotations carried out by non-lexicographers, simpler tie-breaking though voting schemes, and the shift to WordNet as a sense inventory for later Senseval workshops have resulted in dramatically lower inter-annotator agreement figures reported in the literature. For illustration, the Senseval-3 English all-words test, using the WordNet inventory, reported an inter-annotator agreement of 72.5% when experienced annotators manually sense-tagged the gold standard corpus (Snyder and Palmer, 2004).

2.2.5 Sense Inventory Granularity

The low inter-annotator agreement values reported on Senseval tasks using the WordNet sense inventory, coupled with the already high F-scores achieved by the MFS baseline, mean that WSD systems are evaluated in a very narrow performance band.

The poor inter-annotator agreement is seen to stem from the use of WordNet as a sense inventory; doubts as to WordNet's suitability for WSD have been voiced since it was adopted for Senseval-2 (Kilgarriff, 2001). One difficulty is that WordNet is essentially designed as a thesaurus, with its goal being to capture synsets (groups of synonyms), and not to coherently divide the senses of a single word. The result is that the sense distinctions made in WordNet are not necessarily clear or valid.

[14] Edmonds (2000) estimates that a person takes one minute to do one sense annotation; given that corpora should be sense-tagged by two annotators to estimate the inter-rater agreement, and that this agreement is often around 60%, requiring arbitration, he gives a figure of 4 hours to produce a gold standard for 100 word instances.

Cluster	WordNet Sense	Gloss
Financial Institution	1	a financial institution that accepts deposits and channels the money into lending activities: *"he cashed a check at the bank"*
	3	a supply or stock held in reserve for future use (especially in emergencies)
	4	a building in which the business of banking transacted: *"the bank is on the corner of Nassau and Witherspoon"*
	6	a container (usually with a slot in the top) for keeping money at home: *"the coin bank was empty"*
	8	the funds held by a gambling house or the dealer in some gambling games: *"he tried to break the bank at Monte Carlo"*
Land Formation	2	sloping land (especially the slope beside a body of water): *"they pulled the canoe up on the bank"*
	7	a long ridge or pile: *"a huge bank of earth"*
	9	a slope in the turn of a road or track
	10	a flight maneuver: *"the plane went into a steep bank"*
Array	5	an arrangement of similar objects in a row or in tiers: *"he operated a bank of switches"*

Table 2.2: Clustering WordNet senses for the noun *bank* as in (Navigli, 2006)

Several researchers have come to the conclusion that the problem lies in particular with WordNet's sense granularity. WordNet senses are on the level of polysemes, and senses are not grouped hierarchically by homonyms (for example, in Table 2.1, senses 2 and 3 of *table* both refer to types of furniture, but this similarity is not reflected in the organization of WordNet). Thus, for any given word, its sense distinctions in WordNet are often very fine-grained and so subtle as to be difficult for humans, and presumably also machines, to recognize (Edmonds and Kilgarriff, 2002). Snyder and Palmer (2004) argue that most inter-annotator disagreements occur between a few closely related WordNet senses with subtle distinctions. In the same vein, Véronis (1998) cites the poor agreement among amateur annotators using fine-grained sense inventories as evidence that the "average" language user is able to disambiguate word sense without needing to be able to distinguish very fine-grained sense distinctions.

Navigli (2009) suggests that the fine sense resolution of WordNet is the chief obstacle to breaking the ceiling of 70% accuracy for automatic systems, and, indeed, it has been noted that adopting a more coarse-grained sense inventory results in higher inter-annotator agreement: Ng et al. (1999) clustered WordNet senses based on disagreements between sense-tagged corpora and were then able to measure inter-annotator agreements over 90%. They found that the resulting automatically-produced clusters corresponded to intuitive judgements about sense boundaries. The OntoNotes project (Hovy et al., 2006) has put this idea into practice by grouping WordNet senses iteratively until 90% inter-annotator agreement is reached on a sense annotation task.

Consequently, there has been considerable investigation into how to group WordNet senses into coarser clusters. Navigli (2006) produced an automatic mapping of WordNet senses onto homonym entries in the Oxford Dictionary of English (ODE) (Soanes and Stevenson, 2003) by disambiguating words in the gloss definitions of WordNet senses and ODE senses, and then scoring matches based on the number of WordNet relations connecting these gloss words. WordNet senses mapping to the same ODE homonym were then judged to belong to the same sense cluster. Applied to the whole WordNet database, this process resulted in a reduction from 60,302 word senses to 40,079, and a decrease in average polysemy from 3.14 to 2.09. The author reported the mapping to be 80% accurate, as measured using a manually sense-clustered subset of WordNet. Table 2.2 shows the clustering automatically produced in this way for the noun *bank*[15].

The SemEval-2007 Task 7 (further discussed below in Section 2.3.3) used this method to define a coarse-grained English all-words WSD task. For the sense-tagged instances on the task, the sense clusters from (Navigli, 2006) were recreated by hand; the inter-annotator agreement for producing this clustering was measured to be 86.4%. The sense-tagged gold standard for the task had an inter-annotator agreement of 93.8% (Navigli et al., 2007). Average results from systems participating in the task were much higher than on previous SENSEVAL exercises (the best F-score was 82.5%); however, the gap between the best system and the MFS baseline was comparable to that seen on previous all-words exercises.

Snow et al. (2007) used a support vector machine to create a supervised classifier for clustering WordNet senses. The classifier was trained on hand-labelled sense clustering data, and output a similarity measure which allows for adaptive clustering, meaning that the granularity of word sense distinctions can be adjusted to the task at hand. On evaluation using a held-out data set, they found that their classifier outperformed Navigli's clustering approach.

McCarthy (2006) has noted that there are often several ways to group word senses, and that this raises the question of what distinctions are important to keep, and which can be ignored. Instead of creating word sense clusters, she develops a metric for determining how similar one word sense is to another using distributional or semantic similarity to judge the distance between word senses. This metric allows a word sense to be related to others, which are not themselves related. She finds this method performs well under evaluation using a human-produced gold standard. As with the approach in (Snow et al.,

[15] The example clearly shows the fineness of the sense distinctions in WordNet that provides the motivation for sense clustering: the discrimination of $bank_n^1$ (the financial institution) from $bank_n^4$ (the building where banking transactions are performed) would not be useful for most WSD applications. Also of note here is the incorrect assignment of sense 10 to the land formation cluster; this may represent a sense not found in the ODE.

2007), this method allows the granularity of the sense inventory to be freely adjusted as desired.

Erk et al. (2009) introduce a new methodology for studying sense distinctions: they asked three annotators to examine a word in context and then to rate how applicable all the WordNet senses for that word were, using a 5-point scale. The results they collected were significantly correlated with manual annotations previously done for SemCor and Senseval-3; results were also significantly correlated with SemEval-2007 lexical substitution data. They found that different lemmas exhibited different annotation patterns; some (e.g., the verb *win*) tended to show behaviour that would indicate that a single sense was usually most applicable, whereas others (e.g., *different*) showed that many senses seemed to apply to varying degrees. The study examined to what extent coarse-grained sense clusters could account for the sense applicability data and the findings call this paradigm into question. Frequently, there were senses which could not be grouped together into a sense cluster, but which were nonetheless rated *similar*, *very similar*, or *identical* in certain contexts. It thus seems possible that the strategy of attempting to overcome the inadequacies of WordNet as a sense inventory by clustering word senses may be doomed to fail: in the same way that sense inventories need conform to WSD task at hand, there may not be one canonical sense clustering that is acceptable for all WSD applications.

Further research may indicate whether this result poses a real problem to the proposed solution of using a coarser-grained version of WordNet, but the recent evaluations of WSD research indicate that more investigation into the sense inventory is required for improved system performance.

2.3 Common Evaluation Tasks for WSD

This section provides an overview of several tasks commonly used for evaluation of WSD systems.

2.3.1 Senseval-2 English All-Words

The Senseval-2 English all-words task (Edmonds and Cotton, 2001) consisted of approximately 5,800 words of running text taken from three files of the Wall Street Journal section of the Penn Treebank. Figure 2.3 shows an example sentence taken from the task. 2,473 open-class content words were tagged with senses from WordNet version 1.7; 89 of these are tagged as "untaggable" (i.e., the sense intended is not listed in WordNet).

```
"
It
<head id="d01.s23.t02">made</head>
our
<sat id="d01.s23.t05.s0">New</sat>
<head id="d01.s23.t05" sats="d01.s23.t05.s0">Year</head>
,
"
<head id="d01.s23.t08">says</head>
Mr.
Quinlan
.
```

Figure 2.3: A sentence from the Senseval-2 English all-words task

Multi-word expressions are tagged, with the "head" word linked to its "satellite" words. Heads are not tagged with part of speech (pos) or lemmas, and so participating systems must perform pos tagging and lemmatization themselves.

2.3.2 Senseval-3 English All-Words

The Senseval-3 English all-words task (Snyder and Palmer, 2004) closely resembled the Senseval-2 task, except that it used WordNet 1.7.1 senses. The task consisted of 5,000 words of running text from two Wall Street Journal articles (an editorial and a news story) and one excerpt from the Brown Corpus (fiction). 2,037 open-class content words were manually sense-tagged with WordNet senses by two experienced annotators; disagreements were decided by a third annotator. The inter-annotation agreement was reported to be 72.5%, and the organizers noted that disagreements seemed to occur most frequently for words with difficult sense distinctions in WordNet that were too fine for human annotators to tell apart. Even with this caveat, the figure might still be regarded as relatively low[16]; a possible conclusion is that the task itself is inherently difficult compared to other wsd competitions. 26 systems submitted results; the best F_1 figure reported was 65.2%. All of the best performing systems used supervised machine learning algorithms. The mfs baseline on the task was calculated to be 62.4%.

Compared to Senseval-2, a greater number of participating systems achieved results above the mfs baseline level. The organizers concluded that automatic wsd systems had reached a performance barrier at 65–70% accuracy, and that further increases in wsd accuracy would require more a coarser-grained sense inventory than WordNet.

[16] For instance, the random baseline on Senseval-3 is lower than on Senseval-2 (cf. Tables 6.2 and 6.3, page 54), which indicates that the Senseval-3 all-words task has higher polysemy.

2.3.3 SemEval 2007 Task 7: Coarse-Grained English All-Words

The SemEval-2007 Task 7 (Navigli et al., 2007) is a coarse-grained English all-words disambiguation exercise, meaning that all open-class words in the test data must be disambiguated. Each test instance specifies a lemmatized lexical form and its part of speech; a participating system must assign the single most appropriate sense to the word. Figure 2.4 shows an example sentence from the test data.

The task uses a sense inventory based on WordNet 2.1, where senses have been automatically clustered to coarser sense classes as described in (Navigli, 2006); this sense clustering is further discussed below in Section 2.2.5. The sense clustering used for instances in the task test data was manually created by an expert lexicographer, and the automatic clustering for the rest of WordNet was made available to participants. On the coarse-grained sense inventory, the average polysemy of the test set is reported to be 3.06, compared to an average polysemy using the original fine-grained WordNet sense inventory of 6.18.

The test data set consists of 5,377 words of running text from five different articles; 2,269 are open-class content words that are used as test instances. These content words were manually tagged with coarse senses by a lexicographer and constitute the gold standard for evaluation. Using the coarse-grained sense inventory, the pairwise inter-annotator agreement for sense-annotating the task test set was 93.8%.

The random baseline scored an F_1 measure of 52.4%, while the most frequent sense baseline scored 78.9%. The best system result was an F_1 measure of 83.2%; this was achieved by the Structural Semantic Interconnections algorithm (see Chapter 4), participating out of competition.

2.3.3.1 Coarse-Grained Mapping

The coarse-grained sense clustering of Navigli (2006) is given by a mapping of word senses to sets of other word senses. This mapping defines, for each WordNet sense of a given word, the set of other WordNet senses of that word which are equivalent to it. These sets are proper clusters, so that, if a word sense w maps to a given set $\{x, y, z\}$, we can be sure that x also maps to $\{w, y, z\}$, and so on. An example of the sense clustering for the senses of the *bank* is shown in Table 2.2.

Unfortunately, the mapping is not consistent under synset transitivity. This can be clarified using an example, depicted in Figure 2.5. Under the coarse-grained mapping, $harm_n^2$ ("the occurrence of a change for the worse") and $harm_n^3$ ("the act of damaging

```
<sentence id="d002.s020">
If
you
<instance id="d002.s020.t001" lemma="be" pos="v">were</instance>
<instance id="d002.s020.t002" lemma="especially" pos="r">especially</instance>
<instance id="d002.s020.t003" lemma="helpful" pos="a">helpful</instance>
in
a
<instance id="d002.s020.t004" lemma="corrupt" pos="a">corrupt</instance>
<instance id="d002.s020.t005" lemma="scheme" pos="n">scheme</instance>
you
<instance id="d002.s020.t006" lemma="receive" pos="v">received</instance>
not
<instance id="d002.s020.t007" lemma="just" pos="r">just</instance>
<instance id="d002.s020.t008" lemma="cash" pos="n">cash</instance>
in
a
<instance id="d002.s020.t009" lemma="bag" pos="n">bag</instance>
,
but
<instance id="d002.s020.t010" lemma="equity" pos="n">equity</instance>
.
</sentence>
```

Figure 2.4: A sentence from the SemEval 2007 coarse-grained English all-words task

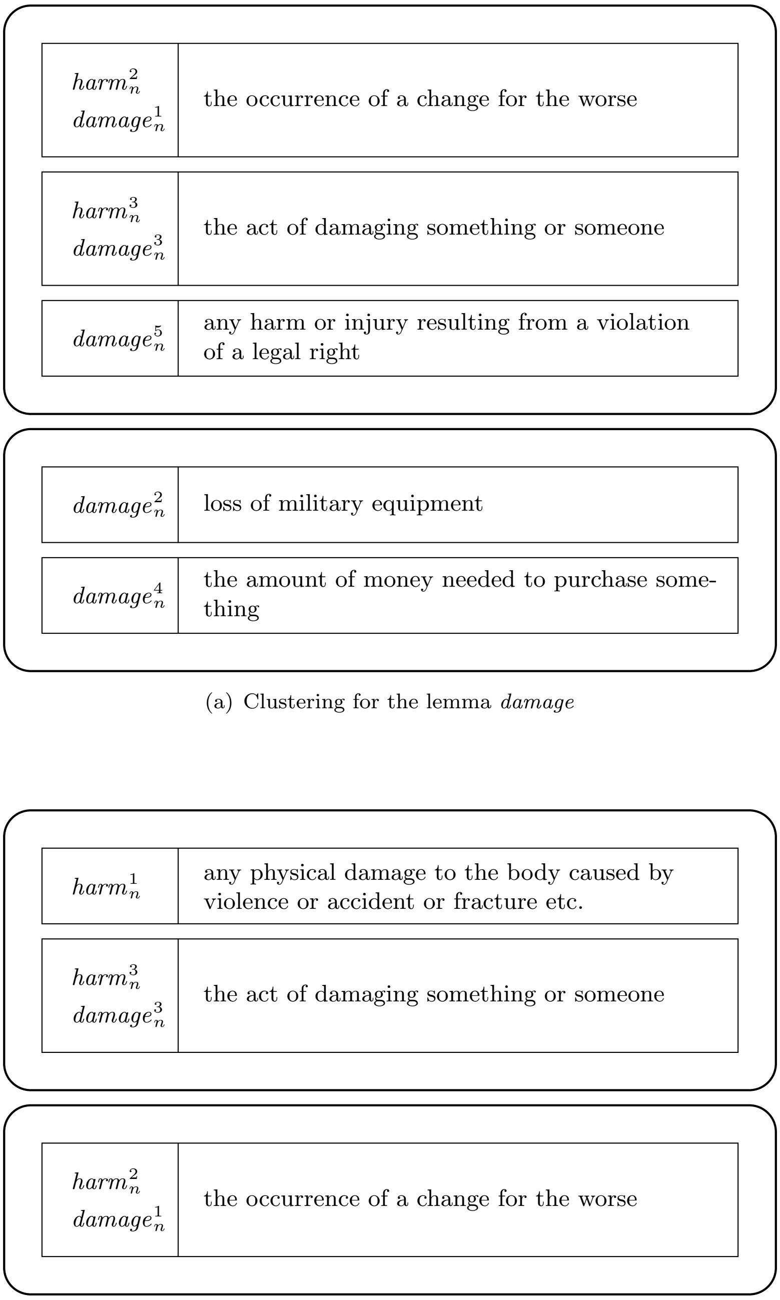
something or someone") are clustered together. These words have the synonyms *damage*^1_n and *damage*^3_n, respectively. Synset transitivity would mean that the coarse-grained mapping would also cluster *damage*^1_n and *damage*^3_n together; however, they are not.

This means that word senses are only clustered in a *shallow* manner, as a function of a given lemma. Other approaches, such as (Snow et al., 2007) and (McCarthy, 2006) avoid this shortcoming.

2.3.4 Domain-WSD Data Set

Koeling et al. (2005) created a set of corpora for testing WSD systems on domain-specific text. Due to the size of the corpora, this was set up as a lexical sample task, where sentences in the corpora are chosen randomly (i.e., not running text), and each sentence contains only one word to be sense-tagged. The data set is in three parts, made up of excerpts from the Sports and Finance sections of the Reuters corpus (Rose et al., 2002), and also from the balanced British National Corpus (BNC) (Leech, 1992). These corpora contain sense-tagged examples of 41 nouns, selected as follows: 18 words having one sense in the financial domain and one sense in the sports domain (F&S cds); 8 words particularly common in the Finance corpus (F sal); 8 words common in the Sports corpus (S sal); and 7 words equally common in both the Finance and Sports corpora (eq sal). These words are shown in Table 2.3.

There are around 100 examples of each word in each domain. The polysemy of the tagged words ranges from 2 to 13 senses; the average is 6.7 senses. Each sentence was sense



$harm_n^2$ $damage_n^1$	the occurrence of a change for the worse
$harm_n^3$ $damage_n^3$	the act of damaging something or someone
$damage_n^5$	any harm or injury resulting from a violation of a legal right

$damage_n^2$	loss of military equipment
$damage_n^4$	the amount of money needed to purchase something

(a) Clustering for the lemma *damage*

$harm_n^1$	any physical damage to the body caused by violence or accident or fracture etc.
$harm_n^3$ $damage_n^3$	the act of damaging something or someone

$harm_n^2$ $damage_n^1$	the occurrence of a change for the worse

(b) Clustering for the lemma *harm*

FIGURE 2.5: Inconsistent sense clustering in (Navigli, 2006)

F&S cds:	*club*, *manager*, *record*, *right*, *bill*, *check*, *competition*, *conversion*, *crew*, *delivery*, *division*, *fishing*, *reserve*, *return*, *score*, *receiver*, *running*, *pitch*
F sal:	*package*, *chip*, *bond*, *market*, *strike*, *bank*, *share*, *target*
S sal:	*fan*, *star*, *transfer*, *striker*, *goal*, *title*, *tie*, *coach*
eq sal:	*will*, *phase*, *half*, *top*, *performance*, *level*, *country*

Table 2.3: Words used in the Domain-WSD data set

annotated by at least three reviewers using senses from WordNet 1.7.1; inter-annotator agreement was 65% (60% for the BNC sentences, 65% for Sports, 69% for Finance).

The Domain-WSD data set is used by different studies in various ways; in this thesis, we follow the approach taken by Agirre et al. (2009). They give the following procedure for creating a gold standard for the Domain-WSD data set: for each test instance, the "correct" (gold-standard) sense is taken to be the sense chosen by the majority of taggers; instances having two majority senses in a tie are discarded.

Chapter 3

Mapping the English Resource Grammar to WordNet

A theme of this project is the strength of the relationship between syntactic knowledge and semantic knowledge. We conducted a simple experiment to test whether semantic similarity entails syntactic similarity.

3.1 Motivation

The experiment used the LinGO HPSG English Resource Grammar (ERG) (Flickinger, 2000), an open-source broad-coverage precision grammar for the Head-Driven Phrase Structure Grammar (HPSG) formalism. The lexicon included in the grammar version from July 2008 lists 33,534 entries, each of which encodes a lexical form and a type; the ERG types are arranged in a hierarchy, and multiple inheritance is used to share features between related types. To estimate how well syntax and semantics correspond to each other, we attempted to assign ERG syntactic types to WordNet synsets. For this experiment, we used version 3.0 of the WordNet database.

The aim of the study was to measure the strength of the relationship between syntax, as encoded in ERG types, and semantic similarity, as encoded in WordNet; if this relationship was strong, then, for instance, the ERG lexicon could be automatically extended using WordNet (since, given a word in the ERG lexicon that is found in a given WordNet synset, one could assume that the other words in that synset have similar syntactic structure). In the context of Word Sense Disambiguation, this would mean that the output of a deep parser (which would yield a syntactic analysis of an ambiguous word) could be used to help select the correct word sense. Some previous work suggests that

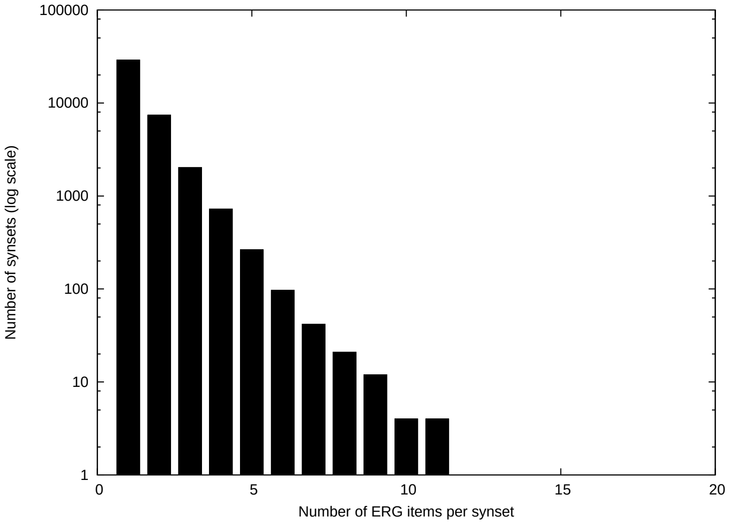


FIGURE 3.1: Number of ERG items per WordNet synset

this kind of semantics-to-syntax mapping is possible; for instance, Roa (2007, p. 26) found that verbs in a given VerbNet class tend to share similar ERG types.

3.2 Generating the Mapping

The ERG contains 33,534 entries which list 24,253 lexically unique stem forms. Each stem has, on average, 1.35 associated grammatical types; in total, there are 921 distinct ERG types used in the ERG lexicon. The ERG stem forms were automatically mapped to WordNet lemmas. Of the 24,253 ERG stem forms, 16,802 (using 609 ERG types) were found in WordNet as they appear in the ERG lexicon. A further 996 forms could have been found in WordNet using the WordNet lemmatizer (this process would map ERG forms such as *brighter* and *brightest* to *bright*, and *charities* to *charity*), but, as these ERG forms would have different syntactic types, this lemmatization was not done. 6,455 ERG forms were not found in WordNet at all (these forms included *Compaq*, *ex wife*, and *Bjørn*).

The 16,802 ERG forms corresponding to WordNet lemmas appear in 39,685 different WordNet synsets (each synset containing, on average, 1.40 ERG items). Figure 3.1 shows a histogram of the number of ERG forms contained in each synset. This set of synsets was filtered to find only those synsets containing two or more ERG items, resulting in a list of 10,607 synsets that contained 10,524 ERG items (average of 2.49 ERG items per synset). This list referenced 554 distinct ERG types.

These synsets were then investigated to discover how consistently they could be labelled with ERG types. Each synset's set of possible ERG types was defined to be the intersection

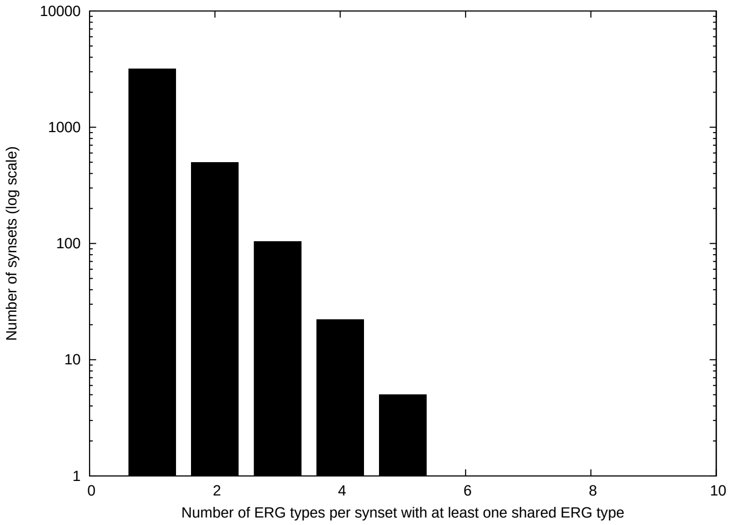


FIGURE 3.2: The number of ERG types per type-consistent synset

POS	Count
Adv	199
Adj	940
Verb	1237
Noun	1419

TABLE 3.1: Number of type-consistent synsets by part of speech

of the sets of possible ERG types for each ERG item contained in that synset. Formally, for a synset s containing n ERG items $\{e_1, e_2, \ldots, e_n\}$, its possible ERG types $t(s)$ are given by

$$t(s) = \bigcap_{i=1}^{n} t(e_i)$$

Note that this study was done using a naïve, strict comparison of ERG types; that is, we made no attempt to define a distance metric on the ERG type hierarchy to quantify how similar or dissimilar two types are.

We now focus on the *type-consistent synsets*, meaning those synsets which had at least one ERG type shared by all the contained ERG items (formally, $\{s : t(s) \neq \emptyset\}$). The 3,795 type-consistent synsets together make use of 145 ERG types; Figure 3.2 shows the number of ERG types per synset (the average is 1.21). The graph would appear to show an inverse exponential relationship. The type-specific synsets together contain a total of 5,159 different ERG items (an average of 2.20 items per synset).

Table 3.1 shows the counts of these synsets by part of speech. Finally, Figure 3.3 shows the sizes of the type-consistent synsets (average size 2.93)[1].

[1] Note that there are two synsets here of size 1; this is due to the fact that ERG items were mapped to WordNet lemmas in a case-insensitive manner, which resulted in confusion of the forms *ana* (a collection of anecdotes about a person or place) and *Ana* (mother of the ancient Irish gods).

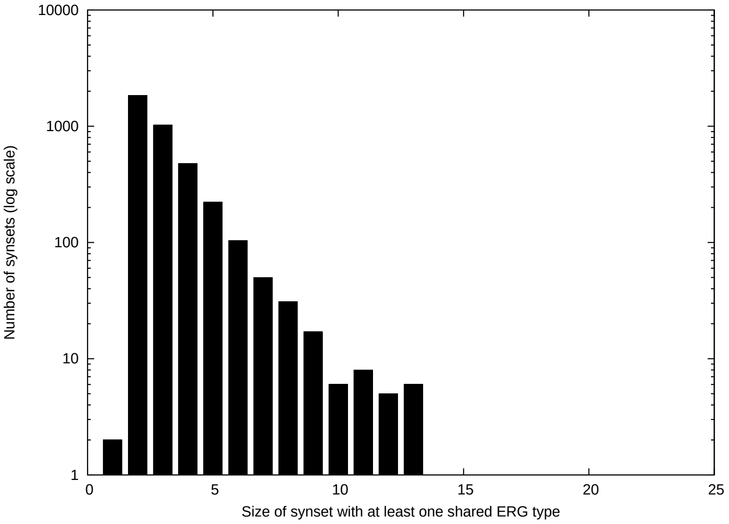


FIGURE 3.3: The size of type-consistent synsets

		Shared ERG Type	
		Yes	No
Same Synset	Yes	10,733	13,835
	No	8,400,896	46,946,562

TABLE 3.2: Dependence between ERG type and WordNet synset for pairs of ERG items

3.3 Evaluation

An examination of the ERG items contained in type-consistent synsets seems to reveal that some are indeed being sorted out according to syntactic differences, while others reflect idiosyncrasies or shortcomings in the ERG lexicon. For example, the terms *fawn*, *inning*, and *versatile* are in type-consistent synsets, while *conjure*, *railing*, and *danger* are not. *Conjure* is not clustered with its synonyms *raise*, *evoke*, and *stir* due to its different subcategorization frame. *Railing* is not included in a type-consistent synset since it is listed in the ERG lexicon as a mass or countable noun, while its synonym *rail* is strictly a countable noun. Finally, *danger* is not included, since its synonym *risk* is listed in the ERG only as a noun requiring a prepositional complement headed by *of* (e.g., *the risk of complications*); arguably, *risk* can also without a prepositional complement.

To more systematically judge the strength of the dependence between synonymy and syntactic type, we conducted a statistical test. Using the 10,524 ERG items found in the 10,607 synsets containing two or more ERG items, all possible pairs of ERG items were drawn; each pair was then checked to see if it could be found in the same synset, and if its two component ERG items had an ERG type in common. The resulting counts (see Table 3.2) show a very strong relationship between ERG type and whether a pair of

ERG items is to be found in the same WordNet synset ($\chi^2(1, N = 55{,}372{,}026) = 15491$, $p < 0.001$).

A final check was made by comparing the part of speech tag of each type-consistent synset with the ERG types assigned by the mapping to that synset. ERG types are made up of three or four fields, separated by underscores and dashes; the first of these fields is a part of speech category. By mapping these part of speech prefixes from the ERG types to WordNet part of speech tags, we were able to compare the fit between the POS listed in WordNet and the POS that would be assigned by the ERG type mapping. Of the 3,795 type-consistent synsets, 87.17% had one or more ERG types assigned which agreed with the WordNet part of speech tag. While it is possible that the organisation of part of speech information in the ERG type hierarchy is more complicated than this simple test assumes, error analysis suggests that the 12.83% of synsets with POS disagreements actually represent cases that should not be mapped to each other (adjective synsets tagged as nouns, nouns tagged as verbs, adverbs tagged as adjectives, etc.).

3.4 Expanding the Semantic Neighbourhood

Two ERG items in the same WordNet synset are much more likely than chance to have the same ERG type. However, the usefulness of this effect is potentially undermined by the relatively low number of synsets containing two or more ERG items: of 16,802 ERG items that are found in WordNet, only 5,159 items are found in type-consistent synsets. This low yield may be due a sparse overlap between the ERG lexicon and WordNet. To see whether this could be overcome, we made use of a common trick for expanding the size of a semantic neighbourhood in WordNet—enlarging each synset by including in it the contents of its hypernym synset. This results, for example, in the synset {*coat*} (*"an outer garment that has sleeves and covers the body from shoulder down; worn outdoors"*) also containing ERG items that are found in its hypernym {*overgarment*, *outer garment*} (*"a garment worn over other garments"*).

The primary effect of including hypernym synsets is that WordNet synsets contain more ERG items than if hypernyms are not included (with hypernyms, WordNet synsets have an average of 2.51 ERG items each, compared with 1.40 items if hypernyms are not included). As Table 3.3 shows, this effect persists through the filtering and type-assignment steps, leading ultimately to more type-consistent synsets found; including hypernyms does not seem to affect the fraction of ERG items that are contained in type-consistent synsets. At the same time, type-consistent synsets seem to have more precisely determined ERG types, and the final mapping of ERG types to synsets uses fewer ERG types. The POS tag check described above shows that in 87.17% of the 6,429

	Not including hypernyms	Including hypernyms
Mean number of ERG items per synset	1.40	2.51
Number of WordNet synsets containing two or more ERG items	10,607	27,250
Number of ERG items contained in these synsets	10,524	13,685
Number of ERG types used in these synsets	554	582
Number of type-consistent synsets	3,795	6,429
ERG types used by type-consistent synsets	145	137
Average number of types per synset	1.21	1.16
Number of ERG items in type-consistent synsets	5,159 (49.0%)	6,636 (48.5%)
Mean number of ERG items per type-consistent synset	2.20	2.35

Table 3.3: Effect of including synset hypernyms in the ERG-WordNet mapping

type consistent synsets, the ERG types assigned agree with the WordNet POS. Thus, this method seems to be an effective way of increasing the coverage of the ERG-WordNet mapping without negatively affecting its accuracy.

3.5 Cross-Validation for Predicting Syntactic Type

We estimated that, using the mapping between the ERG and WordNet, 3,597 new items[2] could potentially be added to the ERG lexicon with very little human effort.

To explore this hypothesis, we conducted a 10-fold cross-validation experiment. For each fold, a test set of 10% of the ERG items found in WordNet were "removed" from the ERG lexicon; the remaining lexical entries were then used to build the mapping to WordNet, using the "hypernym synset inclusion" heuristic discussed above. The mapping allows ERG syntactic types to be predicted for WordNet synsets. After filtering these predictions by part of speech, the syntactic types predicted for the test set items were evaluated.

The mapping produced a mean of 5137.3 type-consistent synsets, which is actually slightly more than the expected value ($6429 \times \frac{9}{10} \times 87.17\% = 5043.7$). On average, syntactic types are predicted for 271.7 of the 1680.2 test items (16.2%). Of these, 58.8% have at least one syntactic type judged correct by the ERG lexicon, and 18.8% have only correctly predicted types. For the syntactic type predictions, we calculated a precision of 49.0% and a recall of 31.5%.

Although the precision measured on this test may seem low, it should be taken with a grain of salt: it is problematic to use the ERG lexicon as a gold standard, since there may be syntactic types that are missing in the ERG lexicon. Overall, it seems that

[2] 9,896 unique forms in the type-consistent synsets in WordNet (using the hypernym synset inclusion described above) less the 5,761 forms already present in the ERG lexicon which overlap with this set gives 4,135 potential new entries. Assuming that checking the part of speech tags on both sides of the mapping (∼87% agreement) will give high-accuracy output, this leaves 3,597 WordNet forms which could be added to the ERG lexicon.

mapping the ERG lexicon to WordNet could very well play a role in the automatic or semi-automatic expansion of the lexicon.

3.6 Conclusion

The mapping between the ERG and WordNet seems very promising thus far. While the quality of the mapping might be improved by enlarging either WordNet or the ERG, it seems that most of the unmapped items are due to particular cases (such as *conjure*) where one word differs syntactically from its synonyms (here, *conjure* takes a different subcategorization frame from *evoke*). Despite these cases, it seems that a significant fraction of the lexicon can be successfully mapped, indicating that, in many cases, semantics does predict syntax.

It remains to be seen how useful this relationship might be, but for some applications it should be adequate. For instance, close examination of cases where synonyms in WordNet have different types in the ERG might provide a useful way to quickly check the correctness of the ERG lexicon.

Chapter 4

Building a Word Sense Disambiguation System

This chapter describes the implementation of a knowledge-based unsupervised algorithm for Word Sense Disambiguation.

The WSD literature displays a growing trend towards exploring *knowledge-based* algorithms, meaning unsupervised methods predominantly based on the use of lexical resources such as machine-readable dictionaries. These methods represent an attractive alternative to supervised systems that require sense-labelled training data, existing corpora of which are few in number and small in size, and which are expensive to produce. *Graph-based* methods, a type of knowledge-based method, operate on semantic networks (graphs), whose nodes represent distinct concepts; often a connectivity measure such as PageRank is then used to identify the "important" nodes in the graph, and these are taken as sense assignments. WSD algorithms of this type have recently attained state of the art performance on standard evaluation metrics (Navigli and Velardi, 2005; Agirre and Soroa, 2009; Ponzetto and Navigli, 2010). A property common to these systems is that they use only lexical semantic information, and are ignorant of syntax and word order. In other words, Bar-Hillel's (1960) example sentences *"the box was in the pen"* and *"the pen was in the box"* would appear identical to a knowledge-based WSD algorithm: both sentences would be represented as {`pen#n`, `box#n`}.

For our WSD implementation, we chose to implement SSI-Dijkstra (Cuadros and Rigau, 2008), a wide-coverage knowledge-based WSD algorithm which bears strong similarities to graph-based approaches. It is a simplification of the Structural Semantic Interconnections (SSI) system (Navigli and Velardi, 2005), and has been used for disambiguating topic signatures, mapping FrameNet to WordNet (Laparra and Rigau, 2009), and integrating ontologies into WordNet (Cuadros et al., 2010).

Network section	Edge count
WordNet relations	371,322
eXtended WordNet glosses	1,177,392
WordNet Domains	199,860
KnowNet concordances*	4,889,678
WordNet++ relations	111,946
Total edge count	6,750,198

* Statistics are given for the largest available version of KnowNet (KnowNet-20), after experiments demonstrated that larger versions of KnowNet increased the SSI algorithm's accuracy.

Table 4.1: Edges in the SSI semantic network

The SSI family of algorithms rely on a large directed graph for disambiguation. The graph is built from WordNet; nodes in the graph are WordNet synsets, and edges between these nodes represent semantic relations listed in WordNet. SSI algorithms use the graph to define a *semantic distance* metric. Given a polysemous word with a set of potential senses, such an algorithm chooses the sense with the least semantic distance to some context; for disambiguating running text, this context includes the monosemous words in the same sentence, as well as words in the same sentence which have already been disambiguated.

As the semantic graph encodes considerable ontological information about the real world, SSI algorithms are fairly stable across domains. Navigli and Velardi (2005) note that SSI performs better on moderately technical text; language that is too technical is not covered by WordNet, and language that is too general is difficult to disambiguate, since the words tend not to be strongly semantically related.

Section 4.1 discusses the construction of the semantic network used by the algorithm; Section 4.2 presents the algorithm itself.

4.1 Building the Semantic Network

The semantic network used for the SSI algorithm is a directed graph, where WordNet synsets are nodes, and semantic relations between synsets are labelled directional edges. Inverse edges are also generated as needed so that all semantic relations are symmetric. A fragment of the completed semantic network is shown in Figure 4.1.

The semantic network incorporates information from a variety of sources, discussed in the sections below. Table 4.1 lists the number of semantic relations derived from each knowledge source.

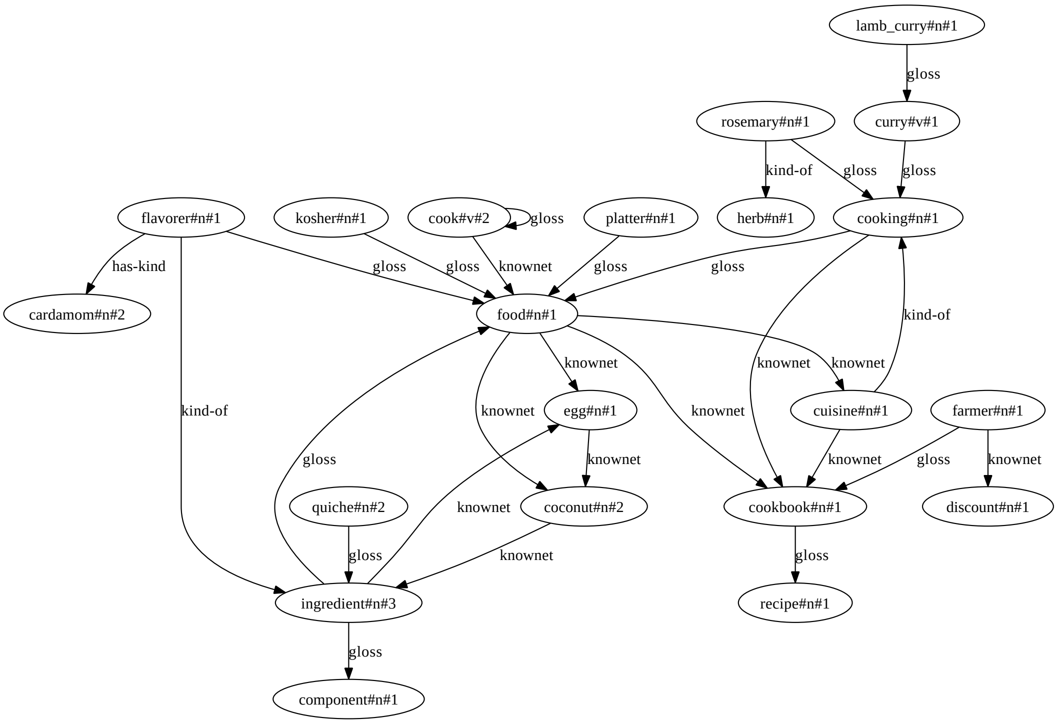


Figure 4.1: A fragment of the SSI semantic network

4.1.1 WordNet

Semantic relations between synsets in WordNet are transformed directly into edges between the synset nodes in the semantic graph. For instance, a hypernymy relation from WordNet would be stored in the network as the edge $poodle_n^1 \xrightarrow{\text{kind}-\text{of}} dog_n^1$. Table 4.2 illustrates the network edges created from the WordNet lexicon[1].

4.1.2 eXtended WordNet

The eXtended WordNet[2] (XWN) (Mihalcea and Moldovan, 2001) is a project to semi-automatically parse and semantically disambiguate the glosses for WordNet's synsets. For each synset, all content words in the gloss are disambiguated to WordNet senses. For example, the gloss for the synset $\{bulk_n^2,\ mass_n^7,\ volume_n^2\}$ is disambiguated to *"the* $property_n^3$ *of* $something_n^1$ *that* is_v^1 $great_a^5$ *in* $magnitude_n^1$*."* XWN also gives the parse tree and a logical form of the gloss for each synset. A small fraction of the database has been manually sense-tagged for checking the disambiguation accuracy. While the remainder of the words in XWN have been disambiguated automatically, the project

[1] The notation $n_1 \overset{e}{\to} n_2$ or (n_1, e, n_2) is used to indicate an edge, where n_1 is the start node, e is the edge label, and n_2 is the end node.

[2] http://xwn.hlt.utdallas.edu

WordNet Relation	Edge Label	Example Graph Edge	Count
antonymy	antonym	`(deny#v#1, antonym, admit#v#1)`	7,656
attribute	attr	`(importance#n#1, attr, unimportant#a#1)`	1,286
causation	cause	`(seat#v#1, cause, sit#v#1)`	219
	cause-1	`(stumble#v#2, cause-1, trip#v#2)`	219
derivationally related	derived	`(diet#n#4, derived, diet#v#1)`	61,246
domain (topic)	cdomain	`(mihrab#n#1, cdomain, islam#n#2)`	6,534
	cdomain-1	`(medicine#n#1, cdomain-1, gauze#n#1)`	6,534
domain (region)	rdomain	`(kamikaze#n#2, rdomain, japan#n#2)`	1,327
	rdomain-1	`(england#n#1, rdomain-1, a_level#n#1)`	1,327
domain (usage)	udomain	`(scissors#n#1, udomain, plural#n#1)`	1,174
	udomain-1	`(slang#n#2, udomain-1, squeeze#n#4)`	1,174
entailment	entails	`(look#v#1, entails, see#v#1)`	409
	entails-1	`(try#v#1, entails-1, succeed#v#1)`	409
hypernymy	has-kind	`(plastic#n#1, has-kind, polypropylene#n#1)`	96,773
hyponymy	kind-of	`(dry_rot#n#2, kind-of, fungus#n#1)`	96,773
member holonymy	has-member	`(rome#n#1, has-member, roman#n#1)`	12,262
member meronymy	member-of	`(militiaman#n#1, member-of, militia#n#1)`	12,262
part holonymy	has-part	`(hand#n#1, has-part, palm#n#1)`	8,874
part meronymy	part-of	`(guangzhou#n#1, part-of, china#n#1)`	8,874
participle	participle	`(dimpled#a#1, participle, dimple#v#1)`	107
	participle-1	`(dump#v#4, participle-1, dumped#a#1)`	107
pertainymy	pertains-to	`(nocturnal#a#2, pertains-to, night#n#1)`	6,691
	pertains-to-1	`(moon#n#1, pertains-to-1, lunar#a#1)`	6,691
see also	see-also	`(unprepared#a#1, see-also, unready#a#1)`	3,219
	see-also-1	`(leap_out#v#1, see-also-1, jump#v#1)`	3,219
similar	similar-to	`(hateful#a#1, similar-to, abominable#a#1)`	22,622
substance holonymy	makes-up	`(iron#n#1, makes-up, steel#n#1)`	793
substance meronymy	made-of	`(aluminum_foil#n#1, made-of, aluminum#n#1)`	793
verb group	vgroup	`(smother#v#2, vgroup, suffocate#v#5)`	1,748
Total			371,322

Table 4.2: Creating graph edges from WordNet

attempts to achieve high precision by combining two WSD systems in a voting scheme, and the accuracy of the whole project is about 73%[3].

Using the XWN database, each synset in the SSI semantic network is connected to all synsets that appear in its gloss. Each of these connections results in an edge with the label "gloss". For instance, chair^1_n ("a seat for one person, with a support for the back") results in the edge $\mathit{chair}^1_n \xrightarrow{\text{gloss}} \mathit{seat}^3_n$. Symmetric inverse edges are also added, so that the graph also contains the edge $\mathit{seat}^3_n \xrightarrow{\text{gloss}-1} \mathit{chair}^1_n$.

4.1.3 WordNet Domains

Magnini and Cavaglià (2000) semi-automatically annotated each noun, verb and adjective synset in WordNet with one or more Subject Field Codes, or domain labels[4], similar to the field labels used in dictionaries (e.g., Medicine or Architecture). The domain

[3] Estimating 100% accuracy on the 14,446 "gold" (i.e., manually sense-annotated) gloss words , 90% on the 57,192 "silver" words (where both automatic WSD systems agreed on the sense tag), and 70% on the 382,137 "normal" words (where only one WSD system's output was used).

[4] http://wndomains.itc.it

labels they use are based on the Dewey Decimal Classification system, and are arranged into a topic hierarchy. Some WordNet synsets (e.g., man_n^3 "the generic use of the word to refer to any human being") are either not specific to a particular domain, or are frequently used in many different contexts, and are labelled with the generic domain of Factotum. While a synset may be assigned multiple domain labels, 95% of all synsets have only a single label[5].

We manually associated the 168 domain labels used in the WordNet Domains project to WordNet word senses; this mapping is given in Appendix A. Each synset with a non-generic domain label is connected to that domain's synset in the network with an edge labelled "domain". For example, $doctor_n^1$ is tagged with the Medicine domain label, and results in the edge $doctor_n^1 \xrightarrow{\text{domain}} medicine_n^3$. Symmetric inverse edges are also added, so that the graph also contains the edge $medicine_n^3 \xrightarrow{\text{domain}-1} doctor_n^1$.

4.1.4 KnowNet

KnowNet[6] (Cuadros and Rigau, 2008) is a large sense-tagged corpus of collocations. It was automatically constructed from the Topic Signatures from the Web (Agirre and López de Lacalle, 2004)[7], a large corpus of collocations acquired from the World Wide Web. For every word sense of every noun in WordNet, a search query was built from the noun's monosemous relatives (refer to the discussion of polysemy and Footnote 7 on page 6), and run on the Google search engine. The text fragments in the search results were then collected and filtered using a text frequency-inverse document frequency (*tf-idf*) metric. This process results in a large ranked list of words for each nominal word sense in WordNet. The full topic signature corpus consists of 35,250 topic signatures, and an average of 6,877 words per signature.

In KnowNet, the first n words (with n varying from five to 20) of each topic signature are automatically disambiguated for part of speech and word sense to WordNet word senses (i.e., both part of speech and WordNet sense number are determined in a single step) using the SSI-Dijkstra algorithm[8]. The semantic disambiguation results in a sense-annotated dictionary of collocations. KnowNet thus contains lists such as

> $party_n^1$: $tammany\_hall_n^1$, $federalist_n^1$, $whig_n^3$, $campaigner_n^1$, $election_n^1$, $bill_n^3$, $reelection_n^1$, $backbencher_n^1$, $political_a^2$, $filibuster_n^1$, $floor_n^9$, $queen_n^1$, $motion_n^6$ ...

[5] WordNet also contains domain information (the relations *cdomain*, *rdomain* and *udomain*), but the coverage is much smaller than that of the WordNet Domains project. WordNet contains approximately 10,000 domain links, while WordNet Domains has around 100,000.

[6] http://adimen.si.ehu.es/web/KnowNet

[7] http://ixa.si.ehu.es/Ixa/resources/sensecorpus

[8] The version of SSI-Dijkstra used to create KnowNet had a semantic network based on WordNet and eXtended WordNet, containing 99,635 nodes and 636,077 edges.

The largest version of the corpus, KnowNet-20, contains around 2,400,000 collocation links between WordNet synsets.

KnowNet is organized as a semantic network, so integrating it into the SSI network is not difficult. Each semantic connection in KnowNet is stored in the network with an edge labelled "knownet". For example, the list above results in the edge $party_n^1 \xrightarrow{\text{knownet}} election_n^1$. KnowNet connections are symmetric, so that $X \xrightarrow{\text{knownet}} Y$ implies $Y \xrightarrow{\text{knownet}} X$.

4.1.5 Wikipedia

Ponzetto and Navigli (2010) describe an automatic mapping of Wikipedia pages to WordNet senses, resulting in a resource they call WordNet++[9]. The work is based on the assumption that Wikipedia pages can be seen as broadly equivalent to word senses.

The automatic mapping is accomplished with use of a Lesk-like metric which calculates the overlap between a semantic context representing a Wikipedia page and a context representing a WordNet sense. Evaluated against a manually-mapped subset of Wikipedia pages, this process achieves an F_1 measure of 84.4 (compared to an inter-annotator agreement on the manual mapping of $\kappa = 0.9$). Following the mapping, all hyperlinks between Wikipedia pages are transferred to semantic relations between WordNet senses. Finally, these relations are filtered using the Wikipedia categories of a given pair of pages; the sense overlap between each pair of Wikipedia categories is computed, and sense relations are retained if they connect word senses in highly related categories.

The authors used this collection of semantic relations with a graph-based WSD algorithm, and showed that the resulting system was competitive with state-of-the-art supervised systems on the SemEval-2007 coarse-grained all-words task (Section 2.3.3), and that it also performed very well on domain-specific texts (Section 2.3.4).

Each of the connections in the WordNet++ resource are stored in the graph as an edge labelled "wikipedia". For example, this results in the edge $anesthesia_n^1 \xrightarrow{\text{wikipedia}} operation_n^5$. Symmetric inverse edges are also added, so that the graph also contains the edge $operation_n^5 \xrightarrow{\text{wikipedia}-1} anesthesia_n^1$.

4.1.6 WordNet Mappings

The resources listed above were all built using different versions of WordNet, while we built our semantic network using WordNet 2.1[10]. To do this, the relationships encoded

[9] http://lcl.uniroma1.it/wordnetplusplus

[10] This version was originally chosen in order to be able to test the algorithm on the SemEval-2007 coarse-grained task.

Algorithm 1 Procedure for mapping a synset s to another version of WordNet

```
if $|\text{map}(s)| = 1$ then
  $\{(w, t)\} \leftarrow \text{map}(s)$
  return $t$
else
  $(w, t) \leftarrow \max \text{map}(s)$
  if $|\text{map}(s)| = 2$ then
    if $w \geq 0.66$ then
      return $t$
    else
      $(w_2, t_2) \leftarrow \min \text{map}(s)$
      return $\{t, t_2\}$
    end if
  else {Source synset $s$ maps to more than 2 target synsets.}
    return $t$
  end if
end if
```

in a given resource were transferred onto WordNet 2.1 synsets using automatically generated mappings between WordNet versions from the Technical University of Catalonia (UPC)[11].

Daudé, Padrú, and Rigau (2000) produced these mappings using Relaxation Labelling, an iterative algorithm which computes a mapping of one set of variables (here, the synsets of one WordNet) to another set of a variables (the synsets of the other WordNet) using a set of constraints. The constraints used rely predominantly on the hypernym/hyponym structure of WordNet, although some constraints based on word overlap in synsets and glosses were used for adjectives and adverbs. The procedure produces, for each synset in the source WordNet, a weighted list of synsets in the target WordNet which the source synset maps to; the weights of the target synsets sum to 1.

For generating the SSI semantic network, when translating synsets from one version of WordNet to another, synsets were mapped according to the following protocol:

- if the source synset maps to only one target synset, choose that synset
- if the source synset maps to two target synsets:
  - if one of the target synsets has a weight of 0.66 or higher, choose only that synset
  - otherwise, choose both
- if the source synset maps to more than two target synsets, choose only the target synset with the highest weight

[11] http://www.lsi.upc.edu/~nlp/web/

Algorithm 2 Procedure for setting up SSI-Dijkstra

```
for all $w_i \in$ batch do
  if $|\text{senses}(w_i)| = 1$ then
    $\{w_i^s\} \leftarrow \text{senses}(w_i)$
    $\text{add}(w_i^s, C)$
  else
    $\text{add}(w_i, P)$
  end if
end for
if $|C| > 0$ then
  return $\text{ssi}(P, C)$
else
  $w^* \leftarrow \arg\min_{w_i \in P} |\text{senses}(w_i)|$
  $\text{remove}(w^*, P)$
  $d^* \leftarrow \infty;\quad C_R^* \leftarrow \emptyset$
  for all $w_i^s \in \text{senses}(w^*)$ do
    $C' \leftarrow C;\quad \text{add}(w_i^s, C')$
    $C_R \leftarrow \text{ssi}(P, C')$
    $d_R \leftarrow \sum_{s_1 \in C_R, s_2 \in C_R, s_1 \neq s_2} \text{dist}_G(s_1, s_2)$
    if $d_R < d^*$ then
      $d^* \leftarrow d_R;\quad C_R^* \leftarrow C_R$
    end if
  end for
  return $C_R^*$
end if
```

Pseudo-code for the procedure of mapping a given synset s is given in Algorithm 1; this algorithm calls a function, map, which returns a list of structures that each contain a weight and a target synset.

4.2 The SSI Algorithm

This section presents the SSI algorithm. The algorithm operates using three data structures:

P the *pending list* of words to be disambiguated;

C the *semantic context*, a list of word senses;

G the semantic network as described above.

For a given word w_i we refer to the set of senses $\{w_i^1, w_i^2, \ldots\}$ that the word can take with $\text{senses}(w_i)$. When a word w_i in the pending list P is assigned a sense, it is removed from P and its chosen sense w_i^* added to the semantic context C.

Algorithm 3 The SSI-Dijkstra algorithm

```
while $|P| > 0$ do
  $d^* \leftarrow \infty$;  $w^* \leftarrow$ None;  $w_i^* \leftarrow$ None
  for all $w_i \in P$ do
    for all $w_i^s \in \text{senses}(w_i)$ do
      $d \leftarrow \sum_{c \in C} \text{dist}_G(c, w_i^s)$
      if $d < d^*$ then
        $d^* \leftarrow d$;  $w^* \leftarrow w_i$;  $w_i^* \leftarrow w_i^s$
      end if
    end for
  end for
  if $d^* \neq \infty$ then
    remove($w^*, P$)
    add($w_i^*, C$)
  else
    return $C$
  end if
end while
return $C$
```

The algorithm disambiguates words in batches. For running text, this batch is the current sentence. For disambiguating a gloss in a dictionary, the batch would be the gloss and the head word, and possibly the hypernyms and hyponyms of the head word and their glosses.

The algorithm consists of two stages: a setup stage, shown in Algorithm 2, and an iterative processing stage, shown in Algorithm 3 (this second stage is called ssi in Algorithm 2). The initialization step places the words $\{w_1, w_2, w_3, \ldots\}$ contained in the current batch into the pending list P. Often, one or more words in the batch are monosemous; formally, this means that $|\text{senses}(w_i)| = 1$. These words can be assigned a sense immediately: their assigned senses are placed in C, and the words themselves are removed from P. Thus, initially, C can either be empty or contain one or more known word senses.

Following this initialization, the algorithm proceeds iteratively in a greedy fashion. On each iteration, for each word w_i in P, the algorithm computes the *semantic distance* under the graph G between the known senses in C and the possible senses that w_i can take. For a given w_i, each sense $w_i^s \in \text{senses}(w_i)$ is given a score, which is the sum of these semantic distances:

$$\text{score}(w_i^s) = \sum_{c \in C} \text{dist}_G(c, w_i^s)$$

The algorithm then chooses the word having the sense with the least score, and assigns that sense to that word:

$$\text{choose } w_i^* = \underset{w_i^s \in \text{senses}(w_i),\ w_i \in P}{\arg\min} \text{score}(w_i^s)$$

w_i is then removed from P and w_i^* added to C, and the algorithm continues. Ties are broken by choosing the sense with the lower sense number (i.e., a bias towards the MFS baseline). The process terminates when P is empty or there is no sense that minimizes a score for any word, as can happen, for example, if a word sense is not found in the semantic network (the algorithm computes $\text{score}(w_i^s) = \infty$ for that sense).

It should be noted that sometimes the algorithm begins with an empty semantic context C, and no words in the pending list P are monosemous. In that case, the algorithm finds the word in P with the least number of senses, and runs multiple times, each time assigning a different possible sense to that word during initialization. Each run R terminates with one or more chosen senses in its semantic context C_R. The algorithm then returns the C_R which minimizes its total internal semantic distance:

$$\text{choose } C_R^* = \underset{C_R}{\arg\min} \sum_{c_1 \in C_R} \sum_{c_2 \in C_R,\, c_1 \neq c_2} \text{dist}_G(c_1, c_2)$$

This can be seen intuitively as selecting the set of word senses which are "closest" to each other, in terms of the semantic distance function.

4.2.1 The Semantic Distance Function

Different descriptions of the SSI algorithm define distance under the semantic network in different ways.

The final publication of the method refers to a context-free grammar which is used to describe which paths through the network represent meaningful semantic relations; that is, the grammar limits which strings of edge labels can be used to draw a connection between two word senses. The grammar is also used to weight the paths found, so that the strength of the connection between two given word senses can be estimated. This strength is then the semantic distance. Navigli and Velardi (2005) write that they used about 50 hand-crafted grammar rules in their implementation, but the complete grammar for the SSI algorithm has unfortunately never been published, making reimplementation of this system very difficult.

SSI-Dijkstra (Cuadros and Rigau, 2008), used in this thesis, is a simpler version that uses Dijkstra's algorithm to find the shortest path through the network between any two word senses and defines the semantic distance to be the length of that path. The use of Dijkstra's algorithm makes the implementation both conceptually uncomplicated and computationally very efficient. In contrast to the original SSI algorithm, this always finds a semantic distance between two word senses, assuming that the graph is connected. Also, SSI-Dijkstra works for all parts of speech (some early versions of the original SSI grammar as in (Navigli and Velardi, 2004) only defined paths between nouns).

Chapter 5

Improving SSI-Dijkstra

This chapter describes a set of improvements to the basic SSI-Dijkstra algorithm introduced in Chapter 4. Section 5.1 discusses the tuning of parameters to optimize the algorithm's performance. Section 5.2 presents a novel way of integrating word sense frequency information into the SSI-Dijkstra algorithm.

5.1 Tuning the Implementation

In order to experiment with methods for improving the performance of the basic SSI-Dijkstra algorithm, the Senseval-2 English all-words task (see Section 2.3.1) was used as a development set[1]. Table 5.1 shows the precision, recall and F_1 values for the random baseline, MFS baseline and for the basic SSI-Dijkstra implementation developed in Chapter 4. Results for the best supervised WSD system (SMUaw) and best unsupervised WSD system (UNED-AW) on the Senseval-2 competition are also reported.

The performance of the basic SSI-Dijkstra algorithm, while much better than the random baseline and competitive with the unsupervised UNED-AW system, lies significantly below the MFS baseline, and more than 10% behind the highest scoring supervised system. This section describes various experiments we carried out to vary parameters of the algorithm with the goal of improving the algorithm's performance.

Condition	Precision	Recall	F_1
Random baseline	42.0	42.0	42.0
MFS baseline	62.7	62.2	62.4
SMUaw	69.0	69.0	68.6
UNED-AW	55.6	55.0	55.2
Basic SSI-Dijkstra	55.8	55.3	55.5

TABLE 5.1: Performance of the basic SSI-Dijkstra algorithm on the SENSEVAL-2 English all-words task

Condition	Precision	Recall	F_1
MFS baseline	62.7	62.2	62.4
No carrying	55.8	55.3	55.5
Carrying 1 sentences	55.6	55.1	55.3
Carrying 2 sentences	53.7	53.3	53.5
Carrying 3 sentences	53.6	53.1	53.4
Carrying 4 sentences	52.7	52.3	52.5
Carrying 5 sentences	53.2	52.8	53.0

TABLE 5.2: Results on the SENSEVAL-2 English all-words task: Increasing the disambiguation context size

5.1.1 Expanding the Disambiguation Context

The first optimization to the system exploits the fact that all-words WSD competitions use running text for input. Navigli and Velardi (2005) found that disambiguation accuracy of the SSI algorithm increased with larger context size, as more information is available to use. The context size for the disambiguation algorithm can be easily extended by including in the semantic context C those word senses which were disambiguated in the previous sentence. Table 5.2 shows evaluation of the algorithm, while varying as a parameter the number of previous sentences "carried" in this way. The results do not show a significant difference between the original algorithm and the condition with one previous sentence's word senses added to the disambiguation context; other evaluations in this chapter (Table 5.7) show a small performance increase with the expanded context size. Hereafter, this carry-the-last-sentence optimization is called *carried sentences*.

Carrying two sentences or more is significantly worse than carrying no sentences ($p < 0.05$)[2].

[1] Note that the SENSEVAL-2 task was mapped to the WordNet 2.1 sense inventory for this evaluation; since a small fraction (about 2%) of the tagged words cannot be mapped due to changes in WordNet, the results obtained are slightly distorted by this process. Using the first sense baseline as an indicator, it would seem that the SENSEVAL-2 values given here for the SSI-Dijkstra system are overestimated by about 2% compared to values published during the SENSEVAL-2 conference.

[2] All significance tests for comparing algorithm performance use the paired McNemar test.

Condition	Precision	Recall	F_1
MFS baseline	62.7	62.2	62.4
Greedy search	55.8	55.3	55.5
Exhaustive search	55.4	55.0	55.2

Table 5.3: Results on the Senseval-2 English all-words task: Exhaustive search

5.1.2 Exhaustive Search

The SSI-Dijkstra algorithm is iterative and greedy in the formulation given in Section 4.2; we conducted an experiment to see if an exhaustive search would improve the results of the program. Such a search checks each possible assignment of senses to words, searching by brute force for the configuration which minimizes the semantic distance between all pairs of assigned senses. Due to varying sentence length and varying polysemy of the words in a sentence, this can pose complexity problems, as the number of possible sense assignments increases exponentially with increasing polysemy. To overcome this, we set a ceiling value of 300,000 sense assignments (about 5 minutes of searching on a 2.8GHz 64-bit processor); sentences that have more possible sense assignments than this are processed using the greedy search.

Table 5.3 shows results on the Senseval-2 task. As can be seen, using the exhaustive search seems to slightly reduce performance; the difference on this test is not significant at the $p < 0.05$ level.

5.1.3 Removing Graph Edges

The next set of results come from an ablation study whereby sections of the semantic network were removed to determine whether any parts of the network were hurting performance. The sections removed are:

domains all "domain" and "domain-1" edges from the WordNet Domains project;

glosses all "gloss" and "gloss-1" edges from the XWN project;

inverse all inverse edges, meaning any edge ending with "-1";

KnowNet all "knownet" edges from the KnowNet collocation corpus;

Wikipedia edges from WordNet++ : "wikipedia", "wikipedia-1"

WordNet domains the domain-related edges from Wordnet: "cdomain", "rdomain", "udomain", and their inverses

Condition	Precision	Recall	F_1
MFS baseline	62.7	62.2	62.4
Complete network	55.6	55.1	55.3
No domains	55.5	55.1	55.3
No glosses	53.9	53.1	53.5
No inverse	53.8	53.3	53.5
No KnowNet	52.3	51.9	52.1
No Wikipedia	55.5	55.0	55.2
No WordNet domains	55.1	54.7	54.9
No WordNet	54.8	54.4	54.6

Table 5.4: Results on the Senseval-2 English all-words task: Removing graph edges

Condition	Precision	Recall	F_1
MFS baseline	62.7	62.2	62.4
KnowNet-5	55.8	55.3	55.5
KnowNet-10	56.1	55.6	55.8
KnowNet-15	57.0	56.5	56.8
KnowNet-20	56.9	56.4	56.6

Table 5.5: Results on the Senseval-2 English all-words task: Increasing the size of KnowNet

WordNet all edges from the WordNet lexicon.

Table 5.4 shows results on the development set with various sections of the graph removed. These results were obtained with the carried sentences optimization from Section 5.1.1. As can be seen in the table, removing sections of the network generally resulted in decreased performance. This drop in performance is statistically significant for several sections: no glosses, no inverse, and no WordNet domains ($p < 0.05$) and no KnowNet ($p < 0.001$). This result fits with findings by Cuadros and Rigau (2008) which show that the main problem facing knowledge-based WSD systems is usually coverage, and that more knowledge usually improves performance, even when that knowledge is not of the highest quality.

5.1.4 Increasing the Size of KnowNet

Table 5.5 shows the effect of increasing the size of the KnowNet corpus used in the network construction. As discussed in Section 4.1.4, KnowNet is available in versions of different size, depending on how many context words are provided for each topic signature. KnowNet-5 contains 231,164 total collocations, KnowNet-10 689,610, KnowNet-15 1,378,286, and KnowNet-20 2,358,927. As can be seen from the table, increasing the number of collocations has a tendency to improve the performance of the algorithm;

Condition	Precision	Recall	F_1
MFS baseline	62.7	62.2	62.4
No edge weighting	55.6	55.1	55.3
Edge weighting $-\log P(n)$	48.8	48.4	48.6
Edge weighting $P^{-1}(n)$	57.1	56.7	56.9

TABLE 5.6: Results on the SENSEVAL-2 English all-words task: Various functions of word frequency information

here, the best performance is attained with KnowNet-15. None of these changes are statistically significant at the $p < 0.05$ level. The pattern seen here, that KnowNet-10 outperforms KnowNet-5, and that KnowNet-15 outperforms KnowNet-20, is also observed in a later experiment in this Chapter (Table 5.7).

5.2 Incorporating Word Frequency Information into the Semantic Distance Function

As shown in Table 5.1, SSI-Dijkstra is outperformed by the MFS baseline, which uses only word sense frequency information. We decided to test the effect of integrating this information into the SSI-Dijkstra algorithm.

The formulation given in Section 4.2.1 of the semantic distance metric used in the algorithm assumes that the edges in the graph structure are unweighted. Thus, it is the number of edges in the shortest path separating two word senses which gives the distance between them. Here, we introduce a scheme whereby graph edges are weighted and conduct an experiment to observe the effect of this change on the algorithm's performance.

Word sense frequency information is collected on the SemCor corpus (refer to Section 2.1.1) with simple Good-Turing smoothing (Gale and Sampson, 1995). These counts give a prior probability distribution over WordNet synsets. Thus, a very common word like *to be*, recorded 13,626 times in SemCor, has a probability of 0.045, whereas a less common word such as *beanbag*, which is not recorded in SemCor, has a probability of 4.9×10^{-7}.

The synset probabilities are used to weight the edges in the graph, such that every edge in the graph ending at a node n has a weight which is a function of the probability $P(n)$ of the node. Since larger edge weights reflect greater semantic distance, it is desirable that edge weights leading to more frequent words should have smaller values. As an initial attempt to achieve this property, we examined edge lengths given by the negative logarithm of the probability, $-\log P(n)$ (called *self-information* in information theory), and the inverse of the probability, $P^{-1}(n)$.

Condition	KnowNet version			
	5	10	15	20
No carried sentences	57.1	58.0	57.8	58.5
Carried sentences	56.9	58.3	58.9	58.1
No carried sentences + Exhaustive search	56.8	57.6	57.2	58.0
Carried sentences + Exhaustive search	57.0	58.1	58.5	57.8

TABLE 5.7: F-scores on the SENSEVAL-2 English all-words task: $P^{-1}(n)$ edge weighting combined with various parameters

Table 5.6 shows the results of testing SSI-Dijkstra on the SENSEVAL-2 all-words task using KnowNet-5 and the "carried sentences" optimization. The experiment tests the effectiveness of using the two functions of synset probabilities for weighting the graph edges. Using the inverse probability function improves both precision and recall compared to the performance without edge weighting. Statistically, the $-\log P(n)$ weighting is significantly worse than no edge weighting ($p < 0.001$), but the improvement seen with the $P^{-1}(n)$ weighting is only significant at the $p < 0.1$ level.

Table 5.7 shows F-scores for a final experiment using the SENSEVAL-2 test which attempts to optimize over all parameters which seemed promising in this section. In this experiment, edge weights are set to the inverse probability $P^{-1}(n)$, and the parameters for KnowNet version, the "carried sentences" optimization and exhaustive search are permuted.

Increasing the KnowNet version from 5 to 10 improves performance independent of other parameters although the only case where this is statistically significant is for the "carried sentences" case ($p < 0.05$). The differences between using KnowNet-10 and 15, and between 15 and 20 are not great. The "carried sentences" optimization here seems to have a slight positive effect overall, while the "exhaustive search" seems to have a slight negative effect overall. Most of the scores given in the table are not significantly different from each other.

The best performance on the SENSEVAL-2 task is 58.9%, achieved with the "carried sentences" optimization using KnowNet-15 and the greedy search algorithm; this is significantly better ($p < 0.01$) than the worst configuration (no "carried sentences", exhaustive search, KnowNet-5). In the optimal configuration, the SSI algorithm achieves an F_1 measure which is only slightly below the MFS baseline and which significantly outperforms the unsupervised UNED-AW algorithm (cf. Table 5.1).

Chapter 6

Subcategorization for WSD

In this chapter, we develop a statistical model that allows predicting verb sense from subcategorization, and integrate it into the SSI-Dijkstra algorithm developed in Chapters 4 and 5 in an effort to boost WSD performance on verbs.

There are two motivations for this. Firstly, subcategorization has not yet become a widely-used knowledge source for WSD, and there are relatively few studies in the literature which examine the marginal effect of analysing the syntactic behaviour of verbs on sense disambiguation performance. As mentioned in Chapter 3, knowledge-based WSD algorithms such as SSI-Dijkstra employ lexical semantic knowledge only, and so should provide an ideal theatre for investigating the effect of adding syntactic information in the form of subcategorization analysis of verbs. Secondly, the SSI family of algorithms is both designed for, and performs best on disambiguation of nouns; it performs worst on verbs (Navigli and Velardi, 2005). Verbs are the part of speech which might be expected to make the most use of syntactic knowledge.

Automatic WSD struggles with verbs in general. Evaluations show that verbs are the hardest words to tag for sense: on Senseval-1, Kilgarriff and Rosenzweig (2000) noted that WSD systems achieve lower precision on verbs (10% lower than nouns, the "easiest" part of speech). While they found nouns to have higher polysemy than verbs, verbs exhibited higher entropy in their sense distributions than other parts of speech (i.e., the multiple senses of verbs are more evenly distributed than the senses of nouns), and entropy correlated more strongly with system performance than polysemy. Even the MFS baseline reflects this trend, and tends to perform best on nouns and worst on verbs. Snyder and Palmer (2004) note that verbs have the lowest inter-annotator agreement when setting up gold standards for traditional evaluations. Chen and Palmer (2005) point out that verb performance by the best system on the Senseval-2 English lexical sample task was 56.6% accuracy, compared to 64.2% on other parts of speech.

By incorporating syntactic features into a supervised WSD system, they were able to achieve 64.6% accuracy on verbs on the test data, showing that it is possible to attain better WSD performance on verbs.

This chapter is organized as follows. Section 6.1 introduces the VerbNet resource, a database encoding syntactic and semantic information about English verbs. Section 6.2 gives an overview of verb subcategorization and discusses the link between subcategorization and verb sense. Section 6.3 describes our joint model of verb sense and subcategorization preference; Section 6.4 shows how this model is integrated into SSI-Dijkstra. Section 6.5 presents evaluation of the SSI-Dijkstra algorithm on the evaluation metrics surveyed in Section 2.3 and discusses results.

6.1 VerbNet

VerbNet (Kipper-Schuler, 2005) is a hierarchical database of English verbs that contains syntactic and semantic information. It is based on Levin's classification of verbs according to alternation behaviour (Levin, 1993), although it has been extended several times with new verb classes (Korhonen and Briscoe, 2004; Kipper et al., 2006). Recently, the SemLink project (Yi et al., 2007) has tagged members of verb classes with WordNet verb senses.

The database contains the following types of information for each verb class (examples reference the HELP class (*72*)):

- Member lexical entries (e.g., *support*, *succor*, *aid*, *abet*, *assist*, *help*).
- Thematic roles with selectional restrictions (e.g., an *Agent* that is either *animate* or an *organization*); a *Beneficiary*, likewise either *animate* or an *organization*; and, a *Theme*).
- Frames (e.g., NP V NP (*"I helped him"*)). The frames label their constituents with thematic roles, and also provide a predicate logic representation.
- Possible subclasses, which inherit from the top-level class but further specify members, thematic roles or syntactic frames. For instance, the word *help* actually belongs to the subclass *72.1*; this subclass licenses the frame NP V PP (*"I helped with the homework"*), which is not licensed for the top-level class (containing *support*).

The latest version of VerbNet (version 3.1) has 270 top-level classes and 200 subclasses; it lists 6,054 different word senses belonging 4,437 different WordNet synsets.

6.2 Verb Subcategorization

Subcategorization Frames (SCFs) are a description of the number and types of arguments taken by a verb, similar to the information encoded by the Frames listed in VerbNet entries. For example, consider the following sentences:

1. I put [the book]$_{NP}$ [on the shelf]$_{PP}$.
2. * I put [the book]$_{NP}$.
3. * I put [on the shelf]$_{PP}$.
4. * I put.

Here, it is clear that the verb *put* can take a noun phrase (NP) and prepositional phrase (PP) as complements, but it is ungrammatical if only one of these is present. The intransitive use of *put* is also not allowed. In fact, verbs in the VerbNet class PUT (*9.1*) all require SCFs like NP-PP (if it can be inferred from the context, the location can be adverbial like *here/there*, or even omitted for verbs such as *stash*). In general, however, because of diathesis alternations, a verb taking one SCF will often take another related one; so that SCFs tend to occur in "families". Models of verb subcategorization preference encode these types of rules about which constituents may or may not appear in a verb phrase; such models are used in lexicalized parsers (e.g., (Collins, 2003)).

Subcategorization is a syntactic phenomenon with implications for semantics as well. Levin's (1993) widely-used verb classification, the basis for VerbNet, is grounded in the hypothesis that a verb's syntactic behaviour and its meaning are strongly connected. Dorr and Jones (1996) show that verbal syntactic features are predicted by verb sense but not by verb lemma, and that the strength of the relationship shows support for Levin's theory. Schulte im Walde and Brew (2002) are able to induce plausible semantic groupings of German verbs by clustering on subcategorization preferences estimated by a statistical parser. Similarly, Stevenson and Merlo (1999) find that syntactic features are good predictors of Levin verb class for English verbs.

Roland and Jurafsky (1998) note that there are significant differences in subcategorization distribution between different corpora, and identify two factors which influence verb subcategorization: *discourse* factors (e.g., the design and type of a corpus), and verb sense (which they term *semantic* factors). Controlling for identified discourse factors indicates that these are responsible for only a fraction of the variation seen in subcategorization between corpora. Further, they show that different verb senses have different subcategorization preferences (for example, the *attack* and *bill* senses of the word *charge* have different SCF probabilities), and that different corpora have different distributions of verb senses.

Roland et al. (2000) continued in this vein, and investigated monosemous verbs from different corpora (Brown, Wall Street Journal, British National Corpus) covering different domains and sub-languages. Instances of these verbs which use non-dominant senses were filtered out. The verb instances were then classified for transitivity by hand; this was used as a simple subcategorization feature. The authors found that 55 of the 64 verbs studied had the same transitivity preferences across all three corpora; they suggested that the remainder of the SCF variation they observed was due to fine-grained variation in verb sense between the corpora. This finding shows that SCF preferences seem to be stable across corpora and between British and American English, when controlled for verb sense. Roland and Jurafsky (2002) conclude that verb sense is the single best predictor for verb subcategorization. While each verb sense does not necessarily have a different SCF, and verb sense is not the only factor which influences subcategorization, the dependency is strong enough to allow predicting syntax from semantics and vice-versa; the authors note the possibility that this relationship could prove useful for WSD.

The link between subcategorization and verb sense has already been exploited for WSD applications. Bikel (2000) trained a probabilistic lexicalized parser with a lexical model that encodes word sense; while the addition of word sense to the parsing model was not found to improve parsing performance on that study, the parser was able to sense disambiguate words, and its WSD performance figures seemed reasonably good. Using a supervised classifier, Chen and Palmer (2005) were able to improve WSD performance on verbs by incorporating syntactic features, including detailed analysis of some subcategorization phenomena.

Andrew et al. (2004) create a joint model of word sense and subcategorization preference using Expectation Maximization to combine a bag of words WSD model with an unlexicalized PCFG parser. This model is trained on the SENSEVAL-2 lexical sample task data (WSD) and the Penn Treebank (SCF). Their joint model delivers a modest performance improvement for both sense disambiguation (accuracy improves from 54.0% to 55.9%) and parsing (as operationalized by identifying SCF, accuracy improves from 59.3% to 61.4%). They find that the joint model helps WSD for some verbs (*begin*, *drive*, *find*, *keep*, *leave*, *work*), but hurts others very slightly. They note the achieved performance increases to be quite small, and speculate that the bag-of-words WSD model is already able to capture much of the SCF information (in particular due to a relative positional weighting technique which they use).

WSD has also been used to improve automatic subcategorization acquisition. Korhonen and Preiss (2003) uses the output of a probabilistic WSD system to significantly improve the performance of a system for automatically acquiring verb subcategorization frames. For subcategorization acquisition, WSD benefits most those verbs which are highly polysemous, and also verbs whose senses differ strongly in terms of subcategorization. This technique was so effective that it was developed as an *in vivo* method for WSD evaluation (Preiss et al., 2002) and ultimately accepted as a task for Senseval-3. Although no teams participated in the task, Preiss and Korhonen (2004) analysed the performance of their own system, and found a very high ($\rho = 0.97$) correlation between gold-standard WSD performance and the amount of improvement seen on the subcategorization acquisition task.

6.3 The Subcategorization Model

We develop our model of subcategorization frame preferences from SemCor (cf. Section 2.1.1). SemCor contains word sense information, but no parse trees. It is true that part of the Brown corpus is available in parsed form in the Penn Treebank; however, this material overlaps with less than half of SemCor[1], which we considered unacceptable, given the problems with data sparseness that SemCor's small size already creates. Therefore, we parsed SemCor using version 1.6.5 of the Stanford Parser (Klein and Manning, 2003)[2], an unlexicalized PCFG parser. The parser was trained on sections 1–21 of the Wall Street Journal corpus, the Genia corpus, parts of the English components of the Chinese Translation Treebank and Arabic Translation Treebanks, as well a small amount of hand-parsed data created at Stanford. Parsing SemCor gives a corpus containing both verb sense information and parse trees.

Using a statistical parser necessarily introduces some noise, which is due to parsing errors and the shallow representations used in Penn Treebank syntactic analyses. For instance, temporal expressions as in *"She saw him yesterday"* are included in the verb phrase as NPs, so that this sentence will be analysed as *see* NP NP (ditransitive); the bracketing notation is not informative enough to allow the temporal expression to be categorized as an adjunct. Nevertheless, this kind of error seems to balance out over SemCor, and is not obviously a significant source of errors.

We use a subset of the subcategorization frames given in (Andrew et al., 2004), which are in turn based on `tgrep`[3] search strings defined in (Roland, 2001). Like Andrew et al.

[1] There are 778,587 words in SemCor, and 331,895 words which overlap between the Penn Treebank-3 and SemCor (just under 43%).

[2] http://nlp.stanford.edu/software/lex-parser.shtml

[3] `tgrep` is a utility for searching treebanks and is distributed with the Penn Treebank.

	VPto	∅	Other	PP	S for to	NP	NP PP	VPing
$appear_v^1$	62	12	12	9	5	3	1	1
$appear_v^2$	0	19	5	54	1	3	1	1

Table 6.1: Counts of verb sense-SCF pairs in SemCor

(2004), we undo passivization[4], but, in contrast to their work, we do not analyse verb particles as arguments, since phrasal verbs are already tagged as multi-word expressions in SemCor. There are 11 SCF types plus a catch-all category called "other", for a total of 12. A list of the subcategorization frames used in this study is given in Appendix B. A verb instance in a parse tree can be categorized into one of these frames by finding the first `tgrep` string that matches.

In this way, 81,461 verb instances in SemCor could be classified for SCF[5], giving counts which allow the estimation of a joint probability model over verb sense and subcategorization. Table 6.1 shows the counts obtained for two senses of the verb *appear*: sense 1 (to "give a certain impression or have a certain outward aspect") selects strongly for *VPto*, whereas sense 2 (to "come into sight or view") instead selects for *PP*.

These counts can be used directly to give a joint model of verb sense and subcategorization. The counts can also be "backed off" to related distributions in two ways: by summing over all possible senses of a verb lemma, we arrive at a distribution relating lemma and SCF; and, by summing over all verb senses belonging to a VerbNet class, we get a distribution relating VerbNet class and SCF. To mitigate problems caused by sparse data, we will interpolate the joint model given by the direct SemCor counts with the joint models for lemma/SCF and VerbNet class/SCF; this is inspired by the hypothesis in (Korhonen, 2002) that verbs in the same VerbNet class will have similar subcategorization preferences.

In the following discussion, we write the number of instances in SemCor tagged with verb sense v_s and subcategorization frame f as $C(v_s, f)$; these counts are smoothed with Good-Turing estimation (Gale and Sampson, 1995). L is a function which takes a verb sense v_s and returns its corresponding lemma l. K is a function which takes a verb sense v_s and returns a set, possibly empty, of VerbNet classes k that list v_s as a member. In this chapter, we use only top-level VerbNet classes. The total number of verb instances is $N = \sum_{v_s} \sum_f C(v_s, f)$.

[4] This seems to be a valid simplification; for example, Stevenson and Merlo (1999) found that the active/passive distinction was not effective for predicting verb class from syntactic features.

[5] There are 88,813 tagged verb instances in SemCor (representing 9,232 verb senses); 243 out of 36,933 sentences were too long to be parsed, and 3,186 verb instances were skipped because they were inside a noun phrase. The 81,461 parsed instances together represent 8,617 verb senses (4,365 verb lemmas).

Now, the Maximum Likelihood Estimate (MLE) for the joint probability of verb sense and subcategorization is given by:

$$P_{MLE}(v_s, f) = \frac{1}{N} C(v_s, f)$$

The probability of a verb sense is, similarly:

$$P_{MLE}(v_s) = \frac{1}{N} \sum_f C(v_s, f)$$

This gives the conditional probability of a SCF f given a verb sense v_s:

$$P_{MLE}(f|v_s) = \frac{P(v_s, f)}{P(v_s)} = \frac{1}{N} C(v_s, f) \frac{N}{\sum_{f'} C(v_s, f')} = \frac{C(v_s, f)}{\sum_{f'} C(v_s, f')}$$

The "back-off" model for lemma and SCF is defined as:

$$P_{MLE}(l, f) = \frac{1}{N} \sum_{\{v_s | L(v_s) = l\}} C(v_s, f)$$

The conditional probability of a SCF under the lemma back-off model is:

$$P_L(f|l) = \frac{\sum_{\{v_s | L(v_s) = l\}} C(v_s, f)}{\sum_{f'} \sum_{\{v_s | L(v_s) = l\}} C(v_s, f')}$$

The "back-off" model for verb class and SCF is slightly different; note that, while every verb sense has a lemma, not every verb sense belongs to a VerbNet class; in the SemCor data, only 56.2% of verb instances could be assigned to a VerbNet class. Furthermore, some verb senses are included in more than one VerbNet class. For example, $moan_v^1$ ("indicate pain, discomfort, or displeasure") belongs to the VerbNet classes COMPLAIN (*37.8*), MANNER OF SPEAKING (*37.3*), NONVERBAL EXPRESSION (*40.2*), SOUND EMISSION (*43.2*), and ANIMAL SOUNDS (*38*). Thus, we define counts[6] by VerbNet class k:

$$C_k(k, f) = \sum_{\{v_s | k \in K(v_s)\}} C(v_s, f)$$

Then, the conditional probability of a SCF under the VerbNet class back-off model is:

[6] These counts are also smoothed with Good-Turing estimation.

$$P_V(f|k) = \frac{C_k(k,f)}{\sum_{f'} C_k(k,f')}$$

To find the conditional probability of a SCF given a verb sense using this model, we average the distributions for all VerbNet classes the sense belongs to:

$$P_V(f|v_s) = \frac{1}{|K(v_s)|} \sum_{\{k \in K(v_s)\}} P_V(f|k)$$

Now we combine the verb sense/SCF model with the two back-off models using linear interpolation. The interpolation is performed by preference between the verb sense/SCF model and the VerbNet class/SCF model; only if a given verb sense is not found in VerbNet do we use the verb lemma/SCF model. This preference for VerbNet reflects the fact that the VerbNet class/SCF model has high precision but relatively low coverage, whereas the lemma/SCF model has complimentary properties. The resulting conditional model is:

$$P^*(f|v_s) = \begin{cases} \alpha P_{MLE}(f|v_s) + (1-\alpha)P_V(f|v_s) & \text{if } v_s \text{ is in VerbNet} \\ \beta P_{MLE}(f|v_s) + (1-\beta)P_L(f|l) & \text{otherwise} \end{cases}$$

The interpolation parameters used were $\alpha = 0.5$, $\beta = 0.55$; these values were estimated by optimization on SemCor using 10-fold cross-validation.

Finally, we use this combined model to get the conditional probability of a verb sense given a lemma and subcategorization frame[7]:

$$P(v_s|l,f) = \frac{P(v_s,l,f)}{P(l,f)} = \frac{P(v_s,f)}{P(l,f)} = \frac{P^*(f|v_s)}{P_{MLE}(l,f)} P_{MLE}(v_s)$$

This model can be used by itself to perform verb sense disambiguation on parsed text. To do this, the combination of lemmatized verb and subcategorization frame are looked up in the model to give a probability distribution over the verb's possible senses; the most probable sense is then chosen. As with the SSI-Dijkstra algorithm, ties are broken by choosing the verb sense with the lowest sense number. On SemCor verbs, this gives an accuracy of 60.8% (again evaluated with 10-fold cross-validation), compared to a random baseline of 19.7% and a MFS baseline of 61.1%.

[7] Note that a verb sense completely determines a verb lemma, and so $P(v_s, l) = P(v_s)$.

System	All	Noun	Verb	Adj	Adv
Random baseline	42.0	45.6	21.9	45.8	60.1
MFS baseline	60.1	71.2	39.0	61.1	75.4
SMUaw	68.6	78.0	52.9	69.9	81.7
UNED-AW	55.2	60.0	38.5	60.2	74.7
Mih05	54.2	36.7	11.6	20.8	14.9
Sinha07	57.6	66.2	34.1	61.8	60.4
Tsatsa07	49.3	—	—	—	—
Agi09	58.6	70.4	38.9	58.3	70.1
SCF Model only	14.0	0.0	40.3	0.0	0.0
SSI-Dijkstra	54.4	60.4	38.6	60.0	68.1
SSI-Dijkstra + edge weighting	58.1	68.1	37.5	63.3	67.7
SSI-Dijkstra + edge weighting + SCF	58.5	68.3	39.2	63.1	67.7

Table 6.2: F-score results on the Senseval-2 English all-words task

6.4 Integrating the Model into SSI-Dijkstra

Using the notion of semantic distance introduced in Section 4.2.1, we can integrate the subcategorization model into the SSI-Dijkstra algorithm in a similar way to the word sense probability model presented in Section 5.2. In our first modification to the basic SSI-Dijkstra algorithm, edges in the semantic graph are given weights derived from a probability distribution over word senses, so that the edges ending at a node n have a weight of $\frac{1}{P(n)}$.

With the subcategorization model, we modify the weights for edges leading to nodes that represent verb senses in the current disambiguation context, so that the edges in the graph ending at a verb node n will now have a weight of $\frac{1}{P(n|l,f)}$; as above, l represents the lemma of the verb, and f its subcategorization frame. This value is found using the posterior probability model defined above. Conceptually, this re-weighting of edges is equivalent to taking nodes that represent verb senses and splitting them into several nodes that represent combinations of word sense and subcategorization frame.

6.5 Evaluation

In this section, we evaluate SSI-Dijkstra on several commonly used English all-words tasks: the Senseval-2 (Section 2.3.1) and Senseval-3 all-words tasks (Section 2.3.2), and the SemEval-2007 coarse-grained all-words task (Section 2.3.3). For these tests, we use KnowNet-10, the "carried sentences" optimization (see Section 5.1.1), and exhaustive

System	All	Noun	Verb	Adj	Adv
Random baseline	34.4	40.6	20.3	45.2	100.0
MFS baseline	61.4	69.4	52.5	65.9	100.0
GAMBL	65.2	70.8	59.3	65.3	100.0
IRST-DDD-00	58.3	62.6	50.3	67.3	100.0
Mih05	52.2	—	—	—	—
Sinha07	53.6	60.8	42.8	54.5	100.0
Nav07	—	61.9	36.1	62.8	—
Agi09	57.4	64.1	46.9	62.6	92.9
SCF Model only	25.3	0.0	49.1	0.0	0.0
SSI-Dijkstra	50.7	55.3	42.3	60.0	100.0
SSI-Dijkstra + edge weighting	53.7	60.2	42.2	66.2	100.0
SSI-Dijkstra + edge weighting + SCF	53.1	60.2	40.7	66.2	100.0

TABLE 6.3: F-score results on the SENSEVAL-3 English all-words task

System	All	Noun	Verb	Adj	Adv
Random baseline	61.3	62.0	52.8	68.5	69.1
MFS baseline	78.9	77.4	75.3	84.3	87.5
UOFR-SSI	83.2	84.1	78.3	85.4	88.5
SUSSX-FR	60.4	68.1	51.0	57.4	49.4
Pon10	81.7	85.5	—	—	—
SCF Model only	29.0	0.0	71.3	0.0	0.0
SSI-Dijkstra	75.9	75.7	70.9	81.2	81.4
SSI-Dijkstra + edge weighting	76.2	76.0	70.7	84.0	79.0
SSI-Dijkstra + edge weighting + SCF	76.3	75.6	71.2	84.0	80.5

TABLE 6.4: F-score results on SemEval-2007 Task 7

System	BNC	Sports	Finance
Random baseline	19.7	19.2	19.5
MFS baseline	37.5	20.7	35.4
Agi09	43.8	35.6	46.9
Pon10	—	42.0	47.8
SSI-Dijkstra	34.8	32.6	40.4
SSI-Dijkstra + edge weighting	39.9	35.8	51.5
SSI-Dijkstra + edge weighting + SCF	40.0	35.9	51.5

TABLE 6.5: F-score results on the WSD-Domain data set

search (Section 5.1.2). The two Senseval tasks require systems to do their own POS-tagging and lemmatization; we used the Stanford POS tagger[8] (Toutanova et al., 2003) for POS-tagging, and the `morpha` tool[9] (Minnen et al., 2001) from the RASP toolkit for lemmatization.

As noted above, the SemEval-2007 coarse-grained task uses a sense inventory based on WordNet (version 2.1) where senses have been clustered to coarser sense classes. For this task, the SSI-Dijkstra algorithm is augmented with knowledge of the sense clustering; formally, we define a function $C(w_s)$ which gives the set of senses that are in the same sense cluster as a word sense w_s. When computing the semantic distance between two word senses w_1 and w_2, the algorithm examines all pairs between the two resulting sense clusters, and finds the minimum distance over these pairs:

$$\text{clustered dist}_G^C(w_1, w_2) = \min_{(w_1', w_2') \in C(w_1) \times C(w_2)} \text{dist}_G(w_1', w_2')$$

Performance by part of speech on Senseval-2, Senseval-3 and SemEval-2007 are shown as F-scores in Tables 6.2, 6.3 and 6.4, respectively. These tables show the random[10] and MFS baselines, and also the results of the best supervised system and unsupervised system at the time of the competition; for example, the best supervised system on Senseval-2 was SMUaw, and the best unsupervised system on that competition was UNED-AW. We also evaluate our algorithm on the Domain-WSD data set (Section 2.3.4); F-scores for the three domains are given in Table 6.5.

For comparison, the tables also show results from recent graph-based WSD methods: Tables 6.2 and 6.3 list results for Mih05 (Mihalcea, 2005), Sinha07 (Sinha and Mihalcea, 2007), Nav07 (Navigli and Lapata, 2007), Tsatsa07 (Tsatsaronis et al., 2007), and the current best graph-based WSD algorithm, Agirre and Soroa's (2009) word-to-word Personalized PageRank (Agi09). Tables 6.4 and 6.5 show results for Agirre et al.'s (2009) Personalized PageRank (Agi09), and Ponzetto and Navigli's (2010) Degree-WordNet++ (Pon10).

The basic SSI-Dijkstra algorithm (without edge weights) performs better than the random baseline, and has good coverage. Its performance on the Domain-WSD data set shows that these qualities are not specific to balanced corpora. Adding the edge weighting scheme results in better disambiguation for nouns, adjectives, and overall score (statistically significant at at least the $p < 0.05$ level on Senseval-2, Senseval-3, and the

[8] http://nlp.stanford.edu/software/tagger.shtml

[9] http://www.informatics.susx.ac.uk/research/groups/nlp/carroll/morph.html

[10] On the Senseval-2 and Senseval-3 tasks, systems are required to POS-tag and lemmatize words by themselves; the random baseline figures given here assume an oracle that always knows the correct POS tag and lemma, and thus have a small advantage over participating systems.

Document	Precision	Recall
d001	76.9	74.1
d002	78.6	74.0
d003	64.8	61.7
d004	75.2	69.0
d005	75.5	71.4

Table 6.6: Performance of the SCF model by document on SemEval-2007 Task 7 verbs

Domain-WSD data set; on SemEval-2007, however, only performance on adjectives seems to be improved). The edge weighting seems to have a slight negative effect on verb disambiguation in the Senseval-2 task, but this is not statistically significant. With edge weighting, the system is often close to the first sense baseline, and on the Domain-WSD data set, SSI-Dijkstra significantly outperforms the first sense baseline. This demonstrates the weakness of the MFS baseline mentioned in Section 2.2.4.1—namely, that it is based on word sense counts from SemCor, a balanced corpus, and that these are not readily applicable to texts in other domains. Although SSI-Dijkstra with edge weighting also makes use of counts from SemCor, it incorporates semantic relations drawn from a wide variety of sources, which we think makes it robust across different domains.

We find the system with edge weighting to be effective, considering the simplicity and ease of implementation of the SSI-Dijkstra algorithm; performance is slightly worse than the best current graph-based methods. On the Senseval-2 competition, the overall F-score of 58.1% beats the best unsupervised system in competition and would have put SSI-Dijkstra fourth overall (out of 22 systems). On Senseval-3, our result places us third among nine unsupervised systems in competition, and fifteenth overall among 26 systems. Performance on the SemEval-2007 task is also respectable, and would have been the second best unsupervised result in competition, placing SSI-Dijkstra seventh out of 15 systems overall.

The SCF model by itself is evaluated on disambiguating verbs in these data sets, where applicable. The SCF model outperforms the basic SSI-Dijkstra method on verbs, although not by a very large margin; its performance is not significantly below the MFS baseline on any test, which is perhaps not surprising, given that both the SCF model and the MFS baseline are based on statistics derived from SemCor. On verbs, the SCF model is competitive with state of the art unsupervised WSD algorithms, and is better than graph-based WSD algorithms; bearing in mind how easy it is to estimate such a model, we hope that subcategorization will be more widely used in the future to access verb sense using syntactic features.

The similarity in performance between the SCF model and the MFS baseline raises the question of whether the SCF model performs differently when tested on domain-specific text. The Domain-WSD data set is unfortunately not useful for measuring this, since it only includes noun instances, which the SCF model is not able to disambiguate. Fortunately, two of the documents on the SemEval-2007 coarse-grained task can be considered domain-specific: d004 is taken from the Wikipedia article on computer programming, while d005 is an excerpt from a book containing biographies of Italian painters. The majority of systems competing on the SemEval-2007 coarse-grained task, including the MFS baseline, scored around 10% worse on d004 than on the other documents; performance was also slightly lower on d005. Table 6.6 shows the performance of the SCF model on verbs by document in the coarse-grained task. Precision and recall on d004 and d005 are in line with the results on the other documents, lending support to Roland and Jurafsky's (2002) thesis that subcategorization preferences conditioned on verb sense are stable across domains.

Integrating the SCF model with edge-weighted SSI-Dijkstra improves results on verb disambiguation to the levels observed for the SCF model by itself. The improvements to verb disambiguation and overall score, however, are not statistically significant. Moreover, on the SENSEVAL-3 task, the addition of the SCF model results in a small drop in verb performance ($p < 0.05$). Despite having conducted error analysis, we are unable to offer an explanation for why the subcategorization model hurts the WSD algorithm in this case.

We think the main reason why adding subcategorization does not significantly improve the performance of SSI-Dijkstra on verb disambiguation is that the SCF model achieves results so close to SSI-Dijkstra. If the SCF model could attain higher scores than the first sense baseline, we expect that the SSI-Dijkstra results on verb senses would be enhanced to a greater degree. It is possible that the subcategorization frames we use here are not specialized enough to capture the subtle distinctions needed for disambiguation to the fine-grained senses listed in WordNet; for instance, we do not analyse adjectival or adverbial adjuncts here, but these might prove useful for some sense distinctions. Future research might also examine the effect of analysing PP adjuncts according to their preposition; this extra level of detail should help distinguish *"the workers are striking for higher wages"* from *"the earthquake struck at midnight"*. Finally, it may be worth investigating other avenues for integrating a subcategorization model into an existing WSD system; for instance, the SSI-Dijkstra algorithm could be altered to output a sense ranking, which could then be combined with syntactic features such as subcategorization in a separate supervised classifier.

Chapter 7

Conclusion

This Chapter summarizes the contributions made by this thesis.

Chapter 3 presented an experiment in mapping WordNet with syntactic types from the English Resource Grammar; this principle could be developed for easily extending the coverage of the ERG lexicon.

In Chapters 4 and 5 we have implemented a simple wide-coverage WSD system, SSI-Dijkstra; with the novel integration of word sense frequencies, we have improved performance to levels competitive with some recent graph-based WSD algorithms. Chapter 6 presented evaluation of our WSD system on a number of commonly used disambiguation tasks, allowing, for the first time, direct comparison of this algorithm to recently published conceptually similar graph-based methods.

Chapter 6 also presented a simple method for estimating a joint probability distribution on verb sense and subcategorization; we have shown that this model is capable of sense disambiguating verbs at a level comparable to the first sense baseline, or to state of the art unsupervised WSD algorithms. Modelling subcategorization in this way is convenient for rapidly accessing statistical information about verb behaviour, and should be useful for other applications, such as learning selectional preferences, or semantic role labelling. Unfortunately, this thesis did not observe a significant improvement to WSD performance as a result of adding the subcategorization model to the SSI-Dijkstra algorithm. We are hopeful that future improvements to the subcategorization model will show a greater effect on the WSD task.

Appendix A

Mapping of WordNet Domains to WordNet Synsets

Domain	WordNet Sense	Gloss
ACOUSTICS	$acoustics_n^1$	The study of the physical properties of sound.
ADMINISTRATION	$administration_n^1$	A method of tending to (especially business) matters.
AGRICULTURE	$agriculture_n^2$	The practice of cultivating the land or raising stock.
ANATOMY	$anatomy_n^1$	The branch of morphology that deals with the structure of animals.
ANIMAL HUSBANDRY	$animal\_husbandry_n^1$	Breeding and caring for farm animals.
ANIMALS	$animal_n^1$	A living organism characterized by voluntary movement.
ANTHROPOLOGY	$anthropology_n^1$	The social science that studies the origins and social relationships of human beings.
APPLIED SCIENCE	$applied\_science_n^1$	The discipline dealing with the art or science of applying scientific knowledge to practical problems.
ARCHAEOLOGY	$archaeology_n^1$	The branch of anthropology that studies prehistoric people and their cultures.
ARCHERY	$archery_n^1$	The sport of shooting arrows with a bow.
ARCHITECTURE	$architecture_n^2$	The discipline dealing with the principles of design and construction and ornamentation of fine buildings.

Domain	WordNet Sense	Gloss
Art	art^2_n	The creation of beautiful or significant things.
Artisanship	$craftsmanship^1_n$	Skill in an occupation or trade.
Astrology	$astrology^1_n$	A pseudoscience claiming divination by the positions of the planets and sun and moon.
Astronautics	$astronautics^1_n$	The theory and practice of navigation through air or space.
Astronomy	$astronomy^1_n$	The branch of physics that studies celestial bodies and the universe as a whole.
Athletics	$athletics^1_n$	An active diversion requiring physical exertion and competition.
Atomic Physic	$atomic\_physics^1_n$	The branch of physics that studies the internal structure of atomic nuclei.
Aviation	$aviation^3_n$	The art of operating aircraft.
Badminton	$badminton^1_n$	A game played on a court with light long-handled rackets used to volley a shuttlecock over a net.
Banking	$banking^1_n$	Engaging in the business of keeping money for savings and checking accounts or for exchange or for issuing loans and credit etc..
Baseball	$baseball^1_n$	A ball game played with a bat and ball between two teams of nine players.
Basketball	$basketball^1_n$	A game played on a court by two opposing teams of 5 players.
Betting	bet^2_v	Stake on the outcome of an issue.
Biochemistry	$biochemistry^1_n$	The organic chemistry of compounds and processes occuring in organisms.
Biology	$biology^1_n$	The science that studies living organisms.
Body Care	$care^1_n$	The work of providing treatment for or attending to someone or something.
Book Keeping	$bookkeeping^1_n$	The activity of recording business transactions.
Bowling	$bowling^1_n$	A game in which balls are rolled at an object or group of objects with the aim of knocking them over or moving them.

Domain	WordNet Sense	Gloss
Boxing	$boxing^1_n$	Fighting with the fists.
Buildings	$building^1_n$	A structure that has a roof and walls and stands more or less permanently in one place.
Card	$card\_game^1_n$	A game played with playing cards.
Chemistry	$chemistry^1_n$	The science of matter.
Chess	$chess^2_n$	A board game for two players who move their 16 pieces according to specific rules.
Cinema	$cinema^1_n$	A medium that disseminates moving pictures.
Color	$color^1_n$	A visual attribute of things that results from the light they emit or transmit or reflect.
Commerce	$commerce^1_n$	Transactions (sales and purchases) having the objective of supplying commodities (goods and services).
Computer Science	$computer\_science^1_n$	The branch of engineering science that studies (with the aid of computers) computable processes and structures.
Cricket	$cricket^2_n$	A game played with a ball and bat by two teams of 11 players.
Cycling	$cycling^1_n$	The sport of traveling on a bicycle or motorcycle.
Dance	$dance^1_n$	An artistic form of nonverbal communication.
Dentistry	$dentistry^1_n$	The branch of medicine dealing with the anatomy and development and diseases of the teeth.
Diplomacy	$diplomacy^1_n$	Negotiation between nations.
Diving	$diving^2_n$	A headlong plunge into water.
Drawing	$drawing^3_n$	The creation of artistic pictures or diagrams.
Earth	$earth\_science^1_n$	Any of the sciences that deal with the earth or its parts.
Economy	$economy^1_n$	The system of production and distribution and consumption.

Domain	WordNet Sense	Gloss
ELECTRICITY	$\textit{electricity}_n^1$	A physical phenomenon associated with stationary or moving electrons and protons.
ELECTRONICS	$\textit{electronics}_n^1$	The branch of physics that deals with the emission and effects of electrons and with the use of electronic devices.
ELECTRO-TECHNOLOGY	$\textit{electrical\_engineering}_n^1$	The branch of engineering science that studies the uses of electricity and the equipment for power generation and distribution and the control of machines and communication.
ENGINEERING	$\textit{engineering}_n^2$	The discipline dealing with the art or science of applying scientific knowledge to practical problems.
ENTERPRISE	$\textit{enterprise}_n^2$	An organization created for business ventures.
ENTOMOLOGY	$\textit{entomology}_n^1$	The branch of zoology that studies insects.
ENVIRONMENT	$\textit{environment}_n^2$	The area in which something exists or lives.
ETHNOLOGY	$\textit{ethnology}_n^1$	The branch of anthropology that deals with the division of humankind into races and with their origins and distribution and distinctive characteristics.
EXCHANGE	$\textit{exchange}_n^8$	Reciprocal transfer of equivalent sums of money especially the currencies of different countries.
FACTOTUM	$\textit{factotum}_n^1$	A servant employed to do a variety of jobs.
FASHION	$\textit{fashion}_n^3$	The latest and most admired style in clothes and cosmetics and behavior.
FENCING	$\textit{fencing}_n^3$	The art or sport of fighting with swords (especially the use of foils or epees or sabres to score points under a set of rules).

Domain	WordNet Sense	Gloss
FINANCE	$\mathit{finance}_n^2$	The branch of economics that studies the management of money and other assets.
FISHING	$\mathit{fishing}_n^1$	The act of someone who fishes as a diversion.
FOLKLORE	$\mathit{folklore}_n^1$	The unwritten literature (stories and proverbs and riddles and songs) of a culture.
FOOD	food_n^2	Any solid substance (as opposed to liquid) that is used as a source of nourishment.
FOOTBALL	$\mathit{football}_n^1$	Any of various games played with a ball (round or oval) in which two teams try to kick or carry or propel the ball into each other's goal.
FREE TIME	$\mathit{free\_time}_n^1$	Time available for hobbies and other activities that you enjoy.
FURNITURE	$\mathit{furniture}_n^1$	Furnishings that make a room or other area ready for occupancy.
GAS	gas_n^1	The state of matter distinguished from the solid and liquid states by: relatively low density and viscosity.
GASTRONOMY	$\mathit{gastronomy}_n^2$	The art and practice of choosing and preparing and eating good food.
GENETICS	$\mathit{genetics}_n^1$	The branch of biology that studies heredity and variation in organisms.
GEOGRAPHY	$\mathit{geography}_n^1$	Study of the earth's surface.
GEOLOGY	$\mathit{geology}_n^1$	A science that deals with the history of the earth as recorded in rocks.
GEOMETRY	$\mathit{geometry}_n^1$	The pure mathematics of points and lines and curves and surfaces.
GOLF	golf_n^1	A game played on a large open course with 9 or 18 holes.
GRAMMAR	$\mathit{grammar}_n^1$	The branch of linguistics that deals with syntax and morphology (and sometimes also deals with semantics or morphology).

Domain	WordNet Sense	Gloss
Graphic Arts	$\mathit{graphic\_art}_n^1$	The arts of drawing or painting or printmaking.
Health	health_n^1	A healthy state of wellbeing free from disease.
Heraldry	$\mathit{heraldry}_n^1$	The study and classification of armorial bearings and the tracing of genealogies.
History	$\mathit{history}_n^4$	The discipline that records and interprets past events involving human beings.
Hockey	hockey_n^2	A game played on an ice rink by two opposing teams of six skaters each who try to knock a flat round puck into the opponents' goal with angled sticks.
Home	home_n^2	Housing that someone is living in.
Humanities	$\mathit{humanities}_n^1$	Studies intended to provide general knowledge and intellectual skills (rather than occupational or professional skills).
Hunting	$\mathit{hunting}_n^1$	The pursuit and killing or capture of wild animals regarded as a sport.
Hydraulics	$\mathit{hydraulics}_n^1$	Study of the mechanics of fluids.
Industry	$\mathit{industry}_n^2$	The organized action of making of goods and services for sale.
Insurance	$\mathit{insurance}_n^1$	Promise of reimbursement in the case of loss.
Jewellery	$\mathit{jewellery}_n^1$	An adornment (as a bracelet or ring or necklace) made of precious metals and set with gems (or imitation gems).
Law	law_n^5	The learned profession that is mastered by graduate study in a law school and that is responsible for the judicial system.
Linguistics	$\mathit{linguistics}_n^1$	The scientific study of language.
Literature	$\mathit{literature}_n^2$	The humanistic study of a body of literature.
Mathematics	$\mathit{mathematics}_n^1$	A science (or group of related sciences) dealing with the logic of quantity and shape and arrangement.

Domain	WordNet Sense	Gloss
MECHANICS	$mechanics_n^1$	The branch of physics concerned with the motion of bodies in a frame of reference.
MEDICINE	$medicine_n^3$	The learned profession that is mastered by graduate training in a medical school and that is devoted to preventing or alleviating or curing diseases and injuries.
METEOROLOGY	$meteorology_n^2$	The earth science dealing with phenomena of the atmosphere (especially weather).
METROLOGY	$metrology_n^1$	The scientific study of measurement.
MILITARY	$military_a^1$	Of or relating to the study of the principles of warfare.
MONEY	$money_n^3$	The official currency issued by a government or national bank.
MOUNTAINEERING	$mountaineering_n^1$	The activity of climbing a mountain.
MUSIC	$music_n^1$	An artistic form of auditory communication incorporating instrumental or vocal tones in a structured and continuous manner.
MYTHOLOGY	$mythology_n^2$	The study of myths.
NAUTICAL	$nautical_a^1$	Relating to or involving ships or shipping or navigation or seamen.
NUMBER	$number_n^1$	The property possessed by a sum or total or indefinite quantity of units or individuals.
NUMISMATICS	$numismatics_n^1$	The collection and study of money (and coins in particular).
OCCULTISM	$occultism_n^1$	The study of the supernatural.
OCEANOGRAPHY	$oceanography_n^1$	The branch of science dealing with physical and biological aspects of the oceans.
OPTICS	$optics_n^1$	The branch of physics that studies the physical properties of light.
PAINTING	$painting_n^1$	Graphic art consisting of an artistic composition made by applying paints to a surface.

Domain	WordNet Sense	Gloss
PALEONTOLOGY	$paleontology^1_n$	The earth science that studies fossil organisms and related remains.
PARANORMAL	$paranormal^1_a$	Seemingly outside normal sensory channels.
PEDAGOGY	$pedagogy^1_n$	The principles and methods of instruction.
PERSON	$person^1_n$	A human being.
PHARMACY	$pharmacy^1_n$	The art and science of preparing and dispensing drugs and medicines,.
PHILATELY	$philately^1_n$	The collection and study of postage stamps.
PHILOLOGY	$philology^1_n$	The humanistic study of language and literature.
PHILOSOPHY	$philosophy^2_n$	The rational investigation of questions about existence and knowledge and ethics.
PHOTOGRAPHY	$photography^1_n$	The act of taking and printing photographs.
PHYSICS	$physics^1_n$	The science of matter and energy and their interactions.
PHYSIOLOGY	$physiology^1_n$	The branch of the biological sciences dealing with the functioning of organisms.
PLANTS	$plant^2_n$	A living organism lacking the power of locomotion.
PLASTIC ARTS	$plastic\_art^1_n$	The arts of shaping or modeling.
PLAY	$game^2_n$	A contest with rules to determine a winner.
POLITICS	$politics^2_n$	The study of government of states and other political units.
POST	$post^{10}_n$	The system whereby messages are transmitted via the post office.
PSYCHIATRY	$psychiatry^1_n$	The branch of medicine dealing with the diagnosis and treatment of mental disorders.

Domain	WordNet Sense	Gloss
Psychoanalysis	$psychoanalysis_n^1$	A set of techniques for exploring underlying motives and a method of treating various mental disorders.
Psychological Features	$psychological\_feature_n^1$	A feature of the mental life of a living organism.
Psychology	$psychology_n^1$	The science of mental life.
Publishing	$publishing_n^1$	The business of issuing printed matter for sale or distribution.
Pure Science	$natural\_science_n^1$	The sciences involved in the study of the physical world and its phenomena.
Quality	$quality_n^1$	An essential and distinguishing attribute of something or someone.
Racing	$racing_n^1$	The sport of engaging in contests of speed.
Radio+Tv	$broadcasting_n^2$	Taking part in a radio or tv program.
Radiology	$radiology_n^1$	The branch of medical science dealing with the medical use of X-rays or other penetrating radiation.
Railway	$railway_n^1$	Line that is the commercial organization responsible for operating a system of transportation for trains that pull passengers or freight.
Religion	$religion_n^1$	A strong belief in a supernatural power or powers that control human destiny.
Roman Catholic	$roman\_catholic_n^2$	The Christian Church based in the Vatican and presided over by a pope and an episcopal hierarchy.
Rowing	$rowing_n^1$	The act of rowing as a sport.
Rugby	$rugby_n^1$	A form of football played with an oval ball.
School	$school_n^1$	An educational institution.
Sculpture	$sculpture_n^1$	A three-dimensional work of plastic art.
Sexuality	$sexuality_n^1$	The properties that distinguish organisms on the basis of their reproductive roles.

Domain	WordNet Sense	Gloss
SKATING	$skating^1_n$	The sport of gliding on skates.
SKIING	$skiing^1_n$	A sport in which participants must travel on skis.
SOCCER	$soccer^1_n$	A football game in which two teams of 11 players try to kick or head a ball into the opponents' goal.
SOCIAL SCIENCE	$social\_science^1_n$	The branch of science that studies society and the relationships of individual within a society.
SOCIOLOGY	$sociology^1_n$	The study and classification of human societies.
SPORT	$sport^1_n$	An active diversion requiring physical exertion and competition.
STATISTICS	$statistics^1_n$	A branch of applied mathematics concerned with the collection and interpretation of quantitative data and the use of probability theory to estimate population parameters.
SUB	$skin\_diving^1_n$	Underwater swimming with a breathing apparatus.
SURGERY	$surgery^1_n$	The branch of medical science that treats disease or injury by operative procedures.
SWIMMING	$swimming^1_n$	The act of swimming.
TABLE TENNIS	$table\_tennis^1_n$	A game (trademark Ping-Pong) resembling tennis but played on a table with paddles and a light hollow ball.
TAX	tax^1_n	Charge against a citizen's person or property or activity for the support of government.
TELECOMMUNICATION	$telecommunication^2_n$	(Often plural) the branch of electrical engineering concerned with the technology of electronic communication at a distance.

Domain	WordNet Sense	Gloss
TELEGRAPHY	$\textit{telegraphy}_n^1$	Communicating at a distance by electric transmission over wire.
TELEPHONY	$\textit{telephony}_n^1$	Transmitting speech at a distance.
TENNIS	$\textit{tennis}_n^1$	A game played with rackets by two or four players who hit a ball back and forth over a net that divides the court.
THEATRE	$\textit{theatre}_n^2$	The art of writing and producing plays.
THEOLOGY	$\textit{theology}_n^3$	The learned profession acquired by specialized courses in religion (usually taught at a college or seminary).
TIME PERIOD	$\textit{time\_period}_n^1$	An amount of time.
TOPOGRAPHY	$\textit{topography}_n^2$	Precise detailed study of the surface features of a region.
TOURISM	$\textit{tourism}_n^1$	The business of providing services to tourists.
TOWN PLANNING	$\textit{town\_planning}_n^1$	Determining and drawing up plans for the future physical arrangement and condition of a community.
TRANSPORT	$\textit{transport}_n^1$	Something that serves as a means of transportation.
UNIVERSITY	$\textit{university}_n^3$	A large and diverse institution of higher learning created to educate for life and for a profession and to grant degrees.
VEHICLES	$\textit{vehicle}_n^1$	A conveyance that transports people or objects.
VETERINARY	$\textit{veterinary}_a^1$	Of or relating to veterinarians or veterinary medicine.
VOLLEYBALL	$\textit{volleyball}_n^1$	A game in which two teams hit an inflated ball over a high net using their hands.
WRESTLING	$\textit{wrestling}_n^2$	The sport of hand-to-hand struggle between unarmed contestants who try to throw each other down.

Appendix B

Subcategorization Frame Types

The `tgrep` search strings given in this Appendix are based strongly on Roland (2001), with some modifications to account for differences between the Penn Treebank format he used in his work and the parse structures output by the Stanford Parser. Roland and Jurafsky (1998) estimate that the error rate of these `tgrep` strings for extracting SCFs from hand-tagged treebanks is between 3% and 7% for all verbs; the single largest source of errors with the `tgrep` strings is identifying quotations as arguments to verbs such as *say*.

Verbs inside of noun phrases are detected with the `tgrep` query `>(>(VP>NP))` and ignored.

Verbs that are not a passive construction are detected with the following `tgrep` query:
`!>(/VB/|MD>(VP<VBN!>NP%(/VB/<$BE_GET)))`
`!>(/VB/|MD>(VP<VBN>(VP!>NP%(/VB/<$BE_GET))))`
`!>(/VB/|MD>(VP<VBN!>NP%(VP<(/VB/<$BE_GET))))`
where the variable `$BE_GET` expands to:
`is|are|was|were|be|am|been|get|gets|got|gotten|getting|being`

Subcat Frame:	**NP** (transitive)
Instances in SemCor:	24,777
SemCor Example:	They **polished** [the windshield].

`tgrep` Queries:	`>((/VB/|MD)>(VP!>NP<NP`
	`!<(NP%..NP|PP|S|SBAR|VP|X)))`

(passive)	`>((/VB/|MD)>(VP!>NP!<(PP!<1(IN<by)))`
	`!%..NP|S|SINV|SBAR|VP|X)`

Subcat Frame:	**Other** (typically bare SBAR sentences)
Instances in SemCor:	12,990
SemCor Example:	Gene Marshall, genial manager of the club, has **announced** [that the Garden of the Gods will open to members Thursday, June 1].
`tgrep` Queries:	None

Subcat Frame:	∅ (intransitive)
Instances in SemCor:	12,846
SemCor Example:	Her little brown face **wrinkled up**, her brown eyes **gleamed**, and with her little gestures she said all the courteous things.
`tgrep` Queries:	`>((/VB/|MD)>(VP!>NP)!%..NP|PP|S|SINV|SBAR|VP|X)`

Subcat Frame:	**PP**
Instances in SemCor:	11,558
SemCor Example:	If he can **bounce back** [with one of those 25 home runs years], the club will have to be better off offensively.

`tgrep` Queries:	`>((/VB/|MD)>(VP!>NP!<NP<PP`
	`!<(/VB/%..NP|S|SBAR|VP|X)))`

Subcat Frame:	**NP-PP**
Instances in SemCor:	10,879
SemCor Example:	A light-colored roof will **reduce** [sun heat] [by 50 per cent].

`tgrep` Queries:	`>((/VB/|MD)>(VP!>NP<(NP%..PP)`
	`!<(NP%..NP|S|SBAR|VP|X)))`

(passive)	`>((/VB/|MD)>(VP!>NP!<NP<(PP!<1(IN<by))`
	`!<(/VB/%..NP|S|SBAR|VP|X)))`

Subcat Frame:	**VPto**
Instances in SemCor:	3,201
SemCor Example:	Dwellers thereabouts **preferred** [to get their apple pies at the local bakery, which had a brick oven fired with redwood billets].

`tgrep` Queries:	`>((/VB/|MD)>(VP!>NP!<NP<(S<(VP<(TO<to)))`
	`!<PP!<(/VB/%..NP|SBAR|VP|X)))`
	`>((/VB/|MD)>(VP!>NP!<NP<(S<1(S<(VP<(TO<to))))`
	`!<PP!<(/VB/%..NP|SBAR|VP|X)))`

Subcat Frame: **NP-SBAR**

Instances in SemCor: 2,011

SemCor Example: A British officer had come aboard and **told** [him] [that in case of enemy air attack he was not to open fire until bombs were actually dropped].

`tgrep` Queries:

```
>((/VB/|MD)>(VP!>NP<(NP%..SBAR)
    !<(NP%..NP|PP|S|VP|X)))
```

(passive)

```
>((/VB/|MD)>(VP!>NP<SBAR!<(PP!<1(IN<by))
    !<(NP|S|VP|X)))
```

Subcat Frame: **NP-NP** (ditransitive)

Instances in SemCor: 1,124

SemCor Example: There would be time enough, perhaps the Old Man reassured himself, to **pay** [the devil] [his due].

`tgrep` Queries:

```
>((/VB/|MD)>(VP!>NP<(NP%..NP)))
```

(passive)

```
>((/VB/|MD)>(VP!>NP<NP!<(PP!<1(IN<by))
    !<(NP%..NP|S|SBAR|VP|X)))
```

Subcat Frame: **NP-VPto**

Instances in SemCor: 1,121

SemCor Example: You can **use** [heat-absorbing glass] [to stop the sun], double glass and insulated glass to combat condensation.

`tgrep` Queries:

```
>((/VB/|MD)>(VP!>NP<(NP%..(S<(VP<(TO<to))))
    !<(NP%..NP|PP|SBAR|VP|X)))
>((/VB/|MD)>(VP!>NP<(S<NP<(VP<(TO<to)))
    !<(NP%..NP|PP|SBAR|VP|X)))
```

(passive)

```
>((/VB/|MD)>(VP!>NP!<NP<(S<(VP<(TO<to)))
    !<(PP!<1(IN<by))!<(/VB/%..NP|SBAR|VP|X)))
```

(passive)

```
>((/VB/|MD)>(VP!>NP!<NP<(S<1(S<(VP<(TO<to))))
    !<(PP!<1(IN<by))!<(/VB/%..NP|SBAR|VP|X)))
```

Subcat Frame: **VPing**

Instances in SemCor: 719

SemCor Example: No sooner had I **started** [drinking] than the driver **started** [zigzagging the truck].

`tgrep` Queries:

```
>((/VB/|MD)>(VP!>NP!<NP<(S!<NP<(VP<VBG))
    !<(/VB/%..NP|PP|SBAR|VP|X)))
>((/VB/|MD)>(VP!>NP!<NP<(VP<VBG)
    !<(/VB/%..NP|PP|S|SBAR|X)))
```

Subcat Frame: **S-for-to**

Instances in SemCor: 338

SemCor Example: One of the agreements **calls** [for the New Eastwick Corp.] [to purchase a 1311 acre tract for $12192865].

`tgrep` Queries:

```
>((/VB/|MD)>(VP!>NP!<NP<(S<(VP<(TO<to)))<PP
    !<(/VB/%..NP|SBAR|VP|X)))
```

Subcat Frame: **NP-VPing** (perceptual complement)

Instances in SemCor: 113

SemCor Example: Rachel had **seen** [me] [watching the young man].

`tgrep` Queries:

```
>((/VB/|MD)>(VP!>NP<NP<(VP<VBG)
    !<(/VB/%..NP|PP|S|SBAR|X)))
>((/VB/|MD)>(VP!>NP<(NP%..VP)
    !<(NP%..NP|PP|S|SBAR|X)))
>((/VB/|MD)>(VP!>NP!<NP<(S<(NP<<,*)<(VP<VBG))
    !<(/VB/%..NP|PP|SBAR|VP|X)))
```

(passive)

```
>((/VB/|MD)>(VP!>NP!<NP<(S<(NP<<,*)<(VP<VBG))
    !<(PP!<1(IN<by))!<(/VB/%..NP|SBAR|VP|X)))
```

(passive)

```
>((/VB/|MD)>(VP!>NP!<NP<(VP<VBG)!<(PP!<1(IN<by))
    !<(/VB/%..NP|S|SBAR|X)))
```

Bibliography

Eneko Agirre and Philip Edmonds, editors. *Word Sense Disambiguation: Algorithms and Applications*. Springer, 2006.

Eneko Agirre and Oier López de Lacalle. Publicly available topic signatures for all WordNet nominal senses. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC)*, Lisbon, Portugal, 2004.

Eneko Agirre and Aitor Soroa. Personalizing PageRank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 33–41, 2009.

Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. Knowledge-based WSD on specific domains: Performing better than generic supervised WSD. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1501–1506, Pasadena, CA, USA, 2009.

Galen Andrew, Trond Grenager, and Christopher D. Manning. Verb sense and subcategorization: Using joint inference to improve performance on complementary tasks. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 150–157, 2004.

Sue Atkins. Tools for computer-aided lexicography: The HECTOR project. *Acta Linguistica Hungarica*, 41:5–72, 1993.

John R. Ayto. On specifying meaning. In Reinhard R. K. Hartmann, editor, *Lexicography: Principles and Practice*, pages 89–98. Academic Press, London, 1983.

Yehoshua Bar-Hillel. The present status of automatic translation of languages. *Advances in Computers*, 1:91–163, 1960.

Tim Berners-Lee, James Hendler, and Ora Lassila. “The Semantic Web”. *Scientific American*, 284(5):28–37, 2001.

Daniel M. Bikel. A statistical model for parsing and word-sense disambiguation. In *Proceedings of the Joint SIGDAT conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, volume 13, pages 155–163, 2000.

Razvan Bunescu and Marius Paşca. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of EACL*, pages 9–16, 2006.

Jinying Chen and Martha Palmer. Towards robust high performance word sense disambiguation of English verbs using rich linguistic features. *Natural Language Processing–IJCNLP 2005*, pages 933–944, 2005.

Michael Collins. Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637, 2003.

Montse Cuadros and German Rigau. KnowNet: Building a large net of knowledge from the Web. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 161–168, 2008.

Montse Cuadros, Egoitz Laparra, German Rigau, Piek Vossen, and Wauter Bosma. Integrating a large domain ontology of species into WordNet. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC '10)*, pages 2310–2317, Valletta, Malta, 2010.

Jordi Daudé, Lluís Padrú, and German Rigau. Mapping WordNets using structural information. In *38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, Hong Kong, 2000.

Bonnie J. Dorr and Doug Jones. Role of word sense disambiguation in lexical acquisition: Predicting semantics from syntactic cues. In *Proceedings of the 16th Conference on Computational Linguistics*, volume 1, pages 322–327, 1996.

Philip Edmonds. Designing a task for SENSEVAL-2. Technical report, University of Brighton, United Kingdom, 2000.

Philip Edmonds and Scott Cotton. SENSEVAL-2: Overview. In *Proceedings of the Second International Workshop on Evaluating Word Sense Diambiguation Systems*, pages 1–6, Toulouse, France, 2001.

Philip Edmonds and Adam Kilgarriff. Introduction to the special issue on evaluating word sense disambiguation systems. *Journal of Natural Language Engineering*, 8(4): 279–291, 2002.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. Investigations on word senses and word usages. In *Proceedings of the Joint Conference of the 47th Annual Meeting of* *the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1*, pages 10–18, 2009.

Christiane Fellbaum, editor. *WordNet: An electronic lexical database.* MIT Press, Cambridge, MA, 1998.

J.R. Firth. *Papers in linguistics, 1934–1951.* Oxford University Press, London, 1957.

Dan Flickinger. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6(1):15–28, 2000.

William A. Gale and Geoffrey Sampson. Good-Turing frequency estimation without tears. *Journal of Quantitative Linguistics*, 2(3):217–237, 1995.

William A. Gale, Kenneth W. Church, and David Yarowsky. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(5):415–439, 1992a.

William A. Gale, Kenneth W. Church, and David Yarowsky. One sense per discourse. In *Proceedings of the Workshop on Speech and Natural Language*, pages 233–237, 1992b.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL*, pages 57–60, New York, NY, 2006.

Nancy Ide and Jean Véronis. Introduction to the special issue on word sense disambiguation: The state of the art. *Computational Linguistics*, 24(1):2–40, 1998.

Nancy Ide and Yorick Wilks. Making sense about sense. In Eneko Agirre and Phil Edmonds, editors, *Word Sense Disambiguation: Algorithms and Applications*, chapter 3, pages 47–73. Springer, 2006.

Adam Kilgarriff. "I don't believe in word senses". *Computers and the Humanities*, 31 (2):91–113, 1997.

Adam Kilgarriff. SENSEVAL: An exercise in evaluating word sense disambiguation programs. In *Proceedings of the First International Conference on Language Resources and Evaluation*, pages 581–585, Granada, Spain, 1998.

Adam Kilgarriff. English lexical sample task description. In *Proceedings of the ACL-SIGLEX* SENSEVAL *Workshop*, pages 17–20, Toulouse, France, 2001.

Adam Kilgarriff. How dominant is the commonest sense of a word? In *Proceedings of Text, Speech and Dialogue*, pages 103–111, Brno, Czech Republic, 2004.

Adam Kilgarriff and Joseph Rosenzweig. English SENSEVAL: Report and results. In *Proceedings of the Second Conference on Language Resources and Evaluation (LREC)*, pages 1239–1244, Athens, Greece, 2000.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. Extending VerbNet with novel verb classes. In *Proceedings of LREC*, 2006.

Karin Kipper-Schuler. *VerbNet: A broad-coverage, comprehensive verb lexicon.* PhD thesis, University of Pennsylvania, 2005.

Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 423–430, 2003.

Rob Koeling, Diana McCarthy, and John Carroll. Domain-specific sense distributions and predominant sense acquisition. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 419–426, 2005.

Anna Korhonen. Subcategorization acquisition. Technical report, University of Cambridge, Computer Laboratory, 2002.

Anna Korhonen and Ted Briscoe. Extended lexical-semantic classification of English verbs. In *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics*, pages 38–45, 2004.

Anna Korhonen and Judita Preiss. Improving subcategorization acquisition using word sense disambiguation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 48–55, 2003.

Robert Krovetz. Homonymy and polysemy in information retrieval. In *Proceedings of the Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 72–79, 1997.

Henry Kučera and W. Nelson Francis. *Computational analysis of present-day American English.* Brown University Press, Providence, RI, 1967.

Egoitz Laparra and German Rigau. Integrating WordNet and FrameNet using a knowledge-based word sense disambiguation algorithm. In *Proceedings of the International Conference, Recent Advances on Natural Language Processing (RANLP 09)*, 2009.

Geoffrey Leech. 100 million words of English: the British National Corpus. *Language Research*, 28(1):1–13, 1992.

Michael Lesk. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, pages 24–26, New York, NY, USA, 1986.

Beth Levin. *English verb classes and alternations: A preliminary investigation.* University of Chicago Press, Chicago, IL, 1993.

Bernardo Magnini and Gabriela Cavaglià. Integrating subject field codes into WordNet. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC 2000)*, pages 1413–1418, 2000.

Diana McCarthy. Lexical substitution as a task for WSD evaluation. In *Proceedings of the ACL Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pages 109–115, Philadelphia, USA, 2002.

Diana McCarthy. Relating WordNet senses for word sense disambiguation. In *Proceedings of the ACL Workshop on Making Sense of Sense*, pages 17–24, Trento, Italy, 2006.

Diana McCarthy and Roberto Navigli. SemEval-2007 Task 10: English lexical substitution task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 48–53, 2007.

Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. Finding predominant word senses in untagged text. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 280–287, Barcelona, Spain, 2004.

Rada Mihalcea. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 411–418, 2005.

Rada Mihalcea. Using Wikipedia for automatic word sense disambiguation. In *Proceedings of NAACL HLT*, pages 196–203, 2007.

Rada Mihalcea and Dan I. Moldovan. eXtended WordNet: Progress report. In *Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*, pages 95–100, Pittsburg, PA, 2001.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. WordNet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–312, 1990.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. A semantic concordance. In *Proceedings of the Workshop on Human Language Technology*, pages 303–308. Princeton, NJ, 1993.

Guido Minnen, John Carroll, and Darren Pearce. Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223, 2001.

Roberto Navigli. Meaningful clustering of senses helps boost word sense disambiguation performance. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 105–112, 2006.

Roberto Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):1–69, 2009.

Roberto Navigli and Mirella Lapata. Graph connectivity measures for unsupervised word sense disambiguation. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1683–1688, 2007.

Roberto Navigli and Paola Velardi. Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. In *Proceedings of* SENSEVAL-3*, Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 179–182, Barcelona, Spain, 2004.

Roberto Navigli and Paola Velardi. Structural semantic interconnections: A knowledge-based approach to word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1075–1086, 2005.

Roberto Navigli, Kenneth C. Litkowski, and Orin Hargraves. SemEval-2007 task 07: Coarse-grained English all-words task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 30–35, 2007.

Hwee Tou Ng, Chung Yong Lim, and Shou King Foo. A case study on inter-annotator agreement for word sense disambiguation. In *Proceedings of the ACL Workship on Standardizing Lexical Resources*, College Park, Maryland, 1999.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. MultiWordNet: Developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet*, pages 21–25, 2002.

Simone Ponzetto and Roberto Navigli. Knowledge-rich word sense disambiguation rivaling supervised systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1522–1531, Uppsala, Sweden, 2010.

Judita Preiss and Anna Korhonen. WSD for subcategorization acquisition task description. In *Proceedings of* SENSEVAL-3*, Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 33–36, 2004.

Judita Preiss, Anna Korhonen, and Ted Briscoe. Subcategorization acquisition as an evaluation method for WSD. In *Proceedings of the Language Resources and Evaluation Conference*, pages 1551–1556, Canary Islands, Spain, 2002.

Philip Resnik and David Yarowsky. Distinguishing systems and distinguishing senses: New evaluation methods for word sense disambiguation. *Natural Language Engineering*, 5(3):113–133, 2000.

Sergio Roa. Learning Bayesian networks for inference of semantic verb classes. Master's thesis, Albert-Ludwigs-University of Freiburg, Germany, 2007.

Douglas Roland. *Verb sense and verb subcategorization probabilities*. PhD thesis, University of Colorado, 2001.

Douglas Roland and Daniel Jurafsky. How verb subcategorization frequencies are affected by corpus choice. In *Proceedings of the 17th International Conference on Computational linguistics*, volume 2, pages 1122–1128, Montreal, Quebec, Canada, 1998.

Douglas Roland and Daniel Jurafsky. Verb sense and verb subcategorization probabilities. In Paola Merlo and Suzanne Stevenson, editors, *The Lexical Basis of Sentence Processing: Formal, Computational, and Experimental Issues*, chapter 16. John Benjamins, 2002.

Douglas Roland, Daniel Jurafsky, Lise Menn, Susanne Gahl, Elizabeth Elder, and Chris Riddoch. Verb subcategorization frequency differences between business-news and balanced corpora: the role of verb sense. In *Proceedings of the Workshop on Comparing Corpora*, volume 9, pages 28–34, Hong Kong, 2000.

Tony G. Rose, Mark Stevenson, and Miles Whitehead. The Reuters corpus volume 1 - from yesterday's news to tomorrow's language resources. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC)*, pages 29–31, Las Palmas de Gran Canaria, Spain, 2002.

Sabine Schulte im Walde and Chris Brew. Inducing German semantic verb classes from purely syntactic subcategorisation information. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 223–230, 2002.

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

Ravi Sinha and Rada Mihalcea. Unsupervised graph-based word sense disambiguation using measures of word semantic similarity. In *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007)*, pages 363–369, Irvine, CA, 2007.

Rion Snow, Sushant Prakash, Daniel Jurafsky, and Andrew Y. Ng. Learning to merge word senses. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007.

Benjamin Snyder and Martha Palmer. The English all-words task. In *Proceedings of the ACL 2004* SENSEVAL-*3 Workshop*, pages 41–43, Barcelona, Spain, 2004.

Catherine Soanes and Angus Stevenson, editors. *Oxford Dictionary of English.* Oxford University Press, 2003.

Mark Stevenson and Yorick Wilks. The interaction of knowledge sources in word sense disambiguation. *Computational Linguistics*, 27(3):321–349, 2001.

Suzanne Stevenson and Paola Merlo. Automatic verb classification using distributions of grammatical features. In *Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 45–52, 1999.

Penelope F. Stock. Polysemy. In *Proceedings of the Exeter Lexicography Conference*, pages 131–140, 1983.

Christopher Stokoe. Differentiating homonymy and polysemy in information retrieval. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 403–410, 2005.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, volume 1, pages 173–180, 2003.

George Tsatsaronis, Michalis Vazirgiannis, and Ion Androutsopoulos. Word sense disambiguation with spreading activation networks generated from thesauri. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence (IJCAI)*, pages 1725–1730, 2007.

Jean Véronis. A study of polysemy judgements and inter-annotator agreement. In *Advanced Papers of the* SENSEVAL *Workshop*, 1998.

David Vickrey, Luke Biewald, Marc Teyssier, and Daphne Koller. Word-sense disambiguation for machine translation. In *Proceedings of the Conference on Human*

Language Technology and Empirical Methods in Natural Language Processing, pages 771–778, Vancouver, Canada, 2005.

Piek Vossen, editor. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, Dordrecht, 1998.

John N. Williams. Processing polysemous words in context: Evidence for interrelated meanings. *Journal of Psycholinguistic Research*, 21(3):193–218, 1992.

David Yarowsky. One sense per collocation. In *Proceedings of the workshop on Human Language Technology*, pages 266–271, 1993.

Szu-ting Yi, Edward Loper, and Martha Palmer. Can semantic roles generalize across genres? In *Proceedings of the 2007 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 548–555, 2007.