The Disambiguation of Nominalizations

Maria Lapata[*]
University of Edinburgh

This article addresses the interpretation of nominalizations, a particular class of compound nouns whose head noun is derived from a verb and whose modifier is interpreted as an argument of this verb. Any attempt to automatically interpret nominalizations needs to take into account: (a) the selectional constraints imposed by the nominalized compound head, (b) the fact that the relation of the modifier and the head noun can be ambiguous, and (c) the fact that these constraints can be easily overridden by contextual or pragmatic factors. The interpretation of nominalizations poses a further challenge for probabilistic approaches since the argument relations between a head and its modifier are not readily available in the corpus. Even an approximation that maps the compound head to its underlying verb provides insufficient evidence. We present an approach that treats the interpretation task as a disambiguation problem and show how we can "re-create" the missing distributional evidence by exploiting partial parsing, smoothing techniques, and contextual information. We combine these distinct information sources using Ripper, a system that learns sets of rules from data, and achieve an accuracy of 86.1% (over a baseline of 61.5%) on the British National Corpus.

1. Introduction

The automatic interpretation of compound nouns has been a long-standing problem for natural language processing (NLP). Compound nouns in English have three basic properties that present difficulties for their interpretation: (a) the compounding process is extremely productive (this means that a hypothetical system would have to interpret previously unseen instances), (b) the semantic relationship between the compound head and its modifier is implicit (this means that it cannot be easily recovered from syntactic or morphological analysis), and (c) the interpretation can be influenced by a variety of contextual and pragmatic factors.

A considerable amount of effort has gone into specifying the set of semantic relations that hold between a compound head and its modifier (Levi 1978; Warren 1978; Finin 1980; Isabelle 1984). Levi (1978), for example, distinguishes two types of compound nouns: (a) compounds consisting of two nouns that are related by one of nine recoverably deletable predicates (e.g., CAUSE relates *onion tears*, FOR relates *pet spray*; see the examples in (1)) and (b) nominalizations, that is, compounds whose heads are nouns derived from a verb and whose modifiers are interpreted as arguments of the related verb (e.g., a *car lover* loves cars; see the examples in (2)–(4)). The prenominal modifier can be either a noun or an adjective (see the examples in (2)). The nominalized verb can take a subject (see (3a)), a direct object (see (3b)) or a prepositional object (see (3c)).

(1)	a. onion tears	CAUSE
	b. vegetable soup	HAVE

[*] Division of Informatics, University of Edinburgh, 2 Buccleuch Place, Edinburgh EH8 9LW, UK. E-mail: mlap@cogsci.ed.ac.uk

	c. music box	MAKE
	d. steam iron	USE
	e. pine tree	BE
	f. night flight	IN
	g. pet spray	FOR
	h. peanut butter	FROM
	i. abortion problem	ABOUT

(2)	a. parental refusal	SUBJ
	b. cardiac massage	OBJ
	c. heart massage	OBJ
	d. sound synthesizer	OBJ

(3)	a. child behavior	SUBJ
	b. car lover	OBJ
	c. soccer competition	AT|IN

(4)	a. government promotion	SUBJ|OBJ
	b. satellite observation	SUBJ|OBJ

Besides Levi (1978), a fair number of researchers (Warren 1978; Finin 1980; Isabelle 1984; Leonard 1984) agree that there is a limited number of regularly recurring relations between a compound head and its modifier. There is far less agreement when it comes to the type and number of these relations. The relations vary from Levi's (1978) recoverably deletable predicates to Warren's (1978) paraphrases and Finin's (1980) role nominals. Leonard (1984) proposes eight relations, and Warren (1978) proposes six basic relations, whereas the number of relations proposed by Finin (1980) is potentially infinite.

The attempt to restrict the semantic relations between the compound head and its modifier to a prespecified number and type has been criticized by Downing (1977), who has shown (through a series of psycholinguistic experiments) that the underlying relations can be influenced by a variety of pragmatic factors and cannot therefore be presumed to be easily enumerable. Sparck Jones (1983, page 4) further notes "that observations about the semantic relation holding between the compound head and its modifier can only be remarks about tendencies and not about absolutes." Consider, for instance, the compound *onion tears* (see (1a)). The relationship CAUSE is one of the possible interpretations the compound may receive. One could easily imagine a context in which the tears are FOR or ABOUT the onion. Consider example[1] (5a), taken from Downing (1977, page 818). Here *apple-juice seat* refers to the situation in which someone is instructed to sit in a seat in front of which a glass of apple juice has been placed. Given this particular state of affairs, none of the relations in (1) can be used to successfully interpret *apple-juice seat*. Such considerations have led Selkirk (1982) to

[1] Unless stated otherwise the example sentences were taken from the British National Corpus and in some cases simplified for purposes of clarity.

claim that only nominalizations are amenable to linguistic characterization, leaving all other compounds to be explained by pragmatics or discourse. A similar approach is put forward by Hobbs et al. (1993) for all types of compounds, including nominalizations: any two nouns can be combined, and the relation between these nouns is entirely underspecified, to be resolved pragmatically.

(5) a. A friend of mine was once instructed to sit in the **apple-juice seat**.

b. By the end of the 1920s, **government promotion** of agricultural development in Niger was limited, consisting mainly of crop trials and model sheep and ostrich farms.

Less controversy arises with regard to nominalizations, perhaps because of the small number of allowable relations. Most approaches follow Levi (1978) in distinguishing nominalizations as a separate class of compounds, the exception being Finin (1980), who claims that most compounds are nominalizations, even in cases in which the head noun is not morphologically derived from a verb (see the examples in (1)). Under Finin's analysis the head *book* in the compound *recipe book* is a role nominal, that is, a noun that refers to a particular thematic role of another concept. This means that *book* refers to the object role of *write*, which is filled by *recipe*. It is not clear, however, how the implicit verb is to be recovered or why *write* is more appropriate than *read* in this example.

Despite the small number of relations between the nominalized head and its modifier, the interpretation of nominalizations can readily change in different contexts. In some cases, the relation of the modifier and the nominalized verb (e.g., subject or object) can be predicted either from the subcategorization properties of the verb or from the semantics of the nominalization suffix of the head noun. Consider (3a), for example. Here *child* can be only the subject of *behavior*, since the verb *behave* is intransitive. In (3b) the agentive suffix *-er* of the head noun *lover* indicates that the modifier *car* is the object of the verb *love*. In other cases, the relation of the modifier and the head noun is genuinely ambiguous. Out of context the compounds *government promotion* and *satellite observation* (see example (4)) can receive either a subject or an object interpretation. One might argue that the preferred analysis for *government promotion* is "government that is promoted by someone." This interpretation can be easily overridden in context, however, as shown in Example (5b): here it is the government that is doing the promotion.

The automatic interpretation of compound nouns poses a challenge for empirical approaches, since the relations between a head and its modifier are not readily available in a corpus, and therefore they have to be somehow retrieved and approximated. Given the data sparseness and the parameter estimation difficulties, it is not surprising that a far greater number of symbolic than probabilistic solutions have been proposed for the automatic interpretation of compound nouns. With the exception of Wu (1993) and Lauer (1995), who use probabilistic models for compound noun interpretation (see Section 7 for details), most algorithms rely on hand-crafted knowledge bases or dictionaries that contain detailed semantic information for each noun; a sequence of rules exploit a knowledge base to choose the correct interpretation for a given compound (Finin 1980; McDonald 1982; Leonard 1984; Vanderwende 1994).

In what follows we develop a probabilistic model for the interpretation of nominalizations. We focus on nominalizations whose prenominal modifier is either the underlying subject or direct object of the verb corresponding to the nominalized compound head. In other words, we focus on examples like (3a, 3b) and ignore for the moment nominalizations whose heads correspond to verbs taking prepositional complements (see example (3c)). Nominalizations are attractive from an empirical perspective: the amount of relations is small (i.e., subject or object, at least if one focuses on direct objects only) and fairly uncontroversial (see the discussion above). Although the relations are not attested in the corpus, they can be retrieved and approximated through parsing. The probabilistic interpretation of nominalizations can provide a lower bound for the difficulty of the compound interpretation task: if we cannot interpret nominalizations successfully, there is little hope for modeling more complex semantic relations stochastically (see the examples in (1)).

We present a probabilistic algorithm that treats the interpretation task as a disambiguation problem. Our approach relies on the simplifying assumption that the relation of the nominalized head and its modifier noun can be approximated by the relation of the latter and the verb from which the head is derived. This approach works insofar as the verb-argument relations from which the nominalizations are derived are attested in the corpus. We show that a large number of verb-argument configurations do not occur in the corpus, something that is perhaps not surprising considering the ease with which novel compounds are created (Levi 1978). We estimate the frequencies of unseen verb-argument pairs by experimenting with three types of smoothing techniques proposed in the literature (back-off smoothing, class-based smoothing, and distance-weighted averaging) and show that their combination achieves good performance. Furthermore, we explore the contribution of context to the disambiguation task and show that performance is increased by taking contextual features into account. Our best results are achieved by combining the predictions of our probabilistic model with contextual information.

The remainder of this article is organized as follows: in Section 2 we present a simple statistical model for the interpretation of nominalizations and describe the procedure used to collect the data for our experiments. Section 3 presents details on how the parameters of the model were estimated and gives a brief overview on the smoothing methods with which we experimented. Section 4 describes the algorithm used for the interpretation of nominalizations, and Section 5 reports the results of several experiments that achieve a combined accuracy of 86.1% on the British National Corpus (BNC). Section 6 discusses the findings. In Section 7 we review related work, and we conclude in Section 8.

2. The Model

2.1 Guessing Argument Relations

As explained in Section 1, nominalizations are compounds whose head noun is a nominalized verb and whose prenominal modifier is derived from either the underlying subject or the underlying object of that verb (Levi 1978). Our goal, given a nominalization, is to develop a procedure for inferring whether the modifier stands in a subject or object relation with respect to the head noun. In other words, we need to assign probabilities to the two different relations (SUBJ, OBJ). For each relation *rel* we calculate the simple expression $P(rel \mid n_1, n_2)$ given in (6).

$$P(rel \mid n_1, n_2) = \frac{f(n_1, rel, n_2)}{f(n_1, n_2)} \tag{6}$$

Since we have a choice between two outcomes we will use a likelihood ratio to compare the two relation probabilities (Mosteller and Wallace 1964; Hindle and Rooth 1993). In particular we will compute the log of the ratio of the probability $P(\text{OBJ} \mid n_1, n_2)$

to the probability $P(\textsc{subj} \mid n_1, n_2)$. We will call this log-likelihood ratio the argument relation (RA) score.

$$RA(rel, n_1, n_2) = \log_2 \frac{P(\textsc{obj} \mid n_1, n_2)}{P(\textsc{subj} \mid n_1, n_2)} \tag{7}$$

Notice, however, that we cannot read off $f(n_1, rel, n_2)$ directly from the corpus. What we can obtain from a corpus (through parsing) is the number of times a noun is the object or the subject of a given verb. By making the simplifying assumption that the relation of the nominalized head and its modifier noun is the same as the relation between the latter and the verb from which the head is derived, we can rewrite (6) as follows:

$$P(rel \mid n_1, n_2) \approx \frac{f(v_{n_2}, rel, n_1)}{\sum\limits_i f(v_{n_2}, rel_i, n_1)} \tag{8}$$

where $f(v_{n_2}, rel, n_1)$ is the frequency with which the modifier noun n_1 is found in the corpus as the subject or object of v_{n_2}, the verb from which the head noun is derived. The sum $\sum_i f(v_{n_2}, rel_i, n_1)$ is a normalization factor.

2.2 Parameter Estimation

2.2.1 Verb-Argument Tuples. We estimated the parameters of the model outlined in the previous section from a part-of-speech-tagged and lemmatized version of the BNC, a 100-million-word collection of samples of written and spoken language from a wide range of sources designed to represent current British English (Burnard 1995). To estimate the term $f(v_{n_2}, rel, n_1)$, the corpus was automatically parsed by Cass (Abney 1996), a robust chunk parser designed for the shallow analysis of noisy text. The main feature of Cass is its finite-state cascade technique. A finite-state cascade is a sequence of nonrecursive levels: phrases at one level are built on phrases at the previous level without containing same-level or higher-level phrases. We used the parser's built-in function to extract tuples of verb subjects and verb objects (see (9)).

(9)	a. change situation	SUBJ
	b. come off heroin	OBJ
	c. deal with situation	OBJ

(10)	a. isolated people	SUBJ
	b. smile good	SUBJ

The tuples obtained from the parser's output are an imperfect source of information about argument relations. Bracketing errors, as well as errors in identifying chunk categories accurately, result in tuples whose lexical items do not stand in a verb-argument relationship. For example, inspection of the original BNC sentences from which (10a) and (10b) were derived revealed that the verb is missing from the former and the noun is missing from the latter (see the sentences in (11)).

(11) a. Wenger found that more than half the childless old people in her study of rural Wales saw a relative, a sibling, niece, nephew or cousin at least once a week, though in inner city London there were more isolated old people.

b. I smiled my best smile down the line.

Table 1
Tuples extracted from the BNC.

	Tokens		Types		
Relation	Parser	Filtering	Tuples	Verbs	Nouns
SUBJ	4,491,386	4,095,578	588,333	10,852	41,336
OBJ	2,631,752	2,598,069	615,328	9,490	35,846

Table 2
Deverbal suffixes.

Suffix	Nominalization
-ER	drink → drinker
-OR	direct → director
-ANT	disinfect → disinfectant
-EE	employ → employee
-ATION	educate → education
-MENT	arrange → arrangement
-AL	refuse → refusal
-ING	hire → hiring

Table 3
Conversion.

Verb	→	Noun
release	→	release
arrest	→	arrest
compromise	→	compromise
attempt	→	attempt

To compile a comprehensive count of verb-argument relations, we tried to eliminate from the parser's output tuples containing erroneous verbs and nouns like those in (10). We did this by matching the verbs contained in the tuples against a list of all words tagged as verbs and nouns in the BNC. Tuples containing words not included in the list were discarded. Furthermore, we discarded tuples containing verbs or nouns attested in a verb-argument relationship only once. This resulted in 588,333 distinct verb-subject pairs and 615,328 distinct verb-object pairs (see Table 1, which contains information about the tuples extracted from the corpus before and after the filtering described earlier in the paragraph).

2.2.2 The Data. So far we have been using the term *nominalization* to refer to two-word compounds whose head is derived from a verb. Morphologically speaking, nominalization is a word formation process by which a noun is derived from a verb, usually by means of suffixation (Quirk et al. 1985). A list of deverbal suffixes (i.e., suffixes that form nouns when attached to verb bases) is given in Table 2. Nominalizations can also be created by **conversion**, the word formation process whereby "an item is adapted or converted to a new word-class without the addition of an affix" (Quirk et al. 1985, page 1009). Examples of conversion are shown in Table 3.

It is beyond the scope of the present study to develop an algorithm that automatically detects nominalizations in a corpus. In the experiments described in the subsequent sections compounds with deverbal heads were obtained as follows:

1. Two-word compound nouns were extracted from the BNC using a heuristic that looks for consecutive pairs of nouns that are neither preceded nor succeeded by a noun (Lauer 1995).
2. A dictionary of deverbal nouns was created using two sources: (a) NOMLEX (Macleod et al. 1998), a dictionary of nominalizations

containing 827 lexical entries, and (b) CELEX (Burnage 1990), a general morphological dictionary that contains 5,111 nominalizations; both dictionaries list the verbs from which the nouns are derived. Sample dictionary entries are given in Tables 2 and 3.

3. Candidate nominalizations were obtained from the compounds acquired from the BNC by selecting noun-noun sequences whose head (i.e., rightmost noun) was one of the deverbal nouns contained in either CELEX or NOMLEX. The procedure resulted in 172,797 potential types of nominalizations.

From these candidate nominalizations a random sample of 1,277 tokens was selected. The sample was manually inspected, and compounds with modifiers whose relation to the head noun was other than subject or object were discarded. In particular nominalizations were discarded if: (a) the relation between the head and the modifier was any of the semantic relations listed in (1) (e.g., CAUSE, HAVE, MAKE); these compounds represented 28.0% of the sample; (b) the head was derived from verbs taking prepositional objects (see example (3c)); these nominalizations represented 9.2% of the sample. After manual inspection the sample contained 796 nominalizations (62.8% of the initial sample). These tokens were used for the experiments described in Section 5.

2.2.3 Mapping. To estimate the frequency, $f(v_{n_2}, rel, n_1)$, the nominalized heads were mapped to their corresponding verbs. Inspection of the frequencies of the verb-argument tuples contained in our data (796 tokens) revealed that 480 verb-noun pairs (60.3%) had a verb-object frequency of zero in the corpus. Similarly, 503 verb-noun pairs (63.2%) had a verb-subject frequency of zero. Furthermore, a total of 373 tuples (46.9%) were not attested at all in the BNC either in a verb-object or verb-subject relation. This finding is not entirely unexpected, considering that compounds are typically used as a text compression device (Marsh 1984), that is, to pack meaning into a minimal amount of linguistic structure. If a nominalization is chosen over a more elaborate structure (i.e., a sentence), then it is not surprising that some verb-argument configurations will not occur in the corpus. Furthermore, some nominalizations are conventionalized (e.g., *business administration*, *health organization*) and are therefore attested more frequently than their verb-subject or verb-object counterparts.

We re-created the frequencies of unseen verb-argument pairs by experimenting with three types of smoothing techniques proposed in the literature: back-off smoothing (Katz 1987), class-based smoothing (Resnik 1993; Lauer 1995), and distance-weighted averaging (Grishman and Sterling 1994; Dagan, Lee, and Pereira 1999). We present these three smoothing variants and their underlying assumptions in the following section.

3. Smoothing

Smoothing techniques have been used in a variety of statistical NLP applications as a means of addressing data sparseness, an inherent problem for statistical methods that rely on the relative frequencies of word combinations. The problem arises when the probability of word combinations that do not occur in the training data needs to be estimated. The smoothing methods proposed in the literature (overviews are provided by Dagan, Lee, and Pereira (1999) and Lee (1999)) can be generally divided into three types: *discounting* (Katz 1987), *class-based smoothing* (Resnik 1993; Brown et al. 1992;

Pereira, Tishby, and Lee 1993), and *distance-weighted averaging* (Grishman and Sterling 1994; Dagan, Lee, and Pereira 1999).

Discounting methods decrease the probability of previously seen events so that the total probability of observed word co-occurrences is less than one, leaving some probability mass to be redistributed among unseen co-occurrences. Class-based smoothing and distance-weighted averaging both rely on an intuitively simple idea: interword dependencies are modeled by relying on the corpus evidence available for words that are similar to the words of interest. The two approaches differ in the way they measure word similarity. Distance-weighted averaging estimates word similarity from lexical co-occurrence information; namely, it finds similar words by taking into account the linguistic contexts in which they occur: two words are similar if they occur in similar contexts. In class-based smoothing, classes are used as the basis according to which the co-occurrence probability of unseen word combinations is estimated. Classes can be induced directly from the corpus using distributional clustering (Pereira, Tishby, and Lee 1993; Brown et al. 1992; Lee and Pereira 1999) or taken from a manually crafted taxonomy (Resnik 1993). In the latter case the taxonomy is used to provide a mapping from words to conceptual classes.

Distance-weighted averaging differs from distributional clustering in that it does not explicitly cluster words. Although both methods make use of the evidence of words similar to the words of interest, distributional clustering assigns to each word a probability distribution over clusters to which it may belong; co-occurrence probabilities can then be estimated on the basis of the average of the clusters to which the words in the co-occurrence belong. This means that word co-occurrences are modeled by taking general word clusters into account and that the same set of clusters is used for different co-occurrences. Distance-weighted averaging does not explicitly create general word clusters. Instead, unseen co-occurrences are estimated by averaging the set of co-occurrences most similar to the target unseen co-occurrence, and a different set of similar neighbors (i.e., distributionally similar words) is used for different co-occurrences.

In language modeling, smoothing techniques are typically evaluated by showing that a language model that uses smoothed estimates incurs a reduction in perplexity on test data over a model that does not employ smoothed estimates (Katz 1987). Dagan, Lee, and Pereira (1999) use perplexity to compare back-off smoothing against distance-weighted averaging methods within the context of language modeling for speech recognition and show that the latter outperform the former. They also compare different distance-weighted averaging methods on a pseudoword disambiguation task in which the language model decides which of two verbs v_1 and v_2 is more likely to take a noun n as its object. The method being tested must reconstruct which of the unseen (v_1, n) and (v_2, n) is a valid verb-object combination. The same task is used by Lee and Pereira (1999) in a detailed comparison between distributional clustering and distance-weighted averaging that demonstrates that the two methods yield comparable results.

In our experiments we re-created co-occurrence frequencies for unseen verb-subject and verb-object pairs using three maximally different approaches: back-off smoothing, class-based smoothing using a predefined taxonomy, and distance-weighted averaging. We preferred taxonomic class-based methods over distributional clustering mainly because we wanted to compare directly methods that use distributional information inherent in the corpus without making external assumptions with regard to how concepts and their similarity are represented with methods that quantify similarity relationships based on information present in a hand-crafted taxonomy. Furthermore, as Lee and Pereira's (1999) results indicate that distributional clustering and distance-weighted averaging obtain similar levels of performance, we restricted ourselves to the latter.

We evaluated the contribution of the different smoothing methods on the nominalization task by exploring how each method and their combination influences disambiguation performance. Sections 3.1–3.3 review discounting, class-based smoothing, and distance-weighted averaging. Section 4 introduces an algorithm that uses smoothed verb-argument tuples to arrive at the interpretation of nominalizations.

3.1 Back-Off Smoothing

Back-off n-gram models were initially proposed by Katz (1987) for speech recognition but have also been successfully used to disambiguate the attachment site of structurally ambiguous prepositional phrases (Collins and Brooks 1995). The main idea behind back-off smoothing is to adjust maximum likelihood estimates like (8) so that the total probability of observed word co-occurrences is less than one, leaving some probability mass to be redistributed among unseen co-occurrences. In general the frequency of observed word sequences is discounted using the Good-Turing estimate (see Katz (1987) and Church and Gale (1991) for details on Good-Turing estimation), and the probability of unseen sequences is estimated by using lower-level conditional distributions. Assuming that the numerator $f(v_{n_2}, rel, n_1)$ in (8) is zero we can approximate $P(rel \mid n_1, n_2)$ by backing off to $P(rel \mid n_1)$:

$$P(rel \mid n_1, n_2) = \alpha \frac{f(rel, n_1)}{f(n_1)} \tag{12}$$

where α is a normalization constant that ensures that the probabilities sum to one. If the frequency $f(rel, n_1)$ is also zero, backing off continues by making use of $P(rel)$.

3.2 Class-Based Smoothing

Generally speaking, taxonomic class-based smoothing re-creates co-occurrence frequencies based on information provided by lexical resources such as WordNet (Miller et al. 1990) or Roget's publicly available thesaurus. In the case of verb-argument tuples, we use taxonomic information to estimate the frequencies $f(v_{n_2}, rel, n_1)$ by substituting for the word n_1 occurring in an argument position the concept with which it is represented in the taxonomy (Resnik 1993). So $f(v_{n_2}, rel, n_1)$ can be estimated by counting the number of times the concept corresponding to n_1 was observed as the argument of the verb v_{n_2} in the corpus.

This would be a straightforward task if each word was always represented in the taxonomy by a single concept or if we had a corpus of verb-argument tuples labeled explicitly with taxonomic information. Lacking such a corpus we need to take into consideration the fact that words in a taxonomy may belong to more than one conceptual class: counts of verb-argument configurations are reconstructed for each conceptual class by dividing the contribution from the argument by the number of classes to which it belongs (Resnik 1993; Lauer 1995):

$$f(v_{n_2}, rel, c) \approx \sum_{n'_1 \in c} \frac{f(v_{n_2}, rel, n'_1)}{|classes(n'_1)|} \tag{13}$$

where $f(v_{n_2}, rel, n'_1)$ is the number of times the verb v_{n_2} was observed with concept $c \in classes(n'_1)$ bearing the argument relation rel (i.e., subject or object) and $|classes(n'_1)|$ is the number of conceptual classes to which n'_1 belongs.

Table 4
Frequency estimation for *group registration* using WordNet.

Verb	Class	$f(v_{n_2},\textsc{obj},n_1)$	$f(v_{n_2},\textsc{subj},n_1)$
register	⟨`abstraction`⟩	16.26	7.28
register	⟨`entity`⟩	14.10	4.50
register	⟨`object`⟩	8.02	1.56
register	⟨`set`⟩	.65	.07
register	⟨`substance`⟩	.70	.08

Consider, for example, the tuple *register group* (derived from the compound *group registration*), which is not attested in the BNC. The word *group* has two senses in WordNet and belongs to five conceptual classes (⟨`abstraction`⟩, ⟨`entity`⟩, ⟨`object`⟩, ⟨`set`⟩, and ⟨`substance`⟩). This means that the frequency $f(v_{n_2}, rel, c)$ will be constructed for each of the five classes, as shown in Table 4. Suppose now that we see the tuple *register patient* in the corpus. The word *patient* has two senses in WordNet and belongs to seven conceptual classes (⟨`case`⟩, ⟨`person`⟩, ⟨`life form`⟩, ⟨`entity`⟩, ⟨`causal agent`⟩, ⟨`sick person`⟩, ⟨`unfortunate`⟩), one of which is ⟨`entity`⟩. This means that we will increment the observed co-occurrence count of *register* and ⟨`entity`⟩ by $\frac{1}{7}$. Since we do not know which is the actual class of the noun *group* in the corpus, we weight the contribution of each class by taking the average of the constructed frequencies for all five classes:

$$f(v_{n_2}, rel, n_1) = \frac{\sum\limits_{c \in classes(n_1)} \sum\limits_{n_1' \in c} \frac{f(v_{n_2}, rel, n_1')}{|classes(n_1')|}}{|classes(n_1)|} \tag{14}$$

Following (14) the frequencies $f(register,\ \textsc{obj},\ group)$ and $f(register,\ \textsc{subj},\ group)$ are $\frac{39.73}{5}$ and $\frac{13.49}{5}$, respectively. Note that the estimation of the frequency $f(v_{n_2}, rel, n_1)$ (see equations (13) and (14)) crucially relies on the simplifying assumption that the argument of a verb is distributed evenly across its conceptual classes. This simplification is necessary unless we have a corpus of verb-argument pairs labeled explicitly with taxonomic information. The task of finding the right class for representing the argument of a given predicate is a research issue on its own (Clark and Weir 2001; Li and Abe 1998; Carroll and McCarthy 2000), and a detailed comparison between different methods for accomplishing this task is beyond the scope of the present study.

3.3 Distance-Weighted Averaging

Distance-weighted averaging induces classes of similar words from word co-occurrences without making reference to a taxonomy. Instead, it is based on the assumption that if a word w_1' is *similar* to word w_1, then w_1' can provide information about the frequency of unseen word pairs involving w_1 (Dagan, Lee, and Pereira 1999). A key feature of this type of smoothing is the function that measures distributional similarity from co-occurrence frequencies.

Several measures of distributional similarity have been proposed in the literature (Dagan, Lee, and Pereira 1999; Lee 1999). We used two measures, the Jensen-Shannon divergence and the confusion probability. The choice of these two measures was motivated by work described in Dagan, Lee, and Pereira (1999), in which the Jensen-Shannon divergence outperforms related similarity measures (such as the confusion probability or the L_1 norm) on a pseudodisambiguation task that uses verb-object pairs. The confusion probability has been used by several authors to smooth word co-occurrence probabilities (Essen and Steinbiss 1992; Grishman and Sterling 1994) and shown to give promising performance. Grishman and Sterling (1994) in particular employ the confusion probability to re-create the frequencies of verb-noun co-occurrences in which the noun is the object or the subject of the verb in question. In the following we describe these two similarity measures and show how they can be used to re-create the frequencies for unseen verb-argument tuples (for a more detailed description see Dagan, Lee, and Pereira (1999)).

3.3.1 Confusion Probability. The confusion probability P_C is an estimate of the probability that a word w_1 can be substituted for a word w_1', in the sense of being found in the same contexts. In other words, the metric expresses how probable it is for word w_1' to occur in contexts in which word w_1 occurs. A large confusion probability value indicates that the two words w_1' and w_1 appear in similar contexts. P_C is estimated as follows:

$$P_C(w_1 \mid w_1') = \sum_s P(w_1 \mid s) P(s \mid w_1') \tag{15}$$

where $P_C(w_1 \mid w_1')$ is the probability that word w_1' occurs in the same contexts s as word w_1, averaged over these contexts. Given a tuple of the form w_1, rel, w_2, we can either treat w_1, rel as context and smooth over the noun w_2 or rel, w_2 as context and smooth over the verb w_1. We opted for the latter for two reasons. Theoretically speaking, it is the verb that imposes the semantic restrictions on its arguments and not vice versa. The idea that semantically similar verbs have similar subcategorizational and selectional patterns is by no means new and has been extensively argued for by Levin (1993). Computational efficiency considerations also favor an approach that treats rel, w_2 as context: the nouns w_2 outnumber the verbs w_1 by a factor of four (see Table 1). When verb-argument tuples are taken into consideration, (8) can be rewritten as follows:

$$\begin{aligned} P_C(w_1 \mid w_1') &= \sum_{rel,w_2} P(w_1 \mid rel, w_2) P(rel, w_2 \mid w_1') \\ &= \sum_{rel,w_2} \frac{f(w_1, rel, w_2)}{f(rel, w_2)} \frac{f(w_1', rel, w_2)}{f(w_1')} \end{aligned} \tag{16}$$

The confusion probability can be computed efficiently, since it involves summation only over the common contexts rel, w_2.

3.3.2 Jensen-Shannon Divergence. The Jensen-Shannon divergence J is an information-theoretic measure. It recasts the concept of distributional similarity into a measure of the "distance" between two probability distributions. The value of the Jensen-Shannon divergence ranges from zero for identical distributions to $\log 2$ for maximally different distributions. J is defined as:

$$J(w_1, w_1') = \frac{1}{2}\left[D\left(w_1 \left\| \frac{w_1 + w_1'}{2}\right.\right) + D\left(w_1' \left\| \frac{w_1 + w_1'}{2}\right.\right)\right] \tag{17}$$

$$D(w_1 \| w_1') = \sum_{rel,w_2} P(rel, w_2 \mid w_1) \log \frac{P(rel, w_2 \mid w_1)}{P(rel, w_2 \mid w_1')} \tag{18}$$

where w_1 is a shorthand for $P(rel, w_2 \mid w_1)$ and w_1' for $P(rel, w_2 \mid w_1')$; D in (17) is the Kullback-Leibler divergence, a measure of the dissimilarity between two probability distributions (see equation (18)) and $(w_1 + w_1')/2$ is a shorthand for the average distribution:

$$\frac{1}{2}(P(rel, w_2 \mid w_1) + P(rel, w_2 \mid w_1')) \tag{19}$$

```
Given a set of nominalizations $n_1$ $n_2$:
    1. map the head noun $n_2$ to the verb $v_{n_2}$ from which it is derived;
    2. retrieve frequencies $f(v_{n_2}, \textsc{obj}, n_1)$ and $f(v_{n_2}, \textsc{subj}, n_1)$ from the BNC;
    3. if $f(v_{n_2}, \textsc{obj}, n_1) < k$ then re-create $f_s(v_{n_2}, \textsc{obj}, n_1)$;
    4. if $f(v_{n_2}, \textsc{subj}, n_1) < k$ then re-create $f_s(v_{n_2}, \textsc{subj}, n_1)$;
    5. calculate probabilities $P(\textsc{obj} \mid n_1, n_2)$ and $P(\textsc{subj} \mid n_1, n_2)$;
    6. compute $RA(rel, n_1, n_2)$;
    7. if $RA \geq j$ then $n_1$ is the object of $n_2$;
    8. else $n_1$ is the subject of $n_2$.
```

Figure 1
Disambiguation algorithm for nominalizations.

Similarly to the confusion probability, the computation of J depends only on the common contexts rel, w_2. Recall that the Jensen-Shannon divergence is a dissimilarity measure. The dissimilarity measure is transformed into a similarity measure using a weight function $W_J(w, w'_1)$:

$$W_J(w_1, w'_1) = 10^{-\beta J(w_1, w'_1)} \tag{20}$$

The parameter β controls the relative influence of the neighbors (i.e., distributionally similar words) closest to w_1: if β is high, only neighbors extremely close to w_1 contribute to the estimate, whereas if β is low, distant neighbors also contribute to the estimate.

We estimate the frequency of an unseen verb-argument tuple by taking into account the similar w_1s and the contexts in which they occur (Grishman and Sterling 1994):

$$f_s(w_1, rel, w_2) = \sum_{w'_1} \mathrm{sim}(w_1, w'_1) f(w'_1, rel, w_2) \tag{21}$$

where $\mathrm{sim}(w_1, w'_1)$ is a function of the similarity between w_1 and w'_1. In our experiments the confusion probability $P_C(w_1 \mid w'_1)$ and the Jensen-Shannon divergence $W_J(w_1, w_1{}')$ were substituted for $\mathrm{sim}(w_1, w'_1)$.

4. The Disambiguation Algorithm

The disambiguation algorithm for nominalizations is summarized in Figure 1. The algorithm uses verb-argument tuples to infer the relation holding between the modifier and its nominalized head. When the co-occurrence frequency of the verb-argument relations is zero, verb-argument tuples are smoothed using one of the methods described in Section 3.

Once frequencies (either actual or reconstructed through smoothing) for verb-argument relations have been obtained, the RA score determines the relation between the head n_1 and its modifier n_2 (see Section 2). The sign of the RA score indicates which relation, subject or object, is more likely: a positive RA score indicates an object relation, whereas a negative score indicates a subject relation. Depending on the task and the data at hand, we can require that an object or subject analysis be preferred only if RA exceeds a certain threshold j (see steps 7 and 8 in Figure 1). We can also impose a threshold k on the type of verb-argument tuples we smooth. If, for instance, we know

Table 5
RA score for verb-argument tuples extracted from the BNC.

Verb-noun	$f(v_{n2},\textsc{obj},n_1)$	$f(v_{n2},\textsc{subj},n_1)$	*RA*
administer student	0	0	.96
establish unit	22	1	.55
promote government	3	10	−1.73

that the parser's output is noisy, then we might choose to smooth not only unseen verb-argument pairs but also pairs with nonzero corpus frequencies (e.g., $f(verb_{n_2}, rel, n_1)$ ≥ 1; see steps 3 and 4 in Figure 1).

Consider, for example, the compound *student administration*: its corresponding verb-noun configuration (e.g., *administer student*) is not attested in the BNC. This is a case in which we need smoothed estimates for both $f(v_{n2}$, $\textsc{obj}$, $n_1)$ and $f(v_{n2}$, $\textsc{subj}$, $n_1)$. The re-created frequencies using the class-based smoothing method described in Section 3.2 are 5.06 and 2.59, respectively, yielding an RA score of .96 (see Table 5), which means that it is more likely that *student* is the object of *administration*. Consider now the compound *unit establishment*: here, we have very little evidence in the corpus with respect to the verb-subject relation (see Table 5, where $f(establish$, $\textsc{subj}$, $unit) = 1)$. Assuming we have set the threshold *k* to 2 (see steps 4 and 5 in Figure 1) we need only re-create the frequency for the subject relation (e.g., 14.99 using class-based smoothing). The resulting *RA* score is again positive (see Table 5), which indicates that there is a greater probability for *unit* to be the object of *establishment* than for it to be the subject. Finally, consider the compound *government promotion*: counts for both subject and object relations are found in the BNC (see Table 5), in which case no smoothing is involved; we need only calculate the *RA* score (see step 6 in Figure 1), which is negative, indicating that *government* is more likely to be the subject of *promotion* than its object.

5. Experiments

5.1 Methodology

The algorithm described in the previous section and the smoothing variants were evaluated on the task of disambiguating nominalizations. As detailed above, the Jensen-Shannon divergence and confusion probability measures are parameterized. This means that we need to establish empirically the best parameter values for the size of the vocabulary (i.e., number of verbs used to find the nearest neighbors) and, for the Jensen-Shannon divergence, the effect of the β parameter. Recall from Section 2.2.2 that we obtained 796 nominalizations from the BNC. From these, 596 were used as training data for finding the optimal parameters for the two variants of distance-weighted averaging. The 596 nominalizations were also used to find the optimal thresholds for the interpretation algorithm. The remaining 200 nominalizations were retained as test data and also to evaluate whether human judges can reliably disambiguate the argument relation between the nominalized head and its modifier (see Experiment 1).

In Experiment 2 we investigate how the different smoothing techniques detailed in Section 3 influence the disambiguation task. As far as class-based smoothing is concerned, we experiment with two concept hierarchies, Roget's thesaurus and WordNet. Although no parameter tuning is necessary for class-based and back-off smoothing, we maintain the train/test data distinction also for these methods to facilitate comparisons with distance-weighted averaging.

We also examine whether knowledge of the semantics of the suffix of the nominalized head can improve performance. We run two versions of the algorithm presented in Section 4: in one version the algorithm assumes no prior knowledge about the semantics of the nominalization suffix (see Figure 1); in the other version the algorithm estimates the probabilities $P(\textsc{obj} \mid n_1, n_2)$ and $P(\textsc{subj} \mid n_1, n_2)$ only for compounds with nominalization suffixes other than *-er*, *-or*, *-ant*, or *-ee*. For compounds with suffixes *-er*, *-or* and *-ant* (e.g., *datum holder*, *car collector*, *water disinfectant*), the algorithm defaults to an object interpretation, and it defaults to a subject analysis for compounds with the suffix *-ee* (e.g., *university employee*). Compounds with heads ending in these four suffixes represented 13.6% of the compounds contained in the train set and 10.8% of the compounds in the test set.

In Experiment 3 we explore how the combination of the different smoothing methods influences disambiguation performance; we also consider context as an additional predictor of the argument relation of a deverbal head and its modifier and combine these distinct information sources using Ripper (Cohen 1996), a machine learning system that induces sets of rules from preclassified examples.

In what follows we briefly describe our study on assessing how well humans agree on disambiguating nominalizations. This study establishes an upper bound for the task against which our automatic methods will be compared. Sections 5.3 and 5.4 present our results on the disambiguation task.

5.2 Experiment 1: Agreement

Two graduate students in linguistics decided whether modifiers were the subject or object of a given nominalized head. The judges were given a page of guidelines but no prior training. The nominalizations were disambiguated in context: the judges were given the corpus sentence in which the nominalization occurred together with the previous and following sentence. We measured the judges' agreement using the kappa coefficient (Siegel and Castellan 1988), which is the ratio of the proportion of times $P(A)$ that k raters agree (corrected by chance agreement $P(E)$) to the maximum proportion of times the raters would agree (corrected for chance agreement):

$$K = \frac{P(A) - P(E)}{1 - P(E)} \tag{22}$$

If there is a complete agreement among the raters, then $K = 1$, whereas if there is no agreement among the raters (other than the agreement that would be expected to occur by chance), then $K = 0$.

The judges' agreement on the disambiguation task was $K = .78$ ($N = 200$, $k = 2$). This translates into a percentage agreement of 89.7%. Although the Kappa coefficient has a number of advantages over percentage agreement (e.g., it takes into account the expected chance interrater agreement; see Carletta (1996) for details), we also report percentage agreement as it allows us to compare straightforwardly the human performance and the automatic methods described below, whose performance will also be reported in terms of percentage agreement. Furthermore, percentage agreement establishes an intuitive upper bound for the task (i.e., 89.7%), allowing us to interpret how well our empirical models are doing in relation to humans.

Finally, note that the level of agreement was good, given that the judges were provided with minimal instructions and no prior training. Even though context was provided to aid the disambiguation task, however, the judges were not in complete

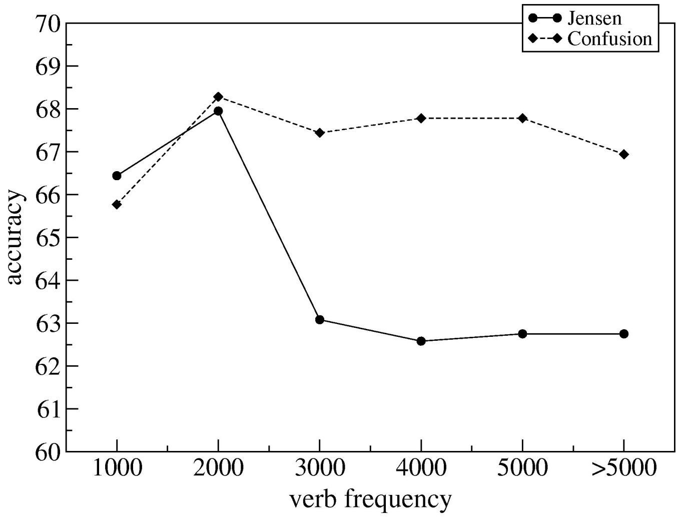

Figure 2
Disambiguation accuracy as the number of similar neighbors (i.e., number of verbs over which the similarity function is calculated) is varied for P_C and J.

agreement. This points to the intrinsic difficulty of the task at hand. Argument relations and consequently selectional restrictions are influenced by several pragmatic factors that may not be readily inferred from the immediate context (see Section 6 for discussion).

5.3 Experiment 2: Comparison of Smoothing Variants

Before reporting the results of the disambiguation task, we describe our initial experiments on finding the optimal parameter settings for the two distance-weighted averaging smoothing methods.

Figure 2 shows how performance on the disambiguation task varies with respect to the number and frequency of verbs over which the similarity function is calculated. The y-axis in Figure 2 shows how performance on the training set varies (for both P_C and J divergence) when verb-argument pairs are selected for the 1,000 most frequent verbs in the corpus, the 2,000 most frequent verbs in the corpus, etc. (x-axis). The best performance for both similarity functions is achieved with the 2,000 most frequent verbs. Furthermore, J and P_C yield comparable performances (68.0% and 68.3%, respectively under that condition). Another important observation is that performance deteriorates less severely for P_C than for J as the number of verbs increases: when all verbs for which verb-argument tuples are extracted from the BNC are used, the accuracy for P_C is 66.9%, whereas the accuracy for J is 62.8%. These results are perhaps unsurprising: verb-argument pairs with low-frequency verbs introduce noise due to the errors inherent in the partial parser. Table 6 shows the 10 closest words to the verb *accept* according to P_C as the number of verbs is varied: the quality of the closest neighbors deteriorates with the inclusion of less frequent verbs.

Finally, we analyzed the role of the parameter β. Recall that β appears in the weight function for the Jensen-Shannon divergence and controls the influence of the most similar words: the contribution of the closest neighbors increases with a high value for β. Figure 3 shows how the value of β affects performance on the disambiguation task when the similarity function is computed for the 1,000 and 2,000 most frequent verbs in the corpus. It is clear that performance is low with high or very low β values

Table 6
Ten closest words to verb *accept* for P_C.

Number of Most Frequent Verbs					
1,000	2,000	3,000	4,000	5,000	>5,000
accept	decline	decline	decline	decline	incl
refuse	accept	tender	tender	re-issued	decline
reject	refuse	accept	abdicate	co-manage	re-issued
submit	delegate	table	accept	tender	co-manage
endorse	reject	disclaim	table	oversubscribe	tender
approve	repudiate	plate	wangle	backdate	goodwill
issue	hitch	shirk	disclaim	abdicate	oversubscribe
implement	shoulder	refuse	plate	accept	pre-arrange
acknowledge	delegate	proffer	shirk	table	backdate
incur	ratify	apportion	disdain	wangle	abdicate

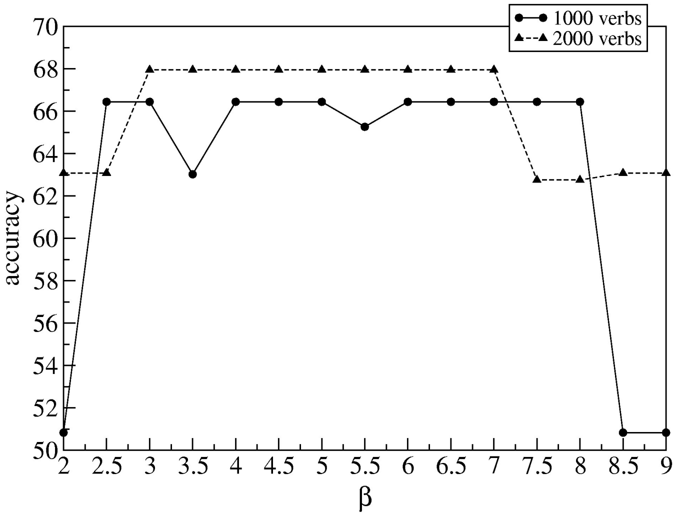

Figure 3
Disambiguation accuracy for *J* as β is varied for the 1,000 and 2,000 most frequent verbs in the BNC.

(e.g., $\beta \in \{2, 9\}$). We chose to set the parameter β to five, and the results shown in Figure 2 have been produced for this value for all verb frequency classes.

Table 7 shows how the three types of smoothing, back-off (*B*), class-based (using WordNet (*Wn*) and Roget (*Ro*)), and distance-weighted averaging (using confusion probability (P_C) and the Jensen-Shannon divergence (*J*)), influence performance in predicting the relation between a modifier and its nominalized head. For the distance-weighted averaging methods we report the results obtained with the optimal parameter settings ($\beta = 5$; 2,000 most frequent verbs). The results in Table 7 were obtained without taking the semantics of the nominalization suffix (*-er*, *-or*, *-ant*, *-ee*) into account (see Section 5.1).

Let us concentrate on the training set first. The back-off method is outperformed by all other methods, although its performance is comparable to that of class-based smoothing using Roget's thesaurus (63.1% and 65.1%, respectively). Distance-weighted averaging methods outperform concept-based methods, although not considerably (accuracy on the training set was 68.3% for P_C and 68.0% for class-based smoothing

Table 7
Disambiguation performance without nominalization suffixes.

Methods	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
B	63.1 ± 1.98	69.6 ± 3.31
P_C	68.3 ± 1.90	75.8 ± 3.08
J	68.0 ± 1.91	69.1 ± 3.33
Wn	68.0 ± 1.91	72.7 ± 3.20
Ro	65.1 ± 1.95	68.6 ± 3.34

Table 8
Disambiguation performance with nominalization suffixes.

Methods	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
B	67.5 ± 1.92	69.6 ± 3.31
P_C	70.6 ± 1.87	76.3 ± 3.06
J	69.0 ± 1.89	69.6 ± 3.31
Wn	70.5 ± 1.87	74.2 ± 3.15
Ro	67.5 ± 1.92	69.6 ± 3.31

using WordNet). Furthermore, the particular concept hierarchy used for class-based smoothing seems to have an effect on disambiguation performance: an increase of approximately 3.0% is obtained by using WordNet instead of Roget's thesaurus. One explanation might be that Roget's thesaurus is too coarse-grained a taxonomy for the task at hand. We used the chi-square statistic to examine whether the observed performance is better than the simple default strategy of always choosing an object relation, which yields an accuracy of 59.0% in the training data (see D in Table 7). The proportion of nominalizations classified correctly was significantly greater than 59.0% ($p < .01$) for all methods but back-off (B) and Roget (Ro).

Similar results are observed on the test set. Again P_C outperforms all other methods, achieving an accuracy of 75.8% (see Table 7). The portion of nominalizations classified correctly by P_C is significantly greater than 61.5% ($\chi^2 = 9.37$, $p < .01$), which is the percentage of object relations in the test set. The second-best method is class-based smoothing using WordNet (see Table 7). WordNet's performance is also significantly better ($\chi^2 = 5.64$, $p < .05$) than the baseline. The back-off method, class-based smoothing using Roget's thesaurus, and J yield comparable results (see Table 7).

Table 8 shows how each method performs when knowledge about the semantics of the nominalization suffix is taken into account. Recall that compounds with agentive and passive suffixes (i.e., *-er*, *-or*, *-ant*, and *-ee*) represent 13.6% of the training data and 10.8% of the test data. A general observation is that knowledge of the semantics of the nominalization suffix does not dramatically influence accuracy. Performance on the test data increases 1.5% for Wn, 1.0% for Ro and 0.5% for distance-weighted averaging (see J and P_C in Table 8). We observe no increase in performance for back-off smoothing (see Tables 7 and 8). These results suggest that the nominalization suffixes do not contribute much additional information to the interpretation task, as their meaning can be successfully retrieved from the corpus.

An interesting question is the extent to which any of the different methods agree in their assignments of subject and object relations. We investigated this by calculating the methods' agreement on the training set using the Kappa coefficient. We calculated the Kappa coefficient for all pairwise combinations of the five smoothing variants. The results are reported in Table 9. The highest agreement is observed for P_C and the class-based smoothing using the WordNet taxonomy ($K = .75$). Agreement between J and P_C as well as agreement between Wn and Ro is rather low ($K = .53$ and $K = .46$, respectively). Note that generally low agreement is observed when B is paired with J, P_C, Wn, or Ro. This is not entirely unexpected, given the assumptions underlying the different smoothing techniques. Both class-based and distance-weighted averaging methods recreate the frequency of unseen word combinations by relying on corpus

Table 9
Agreement between smoothing methods.

	B	*J*	P_C	Wn
J	.31			
P_C	.26	.53		
Wn	.01	.37	.75	
Ro	.25	.26	.49	.46

Table 10
Performance at predicting argument relations.

	Train (%)		Test (%)	
Methods	SUBJ	OBJ	SUBJ	OBJ
B	41.6	78.1	38.0	87.8
P_C	47.4	82.9	54.9	87.8
J	34.7	91.2	35.2	88.6
Wn	47.8	82.1	49.3	86.2
Ro	50.6	74.4	46.5	81.3

evidence for words that are distributionally similar to the words of interest. In distance-weighted averaging smoothing, word similarity is estimated from lexical co-occurrence information, whereas in taxonomic class-based smoothing, similarity emerges from the hierarchical organization of conceptual information. Back-off smoothing, however, incorporates no notion of similarity: unseen sequences are estimated using not similar conditional distributions, but lower-level ones. This also relates to the fact that *B*'s performance is lower than *Wn* and P_C (see Table 7), which suggests that smoothing methods that incorporate linguistic hypotheses (i.e., the notion of similarity) perform better than methods relying simply on co-occurrence distributions. To summarize, the agreement values in Table 9 suggest that methods inducing similarity relationships from corpus co-occurrence statistics are not necessarily incompatible with methods that quantify similarity using manually crafted taxonomies and that different smoothing techniques may be appropriate for different tasks.

Table 10 shows how the different methods compare for the task of predicting the individual argument relations for the training and test sets. A general observation is that all methods are fairly good at predicting object relations. Predicting subject relations is considerably harder: no method exceeds an accuracy of 54.9% (see Table 10). One explanation for this is that selectional constraints imposed on subjects can be more easily overridden by pragmatic and contextual factors than those imposed on objects. Furthermore, selectional constraints on subjects are normally weaker than on objects. *J* is particularly good at predicting object relations, whereas P_C yields the best performance when it comes to predicting subject relations (see Table 10).

5.4 Experiment 3: Using Ripper to Disambiguate Nominalizations

An obvious question is whether a better performance can be achieved by combining the five smoothing variants, given that they seem to provide complementary information for predicting argument relations. For example, *Wn*, *Ro*, and P_C are relatively good for the prediction of subject relations , whereas *J* is best for the prediction of object relations (see Table 10). Furthermore, note that the probabilistic model introduced in Section 2 and the algorithm based on it (see Section 4) ignore contextual information that can provide important cues for disambiguating nominalizations. Consider the nominalization *government promotion* in (23a), which was assigned an object (instead of a subject) interpretation by all smoothing variants except *Wn*. Contextual information could help assign the correct interpretation in cases in which the head of the compound is followed by prepositions such as *of* (see (23a)) or *into* (see (23b)).

(23) a. It was not felt necessary to take account of **government promotion** of unionism.

b. But politicians are calling for the Republic's Government to start a **Court inquiry** into Ross' alleged links with firms in Eire.

In the following we first examine whether combination of the five smoothing variants improves performance at predicting the argument relations for nominalizations (see Section 5.4.1). We then proceed to study the influence of context on the interpretation task; we explore the contribution of context alone (see Section 5.4.2) and in combination with the different smoothing variants (see Section 5.4.3). The different information sources are combined using Ripper (Cohen 1996), a system that induces classification rules from a set of preclassified examples. Ripper takes as input the classes to be learned (in our case the classification is binary, i.e., subject or object), the names and possible values of a set of features, and training data specifying the class and feature values for each training example. In our experiments the features are the smoothing variants and the tokens surrounding the nominalizations in question. The feature vector in (24a) represents the individual predictions of B, Wn, Ro, J, and P_C for the interpretation of *government promotion* (see (23a)). We encode the context surrounding nominalizations using two distinct representations: (a) parts of speech and (b) lemmas. In both cases we encode the position of the tokens with respect to the nominalization in question. The feature vector in (24b) consists of the nominalization *court inquiry* (see (23b)), represented by its parts of speech (NN1 and NN1, respectively) and a context of five words to its right and five words to its left, also reduced to their parts of speech. In (24c) the same tokens are represented by their lemmas.

(24) a. [OBJ, SUBJ, OBJ, OBJ, OBJ]

b. [POS, NN0, TO0, VVI, AJ0, NN1, NN1, PRP, POS, AJ0, NN2, PRP]

c. ['s government to start a court inquiry into Ross 's alleged link]

Ripper is trained on vectors of values like the ones presented in (24) and outputs a classification model for classifying future examples. The model is learned using greedy search guided by an information gain metric and is expressed as an ordered set of if-then rules. For our experiments Ripper was trained on the 596 nominalizations on which the smoothing methods were compared and tested on the 200 unseen nominalizations for which the interjudge agreement was previously calculated (see Section 5.2).

5.4.1 Combination of Smoothing Variants. Table 11 shows Ripper's performance when different combinations of smoothing variants (i.e., features) are used without taking context into account. All results in Table 11 were obtained using the version of the interpretation algorithm that takes suffix semantics into account (see Section 5.3). As shown in Table 11, the combination of all five smoothing variants achieves a performance of 80.4%.[2] Table 11 further reports the accuracy achieved when removing

[2] An anonymous reviewer pointed out that suffix information could be alternatively exploited by including the ending suffix of the nominalization head as an additional feature for the classification task. The latter approach yields comparable performance to our original idea of defaulting to the argument structure denoted by the nominalization suffix. When B, J, P_C, Ro, and Wn are used as features together with nominalization suffixes (*-age*, *-ion*, *-ment*, etc.), Ripper's performance is 79.9% ± 1.65 on the training data and 80.3% ± 2.95% on the test data.

Table 11
Disambiguation performance using the smoothing variants as features.

Features	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
B, *J*, P_C, *Ro*, *Wn*	80.2 ± 1.63	80.4 ± 2.86
B, *J*, P_C, *Wn*	80.2 ± 1.68	80.4 ± 2.88
B, *J*, P_C, *Ro*	78.5 ± 1.68	79.9 ± 2.88
B, *J*, *Wn*, *Ro*	80.7 ± 1.62	79.9 ± 2.88
J, P_C, *Ro*, *Wn*	80.7 ± 1.62	78.4 ± 2.96
B, P_C, *Wn*, *Ro*	79.8 ± 1.64	74.7 ± 3.13

Table 12
Ripper's performance at predicting argument relations.

	Train (%)		Test (%)	
Features	SUBJ	OBJ	SUBJ	OBJ
B, *J*, P_C, *Ro*, *Wn*	66.5	89.7	73.2	84.6
B, *J*, P_C, *Wn*	66.5	89.7	73.2	84.6
B, *J*, *Wn*, *Ro*	71.4	87.2	78.9	80.5
B, *J*, P_C, *Ro*	71.4	87.2	78.9	80.5
J, P_C, *Ro*, *Wn*	69.4	88.6	71.8	82.1
B, P_C, *Wn*, *Ro*	63.3	91.5	50.7	88.6

a single feature. Evaluation on subsets of features allows us to explore the contribution of individual features to the classification task by comparing the subsets to the full feature set. We see that removal of *Ro* has no effect on the results, whereas removal of *J* produces a 5.7% performance decrease. Removing *Wn* or P_C yields the same decrease in performance (i.e., 0.5%). This is not surprising, since P_C and *Wn* tend to agree in their assignments of subject and object relations (see the methods' agreement in Table 9), and therefore their combination is not expected to be very informative. Absence of *J* from the feature set yields the most dramatic performance decrease. This is not unexpected, given that *J* is the best predictor for object relations and that P_C and WordNet behave similarly with respect to their interpretation decisions. In general we observe that the combination of smoothing variants outperforms their individual performances (compare Tables 11 and 8). Comparison of Ripper's best performance (80.4%) against the individual smoothing methods reveals a 10.8% accuracy increase over *B*, *J*, and *Ro*, a 4.1% increase over P_C, and a 6.2% increase over *Wn*.

We further analyzed Ripper's performance at predicting object and subject relations. This information is displayed in Table 12, in which we show how performance varies on the full set of n size features (i.e., five) and each of its $n-1$ size subsets. As can be seen in Table 12, accuracy at predicting subject relations increases when smoothing variants are combined (compare Tables 12 and 10). In fact, combination of *B*, *J*, *Wn*, and *Ro* (or *B*, *J*, P_C, and *Ro*) performs best at predicting subject relations, achieving an increase of 24% over P_C, the best individual predictor for subject relations (see Table 10). In sum, our results show that combination of the different smoothing variants (using Ripper) achieves better results than each individual method. Our overall performance (i.e., 80.4%) outperforms the default baseline significantly, by 18.9% ($\chi^2 = 17.33$, $p < .05$) and is 9.3% lower than the upper bound established in our agreement study (see Section 5.2). In what follows we first examine the independent contribution of context to the disambiguation performance and then turn to its combination with our five smoothing variants.

5.4.2 The Contribution of Context. We evaluated the influence of context by varying both the position and the size of the window of tokens (i.e., lemmas or parts of speech) surrounding the nominalization. We varied the window size parameter between one and five words before and after the nominalization target. We use the symbols l and r for left and right context, respectively, subscripts to denote the context encoding (i.e., lemmas or parts of speech), and numbers to express the size of the window

Table 13
Disambiguation performance using right context encoded as lemmas.

Features	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
$r_l = 1$	70.8 ± 1.86	68.0 ± 3.36
$r_l = 2$	70.1 ± 1.88	68.6 ± 3.34
$r_l = 3$	68.8 ± 1.90	67.5 ± 3.37
$r_l = 4$	68.8 ± 1.90	67.5 ± 3.37
$r_l = 5$	68.8 ± 1.90	67.5 ± 3.37

Table 14
Disambiguation performance using left content encoded as lemmas.

Features	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
$l_l = 1$	66.9 ± 1.93	64.9 ± 3.43
$l_l = 2$	70.5 ± 1.87	67.5 ± 3.37
$l_l = 3$	70.6 ± 1.87	67.0 ± 3.83
$l_l = 4$	67.8 ± 1.92	65.5 ± 3.42
$l_l = 5$	65.3 ± 1.95	63.9 ± 3.46

Table 15
Disambiguation performance using right context encoded as POS tags.

Features	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
$r_p = 1$	64.9 ± 1.96	65.5 ± 3.42
$r_p = 2$	65.8 ± 1.95	62.4 ± 3.49
$r_p = 3$	64.4 ± 1.96	63.4 ± 3.47
$r_p = 4$	65.3 ± 1.95	63.4 ± 3.47
$r_p = 5$	65.9 ± 1.94	62.9 ± 3.48

Table 16
Disambiguation performance using left content encoded as POS tags.

Features	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
$l_p = 1$	63.9 ± 1.97	66.0 ± 3.41
$l_p = 2$	68.1 ± 1.91	64.4 ± 3.45
$l_p = 3$	67.1 ± 1.93	66.5 ± 3.40
$l_p = 4$	65.6 ± 1.95	65.0 ± 3.43
$l_p = 5$	66.6 ± 1.93	61.9 ± 3.50

surrounding the candidate compound. For example, $l_l = 5$ represents a window of five tokens, encoded as lemmas, to the left of the candidate compound.

Tables 13 and 14 show the influence of right and left context, respectively, represented as lemmas. The best peformances are achieved with a window of two words to the right or left of the candidate nominalization (see the features $r_l = 2$ and $l_l = 2$ in Tables 13 and 14, respectively). Combination of the best left and right features ($r_l = 2$, $l_l = 2$) does not increase the disambiguation performance (70.4% ± 1.86% on the training and 66.5% ± 3.41% on the test data). Note that the disambiguation performance simply using contextual features is not considerably worse than the performance of some smoothing variants (see Table 7). Contextual features encoded as lemmas outperform part-of-speech (POS) tags, for which the best performance is achieved with a window of one token to the right or a window of three tokens to the left of the candidate nominalization (see Tables 15 and 16). As in the case of lemmas, combination of the best left and right features ($r_p = 1$, $l_p = 3$) does not yield better results (66.3% ± 1.94% on the training data and 66.5% ± 3.40% on the test data). The lower performance of POS tags is not entirely unexpected: lemmas capture lexical dependencies that are somewhat lost when a more general level of representation is introduced. For example, Ripper assigns a subject interpretation when *for* immediately follows a nominalization head (e.g., *staff requirement for reconnaissance*). This rule cannot be induced when *for* is represented by its part of speech (e.g., PRP), as there are a number of prepositions that can follow the nominalization head, but only a few of them provide cues for its argument structure.

Table 17 shows the performance of the best contextual features for the task of predicting the individual argument relations. The contextual features are consistently better at predicting object than subject relations. This is not surprising, given that ob-

Table 17
Performance at predicting argument relations using context.

	Train (%)		Test (%)	
Methods	SUBJ	OBJ	SUBJ	OBJ
$r_l = 2$	28.0	99.2	20.8	96.7
$l_l = 2$	36.2	94.1	13.8	97.5
$l_p = 3$	33.7	90.1	29.1	88.5
$r_p = 1$	22.6	94.1	20.8	91.8

ject relations represent the majority in both the training and test data; furthermore, identifying superficial features that are good predictors for subject relations is a relatively hard task. For example, even though Ripper identifies prepositions (e.g., *of*, *to*) following the nominalization head and certain frequent nominalization heads (e.g., *behavior*) as indicators of subject relations, it has no means of guessing the transitivity of deverbal heads in the absense of syntactic cues. Consider example (25a), in which neither left nor right context is informative with regard to the fact that *intervene* is intransitive.

Finally, there are some cases in which the syntactic cues can be misleading, as adjacency to the nominalization target does not necessarily indicate argument structure. This is shown in (25b), in which *youth* is classified as the subject of *manager*. Although on the surface *youth manager at* is analogous to nominalizations followed by *of* (e.g., *government promotion of*), the prepositional phrase *at Wimbledon* in (25b) is simply locative and not the argument of *manager*.

(25) a. If the second reminder produces no result or the reply to either reminder seems to indicate the need for **court intervention** the matter will be referred to a master or district judge.

b. He was **youth manager** at Wimbledon when I held a similar position at Palace.

5.4.3 Combination of Context with Smoothing Variants. In this section we investigate whether the combination of surface contextual features with the predictions of the different smoothing methods has an effect on the disambiguation performance. Although context is good at predicting object relations, it performs poorly at guessing subject relations (see Table 17). We expect the combination of context with smoothing variants (some of which, e.g., *Wn*, *Ro*, and P_C, perform relatively well at the predicting subject relations) to improve performance. Recall that the probabilistic model introduced in Section 2.1 and the interpretation algorithm that makes use of it attempt the interpretation of nominalizations without taking contextual cues into account. Here, we examine how well the different smoothing variants perform in the presence of contextual information. Table 18 shows Ripper's performance when the best context (i.e., $r_l = 2$) is combined with a single smoothing method and with all five variants. For the smoothing variants, we used the version of the interpretation algorithm that takes suffix semantics into account (see Table 8).

Comparison between Tables 8 and 18 reveals that the inclusion of context generally increases performance. Combination of B with the best context yields a 6.7% increase over B; an increase of 8.8% (over J) and 7.7% (over *Ro*) is observed when J and *Ro* are combined with context, respectively. No increase in performance is observed when

Table 18
Disambiguation performance using context and smoothing variants.

Methods	Train (%)	Test (%)
D	59.0 ± 2.01	61.5 ± 3.50
$r_l = 2, B$	78.2 ± 1.69	76.3 ± 3.06
$r_l = 2, P_C$	75.0 ± 1.78	76.3 ± 3.06
$r_l = 2, J$	81.5 ± 1.59	78.4 ± 2.96
$r_l = 2, Wn$	88.9 ± 1.29	86.1 ± 2.49
$r_l = 2, Ro$	78.5 ± 1.68	77.3 ± 3.00
$B, J, P_C, Ro, Wn, r_l = 2$	84.4 ± 1.49	85.1 ± 2.57

Table 19
Argument relations using context and smoothing variants.

	Train (%)		Test (%)	
Methods	SUBJ	OBJ	SUBJ	OBJ
$r_l = 2, B$	69.9	83.6	61.3	85.7
$r_l = 2, P_C$	63.9	82.2	54.9	88.6
$r_l = 2, J$	72.9	87.2	66.7	85.7
$r_l = 2, Wn$	87.3	90.0	74.7	93.3
$r_l = 2, Ro$	69.1	84.7	64.0	85.7
$B,\ J,\ P_C,\ Ro,\ Wn,\ r_l = 2$	75.0	90.6	72.0	93.3

context is combined with P_C (see Table 18), whereas combination of *Wn* with context yields a 11.9% increase over *Wn* alone. Combining all five smoothing variants with context yields an increase of 4.7% over just the combination of *B*, *J*, P_C, *Ro*, and *Wn* (see Table 12). Our best performance (i.e., 86.1%) is achieved when *Wn* is combined with right context ($r_l = 2$); this performance is significantly better than the simple strategy of always defaulting to a subject classification, which yields an accuracy of 61.5% ($\chi^2 = 30.64$, $p < .05$), and only 3.6% lower than the upper bound of 89.7%.

As shown in Table 19, the inclusion of context increases accuracy when it comes to the prediction of subject relations (with the exception of P_C, which is relatively good at predicting subject relations, and therefore in that case the inclusion of context does not add much useful information). The combination of *Wn* with $r_l = 2$ achieves the highest accuracy (87.3%) at predicting subject relations.

6. Discussion

We have described an empirical approach for the automatic interpretation of nominalizations. We cast the interpretation task as a disambiguation problem and proposed a statistical model for inferring the argument relations holding between a deverbal head and its modifier. Our experiments revealed that the interpretation task suffers from data sparseness: even an approximation that maps the nominalized head to its underlying verb does not provide sufficient evidence for quantifying the argument relation of a modifier noun and its nominalized head.

We showed how the argument relations (which are not readily available in the corpus) can be retrieved by using partial parsing and smoothing techniques that exploit distributional and taxonomic information. We compensated for the lack of sufficient distributional information using either methods that directly recreate the frequencies of word combinations or contextual features whose distribution in the corpus indirectly provides information about nominalizations. We compared and contrasted a variety of smoothing approaches proposed in the literature and demonstrated that their combination yields satisfactory results for the demanding task of semantic disambiguation. We also explored the contribution of context and showed that it is useful for the disambiguation task. Our approach is applicable to domain-independent unrestricted text and does not require the hand coding of semantic information. In the following sections we discuss our results and their potential usefulness for NLP applications. We also address the limitations of our approach and sketch potential extensions.

6.1 The Interpretation of Nominalizations

Our results indicate that a simple probabilistic model that uses smoothed counts (see the interpretation algorithm in Section 4) yields a significant increase over the baseline without taking context into account. Distance-weighted smoothing using P_C and class-based smoothing using WordNet achieve the best results (76.3% and 74.2%, respectively). Combination of different smoothing methods (using Ripper) yields an overall performance of 80.4%, again without taking context into consideration. Context alone achieves a disambiguation performance of 68.6%, approximating the performance of some of the smoothing variants (see Tables 9 and 13). This result suggests that simple features that can be easily retrieved and estimated from the corpus contain enough information to capture generalizations about the behavior of nominalizations. As expected, the combination of smoothed probabilities with context outperforms the accuracy of individual smoothing variants. The combination of WordNet with a right context of size two achieves an accuracy of 86.1%, compared to an upper bound for the task (i.e., intersubject agreement) of 89.7%. This is an important result considering the simplifications in the system and the sparse data problems encountered in the estimation of the model probabilities. The second-best performance is achieved when J is combined with context (78.4%; see Table 18). This result shows that information inherent in the corpus can make up for the lack of distributional evidence and furthermore that it is possible to extract semantic information from corpora (even if they are not semantically annotated in any way) without recourse to pre-existing taxonomies such as WordNet.

6.2 Limitations and Extensions

To a certain extent the difficulty of interpreting nominalizations is due to their context dependence. Although the approach presented in the previous sections takes immediate context into account, it does so in a shallow manner, without having access to the meaning of the words surrounding the nominalization target, their syntactic dependencies, or the general discourse context within which the compound is embedded. Consider example (26a), in which the compound *computer guidance* receives a subject interpretation (e.g., the computer guides the chef). Our approach cannot detect that the *computer* here is ascribed animate qualities and opts for the most likely interpretation (i.e., an object analysis). In some cases the modifier stands in a metonymic relation to its head. Consider the examples in sentences (26b, 26c), in which the nominalizations *industry reception* and *market acceptance* can be thought of as instances of the metonymic schema "whole for part" (Lakoff and Johnson 1980). In example (26b) it is the industry as a whole that receives the guests rather than LASMO, which is one of its parts, whereas in (26c) the modifier *market* in *market acceptance* refers to the opinion leaders, who are part of the market.

(26) a. Of course, none of this means that the equipment is taking anything away from the chef's own individual skills which are irreplaceable. What it does ensure is that the chef has complete control over some of the most vital tools of his trade, with **computer guidance** as an important aid.

b. The final evening saw more than 300 guests attend an **industry reception**, hosted by LASMO.

c. Marketers interested in the development and introduction of new products will be particularly interested in the attitude of opinion leaders to these products, for their general **market acceptance** can be slowed down or speeded up by the views of such people.

Consider now sentence (27a). The nominalization *student briefing* is ambiguous, even though it is presented within its immediate context. Taking more context into account (see (27a)) does not provide enough disambiguation information either, although perhaps it introduces a slight bias in favor of an object interpretation (i.e., someone is briefing the students). For this particular example, we would have to know what the document within which *student briefing* occurs is about (i.e., a list of teaching guidelines for university lecturers). The sentences in (27) are taken from a document section entitled "Work Experience" that emphasizes the importance of work experience for students. Given all this background information, it becomes apparent that it is not the students who are doing the briefing in (27b).

(27) a. Explain to both students and organisations the role of work experience in personal development and its part in the planned programme.

b. Provide comprehensive guidelines on the work experience which includes a **student briefing**, an employer briefing and a student work checklist.

The observation that discourse or pragmatic context may influence interpretations is by no means new or particular to nominalisations. Sparck Jones (1983) observes that a variety of factors can potentially influence the interpretation of compound nouns in general. These factors range from syntactic analysis (e.g., to arrive at an interpretation of the compound *onion tears*, it is necessary to identify that *tears* is a noun and not the third-person singular of the verb *tear*) to semantic information (e.g., for interpreting *onion tears*, it is important to know that onions cannot be tears or that tears are not made of onions) and pragmatic information. Pragmatic inference may be called for in cases in which syntactic or semantic information is straightforwardly supplied, even where the local text context provides rich information bearing on the interpretation of the compound. Copestake and Lascarides (1997) and Lascarides and Copestake (1998) make the same observation for a variety of constructions such as compound nouns, adjective-noun combinations and verb-argument relations. Consider the sentences in (28)–(30). The discourse in (28) favors the interpretation "bag for cotton clothes" for *cotton bag* over the more likely interpretation "bag made of cotton." Although *fast programmer* is typically a programmer who programs fast, when the adjective-noun combination is embedded in a context like (29a, 29b), the less likely meaning "a programmer who runs fast" is triggered. Finally, although it is more likely to enjoy reading a book rather than eating it, the context in (30) triggers the latter interpretation.

(28) a. Mary sorted her clothes into various bags made from plastic.

b. She put her skirt into the **cotton bag**.

(29) a. All the office personnel took part in the company sports day last week.

b. One of the programmers was a good athlete, but the other was struggling to finish the courses.

c. The **fast programmer** came first in the 100m.

(30) a. My goat eats anything.

b. He really **enjoyed your book**.

Pragmatic context may be particularly important for the interpretation of compound nouns. Because compounds can be used as a text compression device (Marsh 1984), it is plausible that pragmatic inference is required to supply the compound's interpretation. This observation is somewhat supported by our interannotator agreement experiment (see Section 5.2). Even though our participants were provided with some context, the agreement among them was not complete (they reached a K of .78, when absolute agreement is 1). Although our approach takes explicit contextual information into account, it is agnostic to discourse or pragmatic information. Encoding pragmatic information would involve considerable manual effort. Furthermore, a hypothetical statistical learner that takes pragmatic information into account would have not only to deal with data sparseness but furthermore to detect cases in which conflicts arise between discourse information and the likelihood of a given interpretation.

Our experiments focused on nominalizations derived from verbs specifically subcategorizing for direct objects. Although nominalizations whose verbs take prepositional frames (e.g., *oil painting*, *soccer competition*) represent a small fraction of the nominalizations found in the corpus (9.2%), a more general approach would have to take those verbs into account. This task is harder than interpreting direct objects, since to estimate the frequency $f(v_{n_2}, rel, n_1)$, one needs first to determine with some degree of accuracy the attachment site of the prepositional phrase. Taking into account prepositional phrases and their attachment sites can also be useful for the interpretation of compounds other than nominalizations. Consider the compound noun *pet spray* from (1). Assuming that *pet spray* can be "spray for pets," "spray in pets," "spray about pets," or "spray from pets," we can derive the most likely interpretation by looking at which types of prepositional phrases (e.g., *for pets*, *about pets*) are most likely to attach to *spray*. Note that in cases in which the expressions *spray for pets* and *spray in pets* are not attested in the corpus, their respective co-occurrence frequencies can be re-created using the techniques presented in Section 3.

Finally, the approach advocated here can be straightforwardly extended to nominalizations with adjectival modifiers (e.g., *parental refusal*; see the examples in (2)). In most cases the adjective in question is derived from a noun, and any inference process on the argument relations between the head noun and the adjectival modifier could take advantage of this information.

6.3 Relevance for NLP Applications

Robust semantic ambiguity resolution is challenging for current NLP systems. Although general-purpose taxonomies like WordNet or Roget's thesaurus are useful for certain interpretation tasks, such resources are not exhaustive or generally available for languages other than English. Furthermore, the compound noun interpretation task involves acquiring semantic information that is linguistically implicit and therefore not directly available in corpora or taxonomic resources. Indeed, interpreting compound nouns is often analyzed in the linguistics literature in terms of (impractical) general-purpose reasoning with pragmatic information such as real-world knowledge (e.g., Hobbs et al. 1993; see Section 7 for details). We show that it is feasible to learn implicit semantic information automatically from the corpus by utilizing linguistically principled approximations, surface syntactic cues, and (when available) taxonomic information.

The interpretation of compound nouns is important for several NLP tasks, notably machine translation. Consider the nominalization *satellite observation* (taken from (4a)), which may mean "observation by satellite" or "observation of satellites." To translate *satellite observation* into Spanish, we have to work out whether *satellite* is the subject or object of the verb *observe*. In the first case *satellite observation* translates as *observación por satelite* (observation by satellite), whereas in the latter it translates as *observación de satelites* (observation of satellites). In this case the implicit linguistic information has to be retrieved and disambiguated to obtain a meaningful translation. Information retrieval is another relevant application in which again the underlying meaning must be rendered explicit. Consider a search engine faced with the query *cancer treatment*. Presumably one would not like to obtain information about *cancer* or *treatment* in general, but about methods or medicines that help treat cancer. So knowledge about the fact that *cancer* is the object of *treatment* could help rank relevant documents (i.e., documents in which *cancer* is the object of the verb *treat*) before nonrelevant ones or restrict the number of retrieved documents.

7. Related Work

In this section we review previous work on the interpretation of compound nouns and compare it to our own work. Despite the differences among them, most approaches require large amounts of hand-crafted knowledge, place emphasis on the recovery of relations other than nominalizations (see the examples in (1)), contain no quantitative evaluation (the exceptions are Leonard (1984), Vanderwende (1994), and Lauer (1995)), and generally assume that context dependence is either negligible or of little impact. Most symbolic approaches are limited to a specific domain because of the large effort involved in hand-coding semantic information and are distinguished in two main types: concept-based and rule-based.

Under the concept-based approach, each noun in the compound is associated with a concept and various slots. Compound interpretation reduces to slot filling, that is, evaluating how appropriate concepts are as fillers of particular slots. A scoring system evaluates each possible interpretation and selects the highest-scoring analysis. Examples of the approach are Finin (1980) and McDonald (1982). As no qualitative evaluation is reported in these studies, it is difficult to assess how their methods perform, although it is clear that considerable effort needs to be invested in the encoding of the appropriate semantic knowledge.

Under the rule-based approach, interpretation is performed by sequential rule application. A fixed set of rules is applied in a fixed order, and the first rule that is semantically compatible with the nouns forming the compound results in the most plausible interpretation. The approach was introduced by Leonard (1984), was based on a hand-crafted lexicon, and achieved an accuracy of 76.0% (on the training set). Vanderwende (1994) further developed a rule-based algorithm that does not rely on a hand-crafted lexicon but extracts the required semantic information from an on-line dictionary instead. The system achieved an accuracy of 52.0%.

A variant of the concept-based approach uses unification to constrain the semantic relations between nouns represented as feature structures. Jones (1995) used a typed graph–based unification formalism and default inheritance to specify features for nouns whose combination results in different interpretations. Again no evaluation is reported, although Jones points out that ambiguity can be a problem, as all possible interpretations are produced for a given compound. Wu (1993) provides a statistical framework for the unification-based approach and develops an algorithm for approximating the probabilities of different possible interpretations using the maximum-entropy principle. No evaluation of the algorithm's performance is given. The approach remains knowledge intensive, however, as it requires manual construction of the feature structures.

Lauer (1995) provides a probabilistic model of compound noun paraphrasing (e.g., *state laws* are "the laws of the state," *war story* is "a story about war") that assigns probabilities to different paraphrases using a corpus in conjunction with Roget's thesaurus. Lauer does not address the interpretation of nominalizations or compounds with hyponymic relations (see example (1e)) and takes into account only prepositional paraphrases of compounds (e.g., *of*, *for*, *in*, *at*, etc.). Lauer's model makes predictions about the meaning of compound nouns on the basis of observations about prepositional phrases. The model combines the probability of the modifier given a certain preposition with the probability of the head given the same preposition and assumes that these two probabilities are independent.

Consider, for instance, the compound *war story*. To derive the intended interpretation (i.e., "story about war"), the model takes into account the frequency of *story about* and *about war*. For the modifier and head noun are substituted the concepts with which they are represented in Roget's thesaurus, and the frequency of a concept and a preposition is calculated accordingly (see Section 3.2). Lauer's (1995) model achieves an accuracy of 47.0%. The result is difficult to interpret, given that no experiments with humans are performed and therefore the optimal performance on the task is unknown. Lauer acknowledges that data sparseness can be a problem for the estimation of the model parameters and also that the assumption of independence between the head and its modifier is unrealistic and leads to errors in some cases.

Although it is generally acknowledged that context, both intra- and intersentential, may influence the interpretation task, contextual factors are typically ignored, with the exception of Hobbs et al. (1993), who propose that the interpretation of a compound can be achieved via abductive inference. To interpret a compound one must prove the logical form of its constituent parts from what is mutually known. The amount of world knowledge required to work out what is mutually known, however, renders such an approach infeasible in practice. Furthermore, Hobbs et al.'s approach does not capture linguistic constraints on compound noun formation and as a result cannot predict that a noun-noun sequence like *cancer lung* (under the interpretation "cancer in the lung") is odd.

Unlike previous work, we did not attempt to recover the semantic relations holding between a head and its modifier (see (1)). Instead, we focused on the less ambitious task of interpreting nominalizations, that is, compounds whose heads are derived from a verb and whose modifiers are interpreted as its arguments. Similarly to Lauer (1995), we have proposed a simple probabilistic model that uses information about the distributional properties of words and domain-independent symbolic knowledge (i.e., WordNet, Roget's thesaurus). Unlike Lauer, we have addressed the sparse-data problem by directly comparing and contrasting a variety of smoothing approaches proposed in the literature and have shown that these methods yield satisfactory results for the demanding task of semantic disambiguation. Furthermore, we have shown that the combination of different sources of taxonomic and nontaxonomic information (using Ripper) is effective for tasks facing data sparseness. In contrast to previous approaches, we explored the effect of context on the interpretation task and showed that its inclusion generally improves disambiguation performance. We combined different information sources (e.g., contextual features and smoothing variants) using Ripper. Although the use of classifiers has been widespread in studies concerning discourse segmentation (Passonneau and Litman 1997), the disambiguation of discourse cues (Siegel and McKeown 1994), the acquisition of lexical semantic classes (Merlo and Stevenson 1999; Siegel 1999), the automatic identification of user corrections in spoken dialogue systems (Hirschberg, Litman, and Swerts 2001), and word sense disambiguation (Pedersen 2001), the treatment of the interpretation of compound nouns as a classification task is, to our knowledge, novel.

Our approach can be easily adapted to account for Lauer's (1995) paraphrasing task. Instead of assuming that the probability of the compound modifier given a preposition is independent from the probability of the compound head given the same preposition, a more straightforward model would take into account the joint probability of the head, the preposition, and the modifier. In cases in which a certain head, preposition, and modifier combination is not attested in the corpus (e.g., *story about war*), the methodology put forward in Experiments 2 and 3 could be used to re-create its frequency (see also the discussion in Section 6).

Unlike previous approaches, we provide an upper bound for the task. Recall from Section 5.2 that an experiment with humans was performed to evaluate whether the task can be performed reliably. In doing so we took context into account, and as a result we established a higher upper bound for the task than would have been the case if context was not taken into account. Furthermore, it is not clear whether subjects could arrive at consistent interpretations for nominalizations out of context. Downing's (1977) experiments show that, when asked to interpret compounds out of context, participants tend to come up with a variety of interpretations that are not always compatible. For example, for the compound *bullet hole*, the interpretations "a hole made by a bullet," "a hole shaped like a bullet," "a fast-moving hole," "a hole in which to hide bullets," and "a hole into which to throw (bullet) casings" were provided.

8. Conclusions

In this article we presented work on the automatic interpretation of nominalizations (i.e., compounds whose heads are derived from a verb and whose modifiers are interpreted as its arguments). Nominalizations pose a challenge for empirical approaches, as the argument relations between a head and its modifier are not readily available in a corpus, and therefore they have to be somehow retrieved and approximated. Approximating the nominalized head to its corresponding verb and estimating the frequency of verb-noun relations instead of noun-noun relations accounts for only half of the nominalizations attested in the corpus.

Our experiments revealed that data sparseness can be overcome by taking advantage of smoothing methods and surface contextual information. We have directly compared and contrasted a variety of smoothing approaches proposed in the literature and have shown that these methods yield satisfactory results for the demanding task of semantic disambiguation, especially when coupled with contextual information. Our experiments have shown that contextual information that is easily obtainable from a corpus and computationally cheap is good at predicting object relations, whereas the computationally more expensive smoothing variants are better at guessing subject relations. Combination of context with smoothing variants yields better performance over either context or smoothing alone.

We combined different information sources (i.e., contextual features and smoothing variants) using Ripper. Although a considerable body of previous research has treated several linguistic phenomena as classification tasks, the interpretation of compound nouns has so far been based on the availability of symbolic knowledge. We show that the application of probabilistic learning to the interpretation of compound nouns is novel and promising. Finally, our experiments revealed that information inherent in the corpus can make up for the lack of distributional evidence by taking advantage of smoothing methods that rely simply on verb-argument tuples extracted from a large corpus and surface contextual information without strictly presupposing the existence of annotated data or taxonomic information.

Acknowledgments
This work was supported by the Alexander S. Onassis Foundation and ESRC grant number R000237772 (Data Intensive Semantics and Pragmatics). Thanks to Frank Keller, Alex Lascarides, Scott McDonald, and three anonymous reviewers for valuable comments.

References

Abney, Steve. 1996. “Partial parsing via finite-state cascades.” In John Carroll, editor, *Workshop on Robust Parsing*, pages 8–15, Prague. European Summer School in Logic, Language and Information, Prague.

Brown, Peter F., Vincent J. Della Pietra, Peter V. de Souza, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics* 18(4):467–479.

Burnage, Gavin. 1990. “Celex—A guide for users.” Technical Report, Centre for Lexical Information, University of Nijmegan, Nijmegan, Netherlands.

Burnard, Lou. 1995. *Users Guide for the British National Corpus*. British National Corpus Consortium, Oxford University Computing Service, Oxford.

Carletta, Jean. 1996. Agreement on classification tasks: The Kappa statistic. *Computational Linguistics* 22(2):249–254.

Carroll, John and Diana McCarthy. 2000. Word sense disambiguation using automatically acquired verbal preferences. *Computers and the Humanities* 34(1/2):109–114.

Church, Kenneth W. and William A. Gale. 1991. A comparison of the enhanced Good-Turing and deleted estimation methods for estimating probabilities of English bigrams. *Computer Speech and Language* 5:19–54.

Clark, Stephen and David Weir. 2001. “Class-based probability estimation using a semantic hierarchy.” In *Proceedings of the Second North American Annual Meeting of the Association for Computational Linguistics*, pages 95–102, Pittsburgh, Pennsylvania.

Cohen, William W. 1996. “Learning trees and rules with set-valued features.” In *Proceedings of the 13th National Conference on Artificial Intelligence*, pages 709–716, Portland, Oregon. AAAI Press, Menlo Park, California.

Collins, Michael and James Brooks. 1995. “Prepositional phrase attachment through a backed-off model.” In David Yarowsky and Kenneth W. Church, editors, *Proceedings of the Third Workshop on Very Large Corpora*, pages 27–38, Cambridge, Massachusetts.

Copestake, Ann and Alex Lascarides. 1997. “Integrating symbolic and statistical representations: The lexicon pragmatics interface.” In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 136–143, Madrid, Spain.

Dagan, Ido, Lilian Lee, and Fernando C. N. Pereira. 1999. Similarity-based models of word cooccurrence probabilities. *Machine Learning* 34(1–3):43–69.

Downing, Pamela. 1977. On the creation and use of English compound nouns. *Language* 53(4):810–842.

Essen, Ute and Volker Steinbiss. 1992. “Co-occurrence smoothing for stochastic language modeling." In *Proceedings of International Conference on Acoustics Speech and Signal Processing*, volume 1, pages 161–164, San Francisco, California.

Finin, Tim 1980. "The semantic interpretation of nominal compounds." In *Proceedings of First National Conference on Artificial Intelligence*, pages 310–315, Stanford, California. AAAI Press, Menlo Park, California.

Grishman, Ralph and John Sterling. 1994. "Generalizing automatically generated selectional patterns." In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 742–747, Kyoto, Japan.

Hindle, Donald and Mats Rooth. 1993. Structural ambiguity and lexical relations. *Computational Linguistics* 19(1):103–120.

Hirschberg, Julia, Diane Litman, and Marc Swerts. 2001. "Identifying user corrections automatically in spoken dialogue systems." In *Proceedings of the Second North American Annual Meeting of the Association for Computational Linguistics*, pages 208–215, Pittsburgh, Pennsylvania.

Hobbs, Jerry R., Mark Stickel, Douglas Appelt, and Paul Martin. 1993. Interpretation as abduction. *Journal of Artificial Intelligence* 63(1–2):69–142.

Isabelle, Pierre. 1984. "Another look at nominal compounds." In *Proceedings of the 10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*, pages 509–516, Stanford, California.

Jones, Bernard. 1995. "Predicating nominal compounds." In *Proceedings of the 17th Annual Conference of the Cognitive Science Society*, pages 130–135, Pittsburgh, Pennsylvania.

Katz, Slava M. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics Speech and Signal Processing* 33(3):400–401.

Lakoff, George and Mark Johnson. 1980. *Metaphors We Live By*. University of Chicago Press, Chicago.

Lascarides, Alex and Alex Copestake. 1998. Pragmatics and word meaning. *Journal of Linguistics* 34(2):387–414.

Lauer, Mark. 1995. Designing Statistical Language Learners: Experiments on Compound Nouns. Ph.D. dissertation, Macquarie University, Sydney, Australia.

Lee, Lilian. 1999. "Measures of distributional similarity." In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, College Park, Maryland.

Lee, Lilian and Fernando Pereira. 1999. "Distributional similarity models: Clustering vs. nearest neighbors." In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 33–40, College Park, Maryland.

Leonard, Rosemary. 1984. *The Interpretation of English Noun Sequences on the Computer*. North-Holland, Amsterdam.

Levi, Judith N. 1978. *The Syntax and Semantics of Complex Nominals*. Academic Press, New York.

Levin, Beth. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

Li, Hang and Naoki Abe. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics* 24(2):217–244.

Macleod, Catherine, Ralph Grishman, Adam Meyers, Leslie Barrett, and Ruth Reeves. 1998. "Nomlex: A lexicon of nominalizations." In *Proceedings of the Eighth International Congress of the European Association for Lexicography*, pages 187–193, Liège, Belgium.

Marsh, Elaine. 1984. "A computational analysis of complex noun phrases in Navy messages." In *Proceedings of the 10th International Conference on Computational Linguistics*, pages 505–508, Stanford, California.

McDonald, David. 1982. Understanding Noun Compounds. Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, Pennsylvania.

Merlo, Paola and Suzanne Stevenson. 1999. "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 45–51, Bergen, Norway.

Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography* 3(4):235–244.

Mosteller, Frederick and David L. Wallace. 1964. *Inference and Disputed Authorship: The Federalist*. Addison-Wesley, London.

Passonneau, Rebecca J. and Diane J. Litman. 1997. Discourse segmentation by human and automated means. *Computational Linguistics* 23(1):103–140.

Pedersen, Ted. 2001. "A decision tree of bigrams is an accurate predictor of word sense." In *Proceedings of the Second North American Annual Meeting of the Association* *for Computational Linguistics*, pages 79–86, Pittsburgh, Pennsylvania.

Pereira, Fernando, Naftali Tishby, and Lillian Lee. 1993. "Distributional clustering of English words." In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 183–190, Columbus, Ohio.

Quirk, Randolph, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman, London.

Resnik, Philip Stuart. 1993. Selection and Information: A Class-Based Approach to Lexical Relationships. Ph.D. dissertation, University of Pennsylvania, Philadelphia.

Selkirk, Elizabeth. 1982. *The Syntax of Words*. MIT Press, Cambridge, Massachusetts.

Siegel, Eric V. 1999. "Corpus-based linguistic indicators for aspectual classification." In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 112–119, College Park, Maryland.

Siegel, Eric V. and Kathleen R. McKeown. 1994. "Emergent linguistic rules from inducing decision trees: Disambiguating discourse clue words." In *Proceedings of the 12th National Conference on Artificial Intelligence*, pages 820–826, Seattle, Washington. AAAI Press, Menlo Park, California.

Siegel, Sidney and N. John Castellan. 1988. *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill, New York.

Sparck Jones, Karen. 1983. "Compound noun interpretation problems." Technical Report 45, Computer Laboratory, Cambridge University, Cambridge, England.

Vanderwende, Lucy. 1994. "Algorithm for automatic interpretation of noun sequences." In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 782–788, Kyoto, Japan.

Warren, Beatrice. 1978. *Semantic Patterns of Noun-Noun Compounds*. Acta Universitatis Gothoburgensis, Göteborg, Sweden.

Wu, Dekai. 1993. "Approximating maximum-entropy ratings for evidential parsing and semantic interpretation." In *Proceedings of 13th International Joint Conference on Artificial Intelligence*, pages 1290–1296, Chamberry, France. Morgan Kaufmann.