

# Web-Based Models for Natural Language Processing

MIRELLA LAPATA and FRANK KELLER  
University of Edinburgh

---

Previous work demonstrated that Web counts can be used to approximate bigram counts, suggesting that Web-based frequencies should be useful for a wide variety of Natural Language Processing (NLP) tasks. However, only a limited number of tasks have so far been tested using Web-scale data sets. The present article overcomes this limitation by systematically investigating the performance of Web-based models for several NLP tasks, covering both syntax and semantics, both generation and analysis, and a wider range of  $n$ -grams and parts of speech than have been previously explored. For the majority of our tasks, we find that simple, unsupervised models perform better when  $n$ -gram counts are obtained from the Web rather than from a large corpus. In some cases, performance can be improved further by using backoff or interpolation techniques that combine Web counts and corpus counts. However, unsupervised Web-based models generally fail to outperform supervised state-of-the-art models trained on smaller corpora. We argue that Web-based models should therefore be used as a baseline for, rather than an alternative to, standard supervised models.

Categories and Subject Descriptors: I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis; language models*; H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Web-based services*

General Terms: Algorithms, Experimentation, Languages

Additional Key Words and Phrases: Web counts,  $n$ -gram models, evaluation

---

## 1. INTRODUCTION

The Web is increasingly being used as a data source in a wide range of natural language processing (NLP) tasks. Several researchers have explored the potential of Web data for machine translation, either by creating bilingual corpora [Resnik and Smith 2003] or by using the Web to filter out or postedit translation candidates [Grefenstette 1998; Cao and Li 2002; Way and Gough 2003]. Other work discovers semantic relations by querying the Web for lexico-syntactic patterns indicative of hyponymy [Modjeska et al. 2003; Shinzato and Torisawa 2004], entailment [Szpektor et al. 2004], similarity, antonymy, or enablement

---

A preliminary version of this work was published as Lapata and Keller [2004].

Authors' address: School of Informatics, 2 Buccleuch Place, Edinburgh EH8 9LW, UK; email: {mlap,keller}@inf.ed.ac.uk.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or direct commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 1515 Broadway, New York, NY 10036 USA, fax: +1 (212) 869-0481, or [permissions@acm.org](mailto:permissions@acm.org).

© 2005 ACM 1550-4875/05/0200-0001 \$5.00

[Chklovski and Pantel 2004]. A number of studies have investigated the usefulness of the Web for word sense disambiguation [Mihalcea and Moldovan 1999; Rigau et al. 2002; Santamaría et al. 2003], question answering [Dumais et al. 2002; Hildebrandt et al. 2004; Soricut and Brill 2004], and language modeling [Zhu and Rosenfeld 2001; Keller and Lapata 2003; Bulyko et al. 2003].

Keller and Lapata [2003] have undertaken several studies to examine the validity of Web counts for a range of predicate-argument bigrams (verb-object, adjective-noun, and noun-noun bigrams). They presented a simple method for retrieving bigram counts from the Web by querying a search engine and demonstrated that Web counts (a) correlate with frequencies obtained from a carefully edited, balanced corpus such as the 100M words British National Corpus (BNC), (b) correlate with frequencies recreated using smoothing methods in the case of unseen bigrams, (c) reliably predict human plausibility judgments, and (d) yield state-of-the-art performance on pseudo-disambiguation tasks.

Keller and Lapata's [2003] results suggest that Web-based frequencies can be a viable alternative to bigram frequencies obtained from smaller corpora or recreated using smoothing. However, they do not demonstrate that realistic NLP tasks can benefit from Web counts. In order to show this, Web counts would have to be applied to a diverse range of NLP tasks, both syntactic and semantic, involving analysis (e.g., disambiguation) and generation (e.g., selection among competing outputs). Also, it remains to be shown that the Web-based approach scales up to larger  $n$ -grams (e.g., trigrams) and to combinations of different parts of speech (Keller and Lapata [2003] only tested bigrams involving nouns, verbs, and adjectives). Another important question is whether Web-based methods, which are by definition unsupervised, can be competitive alternatives to supervised approaches used for most tasks in the literature. Finally, Keller and Lapata's [2003] work raises the question as to whether Web counts (noisy, but less sparse) can be fruitfully combined with corpus counts (less noisy, but sparse) into a single model.

The present article aims to address these questions. We start by exploring the performance of Web counts on two generation tasks for which the use of large data sets has previously shown promising results: (a) target language candidate selection for machine translation [Grefenstette 1998] and (b) context-sensitive spelling correction [Banko and Brill 2001a, 2001b]. Then we investigate the generality of the Web-based approach by applying it to a range of analysis and generations tasks, involving both syntactic and semantic knowledge: (c) ordering of prenominal adjectives, (d) compound noun bracketing, (e) compound noun interpretation, (f) noun countability detection, (g) article restoration, and (h) PP attachment disambiguation. Table I gives an overview of these tasks and their properties. As the Table illustrates, our choice of tasks covers  $n$ -grams of different sizes and includes a wide variety of parts of speech.

For all tasks listed in Table I, we propose simple, unsupervised  $n$ -gram-based models whose parameters can be estimated using Web counts. We compare these models against identical corpus-based models (whose parameters are estimated from a conventional corpus). The corpus-based models give us a lower limit on the performance for a given task. We also compare the Web-based models against state-of-the-art models reported in the literature; typically, these are

Table I. Overview of the Tasks Investigated in This Article ( $n$ : size of  $n$ -gram; POS: parts of speech; Ling: linguistic knowledge; Type: type of task)

Task	$n$	POS	Ling	Type
MT candidate selection	1, 2	V, N	Sem	Generation
Spelling correction	1, 2, 3	Any	Syn/Sem	Generation
Adjective ordering	1, 2	Adj	Sem	Generation
Article generation	1, 2, 3	Det, any	Sem	Generation
Compound bracketing	1, 2	N	Syn	Analysis
Compound interpretation	1, 2, 3	N, P	Sem	Analysis
Countability detection	1, 2	N, Det	Sem	Analysis
PP attachment	1, 2, 3	V, N, P	Syn/Sem	Analysis

supervised models (which use annotated training data), or unsupervised models which rely on external resources such as taxonomies to recreate missing counts. The models in the literature provide an upper limit on the performance we can expect from Web-based models.

For each of the tasks we investigate, we also explore models that combine Web counts and corpus counts. We propose two combination schemes: backoff and interpolation. Both of these approaches are then compared against purely Web-based and purely corpus-based models. These combined models are weakly supervised: they include parameters that need to be estimated on a separate development set, but they do not require annotated data.

## 2. METHOD

### 2.1 Web Counts

Following Keller and Lapata [2003], we obtain Web counts for  $n$ -grams using a simple heuristic based on queries to a Web search engine. In this approach, the Web count for a given  $n$ -gram is estimated as the number of hits (pages) returned by the search engine for the queries generated for this  $n$ -gram. Two different ways of generating queries for a given  $n$ -gram are employed in the present article:

*Literal queries* use the quoted  $n$ -gram directly as a search term for the search engine (e.g., the bigram *history changes* expands to the query "history changes").

*Inflected queries* are obtained by expanding an  $n$ -gram into all its morphological forms. These forms are then submitted as literal queries, and the resulting hits are summed up (e.g., *history changes* expands to "history change", "history changes", "history changed", "histories change", "histories changed"). John Carroll's suite of morphological tools (morpha, morphg, and ana) is used to generate inflected forms of both verbs and nouns.<sup>1</sup> In certain cases (detailed in the following), determiners are inserted before nouns in order to make it possible to recognize simple NPs. This insertion is limited to *a/an*, *the*, and the empty determiner (for bare plurals).

<sup>1</sup>The tools can be downloaded from <http://www.informatics.susx.ac.uk/research/nlp/carroll/morph.html>.

All queries are performed as exact matches (using quotation marks), and all search terms are submitted to the search engine in lower case. If a query consists of a single, highly frequent word (such as *the*), some search engines return an error message. In these cases, we set the Web count to a large constant ( $10^8$ ). This problem is limited to unigrams which are used in some of the following models. Sometimes the search engine fails to return a hit for a given  $n$ -gram (for any of its morphological variants). We smooth such zero counts by setting them to .5.

For the experiments reported in this article, we use two search engines: Altavista and Google. Google only allows automated querying through the Google Web API. This involves obtaining a license key which then restricts the number of queries to a daily quota of 1000.<sup>2</sup> However, it is possible to apply for an increase of this quota for research purposes.<sup>3</sup>

At the time when we conducted most of the research reported here (September 2003), the Altavista search engine placed no restrictions on automated querying, and thus offered a fast and flexible way of generating Web counts. This is why Altavista was used to obtain the Web counts for all the tasks dealt with in this article (with the exception of article generation). The provider of Altavista changed the database that underlies the search engine in March 2004 (it now uses the same database as Yahoo). Altavista no longer allows unrestricted automated querying; rather the Yahoo API has to be used which involves obtaining a license key that restricts the number of queries to 5000 per day.<sup>4</sup> The Web counts for the article generation task in Section 6 were obtained later than the other data (in August 2004). We therefore decided to use Google to obtain these counts as Google's database is larger than that of Altavista/Yahoo, and both search engines now restrict automatic querying. In general, we would not expect the results to differ dramatically; Keller and Lapata [2003] compare Google and Altavista counts on their bigram data and find that the differences are negligible.

## 2.2 Limitations of Web Counts

As discussed by Keller and Lapata [2003], simple Web-based heuristics such as the ones used here introduce noise in the resulting frequency data. First, the heuristics rely on the assumption that page counts approximate  $n$ -gram frequencies. While Zhu and Rosenfeld [2001] present results that suggest that this assumption is justified, it certainly adds noise to the data. Another problem is that both Google and Altavista disregard punctuation and capitalization when matching a search term (even if the term is quoted). This can lead to false positives, for example, if the match crosses a sentence boundary. Also, the matches are likely to include links, Web addresses, file names, and other nontextual data.

<sup>2</sup>The Google Web API and the license key can be obtained from <http://www.google.com/apis/>.

<sup>3</sup>The authors obtained a quota of 20,000 queries per day by sending a request to [api-support@google.com](mailto:api-support@google.com).

<sup>4</sup>The Yahoo API and the license key can be obtained from <http://developer.yahoo.net/>.

In the method used here, no Web pages are downloaded which means that no tagging, chunking, or parsing of the data can be carried out. This drastically limits the type of counts that can be obtained: we are only dealing with counts of  $n$ -grams of adjacent words as captured by literal queries. Most of the tasks that we deal with would benefit from part-of-speech tagging which could be used to restrict the queries to the linguistic categories that are relevant for the task at hand (see Table I for an overview of the parts of speech involved in our tasks).

Another problem concerns the stability and accuracy of the page counts generated by search engines. There seems to be evidence that the counts returned by Google vary substantially over time due to changes made to the index and the database of the search engine and depending on which Google server is accessed. Also, it has been observed that the Boolean operators supported by Google return unexpected results. For example, the Boolean query *Chirac OR Sarkozy* returns a lower page count than the simple query *Chirac*, contrary to the logic of the OR operator.<sup>5</sup> The negation operator has a similar unexpected effect, for example, the query *applesauce -aosdnao* returns more hits than *applesauce* on its own.<sup>6</sup> It has been hypothesized that such anomalies are due to the optimizations that Google performs when computing the results of Boolean queries.<sup>7</sup>

However, there is also evidence for the reliability of Web counts: Keller and Lapata [2003] show that the counts generated by two search engines (Google and Altavista) are highly correlated with frequencies obtained from two standard corpora for English (the BNC and the North American Newstext Corpus). Note also that this article does not rely on Boolean queries, and therefore the results should be less susceptible to artifacts caused by Google's treatment of index terms.

### 2.3 Corpus Counts

For all tasks, the Web-based models are compared against identical models whose parameters are estimated from the BNC [Burnard 1995]. The BNC is a static 100M word corpus of British English which is about 1000 times smaller than the Web [Keller and Lapata 2003]. Comparing the performance of the same model on the Web and on the BNC allows us to assess how much improvement can be expected simply by using a larger data set. We retrieve BNC counts using the Gsearch corpus query tool [Corley et al. 2001]; the morphological query expansion is the same as for Web queries. Gsearch is used to search solely for adjacent words; no POS information is incorporated in the queries, and no parsing is performed. This ensures that Web counts and corpus counts are as similar as possible.

---

<sup>5</sup>This phenomenon was first investigated by Jean Véronis, see <http://aixtal.blogspot.com/> for details.

<sup>6</sup>This phenomenon was pointed out to us by one of the anonymous reviewers who suggests that including a rare word such as *aosdnao* forces Google to use the whole index instead of performing optimization.

<sup>7</sup>For a detailed discussion of these questions, see the Corpora List (March 2005), <http://torvald.aksis.uib.no/corpora/>.

## 2.4 Model Selection

For all of our tasks, we have to select either the best one of several possible models or the best parameter setting for a single model (in case of model combination, see Section 2.5). We therefore require a separate development set. This is generated by using the gold standard data set from the literature for a given task and randomly dividing it into a development set and a test set (of equal size). We report the test set performance for all models for a given task and indicate which model shows optimal performance on the development set (marked by a “#” in all subsequent tables). We also compare the test set performance of this optimal model to the performance of the models reported in the literature. Here, it is important to note that the performance reported in the literature was typically obtained on the whole gold standard data set and hence may differ from the performance on our test set which is only a random sample thereof. However, we work on the assumption that such differences are negligible.

## 2.5 Combining Web and Corpus Counts

We also examine whether the performance on our tasks can be improved by combining Web counts and corpus counts. Web counts can be expected to contain noise introduced by a number of sources as discussed in Section 2.2. On the other hand, corpus counts are much less noisy but sparser than Web counts. Therefore it seems promising to devise a model that combines the two types of counts: corpus counts are used when they are available, but Web counts are substituted for corpus counts when the latter are too sparse.

The most straightforward way of implementing this idea is in the form of a backoff scheme. If the  $n$ -gram count for an item in the corpus falls below a threshold  $\theta$ , the Web is used to estimate the  $n$ -gram’s frequency, otherwise the corpus counts, are used.<sup>8</sup> Note that this backoff scheme subsumes both a purely Web-based model ( $\theta = 0$ ) and a purely corpus-based model ( $\theta = k$ , where  $k$  is the largest observed corpus count).

An alternative way of combining Web and corpus counts is interpolation. By interpolating Web and corpus counts, we do not discard one type of count completely but instead use a fraction of it in the model. This can be realized straightforwardly using a standard interpolation scheme, as shown in (1).

$$f = \lambda f_{web} + (1 - \lambda) f_{corpus} \quad (1)$$

It can be expected that Web counts are many orders of magnitude larger than corpus counts; the interpolation approach can take this into account by making  $\lambda$  very small. Again, this scheme subsumes both a purely Web-based model ( $\lambda = 1$ ) and a purely corpus-based model ( $\lambda = 0$ ).

The value of the backoff threshold  $\theta$  and the interpolation factor  $\lambda$  must be tuned on a heldout development set. This means that models employing interpolation or backoff are weakly supervised (as opposed to models that purely

<sup>8</sup>In models that use ratios such as  $\frac{f(a,b)}{f(a)}$ , the backoff scheme is only applied to the numerator as the denominator is guaranteed to be larger than the numerator in all cases.

Table II. Fictitious Example for a Contingency Table for a  $\chi^2$  Test Comparing to Models

	Correct	Incorrect
Model 1	870	130
Model 2	746	251

rely on Web or corpus counts and require no parameter tuning at all). To adjust the parameter values, we simply perform exhaustive search. This means that all possible values of  $\lambda$  are tried, with  $0 \leq \lambda \leq 1$  and a stepsize of  $10^{-6}$ . Along the same lines, all possible values of  $\theta$  are tried, with  $0 \leq \theta \leq 150$  and a stepsize of 1.

In Section 3, we will illustrate this search process by plotting the performance of both the backoff and the interpolation model against the parameter settings that are explored.

## 2.6 Significance Testing

In what follows, we present four distinct models for all the tasks we discuss: a Web-based model, a corpus-based model, a backoff model, and an interpolated model. We compare these models to each other to determine the usefulness of Web counts for a given task. However, it is also important to compare the models to valid baselines and to the best models reported in the literature. All these comparisons are only meaningful if statistical tests are performed to determine if differences in performance are significant.

Throughout this article, we report the results of  $\chi^2$  tests that are carried out to determine if a given difference in model performance is statistically significant. Model performance is reported in terms of accuracy, that is, as the number of items correct over the total number of items. Such accuracy figures are effectively frequency data which means that  $\chi^2$  is an appropriate test since it makes no distributional assumptions. It can be carried out by compiling a  $2 \cdot 2$  contingency table: the two models to be compared form the rows of this table, and the number of correct and incorrect items are listed in the columns. This is illustrated by the fictitious data in Table II. Then the  $\chi^2$  coefficient is computed which compares the distributions of the two rows of the table and yields a significant result if they are reliably different.

In the example in Table II, both models are evaluated on the same data set (with a size of 1000 items). However, this is not always the case: throughout this article, we compare models reported in the literature against Web-based and corpus-based models. The former are typically tested on larger data sets than the latter. This is due to the fact that we split the data sets used in the literature into development and test sets as indicated in Section 2.4. Note that the  $\chi^2$  test is robust against such differences in sample size.

In order to facilitate the comparison between models, we use a set of symbols throughout this article, see Table III. These symbols indicate if the corresponding  $\chi^2$  is significant or not.

Table III. Meaning of the Symbols Indicating Statistical Significance ( $\chi^2$  Tests)

Symbols	Meaning
#	best model on development set
*    /#	significantly (not significantly) different from best BNC model
†    /†	significantly (not significantly) different from baseline
‡    /‡	significantly (not significantly) different from best model in literature
\$    /\$	significantly (not significantly) different from best web model

### 3. CANDIDATE SELECTION FOR MACHINE TRANSLATION

Target word selection is a generation task that occurs in machine translation (MT). A word in a source language can often be translated into different words in the target language and the choice of the appropriate translation depends on a variety of semantic, syntactic, and pragmatic factors. The task is illustrated in the following example where there are five translation alternatives for the German noun *Geschichte* listed in curly brackets, the first being the correct one.

- [1.] (a) Die *Geschichte* ändert sich, nicht jedoch die Geographie.  
 (b) {*History, story, tale, saga, strip*} changes but geography does not.

Statistical approaches to target word selection rely on bilingual lexica to provide all possible translations of words in the source language. Once the set of translation candidates is generated, statistical information gathered from target language corpora is used to select the most appropriate alternative [Dagan and Itai 1994]. The task is somewhat simplified by Grefenstette [1998] and Prescher et al. [2000] who do not produce a translation of the entire sentence. Instead, they focus on specific syntactic relations. Grefenstette translates compounds from German and Spanish into English and uses BNC frequencies as a filter for candidate translations. He observes that this approach suffers from an acute data sparseness problem and goes on to obtain counts for candidate compounds through Web searches, thus achieving a translation accuracy of 86–87%.

Prescher et al. [2000] concentrate on verbs and their objects. Assuming that the target language translation of the verb is known, they select from the candidate translations the noun that is semantically most compatible with the verb. The semantic fit between a verb and its argument is modeled using a class-based lexicon that is derived from unlabeled data using the expectation maximization algorithm (verb-argument model). Prescher et al. also propose a refined version of this approach that only models the fit between a verb and its object (verb-object model), disregarding other arguments of the verb. The two models are trained on the BNC and evaluated against two corpora, differing in the degree of translation ambiguity displayed by the object noun. The high ambiguity corpus contains 1,340 bilingual sentence pairs with an average of 8.63 translations for the object noun. The low ambiguity corpus has 814 bilingual sentence pairs with an average of 2.83 translations. Table V lists Prescher et al.'s results on these two corpora for both

Table IV. Performance of Altavista, BNC, Interpolated, and Backoff Models for Candidate Selection for MT (Data From Prescher et al. [2000])

Model	Altavista		BNC		Interpol		Backoff	
	High Ambig	Low Ambig						
$f(v, n)$	46.77	69.40#	52.78#	72.94#	53.07	74.27	52.78#	75.16#
$f(v, n)/f(n)$	46.62#	64.52	49.26	70.06	53.07	74.27	49.26	71.61

Table V. Performance Comparison with the Literature for Candidate Selection for MT (Baseline: Select Most Frequent Target Noun)

Model	High Ambig	Low Ambig
Baseline	31.90*‡\$	45.50*‡\$
Prescher et al. [2000]: Verb-argument	43.30*†‡\$	61.50*†‡\$
Best Altavista	46.62*†‡	69.40*†‡
Prescher et al. [2000]: Verb-object	49.40*†\$	68.20*†\$
Best Backoff	52.78*†‡\$	70.28*†‡\$
Best BNC	52.78†‡\$	72.94†‡\$
Best Interpol	53.07*†‡\$	74.27*†‡\$

models together with a frequency baseline<sup>9</sup> (select the most frequent target noun).

Grefenstette’s [1998] evaluation was restricted to compounds that are listed in a dictionary. These compounds are presumably well-established and fairly frequent which makes it easy to obtain reliable Web frequencies. We wanted to test if the Web-based approach extends from lexicalized compounds to productive syntactic units for which dictionary entries do not exist. We therefore performed our evaluation using Prescher et al.’s [2000] test set of verb-object pairs. Web counts were retrieved for all possible verb-object translations; the most likely one was selected using either co-occurrence frequency ( $f(v, n)$ ) or conditional probability ( $f(v, n)/f(n)$ ). The Web counts were gathered using inflected queries involving the verb, a determiner, and the object (see Section 2.1). Table IV compares the Web-based models against the BNC models on the test set. (Recall from Section 2.4 that we split the data sets in the literature into a development set and a test set. We report the performance on the test set in Table IV and indicate which models performed best on the development set using a “#”. These models then form the basis for the significance tests reported in Table V.)

As explained in Section 2.5, we also test two weakly supervised models that combine Web counts and corpus counts using backoff and interpolation. Table IV includes the performance of these two approaches, listed under the headings Interpol and Backoff. Note that these models require the adjustment of parameters on the development set. Figures 1 and 2 show how model performance varies along different parameter values.

<sup>9</sup>Prescher et al. [2000] only report a random baseline (i.e., they select a target noun at random); the latter achieves an accuracy of 14.20% for the high ambiguity words and 45.90% for the low ambiguity words.

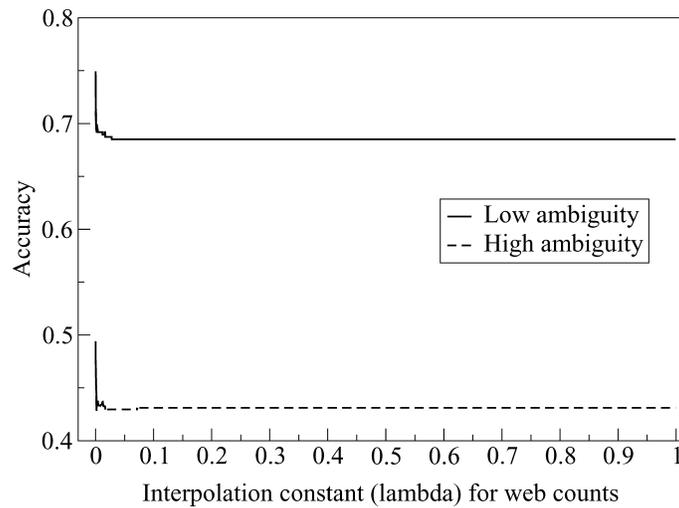


Fig. 1. The performance of the  $f(v, n)$  model for candidate selection for machine translation for different interpolation factors ( $\lambda$ ).

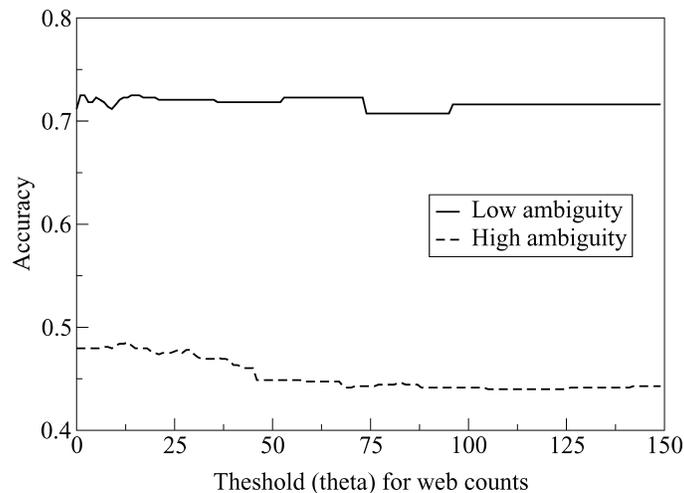


Fig. 2. The performance of the  $f(v, n)$  model for candidate selection for machine translation for different backoff thresholds ( $\theta$ ).

Figure 1 plots model accuracy for the  $f(v, n)$  model against values of  $\lambda$  for the interpolation model. Note that these curves were obtained on the development set, while the performance reported in Table IV is on the test set. For the high ambiguity data set, the best performance was achieved with an interpolation threshold of  $\lambda = 2.1 \cdot 10^{-5}$ ; for the low ambiguity data set, the best performance was achieved with  $\lambda = 4 \cdot 10^{-5}$ . This means that, in both cases, the best model only included a very small fraction of Web counts in the interpolated counts.

Figure 2 plots the performance of the backoff model against different values of  $\theta$  (again, these curves were obtained on the development set). We observe that for the high ambiguity data set, a threshold of  $\theta = 1$  yields the best performance, that is, a model where we use Web counts only when the corpus counts are zero. For the low ambiguity data set, the best performance is obtained for  $\theta = 13$ , that is, for a model that uses Web counts if the corpus counts are less than 13.

Returning to Table IV, we observe that in most cases simple co-occurrence frequency outperforms conditional probability. Furthermore, notice that for both the high and low ambiguity data sets, the performance of the best BNC model is comparable to the best Web-based model; in fact the difference between the two models is not statistically significant (see Table V). Combining the BNC and Web models yields improved performances over the individual models. For low ambiguity words, the interpolated model yields an improvement of approximately 1% over the BNC model, whereas the improvement for the backoff model is approximately 2%. The backoff model yields no improvement over the best BNC model for high ambiguity words; a small improvement of .3% is obtained with the interpolated model.

Table V compares our unsupervised and weakly supervised models with the two sophisticated class-based models previously discussed. The results show that there is no significant difference in performance between the best model reported in the literature and the best Altavista or the best BNC model. The backoff and interpolated models are not significantly different from the best model in the literature either. For high ambiguity words, backoff and interpolation perform significantly better than the best Altavista model. The models based on BNC and Altavista counts, as well as the models based on their combination, significantly outperform the baseline. This holds for both the high and low ambiguity data sets.

#### 4. CONTEXT-SENSITIVE SPELLING CORRECTION

Context-sensitive spelling correction is the task of correcting spelling errors that result in valid words. Such a spelling error is illustrated in the following where *principal* was typed when *principle* was intended.

[2.] Introduction of the dialogue *principal* proved strikingly effective.

Context-sensitive spelling correction can be viewed as a generation task, since it consists of choosing between alternative surface realizations of a word. This choice is typically modeled by *confusion sets* such as {principal, principle} or {then, than} under the assumption that each word in the set could be mistakenly typed when another word in the set was intended. The task is to infer which word in a confusion set is the correct one in a given context. This choice can be either syntactic (as for {then, than}) or semantic (as for {principal, principle}).

A number of machine learning methods have been proposed for context-sensitive spelling correction. These include a variety of Bayesian classifiers [Golding 1995; Golding and Schabes 1996], decision lists [Golding 1995] transformation-based learning [Mangu and Brill 1997], Latent Semantic Analysis (LSA) [Jones and Martin 1997], multiplicative weight update algorithms

Table VI. Performance of Altavista, BNC, Interpolated, and Backoff Models for Context-Sensitive Spelling Correction (Data from Cucerzan and Yarowsky 2002)

Model	Altavista	BNC	Interpol	Backoff
$f(t)$	74.79	71.78	—	—
$f(w_1, t)$	86.13	83.29	86.48	87.54
$f(t, w_1)$	82.36	81.70#	83.67	82.99
$f(w_1, t, w_2)$	89.35#	78.14	89.57#	89.35#
$f(w_1, w_2, t)$	89.24	74.72	88.26	87.38
$f(t, w_2, w_2)$	83.71	73.46	84.36	83.68
$f(w_1, t)/f(t)$	84.44	78.61	83.94	82.33
$f(t, w_1)/f(t)$	75.68	80.19	80.85	80.05
$f(w_1, t, w_2)/f(t)$	86.07	76.64	85.59	85.88
$f(w_1, w_2, t)/f(t)$	85.38	74.51	86.22	85.70
$f(t, w_2, w_1)/f(t)$	81.79	72.47	82.23	82.57
$f(w_1, t, w_2)/f(w_1, t)$	78.02	74.41	78.03	80.16
$f(w_1, t, w_2)/f(t, w_2)$	81.78	75.88	82.07	84.69
$f(w_1, w_2, t)/f(w_2, t)$	71.40	69.61	74.74	78.57
$f(t, w_1, w_2)/f(t, w_1)$	74.68	70.31	71.99	75.48

[Golding and Roth 1999], and augmented mixture models [Cucerzan and Yarowsky 2002]. Despite their differences, most approaches use two types of features: context words and collocations. Context word features record the presence of a word within a fixed window around the target word (bag of words). Collocational features capture the syntactic environment of the target word and are usually represented by a small number of words and/or part-of-speech tags to the left or right of the target word.

The results obtained by a variety of classification methods are given in Table VII. All methods use either the full set or a subset of 18 confusion sets originally gathered by Golding [1995]. Most methods are trained and tested on the Brown corpus, using 80% for training and 20% for testing.<sup>10</sup>

We devised a simple, unsupervised method for performing spelling correction using Web counts. The method takes into account collocational features, that is, words that are adjacent to the target word. For each word in the confusion set, we used the Web to estimate how frequently it co-occurs with a word or a pair of words immediately to its left or right. Disambiguation is then performed by selecting the word in the confusion set with the highest co-occurrence frequency or probability. The Web counts were retrieved using literal queries (see Section 2.1). Ties were resolved by defaulting to the word with the highest unigram frequency in the confusion set. The Web models were compared to their corresponding BNC models. BNC and Web models were further combined using interpolation and backoff, as explained in Section 2.5 (and also Section 3).

Table VI shows the types of collocations we considered and their corresponding accuracy. The baseline ( $f(t)$ ) in Table VI was obtained by always choosing the most frequent unigram in the confusion set. We used the same test set (2,056 tokens from the Brown corpus) and confusion sets as Golding and

<sup>10</sup>An exception is Golding [1995] who uses the entire Brown corpus for training (1M words) and three quarters of the Wall Street Journal corpus [Marcus et al. 1993] for testing.

Table VII. Performance Comparison with the Literature for Context-Sensitive Spelling Correction (Baseline: Select Most Frequent Word in Confusion Set)

Model	Accuracy
Baseline	71.25*‡\$
Golding [1995]	81.40*†‡\$
Best BNC	81.70*†‡\$
Jones and Martin [1997]	84.26*†‡\$
Best Backoff	89.35*†‡\$
Best Altavista	89.35*†‡
Best Interpol	89.57*†‡\$
Golding and Schabes [1996]	89.82*†‡\$
Cucerzan and Yarowsky [2002]	92.20*†‡\$
Mangu and Brill [1997]	92.79*†‡\$
Golding and Roth [1999]	94.23*†\$

Schabes [1996], Mangu and Brill [1997], and Cucerzan and Yarowsky [2002]. A comparison with the literature is given in Table VII.

The best result (89.35%) for the Web-based approach is obtained with a context of one word to the left and one word to the right of the target word  $f(w_1, t, w_2)$ , see Table VI. The BNC-based models perform consistently worse than the Web-based models with the exception of  $f(t, w_1)/t$ ; the best Altavista model performs significantly better than the best BNC model. Table VII shows that both the best Altavista model and the best BNC model outperform their respective baselines. A comparison with the literature reveals that the best Altavista model significantly outperforms Golding [1995], Jones and Martin [1997], and yields results comparable to Golding and Schabes's [1996]. The backoff and interpolation models fail to significantly outperform the best Altavista model.

The highest accuracy on the task is achieved by the class of multiplicative weight-update algorithms such as Winnow [Golding and Roth 1999]. Both the best BNC model and the best Altavista model perform significantly worse than this model. Note that Golding and Roth [1999] use algorithms that can handle large numbers of features and are robust to noise. Our method uses a very small feature set that relies only on co-occurrence frequencies and does not have access to POS information (the latter has been shown to have an improvement on confusion sets whose words belong to different parts of speech). An advantage of our method is that it can be used for a large number of confusion sets without relying on the availability of training data.

## 5. ORDERING OF PRENOMINAL ADJECTIVES

The ordering of prenominal modifiers is important for natural language generation systems where the text must be both fluent and grammatical. For example, the sequence *big fat Greek wedding* is perfectly acceptable, whereas *fat Greek big wedding* sounds odd. The ordering of prenominal adjectives has sparked a great deal of theoretical debate (see Shaw and Hatzivassiloglou [1999] for an overview) and efforts have concentrated on defining rules based

Table VIII. Performance of Altavista, BNC, Interpolated, and Backoff Models for Adjective Ordering (Data From Malouf [2000])

Model	Altavista	BNC	Interpol	Backoff
$f(a_1, a_2) : f(a_2, a_1)$	89.20#	76.40#	91.20#	91.40#
$f(a_1, a_2)/f(a_2) : f(a_2, a_1)/f(a_1)$	80.80	75.80	96.60	86.80
$f(a_1, a_2)/f(a_1) : f(a_2, a_1)/f(a_2)$	79.40	75.80	87.20	87.40

on semantic criteria that account for different orders (e.g., age < color, value < dimension).

Data intensive approaches to the ordering problem rely on corpora for gathering evidence for the likelihood of different orders. They rest on the hypothesis that the relative order of premodifiers is fixed and independent of context and the noun being modified. The simplest strategy is what Shaw and Hatzivassiloglou [1999] call *direct evidence*. Given an adjective pair  $\langle a, b \rangle$ , they count how many times  $\langle a, b \rangle$  and  $\langle b, a \rangle$  appear in the corpus and choose the pair with the highest frequency.

Unfortunately, the direct evidence method performs poorly when a given order is unseen in the training data. To compensate for this, Shaw and Hatzivassiloglou [1999] propose to compute the *transitive closure* of the ordering relation: if  $a < c$  and  $c < b$ , then  $a < b$ . Malouf [2000] reduces adjective ordering to the well-known problem of estimating  $n$ -gram probabilities and proposes a backoff bigram model of adjective pairs for choosing among alternative orders ( $P(\langle a, b \rangle | \{a, b\})$  vs.  $P(\langle b, a \rangle | \{a, b\})$ ). He also uses positional probabilities as a means of estimating how likely it is for a given adjective  $a$  to appear first in a sequence by looking at each pair in the training data that contains the adjective  $a$  and recording its position. Finally, he frames the adjective ordering problem as a classification task and uses memory-based learning for inferring an appropriate order. Morphological and semantic similarities among different adjective orders are expressed using a feature-vector representation. Each adjective pair  $ab$  is encoded as a vector of 16 features (the last eight characters of  $a$  and the last eight characters of  $b$ ) and a class ( $\langle a, b \rangle$  or  $\langle b, a \rangle$ ).

Malouf [2000] extracted 263,838 individual pairs of adjectives from the BNC which he randomly partitioned into test (10%) and training data (90%) and evaluated all the preceding methods for ordering prenominal adjectives. His results showed that a memory-based classifier that uses morphological information as well as positional probabilities as features outperforms all other methods.

For the ordering task, we restricted ourselves to the direct evidence strategy which simply chooses the adjective order with the highest frequency or probability (see Table VIII). Web counts were obtained by submitting literal queries to Altavista (see Section 2.1). We used the same 263,838 adjective pairs that Malouf [2000] extracted from the BNC. These were randomly partitioned into a training (90%) and test corpus (10%). The test corpus contained 26,271 adjective pairs. Given that submitting  $4 \cdot 26,271$  queries to Altavista would be fairly time-consuming, a random sample of 1,000 sequences was obtained from the test corpus, and the Web frequencies of these pairs were retrieved.

Table VIII shows the performance of direct evidence vs. conditional probability models. The best BNC, Altavista, backoff, and interpolated models are

Table IX. Performance Comparison with the Literature for the Ordering of Prenominal Modifiers (Baseline: Taken from Malouf [2000]; Selects Most Frequent Adjective Order in the BNC)

Model	Accuracy
Baseline	78.28 <sup>*/‡</sup> \$
Best BNC	76.40 <sup>/‡</sup> \$
Best Altavista	89.20 <sup>*†‡</sup>
Best Interpol	91.20 <sup>*†‡</sup> \$
Best Backoff	91.40 <sup>*†‡</sup> \$
Malouf [2000]	91.80 <sup>*†</sup> \$

compared against Malouf [2000] in Table IX. We find that direct evidence models generally perform better than conditional models. The best Altavista model significantly outperforms the baseline<sup>11</sup> (reported by Malouf) and the best BNC model; its performance is significantly worse than the best model reported in the literature. The backoff and interpolated models obtain performances that are not significantly different from the best model proposed by Malouf [2000], a supervised method using positional probability estimates from the BNC, and morphological variants.

## 6. ARTICLE GENERATION

Non-native speakers of English often have difficulties with article selection (e.g., *the*, *a*) especially if their native language is a language that lacks articles such as Japanese or Russian. Common mistakes include dropping articles altogether or using *a* or *an* with plural or uncountable nouns (e.g., *a students*, *a research*). Similar mistakes in article choice occur in automatically generated texts that form the output of machine translation or summarization systems, especially if the latter include a sentence regeneration component.

Aiming to improve the quality of the text produced by a Japanese-English machine translation system, Knight and Chander [1994] construct an automatic posteditor for inserting articles into English text. They cast article selection as a binary classification problem and use decision trees to learn whether to generate *the* or *a/an*. The decision trees are trained on a database of 400,000 NP instances derived from the Wall Street Journal, using a variety of lexical (e.g., words before or after the article), syntactic (e.g., parts-of-speech), and semantic (e.g., tense) features. During testing, the postediting process is simulated by removing the articles from texts with grammatical article usage; the decision trees reinsert articles and the resulting text is compared to the original. By constructing decision trees for the most frequent nouns in their data set and guessing *the* for the rest, Knight and Chander [1994] achieve an overall accuracy of 78% (see Table XI).

<sup>11</sup>Malouf's [2000] baseline is the direct evidence model ( $f(a_1, a_2) : f(a_2, a_1)$ ) estimated on the BNC; it differs slightly from our best BNC model since it is computed on a different training/test set split.

Table X. Performance of Google, BNC, Interpolated, and Backoff Models for Article Generation (Data From Lee [2004])

Model	Google	BNC	Interpol	Backoff
Baseline	69.40	69.40	69.40	69.40
$f(art, NP)$	76.20	73.60	78.60	78.70
$f(c_1, art, NP)$	87.30	74.10#	87.40	87.20
$f(c_2, c_1, art, NP)$	91.20#	70.50	91.20#	91.10#
$f(c_1, art, NP)/f(art, NP)$	61.10	68.40	68.40	66.40
$f(c_2, c_1, art, NP)/f(c_1, art, NP)$	79.30	70.40	81.70	81.70

Knight and Chander [1994] also performed experiments with humans in order to estimate an upper bound on the article generation task. The humans mimicked the decision trees: they were given English text without articles and were asked to reinsert them. When given access to the context surrounding the NP in question, the humans achieved accuracies between 94% and 96% (see Table XI). With some context (i.e., two words to the left of the missing article and two words to the right of the head noun), human performance was between 83% to 88%. With limited context (i.e., the head noun following the article and its premodifiers), humans achieved an accuracy between 79% and 80%.

Knight and Chander’s [1994] approach was further extended by Minnen et al. [2000] and Lee [2004] who learn to predict whether to generate *the*, *a/an*, or no article. Using memory-based learning and a feature set larger than that of Knight and Chander [1994], including grammatical functions (e.g., subject, object) and semantic roles (e.g., location, manner), Minnen et al. [2000] achieve an accuracy of 83.6%. Lee [2004] reports an accuracy of 87.7% using a maximum entropy learner and features similar to those employed by Minnen et al. [2000] (see Table XI).

We attempted the article generation task using unsupervised models which were evaluated on Lee’s [2004] test data. The latter was generated from section 23 of the Penn treebank. For our simplest model, we estimate how frequently each noun phrase in the test data co-occurs with *the*, *a/an*, or nothing, and default to the choice with the highest frequency ( $f(art, NP)$ ). We also devised models that take into account one or two words to the left of the unknown article and the core noun phrase ( $f(c_1, art, NP)$  and  $f(c_2, c_1, art, NP)$ ). The Web counts were retrieved using literal queries (see Section 2.1). Table X shows the models we considered and their corresponding accuracy. The baseline in Table X was obtained by choosing the most frequent class (i.e., no article) in the test data.<sup>12</sup>

The best Google model achieves 91.20% on the test data and takes two context words into account ( $f(c_2, c_1, art, NP)$ ). On the BNC, the model that takes one context word into account performs best ( $f(c_1, art, NP)$ ). The Google model significantly outperforms the BNC model as well as the best model in the literature (Lee’s [2004] maximum entropy model). The interpolated and backoff

<sup>12</sup>For this task, we used Google instead of Altavista counts due to the fact that Altavista had changed its database, see Section 2.1 for details.

Table XI. Performance Comparison with the Literature for Article Generation (Baseline: Always Selects No Article, the Most Frequent Class in Lee’s [2004] Data Set)

Model	Accuracy
Baseline	69.40 *‡\$
Best BNC	74.10 †‡\$
Lee [2004]	87.70 *†\$
Best Backoff	91.10 *†‡ \$
Best Google	91.20 *†‡
Best Interpol	91.20 *†‡ \$

models yield results similar to the best Google model but fail to significantly outperform it.

Knight and Chander [1994] obtained upper bounds on the article generation task by asking humans to decide whether the unknown article is *the* or *a/an*. To compare with these upper bounds, we tested the performance of our model on this simpler task by excluding from our test data all instances where the head noun was not preceded by an article. The best Google model reached an accuracy of 95.51% (over the most frequent class baseline which achieved 68.57%). This compares favorably with the 95–96% upper bound cited by Knight and Chander [1994].

## 7. BRACKETING OF COMPOUND NOUNS

The first analysis task we consider is the syntactic disambiguation of compound nouns which has received a fair amount of attention in the NLP literature [Pustejovsky et al. 1993; Resnik 1993; Lauer 1995]. The task can be summarized as such: given a three word compound  $n_1 n_2 n_3$ , determine the correct binary bracketing of the word sequence (see [3.] for an example).

- [3.] (a) [[backup compiler] disk]  
 (b) [backup [compiler disk]]

Previous approaches typically compare different bracketings and choose the most likely one. The *adjacency model* compares  $[n_1 n_2]$  against  $[n_2 n_3]$  and adopts a right branching analysis if  $[n_2 n_3]$  is more likely than  $[n_1 n_2]$ . The *dependency model* compares  $[n_1 n_2]$  against  $[n_1 n_3]$  and adopts a right branching analysis if  $[n_1 n_3]$  is more likely than  $[n_1 n_2]$ .

The simplest model of compound noun disambiguation compares the frequencies of the two competing analyses and opts for the most frequent one [Pustejovsky et al. 1993]. Lauer [1995] proposes an unsupervised method for estimating the frequencies of the competing bracketings based on a taxonomy or a thesaurus. He uses a probability ratio to compare the probability of the left-branching analysis to that of the right-branching (see (2) for the dependency

Table XII. Performance of Altavista, BNC, Interpolated, and Backoff Models for Compound Bracketing (Data from Lauer [1995])

Model	Altavista	BNC	Interpol	Backoff
$f(n_1, n_2) : f(n_2, n_3)$	75.40	69.67	75.40	0.7540
$f(n_1, n_2) : f(n_1, n_3)$	78.68#	76.22#	77.04#	77.04#
$f(n_1, n_2)/f(n_1) : f(n_2, n_3)/f(n_2)$	70.49	63.11	77.04	74.59
$f(n_1, n_2)/f(n_2) : f(n_2, n_3)/f(n_3)$	73.77	70.49	66.39	68.85
$f(n_1, n_2)/f(n_2) : f(n_1, n_3)/f(n_3)$	79.50	77.04	81.14#	77.04

model and (3) for the adjacency model).

$$R_{dep} = \frac{\sum_{t_i \in \text{cats}(w_i)} P(t_1 \rightarrow t_2)P(t_2 \rightarrow t_3)}{\sum_{t_i \in \text{cats}(w_i)} P(t_1 \rightarrow t_3)P(t_2 \rightarrow t_3)} \quad (2)$$

$$R_{adj} = \frac{\sum_{t_i \in \text{cats}(w_i)} P(t_1 \rightarrow t_2)}{\sum_{t_i \in \text{cats}(w_i)} P(t_2 \rightarrow t_3)} \quad (3)$$

Here  $t_1$ ,  $t_2$  and  $t_3$  are conceptual categories in a taxonomy or thesaurus, and the nouns  $w_1 \dots w_i$  are members of these categories. The estimation of probabilities over concepts (rather than words) reduces the number of model parameters and effectively decreases the amount of training data required. The probability  $P(t_1 \rightarrow t_2)$  denotes the modification of a category  $t_2$  by a category  $t_1$ .

Lauer [1995] evaluated both the adjacency and dependency models on 244 compounds extracted from Grolier’s encyclopedia, a corpus of 8 million words. Frequencies for the two models were obtained from the same corpus and from Roget’s Thesaurus (version 1911) by counting pairs of nouns that are either strictly adjacent or co-occur within a window of a fixed size (e.g., two, three, fifty, or hundred words). The majority of the bracketings in the test set were left-branching, yielding a baseline of 66.80% (see Table XIII). Lauer’s best results (77.50%) were obtained with the dependency model and a training scheme which takes strictly adjacent nouns into account. Performance increased further by 3.2% when POS tags were taken into account. The results for this tuned model are also given in Table XIII. Finally, Lauer conducted an experiment with human judges to assess the upper bound for the bracketing task. An average accuracy of 81.50% was obtained.

We replicated Lauer’s [1995] results for compound noun bracketing using the same test set. We compared the performance of the adjacency and dependency models (see (2) and (3)) but instead of relying on a corpus and a thesaurus, we estimated the relevant probabilities using Web counts. The latter were obtained using inflected queries (see Section 2.1). Ties were resolved by defaulting to the most frequent analysis (i.e., left-branching). To gauge the performance of the Web-based models, we compared them against their BNC-based alternatives; we also explored whether the combination of Web and BNC counts yields better results than the individual models (see Table XII). We compare our results against the literature in Table XIII.

Table XIII. Performance Comparison with the Literature for Compound Bracketing (Baseline: Always Selects the Most Frequent Analysis (i.e., Left Branching) in Lauer’s [1995] Data Set)

Model	Accuracy
Baseline	66.80 <sup>*†‡§</sup>
Lauer [1995]: adjacency	68.90 <sup>*†‡§</sup>
Best BNC	76.22 <sup>†‡§</sup>
Best Backoff	77.04 <sup>*†‡§</sup>
Lauer [1995]: dependency	77.50 <sup>*†‡§</sup>
Best Altavista	78.68 <sup>*†‡</sup>
Lauer [1995]: tuned	80.70 <sup>*†§</sup>
Best Interpol	81.14 <sup>*†‡§</sup>
Upper bound	81.50 <sup>*†‡§</sup>

The performance of the best Altavista model was not significantly higher than that of the best BNC model (see Table XIII) even though it significantly outperformed the baseline. Both the best BNC and Altavista models were not significantly different from the best model in the literature (Lauer’s tuned model). Hence our simple unsupervised models achieve the same performance as Lauer without recourse to a predefined taxonomy or a thesaurus. As far as the weakly supervised models are concerned, the backoff’s performance is disappointing, failing to outperform the baseline. The interpolation model performs slightly better than Lauer’s tuned model; however the difference is not statistically significant. Neither the backoff nor the interpolated model significantly outperform the best Altavista model.

## 8. INTERPRETATION OF COMPOUND NOUNS

The second analysis task we consider is the semantic interpretation of compound nouns. Most previous approaches to this problem have focused on the interpretation of two word compounds whose nouns are related via a basic set of semantic relations (e.g., CAUSE relates *onion tears*, FOR relates *pet spray*). The majority of proposals are symbolic and therefore limited to a specific domain due to the large effort involved in hand-coding semantic information (see Lauer [1995] for an extensive overview).

Lauer [1995] is the first to propose and evaluate an unsupervised probabilistic model of compound noun interpretation for domain independent text. By recasting the interpretation problem in terms of paraphrasing, Lauer assumes that the semantic relations of compound heads and modifiers can be expressed via prepositions that (in contrast to abstract semantic relations) can be found in a corpus. For example, in order to interpret *war story*, one needs to find in a corpus related paraphrases: *story about the war*, *story of the war*, *story in the war*, and so on. Lauer uses eight prepositions for the paraphrasing task (*of, for, in, at, on, from, with, about*). A simple model of compound noun paraphrasing is shown in (4):

$$p^* = \arg \max_p P(p|n_1, n_2) \quad (4)$$

Table XIV. Performance of Altavista, BNC, Interpolated, and Backoff Models for Compound Interpretation (Data From Lauer [1995])

Model	Altavista	BNC	Interpol	Backoff
$f(n_1, p)f(p, n_2)$	50.71#	26.42#	50.71#	50.71#
$f(n_1, p, n_2)$	55.71#	10.71	55.71#	55.71#
$f(n_1, p)f(p, n_2)/f(p)$	47.14	32.14	49.28	47.14
$f(n_1, p, n_2)/f(p)$	55.00	08.57	55.00	55.00

Table XV. Performance Comparison with the Literature for Compound Interpretation (Baseline: Always Selects *of*, the Most Frequent Interpretation in Lauer [1995] Data Set)

Model	Accuracy
Best BNC	26.42/†\$
Lauer [1995]: concept-based	28.00/†\$
Baseline	33.00/†\$
Lauer [1995]: word-based	40.00*/\$
Best Altavista	55.71*†‡
Best Interpol	55.71*†‡ \$
Best Backoff	55.71*†‡ \$

Lauer [1995] points out that this model contains one parameter for every triple  $\langle p, n_1, n_2 \rangle$ , and, as a result, hundreds of millions of training instances would be necessary. As an alternative to (4), he proposes the model in (5) which 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.

$$p^* = \arg \max_p \sum_{\substack{t_1 \in \text{cats}(n_1) \\ t_2 \in \text{cats}(n_2)}} P(t_1|p)P(t_2|p) \quad (5)$$

Here,  $t_1$  and  $t_2$  represent concepts in Roget’s Thesaurus. Lauer [1995] also experimented with a lexicalized version of (5) where probabilities are calculated on the basis of word (rather than concept) frequencies which Lauer obtained from Grolier’s Encyclopedia heuristically via pattern matching.

Lauer [1995] tested the model in (5) on 282 compounds that he selected randomly from Grolier’s Encyclopedia and annotated with their paraphrasing prepositions. The preposition *of* accounted for 33% of the paraphrases in this data set (see Baseline in Table XV). The concept-based model (see (5)) achieved an accuracy of 28% on this test set, whereas its lexicalized version reached an accuracy of 40% (see Table XV).

We attempted the interpretation task with the lexicalized version of the bigram model (see (5)) but also tried the more data intensive trigram model (see (4)), again in its lexicalized form. Furthermore, we experimented with several conditional and unconditional variants of (5) and (4) and weakly supervised models (i.e., backoff and interpolation). Co-occurrence frequencies were estimated from the Web using inflected queries (see Section 2.1). Determiners were inserted before nouns resulting in queries of the type *story/stories about* and *about the/a/0 war/wars* for the compound *war story*.

Table XVI. Performance of Altavista, BNC, Interpolated, and Backoff Models for Noun Countability Detection (Data From Baldwin and Bond [2003])

Model	Altavista		BNC		Interpol		Backoff	
	Count	Uncount	Count	Uncount	Count	Uncount	Count	Uncount
$f(n)$	86.94	90.53	87.69#	90.13#	87.69	90.13	87.69	90.30
$f(det, n)$	88.62#	91.53#	50.22	50.80	88.59#	91.42#	88.52#	91.48#

As shown in Table XIV, the best performance was obtained using the Web-based trigram model ( $f(n_1, p, n_2)$ ); this result was matched by the backoff and interpolation models.<sup>13</sup> Comparison with the literature in Table XV reveals that the best Altavista model significantly outperformed both the baseline and the best model in the literature (Lauer’s word-based model). The BNC model, on the other hand, achieved a performance that is not significantly different from the baseline and significantly worse than Lauer’s best model.

## 9. NOUN COUNTABILITY DETECTION

The next analysis task that we consider is the problem of determining the countability of nouns. Countability is the semantic property that specifies whether a noun can occur in singular and plural forms and affects the range of permissible modifiers. In English, nouns are typically either countable (e.g., *one dog, two dogs*) or uncountable (e.g., *some peace, \*one peace, \*two peaces*).

Baldwin and Bond [2003] propose a method for automatically learning the countability of English nouns from the BNC. They obtain information about noun countability by merging lexical entries from COMLEX [Grishman et al. 1994] and the ALTJ/E Japanese-to-English semantic transfer dictionary [Ikehara et al. 1991]. Words are classified into four classes: countable, uncountable, bipartite (e.g., *trousers*), and plural only (e.g., *goods*). A memory-based classifier is used to learn the four-way distinction on the basis of several linguistically motivated features such as: number of the head noun, number of the modifier, subject-verb agreement, plural determiners.

We devised unsupervised models for the countability learning task and evaluated their performance on Baldwin and Bond’s [2003] test data. We concentrated solely on countable and uncountable nouns since they account for the vast majority of the data. Two models were tested: (a) compare the frequency of the singular and plural forms of the noun, (b) compare the frequency of determiner-noun pairs that are characteristic of countable or uncountable nouns; the determiners used were *many* for countable and *much* for uncountable ones.

Unigram and bigram frequencies were estimated from the Web using literal queries. The performance of the BNC and Altavista models on the test set is given in Table XVI; interpolated and backoff models are also shown. Table XVII compares our results with state of the art.

<sup>13</sup>The models  $f(n_1, p)f(n_2, p)$  and  $f(n_1, p, n_2)$  obtained identical accuracies on the development set when using Altavista counts. The same behavior was observed with the backoff and interpolation models. This is why both models are signaled by “#” in Table XIV, although on the test set  $f(n_1, p, n_2)$  outperformed  $f(n_1, p)f(n_2, p)$ .

Table XVII. Performance Comparison with the Literature for Noun Countability Detection (Baseline: Always Selects Countable, the Majority Class in Baldwin and Bond [2003] Data Set)

Model	Count	Uncount
Baseline	74.60*‡\$	78.30*‡\$
Best BNC	87.69†‡\$	90.13†‡\$
Best Backoff	88.52*†‡\$	91.48*†‡\$
Best Interpol	88.59*†‡\$	91.42*†‡\$
Best Altavista	88.62*†‡	91.53*†‡
Baldwin and Bond [2003]	93.90*†	95.20*†\$

The best Altavista model is the determiner-noun model ( $f(det, n)$ ) that achieves 88.62% on countable and 91.53% on uncountable nouns. On the BNC, the simple unigram model performs best. Its performance was not statistically different from that of the best Altavista model. Note that the determiner-noun BNC models perform poorly, presumably due to data sparseness. Neither the interpolated nor the backoff model manage to significantly outperform the BNC or the best Altavista model. Table XVII shows that both the Altavista and BNC models significantly outperform the baseline (relative frequency of the majority class on the gold-standard data). Both models perform significantly worse than the best model in the literature [Baldwin and Bond 2003]; this is a supervised model that uses many more features than just singular/plural frequency and determiner-noun frequency.

## 10. PP ATTACHMENT DISAMBIGUATION

A pervasive problem in natural language analysis is resolving syntactic attachment ambiguities. PP attachment ambiguities in particular have received considerable attention in the NLP literature. The ambiguity is exemplified in the following example, where the PP *with pockets* can be either attached to the noun *shirt* (and mean that the shirt has pockets) or to the verb *bought* (and mean that the pockets were used to purchase the shirt).

[4.] I bought the shirt with pockets.

Previous work has framed the problem as a classification task where the goal is to predict either verb or noun attachment, given the head verb  $v$ , the head noun  $n_1$  of the object of the verb, the preposition  $p$ , and optionally, the head noun  $n_2$  of the object of the preposition. Hindle and Rooth [1993] propose a partially supervised approach in which a parser is used to extract  $\langle v, n_1, p \rangle$  tuples from a corpus. These data are then used to estimate lexical association scores based on which attachment decisions are made. Subsequent work has used  $\langle v, n_1, p, n_2 \rangle$  tuples extracted from the Penn Treebank to train supervised models including a maximum entropy model [Ratnaparkhi et al. 1993], a backoff model [Collins and Brooks 1995], transformation-based [Brill and Resnik 1994], and memory-based learning [Zavrel et al. 1997]. Unsupervised approaches to PP-attachment have been proposed by Ratnaparkhi [1998] and Pantel and Lin [2000].

Table XIX compares the performance of the different approaches on Ratnaparkhi et al.’s [1993] standard data set.<sup>14</sup> The latter was obtained from the Penn Treebank by identifying examples of VPs containing a [V NP PP] sequence. For each such sequence, the head verb, the first head noun, preposition, and second head noun were extracted along with the attachment decision (noun or verb). The training/test set therefore contained solely head words (e.g., the VP [[*joined [the board]] [as a nonexecutive director]] would give the tuple (*joined, board, as, director*)) and the majority of previous models rely primarily on head words for disambiguating PP-attachment. Table XIX also includes two baselines (always choose noun attachment; choose the most likely attachment for a given preposition) and two upper bounds reported by Ratnaparkhi et al. [1993] (human judgments are based on the head words or the whole sentence).*

Recently, Volk [2001] proposed an unsupervised model based on Web counts to resolve PP-attachment ambiguities. His approach was tested on German data and achieved an accuracy of 73.08%. We applied this approach to a random sample of 1000 tokens from the standard test set for English [Ratnaparkhi et al. 1993]. Three models were tested. Model 1 uses Hindle and Rooth’s [1993] lexical association ratio, computed as the probability of the preposition given the first noun divided by the probability of preposition given the verb:<sup>15</sup>

$$R_1 = \frac{P(p|n_1)}{P(p|v)} = \frac{\frac{f(n_1,p)}{f(n_1)}}{\frac{f(v,p)}{f(v)}} \quad (6)$$

Noun attachment is chosen if this ratio is greater than or equal to one, verb attachment if it is less than one. Model 2 was proposed by Volk [2001]; here the lexical association is computed as the joint probability of the first noun, the preposition, and the second noun divided by the joint probability of the verb, the preposition, and the second noun:

$$R_2 = \frac{P(n_1, p, n_2)}{P(v, p, n_2)} = \frac{f(n_1, p, n_2)}{f(v, p, n_2)} \quad (7)$$

Model 3 (again due to Volk [2001]) is the same as Model 2, but the probabilities of the first noun and the verb are included as a way of normalizing the lexical association ratio:

$$R_3 = \frac{P(p, n_2|n_1)}{P(p, n_2|v)} = \frac{\frac{f(n_1,p,n_2)}{f(n_1)}}{\frac{f(v,p,n_2)}{f(v)}} \quad (8)$$

We estimated the probabilities of Models 1–3 using Web counts, employing literal queries or inflected queries. As before, the Web-based models were also compared to the BNC, and we additionally experimented with backoff and interpolated models. Table XVIII gives an overview of the performance of the various

<sup>14</sup>Hindle and Rooth’s [1993] original results are not comparable as they used a different test set.

<sup>15</sup>Hindle and Rooth’s [1993] original model uses smoothed estimates for the probabilities in (6); no smoothing is employed in our experiments since we aim to examine whether Web counts can be used as a substitute for counts that are unseen in a given corpus.

Table XVIII. Performance of Altavista, BNC, Interpolated, and Backoff Models for PP Attachment (Data From Ratnaparkhi et al. [1993])

Model	Altavista	BNC	Interpol	Backoff
<b>Inflected queries</b>				
$f(n_1, p)/f(n_1) : f(v, p)/f(v)$	69.40#	74.40#	69.80#	72.20
$f(n_1, p, n_2) : f(v, p, n_2)$	67.20	60.00	68.40	68.00
$f(n_1, p, n_2)/f(n_1) : f(v, p, n_2)/f(v)$	72.40	63.40	72.40	72.20
<b>Literal queries</b>				
$f(n_1, p)/f(n_1) : f(v, p)/f(v)$	0.664	0.704	0.688	0.702#
$f(n_1, p, n_2) : f(v, p, n_2)$	0.434	0.408	0.436	0.428
$f(n_1, p, n_2)/f(n_1) : f(v, p, n_2)/f(v)$	0.702	0.656	0.700	0.706

Table XIX. Performance Comparison with the Literature for PP Attachment

Model	Accuracy
Always noun attachment	56.90*‡§
Best Altavista	69.40*†‡
Best Interpol	69.80*†‡§
Best Backoff	70.20*†‡§
Most likely for each preposition	72.20*†‡§
Best BNC	74.40†‡§
Ratnaparkhi et al. [1993]	81.60*†‡§
Hindle and Rooth [1993]	82.10*†‡§
Brill and Resnik [1994]	84.10*†‡§
Zavrel et al. [1997]	84.10*†‡§
Pantel and Lin [2000]	84.30*†‡§
Collins and Brooks [1995]	84.50*†‡§
Stetina and Nagao [1997]	88.10*†§
Average Human (4 head words only)	88.20*†§
Average Human (whole sentence)	93.20*†‡§

models. Note that inflected queries generally outperformed literal queries both for the Web and the BNC. This is perhaps not surprising given that literal queries are more vulnerable to sparse data. The Web-based models are compared against the literature in Table XIX.

The highest Web-based performance is 69.40%, achieved by Model 3 with inflected queries. The same model on the BNC obtains an accuracy of 74.40%. The Web-based model is significantly better than the baseline of 56.80% (defaulting to noun attachment). It reaches the performance of the second baseline (always choosing the most likely attachment for a given preposition). Note, however, that this is a supervised baseline as it requires information about the attachment preferences of prepositions. The backoff and interpolation models achieve performances comparable to the best Altavista model. However, none of these three models significantly outperforms the best BNC model.

In general, none of the models attempted here compare favorably with previous approaches in the literature, most of which, including unsupervised ones, rely on the availability of grammatical information (i.e., POS tags) and syntactic knowledge (either in the form of parse trees or syntactic chunks) to provide cues for the choice of the PP-attachment site. Stetina and Nagao [1997] even

Table XX. Overview of the Performance of the Web-Based Models Investigated in this Article

Task	Ling	Type	Base	BNC	Lit
MT candidate selection*	Sem	Generation	△	≡	△/≡
Spelling correction	Syn/Sem	Generation	△	△	▽
Adjective ordering*	Sem	Generation	△	△	≡
Article generation	Sem	Generation	△	△	△
Compound bracketing	Syn	Analysis	△	≡	≡
Compound interpretation	Sem	Analysis	△	△	△
Countability detection	Sem	Analysis	△	≡	▽
PP attachment	Syn/Sem	Analysis	△	≡	▽

Key: Ling: linguistic knowledge; Type: type of task; Base: comparison against baseline; BNC: comparison against BNC-based model; Lit: comparison against best model in the literature. The symbols indicate significance (△: significantly better; ≡: not significantly different; ▽: significantly worse). The \* indicates that the Web-based model used interpolation or backoff.

make use of a semantic dictionary. Neither syntactic nor semantic information is available to our models, and it is a matter of future work to determine whether the naive models presented here can be combined with some of the more sophisticated approaches in the face of data sparseness.

## 11. DISCUSSION

In this article, we examined whether simple unsupervised Web-based models can be devised for a variety of NLP tasks. The tasks were selected so that they cover both syntax and semantics, both generation and analysis, and a wider range of  $n$ -grams and parts of speech than have been previously explored. In order to quantify the effect of the use of Web counts on each task, we compared Web-based models against identical models whose parameters were estimated on the BNC. While the BNC is a relatively large corpus (100M words), it offers considerably less data than the Web.

We also explored the performance of two models that combine Web counts and corpus counts. The backoff model uses corpus counts unless they fall below a threshold in which case the model backs off to Web counts. The interpolation model uses the sum of Web counts and corpus counts with their relative contribution weighted by an interpolation factor.

A summary of our findings is given in Table XX. The Table lists the tasks we attempted, whether they are semantic or syntactic, and whether they concern the analysis or generation of natural language. The Table also compares the best Web-based model for each task against the baseline, the BNC, and the best model in the literature, and states whether significantly lower or higher performance was obtained by the Web-based model. Note that in two cases, this comparison is based on a combined model (using interpolation or backoff) as it outperformed the model based on Web counts alone. These cases are marked with an asterisk.

For all tasks we attempted, the Web-based models significantly outperform the baseline. However, generation and analysis tasks seem to behave differently when it comes to comparing Web-based models with BNC-based models and state-of-the-art models in the literature. We will discuss both types of tasks in turn.

For all generation tasks (with one exception), we found that Web-based models significantly outperform the corpus-based models. An exception is candidate selection for MT, where there was no difference between Web-based and corpus-based models. As for the comparison with the literature, here we obtained a mixed picture: for spelling correction, Web-based models perform worse than the state-of-the-art, while for article generation, they outperform it. For adjective ordering, Web-based models match the state-of-the-art, while for candidate selection, they either match the state-of-the-art (for high ambiguity items) or outperform it (for low ambiguity items).

It is not surprising that the Web performs better at generation tasks than the BNC. It has been argued that corpus-based methods can offer ways to address knowledge gaps and collocational idiosyncrasies in generation while avoiding knowledge intensive hand-coding [Knight and Hatzivassiloglou 1995; Langkilde and Knight 1998]. Since there are many ways of realizing a given semantic content, the likelihood of a relevant construction is directly related to the size of the corpus from which it is estimated. Our results show that Web-scale data can be of benefit to generation applications that exploit corpus-based knowledge. Our results are also consistent with Keller and Lapata [2003] who investigate another generation-related task, the plausibility of predicate argument constructions. Their findings show that Web frequencies provide more accurate estimates of plausibility than BNC frequencies. The reason for this seems to be that the Web is much larger than the BNC (about 1,000 times). The large size seems to compensate for the inadequacies of Web data, that is, for the fact that simple heuristics were used to obtain Web counts, and for the noise caused by tokenization errors, the absence of punctuation, and so on.

Turning now to the analysis tasks in Table XX, we fail to observe an advantage of the Web models over the BNC models (apart from one case). Also, all Web-based models perform worse or equal to the state-of-the-art models reported in the literature (again with one exception). An explanation for this is that analysis tasks involve decisions that are not directly observable in the data. For example, the attachment choice for a PP or the bracketing for a compound noun cannot be reduced to counting alternatives in a corpus unless the corpus is explicitly annotated with syntactic structure. The fact that the Web is much larger than the BNC makes little difference, since neither data source contains the formation that is important for the tasks at hand. Most approaches in the literature therefore take advantage of information that goes beyond the words in the corpus: they are trained on treebanks or on corpora annotated with part-of-speech tags.

It is worth noting that the Web models outperform the BNC and the state-of-the-art in the compound noun interpretation task. Recall that this task was cast by Lauer [1995] as a generation problem. The meaning of a compound was approximated by a paraphrase consisting of a preposition co-occurring with the compound modifier and head (e.g., *war story* can mean *story about the war*, *story of the war*, *story in the war*). We can easily look for such paraphrases in a corpus without any preprocessing (i.e., parsing or POS-tagging). However, since there are many alternative paraphrases to choose from, it is not surprising

that Web-based models significantly outperform models whose parameters are estimated from the BNC or smaller corpora.

For all generation and analysis tasks we attempted, our Web-based models were compared against state-of-the-art models which are reported in the literature and are similar in one important aspect. Most of them are supervised models that have access not only to simple bigram or trigram frequencies, but also to corpus external linguistic information such as part-of-speech tags, semantic restrictions, or context. When unsupervised Web-based models are compared against supervised methods that employ a wide variety of features, we observe that having access to linguistic information makes up for the lack of vast amounts of data. The only two unsupervised models in the literature are Lauer's [1995] models of compound noun bracketing and interpretation. In the case of compound interpretation, Lauer's [1995] model is truly unsupervised, and the Web-based approach outperforms it. In the case of compound noun bracketing, Lauer [1995] makes use of Roget's Thesaurus in order to alleviate data sparseness, hence his model is not truly unsupervised (and the Web-based model yields the same performance).

Furthermore, this article also reported experiments on combining Web counts and corpus counts. We showed that a small performance gain can sometimes be obtained by using interpolation or backoff. For two tasks we found that a combined model outperforms a straight Web model, in the sense that the combined model was significantly better than the corpus-based model or the model in the literature, while the straight Web model failed to achieve a significant difference (see the asterisks in Table XX).

To summarize, our results indicate that large data sets such as those obtained from the Web are not the panacea that they are claimed to be (at least implicitly) by authors such as Grefenstette [1998] and Keller and Lapata [2003]. Rather, in our opinion, Web-based models should be used as a new *baseline* for NLP tasks. The Web baseline indicates how much can be achieved with a simple, unsupervised model based on  $n$ -grams with access to a huge data set. This baseline is more realistic than baselines obtained from standard corpora; it is generally harder to beat as our comparisons with the BNC baseline throughout this article have shown. Note that, for certain tasks, the performance of a Web baseline model might actually be sufficient so that the effort of constructing a sophisticated supervised model and annotating the necessary training data can be avoided: recall that for three tasks, our Web-based models outperformed the best model in the literature (for MT candidate selection, article generation, and compound interpretation, see Table XX).

## 12. FUTURE WORK

An important future direction lies in the application and evaluation of Web models across domains. Rather than looking into different tasks, we will concentrate on a single task and examine whether Web counts (and their combination with corpus counts) yield improvements for different domains (e.g., sports, medicine) and registers (e.g., speech vs. text). Most modeling work to date has focused on written texts, and the effect of Web counts for speech data has not been

rigorously quantified (an exception is language modeling work, e.g., Zhu and Rosenfeld [2001]). Furthermore, the performance of Web-based models has been mainly compared against models trained on established corpora such as the BNC, the Brown corpus, or the Penn Treebank. It remains to be seen whether Web-based models are robust across domains with maximally different vocabularies and stylistic conventions. A related issue is the degree to which different domains are represented in Web data and whether there is a threshold (in terms of the number of pages indexed by a search engine) above which Web counts become useful.

Another possibility that needs further investigation is the combination of Web-based models with supervised methods. This can be done with ensemble learning methods or simply by using Web-based frequencies (or probabilities) as features (in addition to linguistically motivated features) to train supervised classifiers. The latter approach is taken, for example, by Modjeska et al. [2003] who show that an F-measure increase of 11.4% can be achieved when Web-based counts are added to a set of morphosyntactic features for resolving *other-anaphora* in a supervised setting.

Finally in this article, we have compared relatively simple Web-based models against their state-of-the-art alternatives by assessing model performance on the same data sets. A challenge for the future lies in replicating state-of-the-art models using Web-scale data annotated with linguistically rich information such as parts of speech and syntactic structure. It is probably infeasible to store a snapshot of the whole Web and process it linguistically—estimates regarding the size of the Web not only vary (see Kilgariff and Grefenstette [2003]) but are constantly increasing. However, sampling techniques could provide an alternative. For instance, the pages returned by a query could be downloaded and processed linguistically; if a query returns too many hits, then sampling could be used to select the most informative or least similar pages for processing.

#### ACKNOWLEDGMENTS

We thank the anonymous referees for helpful comments and suggestions. We are grateful to Tim Baldwin, Silviu Cucerzan, Mark Lauer, John Lee, Rob Malouf, Detlef Prescher, and Adwait Ratnaparkhi for making their data sets available. A preliminary version of this work was published as Lapata and Keller [2004]; we thank the anonymous reviewers of that paper for their comments.

#### REFERENCES

- BALDWIN, T. AND BOND, F. 2003. Learning the countability of English nouns from corpus data. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan. 463–470.
- BANKO, M. AND BRILL, E. 2001a. Mitigating the paucity-of-data problem: Exploring the effect of training corpus size on classifier performance for natural language processing. In *Proceedings of the 1st International Conference on Human Language Technology Research*. J. Allan, Ed. Morgan Kaufmann, San Francisco.

- BANKO, M. AND BRILL, E. 2001b. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*. Toulouse, France.
- BRILL, E. AND RESNIK, P. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *Proceedings of the 15th International Conference on Computational Linguistics*. Kyoto, Japan. 1198–1204.
- BULYKO, I., OSTENDORF, M., AND STOLCKE, A. 2003. Getting more mileage from Web text sources for conversational speech language modeling using class-dependent mixtures. In *Companion Volume of the Proceedings of HLT-NAACL 2003: Short Papers*. Edmonton, Canada. 7–9.
- BURNARD, L. 1995. *The Users Reference Guide for the British National Corpus*. British National Corpus Consortium, Oxford University Computing Service.
- CAO, Y. AND LI, H. 2002. Base noun phrase translation: Using Web data and the EM algorithm. In *Proceedings of the 19th International Conference on Computational Linguistics*. Taipei, Taiwan. 127–133.
- CHKLOVSKI, T. AND PANTEL, P. 2004. VERBOCEAN: Mining the Web for fine-grained semantic verb relations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, D. Lin and D. Wu, Eds. Barcelona, Spain. 33–40.
- COLLINS, M. AND BROOKS, J. 1995. Prepositional phrase attachment through a backed-off model. In *Proceedings of the 3rd Workshop on Very Large Corpora*. D. Yarowsky and K. W. Church, Eds. Cambridge, MA. 27–38.
- CORLEY, S., CORLEY, M., KELLER, F., CROCKER, M. W., AND TREWIN, S. 2001. Finding syntactic structure in unparsed corpora: The Gsearch corpus query system. *Comput. Humanities* 35, 2, 81–94.
- CUCERZAN, S. AND YAROWSKY, D. 2002. Augmented mixture models for lexical disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. J. Hajič and Y. Matsumoto, Eds. Philadelphia, PA. 33–40.
- DAGAN, I. AND ITAI, A. 1994. Machine translation divergences: A formal description and proposed solution. *Computat. Ling.* 20, 4, 563–597.
- DUMAIS, S., BANKO, M., BRILL, E., LIN, J., AND NG, A. 2002. Web question answering: Is more always better? In *Proceedings of the 25th ACM Conference on Research and Development in Information Retrieval*. Tampere, Finland. 291–298.
- GOLDING, A. R. 1995. A Bayesian hybrid method for context-sensitive spelling correction. In *Proceedings of the 3rd Workshop on Very Large Corpora*. D. Yarowsky and K. W. Church, Eds. Cambridge, MA. 39–53.
- GOLDING, A. R. AND ROTH, D. 1999. A Winnow-based approach to context sensitive spelling correction. *Machine Learn.* 34, 1-3, 1–25.
- GOLDING, A. R. AND SCHABES, Y. 1996. Combining trigram-based and feature-based methods for context-sensitive spelling correction. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*. Santa Cruz, CA. 71–78.
- GREFENSTETTE, G. 1998. The World Wide Web as a resource for example-based machine translation tasks. In *Proceedings of the ASLIB Conference on Translating and the Computer*. London, UK.
- GRISHMAN, R., MACLEOD, C., AND MEYERS, A. 1994. COMPLEX Syntax: Building a computational lexicon. In *Proceedings of the 15th International Conference on Computational Linguistics*. Kyoto, Japan. 268–272.
- HILDEBRANDT, W., KATZ, B., AND LIN, J. 2004. Answering definition questions with multiple knowledge sources. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. Boston, MA. 49–56.
- HINDLE, D. AND ROTH, M. 1993. Structural ambiguity and lexical relations. *Computat. Ling.* 19, 1, 103–120.
- IKEHARA, S., SHIRAI, S., YOKOO, A., AND NAKAIWA, H. 1991. Toward an MT system without pre-editing effects of new methods in ALT-J/E. In *Proceedings of the 3rd Machine Translation Summit*. Washington, DC. 101–106.
- JONES, M. P. AND MARTIN, J. H. 1997. Contextual spelling correction using latent semantic analysis. In *Proceedings of the 5th Conference on Applied Natural Language Processing*. Washington, DC. 166–173.

- KELLER, F. AND LAPATA, M. 2003. Using the Web to obtain frequencies for unseen bigrams. *Computat. Ling.* 29, 3, 459–484.
- KILGARIFF, A. AND GREFFENSTETTE, G. 2003. Introduction to the special issue on the Web as corpus. *Computat. Ling.* 29, 3, 333–347.
- KNIGHT, K. AND CHANDER, I. 1994. Automated postediting of documents. In *Proceedings of 12th National Conference on Artificial Intelligence*. Seattle, WA. 770–784.
- KNIGHT, K. AND HATZIVASSILOPOULOS, V. 1995. Two-level, many paths generation. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. Cambridge, MA. 252–260.
- LANGKILDE, I. AND KNIGHT, K. 1998. Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*. Montréal, Canada. 704–710.
- LAPATA, M. AND KELLER, F. 2004. The Web as a baseline: Evaluating the performance of unsupervised Web-based models for a range of NLP tasks. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. Boston, MA. 121–128.
- LAUER, M. 1995. Corpus statistics meet the noun compound: Some empirical results. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. Cambridge, MA. 47–54.
- LEE, J. 2004. Automatic article restoration. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. Boston, MA. 31–36.
- MALOUF, R. 2000. The order of prenominal adjectives in natural language generation. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*. Hong Kong. 85–92.
- MANGU, L. AND BRILL, E. 1997. Automatic rule acquisition of spelling correction. In *Proceedings of the 14th International Conference on Machine Learning*. Nashville, TN. 187–194.
- MARCUS, M. P., SANTORINI, B., AND MARCINKIEWICZ, M. A. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computat. Ling.* 19, 2, 313–330.
- MIHALCEA, R. AND MOLDOVAN, D. 1999. A method for word sense disambiguation of unrestricted text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, MD. 152–158.
- MINNEN, G., BOND, F., AND COPESTAKE, A. 2000. Memory-based learning for article generation. In *Proceedings of the 4th Workshop on Computational Natural Language Learning*. Lisbon, Portugal. 43–48.
- MODJESKA, N., MARKERT, K., AND NISSIM, M. 2003. Using the Web in machine learning for other-anaphora resolution. In *Proceedings of the 8th Conference on Empirical Methods in Natural Language Processing*. Sapporo, Japan. 176–183.
- PANTEL, P. AND LIN, D. 2000. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*. Hong Kong. 101–108.
- PRESCHER, D., RIEZLER, S., AND ROTH, M. 2000. Using a probabilistic class-based lexicon for lexical ambiguity resolution. In *Proceedings of the 18th International Conference on Computational Linguistics*. Saarbrücken, Germany. 649–655.
- PUSTEJOVSKY, J., BERGLER, S., AND ANICK, P. 1993. Lexical semantic techniques for corpus analysis. *Computat. Ling.* 19, 3, 331–358.
- RATNAPARKHI, A. 1998. Unsupervised statistical models for prepositional phrase attachment. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*. Montréal, Canada. 1079–1085.
- RATNAPARKHI, A., REYNAR, J., AND ROUKOS, S. 1993. A maximum entropy model for prepositional phrase attachment. In *Proceedings of the ARPA Workshop on Human Language Technology*. Plainsboro, NJ.
- RESNIK, P. AND SMITH, N. A. 2003. The Web as a parallel corpus. *Computat. Ling.* 29, 3, 349–380.
- RESNIK, P. S. 1993. Selection and information: A class-based approach to lexical relationships. Ph.D. thesis, University of Pennsylvania.

- RIGAU, G., MAGNINI, B., AGIRRE, E., AND CARROLL, J. 2002. Meaning: A roadmap to knowledge technologies. In *Proceedings of COLING Workshop on a Roadmap for Computational Linguistics*. Taipei, Taiwan.
- SANTAMARÍA, C., GONZALO, J., AND VERDEJO, F. 2003. Automatic association of Web directories with word senses. *Computat. Ling.* 29, 3, 485–502.
- SHAW, J. AND HATZIVASSILOGLOU, V. 1999. Ordering among premodifiers. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, MD. 135–143.
- SHINZATO, K. AND TORISAWA, K. 2004. Acquiring hyponymy relations from Web documents. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. Boston, MA. 73–80.
- SORICUT, R. AND BRILL, E. 2004. Automatic question answering: Beyond the factoid. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. Boston, MA. 57–64.
- STETINA, J. AND NAGAO, M. 1997. Corpus-based PP attachment ambiguity resolution with a semantic dictionary. In *Proceedings of the 5th Workshop on Very Large Corpora*. Beijing, China. 66–80.
- SZPEKTOR, I., TANEV, H., DAGAN, I., AND COPPOLA, B. 2004. Scaling Web-based acquisition of entailment relations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. D. Lin and D. Wu, Eds. Barcelona, Spain. 41–48.
- VOLK, M. 2001. Exploiting the WWW as a corpus to resolve PP attachment ambiguities. In *Proceedings of the Corpus Linguistics Conference*. P. Rayson, A. Wilson, T. McEnery, A. Hardie, and S. Khoja, Eds. Lancaster, UK. 601–606.
- WAY, A. AND GOUGH, N. 2003. wEBMT: Developing and validating an example-based machine translation system using the World Wide Web. *Computat. Ling.* 29, 3, 421–457.
- ZAVREL, J., DAELEMANS, W., AND VEENSTRA, J. 1997. Resolving PP attachment ambiguities with memory-based learning. In *Proceedings of the 1st Workshop on Computational Natural Language Learning*. Madrid, Spain. 136–144.
- ZHU, X. AND ROSENFELD, R. 2001. Improving trigram language modeling with the World Wide Web. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing*. Salt Lake City, Utah.

Received May 2005; accepted June 2005 by Marcello Federico