The Web as a Baseline: Evaluating the Performance of Unsupervised Web-based Models for a Range of NLP Tasks

Mirella Lapata
Department of Computer Science
University of Sheffield
211 Portobello St., Sheffield S1 4DP
mlap@dcs.shef.ac.uk

Frank Keller
School of Informatics
University of Edinburgh
2 Buccleuch Pl., Edinburgh EH8 9LW
keller@inf.ed.ac.uk

Abstract

Previous work demonstrated that web counts can be used to approximate bigram frequencies, and thus should be useful for a wide variety of NLP tasks. So far, only two generation tasks (candidate selection for machine translation and confusion-set disambiguation) have been tested using web-scale data sets. The present paper investigates if these results generalize to tasks covering both syntax and semantics, both generation and analysis, and a larger range of *n*-grams. For the majority of tasks, we find that simple, unsupervised models perform better when *n*-gram frequencies are obtained from the web rather than from a large corpus. However, in most cases, web-based models fail to outperform more sophisticated state-of-the-art models trained on small corpora. We argue that web-based models should therefore be used as a baseline for, rather than an alternative to, standard models.

1 Introduction

Keller and Lapata (2003) investigated 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.

Task	*n*	POS	Ling	Type
MT candidate select.	1,2	V, N	Sem	Generation
Spelling correction	1,2,3	Any	Syn/Sem	Generation
Adjective ordering	1,2	Adj	Sem	Generation
Compound bracketing	1,2	N	Syn	Analysis
Compound interpret.	1,2,3	N, P	Sem	Analysis
Countability detection	1,2	N, Det	Sem	Analysis

Table 1: Overview of the tasks investigated in this paper (*n*: size of *n*-gram; POS: parts of speech; Ling: linguistic knowledge; Type: type of task)

This paper aims to address these questions. We start by using web counts for two generation tasks for which the use of large data sets has shown promising results: (a) target language candidate selection for machine translation (Grefenstette, 1998) and (b) context sensitive spelling correction (Banko and Brill, 2001a,b). 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, and (f) noun countability detection. Table 1 gives an overview of these tasks and their properties.

In all cases, we propose a simple, unsupervised *n*-gram based model whose parameters are estimated using web counts. We compare this model both against a baseline (same model, but parameters estimated on the BNC) and against state-of-the-art models from the literature, which are either supervised (i.e., use annotated training data) or unsupervised but rely on taxonomies to recreate missing counts.

2 Method

Following Keller and Lapata (2003), web counts for *n*-grams were obtained using a simple heuristic based on queries to the search engine Altavista.[1] In this approach, the web count for a given *n*-gram is simply the number of hits (pages) returned by the search engine for the queries generated for this *n*-gram. Three different types of queries were used for the NLP tasks in the present paper:

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

Near queries use Altavista's `NEAR` operator to expand the *n*-gram; `a NEAR b` means that `a` has to occur in the same ten word window as `b`; the window is treated as a bag of words (e.g., *history changes* expands to `"history" NEAR "changes"`).

Inflected queries are performed 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"`, `"histories change"`, `"history changed"`, etc.). John Carroll's suite of morphological tools (`morpha`, `morphg`, and `ana`) was used to generate inflected forms of verbs and nouns.[2] In certain cases (detailed below), determiners were inserted before nouns in order to make it possible to recognize simple NPs. This insertion was limited to *a/an*, *the*, and the empty determiner (for bare plurals).

All queries (other than the ones using the `NEAR` operator) were performed as exact matches (using quotation marks in Altavista). All search terms were submitted to the search engine in lower case. If a query consists of a single, highly frequent word (such as *the*), Altavista will 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 were used in some of the models detailed below. Sometimes the search engine fails to return a hit for a given *n*-gram (for any of its morphological variants). We smooth zero counts by setting them to .5.

For all tasks, the web-based models are compared against identical models whose parameters were 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. The BNC counts were retrieved using the Gsearch corpus query tool (Corley et al., 2001); the morphological query expansion was the same as for web queries; the `NEAR` operator was simulated by assuming a window of five words to the left and five to the right. Gsearch was used to search solely for adjacent words; no POS information was incorporated in the queries, and no parsing was performed.

#	best model on development set
$*$ $\not*$	(not) sign. different from best BNC model on test set
$\dagger$ $\not\dagger$	(not) sign. different from baseline
$\ddagger$ $\not\ddagger$	(not) sign. different from best model in the literature

Table 2: Meaning of diacritics indicating statistical significance (χ^2 tests)

For all of our tasks, we have to select either the best of several possible models or the best parameter setting for a single model. We therefore require a separate development set. This was achieved 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 then compare the test set performance of this optimal model to the performance of the models reported in the literature. It is important to note that the figures taken from the literature were typically obtained on the whole gold standard data set, and hence may differ from the performance on our test set. We work on the assumption that such differences are negligible.

We use χ^2 tests to determine whether the performance of the best web model on the test set is significantly different from that of the best BNC model. We also determine whether both models differ significantly from the baseline and from the best model in the literature. A set of diacritics is used to indicate significance throughout this paper, see Table 2.

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 and pragmatic factors. The task is illustrated in (1) 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

[1] We did not use Google counts, as Google limits the number of queries to 1000 per day, which makes the process of retrieving a large number of web counts very time consuming.

[2] The tools can be downloaded from `http://www.cogs.susx.ac.uk/lab/nlp/carroll/morph.html`.

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 of 1,340 and 814 bilingual sentence pairs, with an average of 8.63 and 2.83 translations for the object noun, respectively. Table 4 lists Prescher et al.'s results for the two corpora and for both models together with a random baseline (select a target noun at random) and a frequency baseline (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). Table 3 compares the web-based models against the BNC models. For both the high ambiguity and the low ambiguity data set, we find that the performance of the best Altavista model is not significantly different from that of the best BNC model. Table 4 compares our simple, unsupervised methods with the two sophisticated class-based models discussed above. 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. However, both models significantly outperform the baseline. This holds for both the high and low ambiguity data sets.

	Altavista		BNC	
Model	high ambig	low ambig	high ambig	low ambig
$f(v,n)$	45.74	68.73#$\not\ast$	45.89#	70.06#
$f(v,n)/f(n)$	45.16#$\not\ast$	64.96	46.18	66.07

Table 3: Performance of Altavista counts and BNC counts for candidate selection for MT (data from Prescher et al. 2000)

Model	high ambig	low ambig
Random baseline	14.20	45.90
Frequency baseline	31.90	45.50
Prescher et al. (2000): verb-argument	43.30	61.50
Best Altavista	45.16†$\not\ddagger$	68.73†$\not\ddagger$
Best BNC	45.89†$\not\ddagger$	70.06†$\not\ddagger$
Prescher et al. (2000): verb-object	49.40	68.20

Table 4: Performance comparison with the literature for candidate selection for MT

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 (4) where *principal* was typed when *principle* was intended.

(2) Introduction of the dialogue *principal* proved strikingly effective.

The task can be viewed as generation task, as 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 (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 6. 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

Model	Alta	BNC	Model	Alta	BNC
$f(t)$	72.98	70.00	$f(w_1,t,w_2)/f(t)$	87.77	76.33
$f(w_1,t)$	84.40	83.02	$f(w_1,w_2,t)/f(t)$	86.27	74.47
$f(t,w_1)$	84.89	82.74	$f(t,w_2,w_2)/f(t)$	84.94	74.23
$f(w_1,t,w_2)$	89.24#*	77.13	$f(w_1,t,w_2)/f(w_1,t)$	80.70	73.69
$f(w_1,w_2,t)$	87.13	74.89	$f(w_1,t,w_2)/f(t,w_2)$	82.24	75.10
$f(t,w_1,w_2)$	84.68	75.08	$f(w_1,w_2,t)/f(w_2,t)$	72.11	69.28
$f(w_1,t)/f(t)$	82.81	77.84	$f(t,w_1,w_2)/f(t,w_1)$	75.65	72.57
$f(t,w_1)/f(t)$	77.49	80.71#			

Table 5: Performance of Altavista counts and BNC counts for context sensitive spelling correction (data from Cucerzan and Yarowsky 2002)

Model	Accuracy
Baseline BNC	70.00
Baseline Altavista	72.98
Best BNC	80.71†‡
Golding (1995)	81.40
Jones and Martin (1997)	84.26
Best Altavista	89.24†‡
Golding and Schabes (1996)	89.82
Mangu and Brill (1997)	92.79
Cucerzan and Yarowsky (2002)	92.20
Golding and Roth (1999)	94.23

Table 6: Performance comparison with the literature for context sensitive spelling correction

the Brown corpus, using 80% for training and 20% for testing.[3]

We devised a simple, unsupervised method for performing spelling correction using web counts. The method takes into account collocational features, i.e., 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). Ties are resolved by comparing the unigram frequencies of the words in the confusion set and defaulting to the word with the highest one. Table 5 shows the types of collocations we considered and their corresponding accuracy. The baseline ($f(t)$) in Table 5 was obtained by always choosing the most frequent unigram in the confusion set. We used the same test set (2056 tokens from the Brown corpus) and confusion sets as Golding and Schabes (1996), Mangu and Brill (1997), and Cucerzan and Yarowsky (2002).

Table 5 shows that the best result (89.24%) 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 ($f(w_1,t,w_2)$). 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 6 shows that both the best Altavista model and the best BNC model outperform their respective baselines. A comparison with the literature shows that the best Altavista model outperforms Golding (1995), Jones and Martin (1997) and performs similar to Golding and Schabes (1996). 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, it 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 on semantic criteria that account for different orders (e.g., age $\prec$ color, value $\prec$ 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 $\{a,b\}$, 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 $a \prec c$ and $c \prec b$, then $a \prec b$. Malouf (2000) further proposes a back-off 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 proposes 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 uses memory-based learning as a means to encode morphological and semantic similarities among different adjective orders. 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

[3] An exception is Golding (1995), who uses the entire Brown corpus for training (1M words) and 3/4 of the Wall Street Journal corpus (Marcus et al., 1993) for testing.

Model	Altavista	BNC
$f(a_1,a_2) : f(a_2,a_1)$	89.6#$*\not\ddagger$	80.4#‡
$f(a_1,a_2)/f(a_2) : f(a_2,a_1)/f(a_1)$	83.2	77.0
$f(a_1,a_2)/f(a_1) : f(a_2,a_1)/f(a_2)$	80.2	80.6
Malouf (2000): memory-based	–	91.0

Table 7: Performance of Altavista counts and BNC counts for adjective ordering (data from Malouf 2000)

$\langle b,a \rangle$).

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 above 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 (see Table 7). 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 7). Web counts were obtained by submitting literal queries to Altavista (see Section 2). We used the same 263,838 adjective pairs that Malouf 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 26,271 queries to Altavista would be fairly time-consuming, a random sample of 1000 sequences was obtained from the test corpus and the web frequencies of these pairs were retrieved. The best Altavista model significantly outperformed the best BNC model, as indicated in Table 7. We also found that there was no significant difference between the best Altavista model and the best model reported by Malouf, a supervised method using positional probability estimates from the BNC and morphological variants.

6 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 follows: given a three word compound n_1 n_3 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 (4) for the dependency model and (5) for the adjacency model).

Model	Alta	BNC
Baseline	63.93	63.93
$f(n_1,n_2) : f(n_2,n_3)$	77.86	66.39
$f(n_1,n_2) : f(n_1,n_3)$	78.68#$*$	65.57
$f(n_1,n_2)/f(n_1) : f(n_2,n_3)/f(n_2)$	68.85	65.57
$f(n_1,n_2)/f(n_2) : f(n_2,n_3)/f(n_3)$	70.49	63.11
$f(n_1,n_2)/f(n_2) : f(n_1,n_3)/f(n_3)$	80.32	66.39
$f(n_1,n_2) : f(n_2,n_3)$ (NEAR)	68.03	63.11
$f(n_1,n_2) : f(n_1,n_3)$ (NEAR)	71.31	67.21
$f(n_1,n_2)/f(n_1) : f(n_2,n_3)/f(n_2)$ (NEAR)	61.47	62.29
$f(n_1,n_2)/f(n_2) : f(n_2,n_3)/f(n_3)$ (NEAR)	65.57	57.37
$f(n_1,n_2)/f(n_2) : f(n_1,n_3)/f(n_3)$ (NEAR)	75.40	68.03#

Table 8: Performance of Altavista counts and BNC counts for compound bracketing (data from Lauer 1995)

Model	Accuracy
Baseline	63.93
Best BNC	68.03$\not\dagger$‡
Lauer (1995): adjacency	68.90
Lauer (1995): dependency	77.50
Best Altavista	78.68†$\not\ddagger$
Lauer (1995): tuned	80.70
Upper bound	81.50

Table 9: Performance comparison with the literature for compound bracketing

$$(4) \qquad R_{dep} = \frac{\sum_{t_i \in cats(w_i)} P(t_1 \rightarrow t_2) P(t_2 \rightarrow t_3)}{\sum_{t_i \in cats(w_i)} P(t_1 \rightarrow t_3) P(t_2 \rightarrow t_3)}$$

$$(5) \qquad R_{adj} = \frac{\sum_{t_i \in cats(w_i)} P(t_1 \rightarrow t_2)}{\sum_{t_i \in cats(w_i)} P(t_2 \rightarrow t_3)}$$

Here t_1, t_2 and t_3 are conceptual categories in the 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) tested 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 our test set were left-branching, yielding a baseline of 63.93% (see Table 9). 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 9. 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 (4) and (5)), 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) and Altavista's `NEAR` operator. 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; the performance of the best Altavista model was significantly higher than that of the best BNC model (see Table 8). A comparison with the literature (see Table 9) shows that the best BNC model fails to significantly outperform the baseline, and it performs significantly worse than the best model in the literature (Lauer's tuned model). The best Altavista model, on the other hand, is not significantly different from Lauer's tuned model and significantly outperforms the baseline. Hence we achieve the same performance as Lauer without recourse to a predefined taxonomy or a thesaurus.

7 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*, etc. 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 (6):

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

Lauer (1995) points out that the above 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 (6), he proposes the model in (7) 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.

Model	Altavista	BNC
$f(n_1, p) f(p, n_2)$	50.71	27.85#
$f(n_1, p, n_2)$	55.71#*	11.42
$f(n_1, p) f(p, n_2) / f(p)$	47.14	26.42
$f(n_1, p, n_2) / f(p)$	55.00	10.71

Table 10: Performance of Altavista counts and BNC counts for compound interpretation (data from Lauer 1995)

Model	Accuracy
Best BNC	27.85$\not\dagger$‡
Lauer (1995): concept-based	28.00
Baseline	33.00
Lauer (1995): word-based	40.00
Best Altavista	55.71†‡

Table 11: Performance comparison with the literature for compound interpretation

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

Here, t_1 and t_2 represent concepts in Roget's thesaurus. Lauer (1995) also experimented with a lexicalized version of (7) 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 (7) 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 11). The concept-based model (see (7)) achieved an accuracy of 28% on this test set, whereas its lexicalized version reached an accuracy of 40% (see Table 11).

We attempted the interpretation task with the lexicalized version of the bigram model (see (7)), but also tried the more data intensive trigram model (see (6)), again in its lexicalized form. Furthermore, we experimented with several conditional and unconditional variants of (7) and (6). Co-occurrence frequencies were estimated from the web using inflected queries (see Section 2). 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*. As shown in Table 10, the best performance was obtained using the web-based trigram model ($f(n_1, p, n_2)$); it significantly outperformed the best BNC model. The comparison with the literature in Table 11 showed 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,

	Altavista		BNC	
Model	Count	Uncount	Count	Uncount
$f(n)$	87.01	90.13	87.32#	90.39#
$f(det,n)$	88.38#$\not\ast$	91.22#$\not\ast$	51.01	50.23
$f(det,n)/f(n)$	83.19	85.38	50.95	50.23
Backoff	87.01	89.80	–	–

Table 12: Performance of Altavista counts and BNC counts for noun countability detection (data from Baldwin and Bond 2003)

achieved a performance that is not significantly different from the baseline, and significantly worse than Lauer's best model.

8 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 determines 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, as they account for the vast majority of the data. Four 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; (c) same as model (b), but the det-noun frequencies are normalized by the frequency of the noun; (d) backoff: try to make a decision using det-noun frequencies; if these are too sparse, back off to singular/plural frequencies.

Unigram and bigram frequencies were estimated from the web using literal queries; for models (a)–(c) a threshold parameter was optimized on the development set (this parameter determines the ratio of singular/plural frequencies or det-noun frequencies above which a noun was considered as countable). For model (b), an additional backoff parameter was used, specifying the minimum frequency that triggers backoff.

The models and their performance on the test set are listed in Table 12. The best Altavista model is the conditional det-noun model ($f(det,n)/f(n)$), which achieves 88.38% on countable and 91.22% on uncountable nouns. On the BNC, the simple unigram model performs best. Its performance is not statistically different from that of the best Altavista model. Note that for the BNC models, data sparseness means the det-noun models perform poorly, which is why the backoff model was not attempted here. Table 13 shows that both the Altavista model and BNC model significantly outperform the baseline (relative frequency of the majority class on the gold-standard data). The comparison with the literature shows that both the Altavista and the BNC model perform significantly worse than the best model proposed by Baldwin and Bond (2003); this is a supervised model that uses many more features than just singular/plural frequency and det-noun frequency.

Model	Count	Uncount
Baseline	74.60	78.30
Best BNC	87.32†‡	90.39†‡
Best Altavista	88.38†‡	91.22†‡
Baldwin and Bond (2003)	93.90	95.20

Table 13: Performance comparison with the literature for noun countability detection

9 Conclusions

We showed that simple, unsupervised models using web counts 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 than have been previously used.

For all but two tasks (candidate selection for MT and noun countability detection) we found that simple, unsupervised models perform significantly better when *n*-gram frequencies are obtained from the web rather than from a standard large corpus. This result is consistent with Keller and Lapata's (2003) findings that the web yields better counts than the BNC. The reason for this seems to be that the web is much larger than the BNC (about 1000 times); the size seems to compensate for the fact that simple heuristics were used to obtain web counts, and for the noise inherent in web data.

Our results were less encouraging when it comes to comparisons with state-of-the-art models. We found that in all but one case, web-based models fail to significantly outperform the state of the art. The exception was compound noun interpretation, for which the Altavista model was significantly better than the Lauer's (1995) model. For three tasks (candidate selection for MT, adjective ordering, and compound noun bracketing), we found that the performance of the web-based models was not significantly different from the performance of the best models reported in the literature.

Note that for all the tasks we investigated, the best performance in the literature was obtained by supervised models that have access not only to simple bigram or trigram frequencies, but also to linguistic information such as part-of-speech tags, semantic restrictions, or context (or a thesaurus, in the case of Lauer's models). 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.

Our results therefore 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 paper 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. 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.

Acknowledgments

We are grateful to Tim Baldwin, Silviu Cucerzan, Mark Lauer, Rob Malouf, Detelef Prescher, and Adwait Ratnaparkhi for making their data sets available.

References

Baldwin, Timothy and Francis Bond. 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, pages 463–470.

Banko, Michele and Eric Brill. 2001a. Mitigating the paucity-of-data problem: Exploring the effect of training corpus size on classifier performance for natural language processing. In James Allan, editor, *Proceedings of the 1st International Conference on Human Language Technology Research*. Morgan Kaufmann, San Francisco.

Banko, Michele and Eric Brill. 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.

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

Corley, Steffan, Martin Corley, Frank Keller, Matthew W. Crocker, and Shari Trewin. 2001. Finding syntactic structure in unparsed corpora: The Gsearch corpus query system. *Computers and the Humanities* 35(2):81–94.

Cucerzan, Silviu and David Yarowsky. 2002. Augmented mixture models for lexical disambiguation. In Jan Hajič and Yuji Matsumoto, editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Philadelphia, PA, pages 33–40.

Dagan, Ido and Alon Itai. 1994. Machine translation divergences: A formal description and proposed solution. *Computational Linguistics* 20(4):563–597.

Golding, Andrew R. 1995. A Bayesian hybrid method for context-sensitive spelling correction. In David Yarowsky and Kenneth W. Church, editors, *Proceedings of the 3rd Workshop on Very Large Corpora*. Cambridge, MA, pages 39–53.

Golding, Andrew R. and Dan Roth. 1999. A winnow-based approach to context sensitive spelling correction. *Machine Learning* 34(1–3):1–25.

Golding, Andrew R. and Yves Schabes. 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, pages 71–78.

Grefenstette, Gregory. 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.

Grishman, Ralph, Catherine Macleod, and Adam Meyers. 1994. COMLEX syntax: Building a computational lexicon. In *Proceedings of the 15th International Conference on Computational Linguistics*. Kyoto, Japan, pages 268–272.

Ikehara, Satoru, Satoshi Shirai, Akio Yokoo, and Hiromi Nakaiwa. 1991. Toward an MT system without pre-editing effects of new methods in ALT-J/E. In *Proceedings of the Third Machine Translation Summit*. Washington, DC, pages 101–106.

Jones, Michael P. and James H. Martin. 1997. Contextual spelling correction using latent semantic analysis. In *Proceedings of the 5th Conference on Applied Natural Language Processing*. Washington, DC, pages 166–173.

Keller, Frank and Mirella Lapata. 2003. Using the web to obtain frequencies for unseen bigrams. *Computational Linguistics* 29(3):459–484.

Lauer, Mark. 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, pages 47–54.

Malouf, Robert. 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, pages 85–92.

Mangu, Lidia and Eric Brill. 1997. Automatic rule acquisition of spelling correction. In *Proceedings of the 14th International Conference on Machine Learning*. Nashville, Tennessee, pages 187–194.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics* 19(2):313–330.

Prescher, Detlef, Stefan Riezler, and Mats Rooth. 2000. Using a probabilistic class-based lexicon for lexical ambiguity resolution. In *Proceedings of the 18th International Conference on Computational Linguistics*. Saarbrücken, Germany, pages 649–655.

Pustejovsky, James, Sabine Bergler, and Peter Anick. 1993. Lexical semantic techniques for corpus analysis. *Computational Linguistics* 19(3):331–358.

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

Shaw, James and Vassilis Hatzivassiloglou. 1999. Ordering among premodifiers. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, MD, pages 135–143.