Towards Full Automation of Lexicon Construction

Richard Rohwer
Fair Isaac Corporation
RichardRohwer@fairisaac.com

Dayne Freitag
Fair Isaac Corporation
DayneFreitag@fairisaac.com

Abstract

We describe work in progress aimed at developing methods for automatically constructing a lexicon using only statistical data derived from analysis of corpora, a problem we call *lexical optimization*. Specifically, we use statistical methods alone to obtain information equivalent to syntactic categories, and to discover the semantically meaningful units of text, which may be multi-word units or polysemous terms-in-context. Our guiding principle is to employ a notion of "meaningfulness" that can be quantified information-theoretically, so that plausible variants of a lexicon can be judged relative to each other. We describe a technique of this nature called *information theoretic co-clustering* and give results of a series of experiments built around it that demonstrate the main ingredients of lexical optimization. We conclude by describing our plans for further improvements, and for applying the same mathematical principles to other problems in natural language processing.

1 Introduction

A lexicon is a key resource for natural language processing, providing the link between the terms of a language and the semantic and syntactic properties with which they are associated. Like most resources of considerable value, a good lexicon can be difficult or expensive to obtain. This is particularly true if the lexicon needs to be specialized to a technical subject, an obscure language or dialect, or a highly idiomatic writing style. Motivated by the practical importance of these cases as well as the theoretical interest inherent to the problem, we have set out to develop methods for building a lexicon automatically, given only a corpus of text representative of the domain of interest.

We represent the semantics of a term by an associated probability distribution over what we call a *grounding space*, which we define in various relatively conventional ways involving terms that occur in text in the vicinity of the term in question. It is well-known that such distributions can represent meaning reasonably well, at least for meaning-comparison purposes (Landauer and Dumais, 1997). We add to this framework the notion that the more information such a *distributional lexicon* can capture, the more useful it is. This provides us with a mathematical concept of *lexical optimization*.

We begin the lexicon construction process by applying a distributional clustering technique called *information theoretic co-clustering* to make a first pass at grouping the most frequent terms in the corpus according to their most common syntactic part of speech category, as described in Section 2 along with illustrative results. We briefly describe the co-clustering algorithm in Section 2.1. In Section 3.1, we show that novel terms can be sensibly assigned to previously defined clusters using the same information theoretic criterion that the co-clustering uses.

Even though term clustering crudely ignores the fact that a term's part of speech generally varies with its context, it is clear from inspection that the clusters themselves correspond to corpus-adapted part-of-speech categories, and can be used as such. In Section 3.2, we examine two approaches to incorporating context information. The most direct is to partition the contexts in which a term occurs into classes according to the informatic criterion used in co-clustering, creating sense-disambiguated word-with-context-class "pseudo-terms". We also discuss the use of Hidden Markov Models (HMMs) to capture contextual information. In Section 3.3 we apply the same principle "in reverse" to find multi-word units.

We conclude in Section 3.5 with a discussion of possible improvements to our approach, and possible extensions of it.

2 Co-clustering to define surrogate syntactic tags

Many applications of text processing rely on or benefit from information regarding the parts of speech of individual terms. While part of speech is a somewhat fluid notion, the computational linguistics community has converged on a handful of standard "tag sets," and taggers are now available in a number of languages. Since some high-quality taggers are in the public domain, any application that could benefit from part-of-speech information should have access to it.

However, using a specific tagger and its tag set entails adopting the assumptions it embodies, which may not be appropriate for the target application. In the worst case, the domain of interest may include text in a language not covered by available taggers. Even when a tagger is available, the domain may involve usages substantially different from those in the corpus for which the tagger was developed. Many current taggers are tuned to relatively formal corpora, such as newswire, while many interesting domains, such as email, netnews, or physicians' notes, are replete with elisions, jargon, and neologisms. Fortunately, using distributional characteristics of term contexts, it is feasible to induce part-of-speech categories directly from a corpus of sufficient size, as several papers have made clear (Brown et al., 1992; Schütze, 1993; Clark, 2000).

Distributional information has uses beyond part of speech induction. For example, it is possible to augment a fixed syntactic or semantic taxonomy with such information to good effect (Hearst and Schütze, 1993). Our objective is, where possible, to work directly with the inferred syntactic categories and their underlying distributions. There are many applications of computational linguistics, particularly those involving "shallow" processing, such as information extraction, which can benefit from such automatically derived information, especially as research into acquisition of grammar matures (e.g., (Clark, 2001)).

2.1 The Co-clustering Algorithm.

Our approach to inducing syntactic clusters is closely related to that described in Brown, et al, (1992) which is one of the earliest papers on the subject. We seek to find a partition of the vocabulary that maximizes the mutual information between term categories and their contexts. We achieve this in the framework of *information theoretic co-clustering* (Dhillon et al., 2003), in which a space of entities, on the one hand, and their contexts, on the other, are alternately clustered in a way that maximizes mutual information between the two spaces. By treating the space of terms and the space of contexts as separate, we part ways with Brown, et al. This allows us to experiment with the notion of context, as well as to investigate whether pooling contexts is useful, as has been assumed.

2.2 Definitions

Given a corpus, and some notion of *term* and *context*, we derive co-occurrence statistics. More formally, the input to our algorithm is two finite sets of symbols, say $X = \{x_0, x_1, ..., x_{N_X}\}$ and $Y = \{y_0, y_1, ..., y_{N_Y}\}$, together with a set of co-occurrence count data consisting of a non-negative integer $n_{x_i y_j}$ for every pair of symbols (x_i, y_j), that can be drawn from X and Y. The output is two sets of sets: $X^* = \{x^*_0, ..., x^*_{N_{X^*}}\}$ and $Y^* = \{y^*_0, ..., y^*_{N_{Y^*}}\}$, where each x^*_i is a subset of X (a "cluster"), none of the x^*_i intersect each other, the union of all the x^*_i is X (similar remarks apply to the y^*_j and Y). The co-clustering algorithm chooses the partitions X^* and Y^* to (locally) maximize the expected mutual information between them.

The multinomial parameters p_{xy} of a joint distribution over X and Y may be estimated from this co-occurrence data as $p_{xy} = n_{xy} / \sum_{x,y} n_{xy}$, using the naive maximum likelihood method. We follow a more fully Bayesian procedure to obtain "pseudo-counts" u_{xy} that are added to the counts n_{xy} to obtain "smoothed" estimates. Due to space limitations, we define but do not fully discuss our procedure here. We apply the "Evidence" method in the "Dice Factory" setting of MacKay and Peto (1994), to obtain a pseudo-count $u_{x\cdot}$ for every symbol $x \in X$ by treating each $y \in Y$ as a sample *of* (not *from*) a random process $P(x|y)$, in a Multinomial/Dirichlet setting. By a symmetric procedure, we also obtain pseudo-counts $u_{\cdot y}$ for each $y \in Y$. These are combined according to $u_{xy} = \frac{1}{2}(u_X + u_Y)(u_{x\cdot} u_{\cdot y})/(u_X u_Y)$, and then the totals $n_{xy} + u_{xy}$ are rescaled by $n/(u + n)$, where $u_X = \sum_x u_{x\cdot}$, $u_Y = \sum_y u_{\cdot y}$, $u = \sum_{xy} u_{xy}$, and $n = \sum_{xy} n_{xy}$.

The *entropy* or *Shannon information* of a discrete distribution is:

$$\mathrm{I}_X = -\sum_x \mathrm{P}(x) \ln \mathrm{P}(x). \tag{1}$$

This quantifies average improvement in one's knowledge upon learning the specific value of an event drawn from X. It is large or small depending on whether X has many or few probable values.

The *mutual information* between random variables X and Y can be written:

$$\mathrm{M}_{XY} = \sum_{xy} \mathrm{P}(x, y) \ln \frac{\mathrm{P}(x, y)}{\mathrm{P}(x)\mathrm{P}(y)} \tag{2}$$

This quantifies the amount that one expects to learn indirectly about X upon learning the value of Y, or vice versa. The following relationship holds between the information of a joint distribution and the information of the marginals and mutual information:

$$\mathrm{I}_{XY} = \mathrm{I}_X + \mathrm{I}_Y - \mathrm{M}_{XY} \qquad (3)$$

From this we see that the expected amount one can learn upon hearing of a joint event (x, y) is bounded by what one can learn about X and Y separately. Combined with another elementary result, $I_X \geq M_{XY} \geq 0$ and symmetrically $I_Y \geq M_{XY} \geq 0$, we see that a joint event (x, y) yields at least as much information as either event alone, and that one cannot learn more about an event y from Y by hearing about an event x from X than one would know by hearing about y explicitly.

2.3 The Algorithm

The co-clustering algorithm seeks partitions X^* of X and Y^* of Y with maximal mutual information $M_{X^*Y^*}$, under a constraint limiting the total number of clusters in each partition. The mutual information is computed from the distributions estimated as discussed in Section 2.2, by summing $P(x, y)$ over the elements within each cluster to obtain $P(x^*, y^*)$.

We perform an approximate maximization of $M_{X^*Y^*}$ using a simulated annealing procedure in which each trial move takes a symbol x or y out of the cluster to which it is tentatively assigned and places it into another. It is straightforward to obtain a formula for the change in $M_{X^*Y^*}$ under this operation that does not involve its complete re-computation. We use an *ad hoc* adaptive cooling schedule that seeks to continuously reduce the rejection rate of trial moves from an initial level near 50%, staying at each target rejection rate long enough to visit a fixed fraction of the possible moves with high probability. After achieving one rejection rate target for the required number of moves, the target is lowered. The temperature is also lowered, but will be raised again to an intermediate value if the resulting rejection rate is below the next target, or lowered further if the rejection rate remains above the next target.

Candidate moves are chosen by selecting a non-empty cluster uniformly at random, randomly selecting one of its members, then randomly selecting a destination cluster other than the source cluster. When temperature 0 is reached, all possible moves are repeatedly attempted until no move leads to an increase in the objective function.

2.4 Co-clustering for Term Categorization

Applying co-clustering to the problem of part of speech induction is straightforward. We define X to be the space of terms under some tokenization of the corpus, and Y to be the space of contexts of those terms, which are a function of the close neighborhood of occurrences from X. Members of Y are also typically terms, but we have also experimented with concatenations of terms, and more complex definitions based on relative position. The results reported here are based on the simple context definition of one term to the left and one to the right, regarded as separate events.

Experiment	Time
No Conj. Clusters	74:17:31
Conj. Clusters	12:07:43

Table 1: Time to complete clustering, with and without conjugate clusters in hours:minutes:seconds.

Given a particular tokenization and method for defining context, we can derive input for the co-clustering algorithm. Sparse co-occurrence tables are created for each term of interest; each entry in such a table records a context identifier and the number of times the corresponding context occurred with the reference term. For expediency, and to avoid problems with sparse statistics, we retain only the most frequent terms and contexts. (We chose the top 5000 of each.) In Section 3.1, we show that we can overcome this limitation through subsequent processing.

2.5 Experimental details and results

We conducted experiments with the Reuters-21578 corpus—a relatively tiny one for such experiments. Clark (2000) reports results on a corpus containing 12 million terms, Schütze (1993) on one containing 25 million terms, and Brown, et al, (1992) on one containing 365 million terms. In contrast, we count approximately 2.8 million terms in Reuters-21578.

Only the bodies of articles in the corpus were considered. Each such article was segmented into paragraphs, but not sentences. Paragraphs were then converted into token arrays, with each token corresponding to one of the following: an unbroken string of alphabetic characters or hyphens, possibly terminated by an apostrophe and additional alphabetic characters; a numeric expression; or a single occurrence and unit of punctuation presumed to be syntactically significant (e.g., periods, commas, and question marks). Alphabetic tokens were case-normalized, and all numeric expressions were replaced with the special token `<num>`. For the purposes of constructing context distributions, special contexts (`<bop>` and `<eop>`) were inserted at the beginnings and endings of each such array.

We applied the co-clustering algorithm to the most frequent 5000 terms and most frequent 5000 contexts in the corpus, clustering each into 200 categories.

Co-clustering—alternately clustering terms and contexts—is faster than simple clustering against the full set of contexts. Table 1 presents computation times for experiments with one grounding space on the same

Clust.	Terms
37	may employs
71	because out ahead comprised consists ...
96	he she fitzwater mulford azpurua ...
145	reported announced showed follows owns ...
159	set available used asked given paid taken ...
161	are were am
179	operations funds figurers results issues ...
180	on until upon regarding governing
186	business investment development sugar ...
194	to
195	of
199	the japan's today's brazil's canada's ...

Table 2: Selected clusters from experiment on the full corpus. Clusters are ordered according to their impact on mutual information, least to greatest ascending. Within each cluster, terms are ordered most frequent to least.

machine under similar loads. While the exact time to completion is a function of particularities such as machine speed, cluster count, and annealing schedule, the relative durations (co-clustering finishes in 1/6 the time) are representative. This may be counter-intuitive, since co-clustering involves two parallel clustering runs, instead of a single one. However, the savings in the time it takes to compute the objective function (in this case, mutual information with 200 contexts, instead of 5000) typically more than compensates for the additional algorithmic steps.

Table 2 lists clusters that illustrate both strengths and weaknesses of our approach. While many of the clusters correspond unambiguously to some part of speech, we can identify four phenomena that sometimes prevent the clusters from corresponding to unique part-of-speech tags:

1. **Lack of distributional evidence**. In several cases, the grounding space chosen provides no evidence for a distinction made by the tagger. Examples of this are cluster 199, where "the" is equated with the possessive form of many nouns; cluster 145, where present tense and past tense verbs are both represented; and cluster 96, where personal pronouns are equated with surnames.[1]

2. **Predominant idioms and contexts**. If a term is used predominantly in a particular idiom, then the context supplied by that idiom may have the strongest influence on its cluster assignment, occasionally leading to counter-intuitive clusters. An obvious example of this is cluster 71. All of the terms in this cluster are typically followed by the context "of."

3. **Lexical ambiguity**. If a term has two or more frequent syntactic categories, the algorithm assigns it (in the best case) to a cluster corresponding to its more frequent sense, or (in the worst case) to a "junk" or singleton cluster. This happens with the word "may" (cluster 37, above) in all our experiments.

4. **Multi-token lexemes**. In order to tally context distributions, we must commit to an initial fixed segmentation of the corpus. While English orthography insures that this is not difficult, there exist nevertheless fixed collocations (commonly called multi-word units, MWUs), such as "New York," which inject statistical noise under the default segmentation.

Of these four "problems," the last two are probably more serious, since they give rise to specious distinctions. Depending on the application, problems 1 and 2 may not be problems at all. In this corpus, for example, the term "regarding" (cluster 180) may never be used in any but a quasi-prepositional sense. And proper nouns in the possessive arguably do share a syntactic function with "the."

3 Refinements

Lexical categorizations, such as those provided by a part of speech tagger or a semantic resource like Wordnet, are usually a means to an end, almost never applications in their own right. While it is interesting to measure how faithfully an unsupervised algorithm can reconstruct prior categories, we neither expect to achieve anything like perfect performance on this task, nor believe that it is necessary to do so. In fact, adherence to a specific tag set can be seen as an impediment, inasmuch as it introduces brittleness and susceptibility to noise in categorization.

It is nevertheless interesting to ignore the confounding factors enumerated in Section 2.5 and measure the agreement between term categories induced by co-clustering and the tags assigned by a tagger. Using the tagger from The XTag Project (Project, 2003), we measured the agreement between our clusters and the tagger output over the terms used in clustering. We found that the clusters captured 85% of the information in the tagged text (the tagged data had an entropy of 2.68, while mutual information between clusters and tags is 2.23). In a theoretical optimal classifier, this yields a ninefold increase in accuracy over the default rule of always choosing the most frequent tag.

In order to make our *distributional lexicon* useful, however, we need to extend its reach beyond the few thou-

[1] Far from a bad thing, however, this last identification suggests some avenues for research in unsupervised pronominal reference resolution.

sand most frequent terms, on the one hand, and adjust for phenomena that lead to sub-optimal performance, on the other. We call the process of expanding and adjusting the lexicon after its initial creation *lexicon optimization*.

3.1 Increasing Lexicon Coverage

For tractability, the initial classes are induced using only the most frequent terms in a corpus. (While we cluster using only the 5000 most frequent terms, the corpus contains approximately 41,000 distinct word-forms.) This yields consistent results and broad coverage of the corpus, but leaves us unable to categorize about 5% of tokens. Clearly, in order for our automatically constructed resource to be useful, we must introduce these uncovered terms into the lexicon, or better still, find a way to apply it to individual novel tokens.

3.1.1 HMM tagging

In light of the current state of the art in part of speech tagging, the occurrence of these unknown terms does not pose a significant problem. It has been known for some years that good performance can be realized with partial tagging and a hidden Markov model (Cutting et al., 1992). Note that the notion of partial tagging described in Cutting, et al, is essentially different from what we consider here. Whereas they assume a lexicon which, for every term in the vocabulary, lists its possible parts of speech, we *construct* a lexicon which imposes a single sense (or a few senses; see Section 3.2) on each of the few thousand most frequent terms, but provides no information about other terms.

As in Cutting, et al, however, we can use Baum-Welch re-estimation to extract information from novel terms, and apply the Viterbi algorithm to dispose of a particular occurrence. While the literature suggests that Baum-Welch training can degrade performance on the tagging task (Elworthy, 1994; Merialdo, 1994), we have found in early experiments that agreement between a tagger trained in this way and the tagger from the XTag Project consistently increases with each iteration of Baum-Welch, eventually reaching a plateau, but not decreasing. We attribute this discrepancy to the different structure of our problem.

3.1.2 Lexicon expansion

Note that a HMM is under no constraint to handle a given term in a consistent fashion. A single model can and often does assign a single term to multiple classes, even in a single document. When a term is sufficiently frequent, a more robust approach may be to assign it to a category using only its summary co-occurrence statistics. The idea is straightforward: Create an entry in the lexicon for the novel term and measure the change in mutual information associated with assigning it to each of the available categories. Assign it to the category for which this change is maximized.

Term	Freq.	Cluster Example Terms
weizsaecker	30	baker morgan shearson
provoke	20	take buy make
glut	10	price level volume
councils	5	prices markets operations
stockbuilding	3	earnings income profits
ammonia	2	energy computer petroleum
unwise	2	expected likely scheduled

Table 3: Assigning novel terms to clusters using the mutual information objective function. Each row shows a term not present in the initial clustering, its corpus frequency, and example terms from the cluster to which it is assigned.

As Table 3 demonstrates, this procedure works surprisingly well, even for words with low corpus frequencies. Of course, as frequencies are reduced, the likelihood of making a sub-optimal assignment increases. At some point, the decision is better made on an individual basis, by a classifier trained to account for the larger context in which a novel term occurs, such as an HMM. We are currently investigating how to strike this trade-off, in a way that best exploits the two available techniques for accommodating novel tokens.

Lexical ambiguity (or polysemy) and fixed collocations (multi-word units) are two phenomena which clearly lead to sub-optimal clusters. We have achieved promising results resolving these problems while remaining within the co-clustering framework. The basic idea is as follows: If by treating a term as two distinct lexemes (or, respectively, a pair of commonly adjacent terms as a distinct lexeme), we can realize an increase in mutual information, then the term is lexically ambiguous (respectively, a fixed collocation). In the case of polysemy resolution, this involves factoring the context distribution into two or more clusters. In the case of a fixed collocation, we consider the effect of treating an n-gram as a lexical unit.

3.2 Polysemy Resolution

To determine whether a term is polysemous we must determine whether the lexicon's mutual information can be increased by treating the term as two distinct lexemes. Given a particular term, we make this determination by attempting to factor its context distribution into every possible pair of distinct clusters.[2] Faced with a candidate pair, we posit two senses of the target term, one in each

[2] In this discussion, we assume exactly two senses, but the approach is easily extended to handle more than two.

Term	Δ MI	Cluster Example Terms
may	8.75e-5	april march june
		would could should
act	6.51e-5	continue remain come
		board committee court
vote	4.32e-5	continue remain come
		meeting report
france	-1.2e-6	japan canada brazil
		and
would	-0.0008	will
		would could should

Table 4: The result of polysemy resolution run on some representative terms. The third column lists sample terms from the two clusters into which each term is divided.

Phrase	Example Cluster Terms
cubic feet	francs barrels ounces
hong kong	london tokyo texas
pointed out	added noted disclosed
los angeles	london tokyo texas
merrill lynch	texaco chrysler ibm
we don’t	we i you
saudi arabia	japan canada brazil
morgan stanley	texaco chrysler ibm
managing director	president chairman
smith barney	underwriters consumers

Table 5: The ten highest-scoring two-word multi-word units in Reuters, along with example terms from the cluster to which each was assigned.

cluster. The probability mass associated with each event type in the term's context distribution is then assigned to one or the other hypothetical sense, always to the one that improves mutual information the most (or hurts it the least). Once the probability mass of the original term has been re-apportioned in this way, the resulting change in mutual information reflects the quality of the hypothetical sense division. The maximum change in mutual information over all such cluster pairs is then taken to be the polysemy score for the target term.

Table 4 shows how this procedure handles selected terms from the Reuters corpus. Positive changes in mutual information clearly correspond to polysemy in the target term. In the Reuters corpus, there are a fair number of terms that have a noun and a verb sense, such as "act" and "vote" in the table. Note, too, the result of polysemy resolution run on unambiguous terms–either a nonsensical division, as with "france," or division into two closely related clusters, in both cases, however, with a decrease in mutual information.

Note that the problem of lexical ambiguity has been studied elsewhere. Schütze (1993; 1995) proposes two distinct methods by which ambiguity may be resolved. In one paper, a separate model (a neural network) is trained on the results of clustering in order to classify individual term occurrences. In the other, the individual occurrences of a term are "tagged" according to the distributional properties of their neighbors. Clark (2000) presents a framework which in principle should accommodate lexical ambiguity using mixtures, but includes no evidence that it does so. Furthermore, a mixture distribution specifies the proportion of occurrences of a term that should be tagged one way or another, but does not prescribe what to do with every individual event. In contrast to the above approaches, we derive a lexicon which succinctly lists the possible syntactic senses for a term and provides a means to disambiguate the sense of a single occurrence. Moreover, a shortcoming of occurrence-based methods of polysemy resolution is that a given term may be assigned to an implausibly large number of categories. By analyzing this behavior at the *type level*, rather than the *token level*, we not only can exploit the corpus-wide behavior of a term, but we can enforce the linguistically defensible constraint that it have only a few senses.

3.3 Multi-Word Units

In English, orthography provides a convenient clue to textual word segmentation. Doing little more than breaking the text on whitespace boundaries, it is possible to perform a linguistically meaningful statistical analysis of a corpus. Multi-word units (MWUs) are the exception to this rule. Treating terms such as "York"—terms which in a particular corpus may not be meaningful in isolation—gives rise to highly idiosyncratic context distributions, which in turn add noise to cluster statistics or lead to the production of "junk" clusters.

In order to recognize such cases, we apply a variant of our by now familiar lexicon optimization rule: We posit a lexical entry for a given candidate MWU, find the cluster to which it is best suited, and ask whether creating the lexeme improves the situation. In principle, we can conduct this process in the same way as with novel terms and polysemy. Here, however, we report the results of a simple surrogate technique. After assembling the context distribution of the candidate MWU (an n-gram), we compute the Hellinger distance between this distribution and that of each cluster. The Hellinger distance between two distributions P and Q is defined as:

$$D\left[P, Q\right] = 1 - \sum_i \sqrt{p_i}\sqrt{q_i} \tag{4}$$

The candidate MWU is then tentatively assigned to the cluster for which this quantity is minimized and its distance to this cluster is noted (call this distance

Score Band	% in Wordnet
> 0.5	55
0.25 – 0.5	37
0 – 0.25	21
-0.25 – 0	11
-0.5 – -0.25	5
-1 – -0.5	1.4
< -1	1.9

Table 6: Fraction of two-word collocations present in Wordnet in each MWU score band.

D_{ngram}). We then compute the distance between each of the n-gram's constituent terms and its respective cluster ($D_{\text{term}_1} \cdots D_{\text{term}_n}$). The MWU score is the difference between the maximum term distance and the n-gram distance, or $\max_i D_{\text{term}_i} - D_{\text{ngram}}$. In other words, the score of a candidate MWU increases with its closeness of fit to its cluster and the lack of fit of its constituent terms.

Table 5 shows the ten bi-grams that score highest using this heuristic. Note that they come from a number of syntactic categories. In this list, the only error is the phrase "we don't," which is determined to be syntactically substitutable for pronouns. Note, however, that this is the only collocation in this list consisting entirely of closed-class terms. To the extent that we can recognize such terms, it is easy to rule out such cases.

Table 6 benchmarks this technique against Wordnet. Breaking the range of MWU scores into bands, we ask what fraction of n-grams in each band can be found in Wordnet. The result is a monotonic decrease in Wordnet representation. Investigating further, we find that almost all of the missing n-grams that score high are absent because they are corpus-specific concepts, such as "Morgan Stanley" and "Smith Barney." On the other end, we find that low-scoring n-grams present in Wordnet are typically included for reasons other than their ability to serve as independent lexemes. For example, "on that" appears to have been included in Wordnet because it is a synonym for "thereon."

3.4 Directions

We have begun research into characterizing more precisely the grammatical roles of the clusters found by our methods, with an eye to identifying the lowest-level expansions in the grammar responsible for the text. Inasmuch as information extraction can rely on shallow methods, the ability to produce a shallow parser without supervision should enable rapid creation of information extraction systems for new subject domains and languages.

Highest		Lowest	
Term	Entropy	Term	Entropy
and	6.67	swedish	3.50
,(comma)	6.31	june	3.50
to	6.27	apparel	3.50
for	6.01	giant	3.50
was	5.92	modified	3.50

Table 7: Five most entropic and five least entropic terms among the 5000 most frequent terms, using the "-1 $\vee$ +1" grounding space. In general, closed-class terms have higher entropy.

We have had some success distinguishing open-class clusters from closed-class clusters, on the one hand, and "head" clusters from modifier clusters, on the other. Schone and Jurafsky (2001) list several universal characteristics of language that can serve as clues in this process, some of which we exploit. However, their use of "perfect" clusters renders some of their algorithmic suggestions problematic. For example, they propose using the tendency of a cluster to admit new members as an indication that it contains closed-class (or function) terms. While we do find large clusters corresponding to open classes and small clusters to closed classes, the separation is not always clean (e.g., cluster 199 in Table 2). Small clusters often contain open-class terms with predominant corpus-specific idiomatic usages. For example, Reuters-21578 has special usages for the terms "note," "net," and "pay," in additional to their usual usages.

While the size of its cluster is a useful clue to the open- or closed-class status of a term, we are forced to search for other sources of evidence. Once such indicator is the entropy of the term's context distribution. Table 7 lists the five most and least entropic among the 5000 most frequent terms in Reuters-21578. Function terms have higher entropy not only because they are more frequent than non-function terms, but also because a function term must participate syntactically with a wide variety of content-carrying terms. While entropy alone also does not yield a clean separation between "function" and "content" terms, it may be possible to use it in combination with the suggestion of Schone and Jurafsky to produce a reliable separation.

3.5 Conclusion

It seems clear that practical constraints will necessitate the development of powerful corpus-driven methods for meaning representation, particularly when dealing with diverse languages, subject matter, and writing styles. Although it remains to be fully developed and tested, the evidence assembled thus far seems sufficient to conclude that our *lexical optimization* approach offers this prospect.

The approach follows a simple information-theoretic principle: A lexicon can be judged by the amount of information it captures about a suitably chosen "grounding space". The process results in a *distributional lexicon* suitable for semantic comparison of sense-disambiguated terms, multi-word units, and most likely, larger units of text such as short phrases.

One can initialize the lexical optimization process by applying a distributional clustering algorithm such as co-clustering to obtain term classes that have the properties of syntactic tags, regardless of the fact that many of the terms in a typical cluster will, in many contexts, fail to exhibit the syntactic class that the cluster implicitly represents. This starting point is sufficient to support incremental refinements including sense disambiguation, multi-word-unit detection, and the incorporation of novel terms into the lexicon. The preliminary evidence also suggests that this approach can be extended to capture shallow parsing information. Although we have yet to conduct such experiments, it also seems clear that given a set of refinements based on one co-clustering run, it becomes possible to re-analyze the corpus in terms of the improved lexicon and generate an improved co-clustering, etc. It remains to be seen how far such an approach can be productively pursued.

Acknowledgements

This work was supported in full by the Advanced Research and Development Activity (ARDA) under contract number 2003-H265500-000.

References

P.F. Brown, V.J. Della Pietra, P.V. deSouza, J.C. Lai, and R.L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

A. Clark. 2000. Inducing syntactic categories by context distribution clustering. In *CoNLL 2000*, September.

A. Clark. 2001. Unsupervised induction of stochastic context-free grammars using distributional clustering. In *CoNLL 2001*, July.

D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*.

I. S. Dhillon, S. Mallela, and D. S. Modha. 2003. Information-theoretic co-clustering. Technical Report TR-03-12, Dept. of Computer Science, U. Texas at Austin.

D.J.C. MacKay. 1994. A Hierarchical Dirichlet Language Model. *Natural Language Engineering*, 1.

D. Elworthy. 1994. Does Baum-Welch re-estimation help taggers? In *Proc. 4th ACL Conference on Applied Natural Language Processing*.

M.A. Hearst and H. Schütze. 1993. Customizing a lexicon to better suit a computational task. In *Proc. ACL SIGLEX Workshop on Acquisition of Lexical Knowledge from Text*.

T.K. Landauer and S.T. Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2):211–240.

B. Merialdo. 1994. Tagging text with a probabilistic model. *Computational Linguistics*, 20(2):155–171.

Xtag Project. 2003. www.cis.upenn.edu/∼xtag/.

P. Schone and D. Jurafsky. 2001. Language-independent induction of part of speech class labels using only language universals. In *Proc. IJCAI-2001 Workshop 'Text Learning: Beyond Supervision'*.

H. Schütze. 1993. Part-of-speech induction from scratch. In *Proc. 31st Annual Meeting of the ACL (ACL-93)*.

H. Schütze. 1995. Distributional part-of-speech tagging. In *Proc. 7th EACL Conference (EACL-95)*, March.