NAACL 2009 Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics

Corpus-based Semantic Lexicon Induction with Web-based Corroboration

Sean P. Igo
Center for High Performance Computing
University of Utah
Salt Lake City, UT 84112 USA
Sean.Igo@utah.edu

Ellen Riloff
School of Computing
University of Utah
Salt Lake City, UT 84112 USA
riloff@cs.utah.edu

Abstract

Various techniques have been developed to automatically induce semantic dictionaries from text corpora and from the Web. Our research combines corpus-based semantic lexicon induction with statistics acquired from the Web to improve the accuracy of automatically acquired domain-specific dictionaries. We use a weakly supervised bootstrapping algorithm to induce a semantic lexicon from a text corpus, and then issue Web queries to generate co-occurrence statistics between each lexicon entry and semantically related terms. The Web statistics provide a source of independent evidence to confirm, or disconfirm, that a word belongs to the intended semantic category. We evaluate this approach on 7 semantic categories representing two domains. Our results show that the Web statistics dramatically improve the ranking of lexicon entries, and can also be used to filter incorrect entries.

1 Introduction

Semantic resources are extremely valuable for many natural language processing (NLP) tasks, as evidenced by the wide popularity of WordNet (Miller, 1990) and a multitude of efforts to create similar "WordNets" for additional languages (e.g. (Atserias et al., 1997; Vossen, 1998; Stamou et al., 2002)). Semantic resources can take many forms, but one of the most basic types is a dictionary that associates a word (or word sense) with one or more semantic categories (hypernyms). For example, *truck* might be identified as a VEHICLE, and *dog* might be identified as an ANIMAL. Automated methods for generating such dictionaries have been developed under the rubrics of lexical acquisition, hyponym learning, semantic class induction, and Web-based information extraction. These techniques can be used to rapidly create semantic lexicons for new domains and languages, and to automatically increase the coverage of existing resources.

Techniques for semantic lexicon induction can be subdivided into two groups: corpus-based methods and Web-based methods. Although the Web can be viewed as a (gigantic) corpus, these two approaches tend to have different goals. Corpus-based methods are typically designed to induce domain-specific semantic lexicons from a collection of domain-specific texts. In contrast, Web-based methods are typically designed to induce broad-coverage resources, similar to WordNet. Ideally, one would hope that broad-coverage resources would be sufficient for any domain, but this is often not the case. Many domains use specialized vocabularies and jargon that are not adequately represented in broad-coverage resources (e.g., medicine, genomics, etc.). Furthermore, even relatively general text genres, such as news, contain subdomains that require extensive knowledge of specific semantic categories. For example, our work uses a corpus of news articles about terrorism that includes many arcane weapon terms (e.g., *M-79*, *AR-15*, *an-fo*, and *gelignite*). Similarly, our disease-related documents mention obscure diseases (e.g., *psittacosis*) and contain many informal terms, abbreviations, and spelling variants that do not even occur in most medical dictionaries. For example, *yf* refers to yellow fever, *tularaemia* is an alternative spelling for *tularemia*, and *nv-cjd* is frequently used to refer to *new variant Creutzfeldt Jacob Disease*.

The Web is such a vast repository of knowledge that specialized terminology for nearly any domain probably exists in some niche or cranny, but finding the appropriate corner of the Web to tap into is a challenge. You have to know where to look to find specialized knowledge. In contrast, corpus-based methods can learn specialized terminology directly from a domain-specific corpus, but accuracy can be a problem because most corpora are relatively small.

In this paper, we seek to exploit the best of both worlds by combining a weakly supervised corpus-based method for semantic lexicon induction with statistics obtained from the Web. First, we use a bootstrapping algorithm, Basilisk (Thelen and Riloff, 2002), to automatically induce a semantic lexicon from a domain-specific corpus. This produces a set of words that are hypothesized to belong to the targeted semantic category. Second, we use the Web as a source of corroborating evidence to confirm, or disconfirm, whether each term truly belongs to the semantic category. For each candidate word, we search the Web for pages that contain both the word and a semantically related term. We expect that true semantic category members will co-occur with semantically similar words more often than non-members.

This paper is organized as follows. Section 2 discusses prior work on weakly supervised methods for semantic lexicon induction. Section 3 overviews our approach: we briefly describe the weakly supervised bootstrapping algorithm that we use for corpus-based semantic lexicon induction, and then present our procedure for gathering corroborating evidence from the Web. Section 4 presents experimental results on seven semantic categories representing two domains: Latin American terrorism and disease-related documents. Section 5 summarizes our results and discusses future work.

2 Related Work

Our research focuses on semantic lexicon induction, where the goal is to create a list of words that belong to a desired semantic class. A substantial amount of previous work has been done on weakly supervised and unsupervised creation of semantic lexicons. Weakly supervised corpus-based methods have utilized noun co-occurrence statistics (Riloff and Shepherd, 1997; Roark and Charniak, 1998), syntactic information (Widdows and Dorow, 2002; Phillips and Riloff, 2002; Pantel and Ravichandran, 2004; Tanev and Magnini, 2006), and lexico-syntactic contextual patterns (e.g., *“resides in <location>”* or *“moved to <location>”*) (Riloff and Jones, 1999; Thelen and Riloff, 2002). Due to the need for POS tagging and/or parsing, these types of methods have been evaluated only on fixed corpora[1], although (Pantel et al., 2004) demonstrated how to scale up their algorithms for the Web. The goal of our work is to improve upon corpus-based bootstrapping algorithms by using co-occurrence statistics obtained from the Web to re-rank and filter the hypothesized category members.

Techniques for semantic class learning have also been developed specifically for the Web. Several Web-based semantic class learners build upon Hearst’s early work (Hearst, 1992) with hyponym patterns. Hearst exploited patterns that explicitly identify a hyponym relation between a semantic class and a word (e.g., *“such authors as <X>”*) to automatically acquire new hyponyms. (Paşca, 2004) applied hyponym patterns to the Web and learned semantic class instances and groups by acquiring contexts around the patterns. Later, (Pasca, 2007) created context vectors for a group of seed instances by searching Web query logs, and used them to learn similar instances. The KnowItAll system (Etzioni et al., 2005) also uses hyponym patterns to extract class instances from the Web and evaluates them further by computing mutual information scores based on Web queries. (Kozareva et al., 2008) proposed the use of a doubly-anchored hyponym pattern and a graph to represent the links between hyponym occurrences in these patterns.

Our work builds upon Turney’s work on semantic orientation (Turney, 2002) and synonym learning (Turney, 2001), in which he used a PMI-IR algorithm to measure the similarity of words and phrases based on Web queries. We use a similar PMI (pointwise mutual information) metric for the purposes of semantic class verification.

There has also been work on fully unsupervised

[1] Meta-bootstrapping (Riloff and Jones, 1999) was evaluated on Web pages, but used a precompiled corpus of downloaded Web pages.

semantic clustering (e.g., (Lin, 1998; Lin and Pantel, 2002; Davidov and Rappoport, 2006; Davidov et al., 2007)), however clustering methods may or may not produce the types and granularities of semantic classes desired by a user. Another related line of work is automated ontology construction, which aims to create lexical hierarchies based on semantic classes (e.g., (Caraballo, 1999; Cimiano and Volker, 2005; Mann, 2002)).

3 Semantic Lexicon Induction with Web-based Corroboration

Our approach combines a weakly supervised learning algorithm for corpus-based semantic lexicon induction with a follow-on procedure that gathers corroborating statistical evidence from the Web. In this section, we describe both of these components. First, we give a brief overview of the Basilisk bootstrapping algorithm that we use for corpus-based semantic lexicon induction. Second, we present our new strategies for acquiring and utilizing corroborating statistical evidence from the Web.

3.1 Corpus-based Semantic Lexicon Induction via Bootstrapping

For corpus-based semantic lexicon induction, we use a weakly supervised bootstrapping algorithm called Basilisk (Thelen and Riloff, 2002). As input, Basilisk requires a small set of *seed words* for each semantic category, and a collection of (unannotated) texts. Basilisk iteratively generates new words that are hypothesized to belong to the same semantic class as the seeds. Here we give an overview of Basilisk's algorithm and refer the reader to (Thelen and Riloff, 2002) for more details.

The key idea behind Basilisk is to use pattern contexts around a word to identify its semantic class. Basilisk's bootstrapping process has two main steps: Pattern Pool Creation and Candidate Word Selection. First, Basilisk applies the AutoSlog pattern generator (Riloff, 1996) to create a set of lexicosyntactic patterns that, collectively, can extract every noun phrase in the corpus. Basilisk then ranks the patterns according to how often they extract the seed words, under the assumption that patterns which extract known category members are likely to extract other category members as well. The highest-ranked patterns are placed in a *pattern pool*.

Second, Basilisk gathers every noun phrase that is extracted by at least one pattern in the pattern pool, and designates each head noun as a *candidate* for the semantic category. The candidates are then scored and ranked. For each candidate, Basilisk collects all of the patterns that extracted that word, computes the logarithm of the number of seeds extracted by each of those patterns, and finally computes the average of these log values as the score for the candidate. Intuitively, a candidate word receives a high score if it was extracted by patterns that, on average, also extract many known category members.

The N highest ranked candidates are automatically added to the list of *seed words*, taking a leap of faith that they are true members of the semantic category. The bootstrapping process then repeats, using the larger set of seed words as known category members in the next iteration.

Basilisk learns many good category members, but its accuracy varies a lot across semantic categories (Thelen and Riloff, 2002). One problem with Basilisk, and bootstrapping algorithms in general, is that accuracy tends to deteriorate as bootstrapping progresses. Basilisk generates candidates by identifying the contexts in which they occur and words unrelated to the desired category can sometimes also occur in those contexts. Some patterns consistently extract members of several semantic classes; for example, *"attack on <NP>"* will extract both people (*"attack on the president"*) and buildings (*"attack on the U.S. embassy"*). Idiomatic expressions and parsing errors can also lead to undesirable words being learned. Incorrect words tend to accumulate as bootstrapping progresses, which can lead to gradually deteriorating performance.

(Thelen and Riloff, 2002) tried to address this problem by learning multiple semantic categories simultaneously. This helps to keep the bootstrapping focused by flagging words that are potentially problematic because they are strongly associated with a competing category. This improved Basilisk's accuracy, but by a relatively small amount, and this approach depends on the often unrealistic assumption that a word cannot belong to more than one semantic category. In our work, we use the single-category version of Basilisk that learns each semantic category independently so that we do not need to make this assumption.

3.2 Web-based Semantic Class Corroboration

The novel aspect of our work is that we introduce a new mechanism to independently verify each candidate word's category membership using the Web as an external knowledge source. We gather statistics from the Web to provide evidence for (or against) the semantic class of a word in a manner completely independent of Basilisk's criteria. Our approach is based on the *distributional hypothesis* (Harris, 1954), which says that words that occur in the same contexts tend to have similar meanings. We seek to corroborate a word's semantic class through statistics that measure how often the word co-occurs with semantically related words.

For each candidate word produced by Basilisk, we construct a Web query that pairs the word with a semantically related word. Our goal is not just to find Web pages that contain both terms, but to find Web pages that contain both terms in close proximity to one another. We consider two terms to be collocated if they occur within ten words of each other on the same Web page, which corresponds to the functionality of the NEAR operator used by the AltaVista search engine[2]. Turney (Turney, 2001; Turney, 2002) reported that the NEAR operator outperformed simple page co-occurrence for his purposes; our early experiments informally showed the same for this work.

We want our technique to remain weakly supervised, so we do not want to require additional human input or effort beyond what is already required for Basilisk. With this in mind, we investigated two types of collocation relations as possible indicators of semantic class membership:

Hypernym Collocation: We compute co-occurrence statistics between the candidate word and the name of the targeted semantic class (i.e., the word's hypothesized hypernym). For example, given the candidate word *jeep* and the semantic category VEHICLE, we would issue the Web query "*jeep* NEAR *vehicle*". Our intuition is that such queries would identify definition-type Web hits. For example, the query "*cow* NEAR *animal*" might retrieve snippets such as *"A cow is an animal found on dairy farms"* or *"An animal such as a cow has..."*.

Seed Collocation: We compute co-occurrence statistics between the candidate word and each seed word that was given to Basilisk as input. For example, given the candidate word *jeep* and the seed word *truck*, we would issue the Web query "*jeep* NEAR *truck*". Here the intuition is that members of the same semantic category tend to occur near one another - in lists, for example.

As a statistical measure of co-occurrence, we compute a variation of Pointwise Mutual Information (PMI), which is defined as:

$$PMI(x, y) = log(\frac{p(x,y)}{p(x)*p(y)})$$

where $p(x, y)$ is the probability that x and y are collocated (near each other) on a Web page, $p(x)$ is the probability that x occurs on a Web page, and $p(y)$ is the probability that y occurs on a Web page.

$p(x)$ is calculated as $\frac{count(x)}{N}$, where $count(x)$ is the number of hits returned by AltaVista, searching for x by itself, and N is the total number of documents on the World Wide Web at the time the query is made. Similarly, $p(x, y)$ is $\frac{count(x\ NEAR\ y)}{N}$. Given this, the PMI equation can be rewritten as:

$$log(N) + log(\frac{count(x\ NEAR\ y)}{count(x)*count(y)})$$

N is not known, but it is the same for every query (assuming the queries were made at roughly the same time). We will use these scores solely to compare the relative goodness of candidates, so we can omit N from the equation because it will not change the relative ordering of the scores. Thus, our PMI score[3] for a candidate word and related term (hypernym or seed) is:

$$log(\frac{count(x\ NEAR\ y)}{count(x)*count(y)})$$

Finally, we created three different scoring functions that use PMI values in different ways to capture different types of co-occurrence information:

Hypernym Score: PMI based on collocation between the hypernym term and candidate word.

[2] http://www.altavista.com

[3] In the rare cases when a term had a zero hit count, we assigned -99999 as the PMI score, which effectively ranks it at the bottom.

Average of Seeds Score: The mean of the PMI scores computed for the candidate and each seed word:

$$\frac{1}{|seeds|} \sum_{i=1}^{|seeds|} PMI(candidate, seed_i)$$

Max of Seeds Score: The maximum (highest) of the PMI scores computed for the candidate and each seed word.

The rationale for the Average of Seeds Score is that the seeds are all members of the semantic category, so we might expect other members to occur near many of them. Averaging over all of the seeds can diffuse unusually high or low collocation counts that might result from an anomalous seed. The rationale for the Max of Seeds Score is that a word may naturally co-occur with some category members more often than others. For example, one would expect *dog* to co-occur with *cat* much more frequently than with *frog*. A high Max of Seeds Score indicates that there is at least one seed word that frequently co-occurs with the candidate.

Since Web queries are relatively expensive, it is worth taking stock of how many queries are necessary. Let N be the number of candidate words produced by Basilisk, and S be the number of seed words given to Basilisk as input. To compute the Hypernym Score for a candidate, we need 3 queries: $count(hypernym)$, $count(candidate)$, and $count(hypernym\ NEAR\ candidate)$. The first query is the same for all candidates, so for N candidate words we need $2N + 1$ queries in total. To compute the Average or Max of Seeds Score for a candidate, we need S queries for $count(seed_i)$, S queries for $count(seed_i\ NEAR\ candidate)$, and 1 query for $count(candidate)$. So for N candidate words we need $N * (2S + 1)$ queries. S is typically small for weakly supervised algorithms (S=10 in our experiments), which means that this Web-based corroboration process requires $O(N)$ queries to process a semantic lexicon of size N.

4 Evaluation

4.1 Data Sets

We ran experiments on two corpora: 1700 MUC-4 terrorism articles (MUC-4 Proceedings, 1992) and a combination of 6000 disease-related documents, consisting of 2000 ProMed disease outbreak reports (ProMed-mail, 2006) and 4000 disease-related PubMed abstracts (PubMed, 2009). For the terrorism domain, we created lexicons for four semantic categories: Building, Human, Location, and Weapon. For the disease domain, we created lexicons for three semantic categories: Animal[4], Disease, and Symptom. For each category, we gave Basilisk 10 seed words as input. The seeds were chosen by applying a shallow parser to each corpus, extracting the head nouns of all the NPs, and sorting the nouns by frequency. A human then walked down the sorted list and identified the 10 most frequent nouns that belong to each semantic category[5]. This strategy ensures that the bootstrapping process is given seed words that occur in the corpus with high frequency. The seed words are shown in Table 1.

Building:	embassy office headquarters church offices house home residence hospital airport
Human:	people guerrillas members troops Cristiani rebels president terrorists soldiers leaders
Location:	country El_Salvador Salvador United_States area Colombia city countries department Nicaragua
Weapon:	weapons bomb bombs explosives arms missiles dynamite rifles materiel bullets
Animal:	bird mosquito cow horse pig chicken sheep dog deer fish
Disease:	SARS BSE anthrax influenza WNV FMD encephalitis malaria pneumonia flu
Symptom:	fever diarrhea vomiting rash paralysis weakness necrosis chills headaches hemorrhage

Table 1: Seed Words

To evaluate our results, we used the gold standard answer key that Thelen & Riloff created to evaluate Basilisk on the MUC4 corpus (Thelen and Riloff, 2002); they manually labeled every head noun in the corpus with its semantic class. For the ProMed / PubMed disease corpus, we created our own answer key. For all of the lexicon entries hypothesized by Basilisk, a human annotator (not any of the authors)

[4] Animal was chosen because many of the ProMed disease outbreak stories concerned outbreaks among animal populations.

[5] The disease domain seed words were chosen from a larger set of ProMed documents, which included the 2000 used for lexicon induction.

	Building				Human				Location				Weapon			
N	Ba	Hy	Av	Mx	Ba	Hy	Av	Mx	Ba	Hy	Av	Mx	Ba	Hy	Av	Mx
25	.40	**.56**	.52	**.56**	.40	.72	.80	**.84**	.68	.88	.88	**1.0**	.56	.84	**1.0**	**1.0**
50	.44	**.56**	.46	.40	.56	.80	**.88**	.86	.80	.86	.84	**.98**	.52	.74	.76	**.90**
75	.44	**.45**	.41	.39	.65	.84	**.85**	**.85**	.80	.88	.80	**.99**	.52	.63	.65	**.79**
100	**.42**	.41	.38	.36	.69	.81	.80	**.87**	.81	.85	.78	**.95**	.55	.55	.56	**.63**
300	.22				.82				.75				.26			

	Animal				Disease				Symptom			
N	Ba	Hy	Av	Mx	Ba	Hy	Av	Mx	Ba	Hy	Av	Mx
25	.48	.88	**.92**	**.92**	.64	**.84**	.80	**.84**	.64	.84	**.92**	.80
50	.58	.82	**.84**	.80	.72	**.84**	.60	.82	.62	.76	**.90**	.74
75	.55	.68	.67	**.69**	.69	**.83**	.59	.81	.61	.68	**.79**	.71
100	.45	.55	.54	**.57**	.69	.78	.58	**.80**	.59	.71	**.77**	.64
300	.20				.62				.38			

Table 2: Ranking results for 7 semantic categories, showing accuracies for the top-ranked N words. (Ba=Basilisk, Hy=Hypernym Re-ranking, Av=Average of Seeds Re-ranking, Mx=Max of Seeds Re-ranking

labeled each word as either correct or incorrect for the hypothesized semantic class. A word is considered to be correct if any sense of the word is semantically correct.

4.2 Ranking Results

We ran Basilisk for 60 iterations, learning 5 new words in each bootstrapping cycle, which produced a lexicon of 300 words for each semantic category. The columns labeled Ba in Table 2 show the accuracy results for Basilisk.[6] As we explained in Section 3.1, accuracy tends to decrease as bootstrapping progresses, so we computed accuracy scores for the top-ranked 100 words, in increments of 25, and also for the entire lexicon of 300 words.

Overall, we see that Basilisk learns many correct words for each semantic category, and the top-ranked terms are generally more accurate than the lower-ranked terms. For the top 100 words, accuracies are generally in the 50-70% range, except for LOCATION which achieves about 80% accuracy. For the HUMAN category, Basilisk obtained 82% accuracy over all 300 words, but the top-ranked words actually produced lower accuracy.

Basilisk's ranking is clearly not as good as it could be because there are correct terms co-mingled with incorrect terms throughout the ranked lists. This has two ramifications. First, if we want a human to manually review each lexicon before adding the words to an external resource, then the rankings may not be very helpful (i.e., the human will need to review all of the words), and (2) incorrect terms generated during the early stages of bootstrapping may be hindering the learning process because they introduce noise during bootstrapping. The HUMAN category seems to have recovered from early mistakes, but the lower accuracies for some other categories may be the result of this problem. The purpose of our Web-based corroboration process is to automatically re-evaluate the lexicons produced by Basilisk, using Web-based statistics to create more separation between the good entries and the bad ones.

Our first set of experiments uses the Web-based co-occurrence statistics to re-rank the lexicon entries. The Hy, Av, and Mx columns in Table 2 show the re-ranking results using each of the Hypernym, Average of Seeds, and Maximum of Seeds scoring functions. In all cases, Web-based re-ranking outperforms Basilisk's original rankings. Every semantic category except for BUILDING yielded accuracies of 80-100% among the top candidates. For each row, the highest accuracy for each semantic category is shown in boldface (as are any tied for highest).

Overall, the Max of Seeds Scores were best, performing better than or as well as the other scoring functions on 5 of the 7 categories. It was only out-

[6] These results are not comparable to the Basilisk results reported by (Thelen and Riloff, 2002) because our implementation only does single-category learning while the results in that paper are based on simultaneously learning multiple categories.

BUILDING	HUMAN	LOCATION	WEAPON	ANIMAL	DISEASE	SYMPTOM
consulate	guerrilla	San_Salvador	shotguns	bird-to-bird	meningo-encephalitis	nausea
pharmacies	extremists	Las_Hojas	carbines	cervids	bse).austria	diarrhoea
aiport	sympathizers	Tejutepeque	armaments	goats	inhalational	myalgias
zacamil	assassins	Ayutuxtepeque	revolvers	ewes	anthrax_disease	chlorosis
airports	patrols	Copinol	detonators	ruminants	otitis_media	myalgia
parishes	militiamen	Cuscatancingo	pistols	swine	airport_malaria	salivation
Masariegos	battalion	Jiboa	car_bombs	calf	taeniorhynchus	dysentery
chancery	Ellacuria	Chinameca	calibers	lambs	hyopneumonia	cramping
residences	rebel	Zacamil	M-16	wolsington	monkeypox	dizziness
police_station	policemen	Chalantenango	grenades	piglets	kala-azar	inappetance

Table 3: Top 10 words ranked by Max of Seeds Scores.

performed once by the Hypernym Scores (BUILDING) and once by the Average of Seeds Scores (SYMPTOM).

The strong performance of the Max of Seeds scores suggests that one seed is often an especially good collocation indicator for category membership – though it may not be the same seed word for all of the lexicon words. The relatively poor performance of the Average of Seeds scores may be attributable to the same principle; perhaps even if one seed is especially strong, averaging over the less-effective seeds' scores dilutes the results. Averaging is also susceptible to damage from words that receive the special-case score of -99999 when a hit count is zero (see Section 3.2).

Table 3 shows the 10 top-ranked candidates for each semantic category based on the Max of Seeds scores. The table illustrates that this scoring function does a good job of identifying semantically correct words, although of course there are some mistakes. Mistakes can happen due to parsing errors (e.g., *bird-to-bird* is an adjective and not a noun, as in *bird-to-bird transmission*), and some are due to issues associated with Web querying. For example, the nonsense term *"bse).austria"* was ranked highly because Altavista split this term into 2 separate words because of the punctuation, and *bse* by itself is indeed a disease term (*bovine spongiform encephalitis*).

4.3 Filtering Results

Table 2 revealed that the 300-word lexicons produced by Basilisk vary widely in the number of true category words that they contain. The least dense category is ANIMAL, with only 61 correct words, and the most dense is HUMAN with 247 correct words. Interestingly, the densest categories are not always the easiest to rank. For example, the HUMAN category is the densest category but Basilisk's ranking of the human terms was poor.

θ	*Category*	*Acc*	*Cor/Tot*
-22	WEAPON	**.88**	46/52
	LOCATION	**.98**	59/60
	HUMAN	.80	8/10
	BUILDING	**.83**	5/6
	ANIMAL	**.91**	30/33
	DISEASE	**.82**	64/78
	SYMPTOM	**.65**	64/99
-23	WEAPON	.79	59/75
	LOCATION	.96	82/85
	HUMAN	.85	23/27
	BUILDING	.71	12/17
	ANIMAL	.87	40/46
	DISEASE	.78	82/105
	SYMPTOM	.62	86/139
-24	WEAPON	.63	63/100
	LOCATION	.93	111/120
	HUMAN	**.87**	54/62
	BUILDING	.45	17/38
	ANIMAL	.75	47/63
	DISEASE	.74	94/127
	SYMPTOM	.60	100/166

Table 4: Filtering results using the Max of Seeds Scores.

The ultimate goal behind a better ranking mechanism is to completely automate the process of semantic lexicon induction. If we can produce high-quality rankings, then we can discard the lower ranked words and keep only the highest ranked words for our semantic dictionary. However, this presupposes that we know where to draw the line between the good and bad entries, and Table 2 shows that this boundary varies across categories. For HUMANS, the top 100 words are 87% accurate, and in fact we get 82% accuracy over all 300 words. But for ANIMALS we achieve 80% accuracy only for the top 50 words. It is paramount for semantic dictionaries to have high integrity, so accuracy must be high if we want to use the resulting lexicons without manual review.

As an alternative to ranking, another way that we could use the Web-based corroboration statistics is to automatically filter words that do not receive a high score. The key question is whether the values of the scores are consistent enough across categories to set a single threshold that will work well across the different categories.

Table 4 shows the results of using the Max of Seeds Scores as a filtering mechanism: given a threshold θ, all words that have a score $< \theta$ are discarded. For each threshold value θ and semantic category, we computed the accuracy (Acc) of the lexicon after all words with a score $< \theta$ have been removed. The Cor/Tot column shows the number of correct category members and the number of total words that passed the threshold.

We experimented with a variety of threshold values and found that θ=-22 performed best. Table 4 shows that this threshold produces a relatively high-precision filtering mechanism, with 6 of the 7 categories achieving lexicon accuracies $\geq$ 80%. As expected, the *Cor/Tot* column shows that the number of words varies widely across categories. Automatic filtering represents a trade-off: a relatively high-precision lexicon can be created, but some correct words will be lost. The threshold can be adjusted to increase the number of learned words, but with a corresponding drop in precision. Depending upon a user's needs, a high threshold may be desirable to identify only the most confident lexicon entries, or a lower threshold may be desirable to retain most of the correct entries while reliably removing some of the incorrect ones.

5 Conclusions

We have demonstrated that co-occurrence statistics gathered from the Web can dramatically improve the ranking of lexicon entries produced by a weakly-supervised corpus-based bootstrapping algorithm, without requiring any additional supervision. We found that computing Web-based co-occurrence statistics across a set of seed words and then using the highest score was the most successful approach. Co-occurrence with a hypernym term also performed well for some categories, and could be easily combined with the Max of Seeds approach by choosing the highest value among the seeds as well as the hypernym.

In future work, we would like to incorporate this Web-based re-ranking procedure into the bootstrapping algorithm itself to dynamically “clean up” the learned words before they are cycled back into the bootstrapping process. Basilisk could consult the Web-based statistics to select the best 5 words to generate before the next bootstrapping cycle begins. This integrated approach has the potential to substantially improve Basilisk's performance because it would improve the precision of the induced lexicon entries during the earliest stages of bootstrapping when the learning process is most fragile.

Acknowledgments

Many thanks to Julia James for annotating the gold standards for the disease domain. This research was supported in part by Department of Homeland Security Grant N0014-07-1-0152.

References

J. Atserias, S. Climent, X. Farreres, G. Rigau, and H. Rodriguez. 1997. Combining Multiple Methods for the Automatic Construction of Multilingual WordNets. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*.

S. Caraballo. 1999. Automatic Acquisition of a Hypernym-Labeled Noun Hierarchy from Text. In *Proc. of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 120–126.

P. Cimiano and J. Volker. 2005. Towards large-scale, open-domain and ontology-based named entity classification. In *Proc. of Recent Advances in Natural Language Processing*, pages 166–172.

D. Davidov and A. Rappoport. 2006. Efficient unsupervised discovery of word categories using symmetric patterns and high frequency words. In *Proc. of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL.*

D. Davidov, A. Rappoport, and M. Koppel. 2007. Fully unsupervised discovery of concept-specific relationships by web mining. In *Proc. of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 232–239, June.

O. Etzioni, M. Cafarella, D. Downey, A. Popescu, T. Shaked, S. Soderland, D. Weld, and A. Yates. 2005. Unsupervised named-entity extraction from the web: an experimental study. *Artificial Intelligence*, 165(1):91–134, June.

Z. Harris. 1954. Distributional Structure. In J. A. Fodor and J. J. Katz, editor, *The Structure of Language*, pages 33–49. Prentice-Hall.

M. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proc. of the 14th International Conference on Computational Linguistics (COLING-92)*.

Z. Kozareva, E. Riloff, and E. Hovy. 2008. Semantic Class Learning from the Web with Hyponym Pattern Linkage Graphs. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08)*.

D. Lin and P. Pantel. 2002. Concept discovery from text. In *Proc. of the 19th International Conference on Computational linguistics*, pages 1–7.

D. Lin. 1998. Dependency-based Evaluation of MINIPAR. In *Workshop on the Evaluation of Parsing Systems, Granada, Spain*.

G. Mann. 2002. Fine-grained proper noun ontologies for question answering. In *Proc. of the 19th International Conference on Computational Linguistics*, pages 1–7.

G. Miller. 1990. Wordnet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4).

MUC-4 Proceedings. 1992. *Proceedings of the Fourth Message Understanding Conference (MUC-4)*. Morgan Kaufmann.

M. Paşca. 2004. Acquisition of categorized named entities for web search. In *Proc. of the Thirteenth ACM International Conference on Information and Knowledge Management*, pages 137–145.

P. Pantel and D. Ravichandran. 2004. Automatically labeling semantic classes. In *Proc. of Conference of HLT / North American Chapter of the Association for Computational Linguistics*, pages 321–328.

P. Pantel, D. Ravichandran, and E. Hovy. 2004. Towards terascale knowledge acquisition. In *Proc. of the 20th international conference on Computational Linguistics*, page 771.

M. Pasca. 2007. weakly-supervised Discovery of Named Entities using Web Search Queries. In *CIKM*, pages 683–690.

W. Phillips and E. Riloff. 2002. Exploiting Strong Syntactic Heuristics and Co-Training to Learn Semantic Lexicons. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 125–132.

ProMed-mail. 2006. *http://www.promedmail.org/*.

PubMed. 2009. *http://www.ncbi.nlm.nih.gov/sites/entrez*.

E. Riloff and R. Jones. 1999. Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*.

E. Riloff and J. Shepherd. 1997. A Corpus-Based Approach for Building Semantic Lexicons. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124.

E. Riloff. 1996. Automatically Generating Extraction Patterns from Untagged Text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 1044–1049. The AAAI Press/MIT Press.

B. Roark and E. Charniak. 1998. Noun-phrase Co-occurrence Statistics for Semi-automatic Semantic Lexicon Construction. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics*, pages 1110–1116.

Sofia Stamou, Kemal Oflazer, Karel Pala, Dimitris Christoudoulakis, Dan Cristea, Dan Tufis, Svetla Koeva, George Totkov, Dominique Dutoit, and Maria Grigoriadou. 2002. Balkanet: A multilingual semantic network for the balkan languages. In *Proceedings of the 1st Global WordNet Association conference*.

H. Tanev and B. Magnini. 2006. Weakly supervised approaches for ontology population. In *Proc. of 11st Conference of the European Chapter of the Association for Computational Linguistics*.

M. Thelen and E. Riloff. 2002. A Bootstrapping Method for Learning Semantic Lexicons Using Extraction Pa ttern Contexts. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 214–221.

Peter D. Turney. 2001. Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 491–502, London, UK. Springer-Verlag.

P. D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, pages 417–424.

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

D. Widdows and B. Dorow. 2002. A graph model for unsupervised lexical acquisition. In *Proc. of the 19th International Conference on Computational Linguistics*, pages 1–7.