Proceedings of the Australasian Language Technology Workshop 2007, pages 66-74

Experiments in Mutual Exclusion Bootstrapping

Tara Murphy and **James R. Curran**
School of Information Technologies
University of Sydney
NSW 2006, Australia
{tm, james}@it.usyd.edu.au

Abstract

Mutual Exclusion Bootstrapping (MEB) was designed to overcome the problem of *semantic drift* suffered by iterative bootstrapping, where the meaning of extracted terms quickly drifts from the original seed terms (Curran et al., 2007). MEB works by extracting mutually exclusive classes in parallel which constrain each other.

In this paper we explore the strengths and limitations of MEB by applying it to two novel lexical-semantic extraction tasks: extracting bigram named entities and WordNet lexical file classes (Fellbaum, 1998) from the Google Web 1T 5-grams.

1 Introduction

Extracting lexical semantic resources from text with minimal supervision is critical to overcoming the knowledge bottleneck in Natural Language Processing (NLP) tasks ranging from Word Sense Disambiguation to Question Answering.

Template-based extraction is attractive because it is reasonably efficient, works on small and large datasets, and requires minimal linguistic preprocessing, making it fairly language independent. Hearst (1992) proposed template-based extraction for identifying hyponyms using templates like X, Y, and/or other Z where X and Y are hyponyms of Z.

Riloff and Shepherd (1997) proposed *iterative bootstrapping* where frequent neighbours to terms from a given semantic class are extracted in multiple bootstrap iterations. Roark and Charniak (1998) improved its accuracy by optimising the bootstrapping parameters. In *mutual bootstrapping* (Riloff and Jones, 1999) the terms, and the contexts they occur in, are extracted. Similar approaches have been used in Information Extraction (IE) for identifying company headquarters (Agichtein et al., 2000) and acronym expansions (Sundaresan and Yi, 2000).

In *Mutual Exclusion Bootstrapping* (MEB), we assume the semantic classes partition terms into disjoint sets, that is, the classes are *mutually exclusive* (Curran et al., 2007). Each class is extracted in parallel using separate bootstrapping loops that each race to collect terms and contexts. Although this assumption is clearly false, it significantly reduces the extraction errors of existing approaches.

This paper presents two applications of MEB that allow some insight into MEB's strengths and limitations. First, we extend MEB to extracting *bigram* BBN named entity types (Weischedel and Brunstein, 2005). We discover that both unigram and bigram MEB are very sensitive to the context window surrounding the extracted terms. Surprisingly, MEB is insensitive to the order the semantic classes are presented and the noise in the terms themselves.

Second, we extract common nouns using 25 semantic classes defined by the WordNet lexical files (Fellbaum, 1998). We use a closed vocabulary of WordNet unigram nouns, so the evaluation can be performed automatically against WordNet. We find that MEB performs well on classes with narrow definitions and thus more coherent contexts, such as animal, but performs poorly on classes like cognition. We also find that increasing the number of seed terms increases the accuracy significantly.

2 Mutual Exclusion Bootstrapping

Mutual bootstrapping (Riloff and Jones, 1999) has the advantage that it can identify new templates or *contexts*, which in turn identify new terms, significantly increasing recall. Unfortunately, erroneously adding a term with a different predominant sense or a context that weakly constrains the terms, quickly leads to *semantic drift*, where erroneous terms or contexts infect the semantic class.

Mutual Exclusion Bootstrapping (MEB) attempts to minimise semantic drift in both the terms and contexts (Curran et al., 2007). It does this by extracting all of the semantic classes in parallel, using an independent bootstrapping loop for each class, with the constraint that a term or context *must only be used by one* class. We assume that each term has only one sense and that each context only extracts terms with one sense, that is, the semantic classes are *mutually exclusive* with respect to terms and contexts.

This assumption is far from realistic, but it is very effective at reducing the degree of semantic drift. For many terms, especially the bigram named entities, there is a clearly dominant semantic class. However, for some pairs of semantic classes, e.g. nationalities and languages, there is a significant lexical overlap and so they are far from mutually exclusive.

The MEB algorithm is shown in Algorithm 1. In each iteration, contexts and then terms are added to each semantic class. If more than one class attempts to extract a context or term in the current iteration then it is eliminated, leading to mutual exclusion between the semantic classes. The terms and contexts are ranked in the same way as Riloff and Jones (1999), our only addition in MEB is the parallel mutual exclusion constraint.

Mutual exclusion is very strict and many terms and contexts are discarded. This is not a major issue when precision is paramount and we are using a large dataset, e.g. Web 1T, but it can be problematic on smaller datasets. It is a significant problem when the semantic classes are far from mutually exclusive because many viable contexts are rejected when the terms they extract are polysemous, even though the contexts themselves reliably select one sense.

MEB is potentially sensitive to the order the contexts and terms are added to semantic classes, since once they are added to a class they cannot be added elsewhere (by the mutual exclusion assumption). In this sense, the individual bootstrapping loops compete in parallel to reach a term or context first, and claim it for themselves. Polysemous terms may be added to just one semantic class if it is not identified by contexts from multiple semantic classes simultaneously, and this also applies for contexts. There is no guarantee that the predominant sense of a term will be reached first, although if it is significantly more frequent, it is likely to be reached first since it will appear in more contexts.

in : Seed word lists S_k $\forall$ categories k
in : Raw contexts $\mathscr{C}$ and terms $\mathscr{T}$
in : # terms N_T and contexts N_C per iteration
out: Term T_k and context C_k lists $\forall$ categories k
$T_k \longleftarrow S_k \forall$ categories k;
foreach *iteration* **do**
  foreach $c \in \mathscr{C}$ **do**
    count # times c occurs with each $t \in T_k$;
    discard c if occurs with multiple classes;
  foreach *class* k **do**
    sort set of c by above occurrence counts;
    add top N_C contexts to C_k;
  foreach $t \in \mathscr{T}$ **do**
    count # times t occurs with each $c \in C_k$;
    discard t if occurs with multiple classes;
  foreach *class* k **do**
    sort set of t by above occurrence counts;
    add top N_T terms to T_k;

Algorithm 1: Mutual Exclusion Bootstrapping

3 Using the Google Web 1T n-grams

Riloff and Jones (1999) used contexts extracted from POS tagged and chunked text by AutoSlog-TS (Riloff, 1996). Our goal was to keep MEB language independent to maintain this key advantage of template-based approaches. We also wanted to demonstrate that MEB scales efficiently to extremely large datasets and vocabularies.

Google has collected the Web 1T corpus (Brants and Franz, 2006), which consists of unigram to 5-gram counts calculated over 1 trillion words of web text collected during January 2006. The text was tokenised using Penn Treebank tokenisation, except that words are usually split on hyphens; and dates, email addresses, and URLs are kept as single tokens. Sentence boundaries were detected using sta-

tistical techniques. The individual words in the n-grams occurred ≥200 times, otherwise they were replaced with <UNK>. Each n-gram appears ≥40 times. There is 25GB of compressed data.

We use the 3-, 4-, or 5-grams from Web 1T as our raw data, depending on the experiment. The middle token (for unigrams) or tokens (for bigrams) form the *term* and the one or two tokens on either side form the *context*. This context definition is quite language independent (except for languages without word segmentation). Unfortunately, we can only extract terms consisting of one or two words, and the contexts are noisier than those extracted from parsed text, cf. Curran (2004).

For the bigram experiments we follow the process described in Curran et al. (2007). We removed n-grams with non-titlecase middle token(s) because we only extract proper noun named entity types, and we removed all contexts containing numbers. For the WordNet experiments we only included n-grams where the middle token(s) were a term in WordNet. In every experiment, we eliminate contexts that only appear with one term, and thus terms that only appear in one context, since they cannot be reached.

The size of the resulting dataset varied depending on the experiment from 176MB (for the bigrams heavily filtered using the t-test) to 1.2GB (for the bigrams with a window of one word either side and the WordNet experiments). All of the data must be loaded into memory and for the largest experiments this requires 1.6GB of RAM using our space-optimised C++ implementation.

4 Named Entity Classes

In our first set of experiments we continue our previous work on proper-noun named entities. We based our semantic classes on the 29 entity types used to annotated the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein, 2005). We ignored entity types that did not primarily include proper nouns, for example the DESCRIPTION types, CHEMICALS and QUANTITIES.

For the unigram experiments we reused our previous classification where we ignored entity types that were almost exclusively multi-word terms, for example WORKS OF ART and LAWS. We also split the PERSON class into MALE and FEMALE first names and LAST names to investigate more fine-grained distinctions for this class.

LABEL	UNI	BI	DESCRIPTION
NAME		•	Person: name *'Katie Holmes' 'Adam Smith'*
FEM	•		Person: female first name *Mary Patricia Linda Elizabeth*
MALE	•		Person: male first name *James John Robert Michael William*
LAST	•		Person: last name *Smith Johnson Williams Jones Brown*
TITLE	•	•	Honorific title *President Dr Lord Miss Major*
NORP	•	•	Nationality, Religion, Political (adj) *Republican Christian 'South African'*
FAC	•	•	Facility: names of man-made structures *Broadway Legoland 'Golden Gate'*
ORG	•	•	Organisation: e.g. companies, gov. *Intel Microsoft 'American Express'*
GPE	•	•	Geo-political entity *Canada China London 'Los Angeles'*
LOC	•	•	Locations other than GPEs *Africa Asia Pacific Earth 'Middle East'*
DATE	•	•	Reference to a date or period *January May Friday 'Easter Day'*
LANG	•	•	Any named language *English Arabic Hebrew 'Scots Gaelic'*
EVENT		•	Battles, sporting, and other named events *'World War' 'Hurricane Katrina'*
LAW		•	Document that has been made into a law *'Reform Act' 'First Amendment'*

Table 1: The semantic classes used for the proper noun unigram (Column 2) and bigram (Column 3) experiments. Bigram examples are shown in quotes.

For the bigram experiments, we kept a single class NAME for person name, and reintroduced the LAW and EVENT classes. Most classes are common to both the unigram and bigram experiments. As in our previous experiments, some classes were easier to evaluate manually because we were only extracting unigrams, whilst others were more difficult. Similar difficulties exist in the bigram classes as well. The complete list of semantic classes used in the named entity experiments are summarised in Table 1.

5 Named Entity Evaluation

Our evaluation followed the manual inspection process used in our previous experiments. To make this more efficient, we stored a cache of previous evaluator decisions for each class, so that once a decision had been made for a particular term in a particular class it would be made automatically in future instances. This dramatically reduces the effort re-

quired for manual evaluation.

Although the seed lists were mutually exclusive, for the purposes of evaluation, ambiguous words such as *French* were counted as correct if they appeared in either valid category (NORP or LANG).

If a single word was an clearly part of a multi-word term we counted it as correct (e.g. *Coast* as a LOC) with the exception of the mixed unigram-bigram experiments. If the word was not strongly indicative of a semantic class (e.g. *The*) it was not counted as correct. Mis-spellings of words (e.g. *Januray*) were also counted as correct. The extracted terms that were unrecognised by the human evaluator were checked using Wikipedia and Google.

We calculate accuracy at n – the percentage of correct terms in the top n ranked terms, following previous bootstrapping work. This is averaged over the semantic classes (Av(n)). We manually evaluated all semantic classes down to $n = 50$, which adequately discriminates between most configurations. We vary the number of seeds (nS), and terms (nT) and context (nC) added in each iteration.

6 Named Entity Experiments

Our initial expectation was that bigram named entities would be an easier task than unigram named entities because they had fewer senses and so better satisfied the mutual exclusion assumption. Also, we expected them to be easier to evaluate since they were less ambiguous. However, the results did not match our intuition and so we experimented with unigrams and bigrams to determine the cause.

6.1 Context Geometry

A major disadvantage for the bigram and longer n-gram experiments is that the size of the context must be reduced to accommodate the term itself within a fixed sized n-gram (e.g. the Web 1T 5-grams). Even if longer n-grams were collected for bootstrapping, there would still be the problem of sparser counts (even from one trillion words).

We started by repeating our original unigram named entity experiments but this time we reduced the context window to one token on the left and/or right, as shown in Table 2. Table 3 shows the impact of context geometry on unigram accuracy. Our previous best unigram results are UNI5GMS with 78%. Removing a token from the right context (UNI4LEFT) makes almost no difference to the results, but removing a token from the left context (UNI4RIGHT) makes an enormous difference (a loss of almost 30%). The effect of removing both (UNI3GMS) is slightly worse again.

NAME	TEMPLATE
UNI5GMS	$w_1\ w_2\ \mathbf{X}\ w_3\ w_4$
UNI4LEFT	$w_1\ w_2\ \mathbf{X}\ w_3$
UNI4RIGHT	$w_1\ \mathbf{X}\ w_2\ w_3$
UNI3GMS	$w_1\ \mathbf{X}\ w_2$
BI5LEFT	$w_1\ w_2\ \mathbf{X}\ \mathbf{X}\ w_3$
BI5RIGHT	$w_1\ \mathbf{X}\ \mathbf{X}\ w_2\ w_3$
BI4GMS	$w_1\ \mathbf{X}\ \mathbf{X}\ w_2$

Table 2: Unigram and bigram Web 1T templates.

	nS	nT	nC	Av(10)	Av(50)
UNI5GMS	5	5	10	90	**78**
UNI4LEFT	5	5	10	86	74
UNI4RIGHT	5	5	10	76	49
UNI3GMS	5	5	10	74	48

Table 3: Effect of context geometry on unigrams.

	nS	nT	nC	Av(10)	Av(50)
BI5LEFT	5	5	10	92	**68**
BI5RIGHT	5	5	10	83	51
BI4GMS	5	5	10	77	48

Table 4: Effects of context geometry on bigrams.

We should also note that the UNI3GMS and UNI4GMS experiments use the Web 1T 3- and 4-gram data, so the counts are larger and more reliable, and the chance of shared contexts is greater. This suggests that the impact of removing the left context is even greater than these results indicate.

We also considered the minimum number of contexts a term had to appear in to be included. Our previous experiments required two contexts – otherwise a term cannot be discovered. Increasing this cutoff to 10 made no significant difference.

The impact of context geometry on bigram accuracy is shown in Table 4. The penalty for removing some left context was not as great for bigrams, dropping from 68% with BI5GMS to 48% with BI4GMS. The remaining unigram experiments use UNI5GMS and the bigram experiments use BI5LEFT.

STOP	mean Av(50)	σ Av(50)
NO	69	2.9
YES	**75**	3.0

Table 5: Effects of category order on unigrams.

6.2 Class Ordering and Stop Classes

One criticism of the MEB algorithm is that it may be highly dependant on the order in which the classes are considered. Because of the mutual exclusion (i.e. once a word has been assigned to a particular class, it can't be assigned to any other class) the class order clearly has the potential to impact the results.

To test this we have run a set of ten unigram experiments with the classes arranged in random permutations. The results are shown in Table 5. The standard deviation of the ten sets is 3.0, around a mean of 75. This shows that although it has some impact, MEB is reasonably robust to changes in the category order. The standard deviation from these experiments can be used as an indication of the scatter across the MEB experiments in general.

We also compared the accuracy of using, and not using, the stop classes, which are used to constrain specific semantic drift problems (Curran et al., 2007). When the stop classes were used, they always appeared first in the same order, before the randomly permuted semantic classes. The difference in the means in Table 5 between the set with and without stop classes shows that using stop classes does improve the accuracy of MEB.

6.3 Number of Contexts

Our experiments with unigrams in Curran et al. (2007) showed that the best results were obtained by adding 10 contexts per iteration of the bootstrapping process. We have repeated this experiment for bigrams, with the results shown in Table 6.

These experiments show that that best results are obtained for numbers of contexts between 5 and 20. There is a significant drop-off in accuracy ($\sim 10-20\%$) for values of nC less than 5 or greater than 20. This demonstrates a preference for having more evidence for new terms being reliable than for simply adding more terms in each iteration. It also shows that keeping the number of terms added per iteration (nT) and the number of contexts (nC) added per iteration reasonably well balanced is the best strategy. This makes sense if we consider the extreme cases: adding 5 terms and only 1 new context per iteration would mean that it was difficult for the system to expand into new space; adding 5 terms and 100 new contexts per iteration would mean that many of the contexts may not be representative of the contexts that those 5 terms appear in.

nS	nT	nC	Av(10)	Av(50)
5	5	1	73	52
5	5	2	76	50
5	5	5	92	71
5	5	10	92	**73**
5	5	20	93	70
5	5	50	87	58
5	5	100	84	59

Table 6: Effects of changing the number of contexts added per iteration.

6.4 Filtering Using Collocations

One issue that arises when extracting bigrams (or longer n-grams) is the possibility that random combinations of tokens may be selected by chance in the MEB process. To investigate this we have carried out a series of experiments on data that was pre-filtered using collocation statistics.

We filtered the Web 1T data so that we only kept bigrams that were significant collocations based on their frequency in the Web 1T corpus. We chose the t-test as our measure of significance as it is simple to calculate and we do not have any low frequency values (< 5) for which the t-test is known to perform badly. Our calculation follows Manning and Schütze (1999, pg. 165). If $f(w)$ and $f(w_1, w_2)$ are the unigram and bigram frequencies from Web 1T, then the t-test is:

$$t = \frac{p(w_1, w_2) - p(w_1)p(w_2)}{\sqrt{\frac{p(w_1, w_2)}{N-1}}} \quad (1)$$

where N is the number of tokens. Using the Maximum Likelihood Estimates (MLE) we have:

$$p(w_1, w_2) = \frac{f(w_1, w_2)}{N-1} \quad (2)$$

$$p(w) = \frac{f(w)}{N} \quad (3)$$

The results for cutoffs at different significance levels are shown in Table 7. These experiments

t	nS	nT	nC	Av(10)	Av(50)
50	5	5	10	86	69
100	5	5	10	87	**70**
250	5	5	10	85	68
500	5	5	10	81	58

Table 7: Effects of using only significant collocations. A value of 100 in column 1 means that only bigrams with a significance of $t \geq 100$ were used.

show that the filtering had no statistically significant result on the accuracy of MEB. In a sense, this is not surprising, as the MEB process of ranking new terms on the number of contexts they occur in is already performing a form of significance testing.

However, filtering on collocations does have the advantage of significantly reducing the size of the vocabulary without a significant loss of accuracy at the Av(50) level. For example the number of unique bigram terms in the BI5LEFT experiments in previous sections is 1 858 097, compared to 482 053 for the $t \geq 100$ filtered subset ($\sim 25\%$) and 87 537 for the $t \geq 250$ filtered subset ($\sim 5\%$). This is particularly important when dealing with massive corpora.

6.5 Multi-word Expressions

Of greater interest than extracting unigrams or bigrams alone is the application of MEB to the general case of extracting n-grams of any length. Since the maximum length of the term and context in the Web 1T corpus is five tokens, and given the decline in accuracy that comes with reducing the length of the context (see Tables 3 and 4) it would be impractical to extract terms with more than two tokens.

Hence our final experiment with proper noun named entity extraction combines the unigram and bigram data together. This serves as an initial test of extracting multi-word expressions as it is not specific to only unigrams or only bigrams. The data consists of that used for UNI4LEFT and BI5LEFT, so that the context surrounding the unigram or bigram has the same length and geometry.

The categories we used for this experiment are those in Table 1 that are marked as suitable for both unigrams and bigrams. The results for this experiment are shown in Table 8. These are comparable to our best results for bigram extraction.

nS	nT	nC	Av(10)	Av(50)
5	5	10	84	69

Table 8: Results for extracting bi- and unigrams

	UNI5GMS	WordNet
terms	263 613	29 157
contexts	10 449 412	18 832 474
terms-contexts	42 039 483	88 178 856

Table 9: Comparison of the datasets used in the UNI5GMS and WordNet experiments. The number of unique terms, unique contexts and unique term-context combinations is shown.

7 WordNet Common Nouns

In our second set of experiments we investigate the application of MEB to common nouns. For these experiments we used the noun classes from WordNet, as described in the next section. We expected the performance on this task to be worse than for proper nouns for a number of reasons. Firstly, common nouns have a larger number of senses, on average, compared to proper nouns. This breaks the mutual exclusion assumption that is central to MEB's success. Secondly, common nouns are likely to occur in a wider range of contexts than many proper nouns. Thirdly, the common noun categories are more general and less well defined than for proper nouns, and abstract nouns are also likely to be harder to categorise than concrete nouns.

One factor that favours common noun extraction is that the WordNet classes are designed to have reasonably complete coverage of the semantic space. This is not the case in the BBN named entity categories, which is one of the reasons why we introduced stop classes (Curran et al., 2007).

Table 9 compares the size of the initial dataset for the UNI5GMS experiments (Section 6) and the WordNet common noun experiments. Even though we have ~ 10 times fewer unique terms in the WordNet dataset, the number of unique term-context combinations is double that in the UNI5GMS dataset. The total size of the dataset used for the common noun experiments is 1.2GB.

7.1 WordNet Categories

For common nouns we used 25 noun categories from WordNet 3.0. These come from the broad seman-

CATEGORY	# WORDS	# UNI	# BI
act	6650	4917	1512
animal	7509	5010	2227
artifact	11587	7176	4163
attribute	3039	2646	322
body	2016	930	898
cognition	2964	2118	724
communication	5607	3788	1557
event	1074	844	216
feeling	428	396	31
food	2573	1347	1131
group	2624	1218	1012
location	3209	2272	788
motive	42	31	9
object	1545	1000	455
person	11087	9426	1516
phenomenon	641	332	285
plant	8030	4200	3382
possession	1061	492	514
process	770	594	162
quantity	1275	806	287
relation	437	266	150
shape	341	252	78
state	3544	2403	991
substance	2983	1869	1060
time	1028	574	375

Table 10: Noun categories in WordNet and the number of words, unigrams and bigrams in each.

tic classes employed by lexicographers in the initial phase of inserting words into the WORDNET hierarchy, called *lexicographer files* (*lex files*). For the noun hierarchy, there are 25 lex files and a file containing the top level nodes in the hierarchy called Tops. Lex files form a set of coarse-grained sense distinctions within WORDNET. These categories and the number of WordNet words in each category are shown in Table 7.1.

7.2 WordNet Evaluation

These experiments only involved unigrams seen in WordNet and hence we could evaluate directly against WordNet as a complete gold standard. We extracted the unigrams from all of the noun categories in WordNet. We then filtered the Web 1T corpus to extract only contexts where a WordNet unigram was the central token. The rest of the filtering, evaluation and scoring details follow the principles described in Section 5.

nS	nT	nC	Av(10)	Av(50)
5	5	10	29	22
10	5	10	51	43
20	5	10	67	52
100	5	10	**73**	**59**

Table 11: Effects of number of seed words.

Each proposed term was marked as correct if it appeared in that WordNet semantic category. The advantage of a closed system is the ease of evaluating the results. However, an obvious disadvantage is that the system cannot be marked correct for valid unigrams it discovers in a category, that are not listed under that category in WordNet. A full manual evaluation may produce better results.

8 WordNet Experiments

Creating seed lists using the Web 1T frequencies, as we had done in previous experiments, was complicated by skew towards web-related senses. For example, thumbnail was the 5th most frequent word in the body category and site was the 2nd most frequent word in the location category. In the number of seeds experiments we chose the seeds based on their frequency alone, but in the remaining experiments we manually created seed lists.

8.1 Number of Seed Words

We use the *n* most frequent words that were unique to each category as seeds, regardless of whether they have obvious web-related senses. The results for increasing the number of seed words are shown in Table 11. Note that the seed words are not included in the accuracy calculation. The limited number of terms in some categories (in particular, motive) causes a decrease in accuracy when more seeds are used because many of the correct proposed synonyms are now seed words.

There is a substantial increase in accuracy as the number of seeds is increased. This shows that even though the choice of seeds is far from optimal, and is strongly affected by interference, the results are still reasonable as long as a large number of seed words is used.

CATEGORY	Av(10)	Av(50)
animal	100	92
communication	100	94
food	100	96
location	100	98
cognition	0	12
feeling	60	24
object	40	20
relation	60	22
Mean	**62**	**44**

Table 12: Results for a selection of high and low performing common noun categories. The mean was calculated across all the semantic classes. The other parameters were (nS, nT, nC) = (5, 5, 10).

8.2 Comparison of Semantic Classes

To compare performance across semantic classes, we manually selected 5 seed words from the 20 most frequent words in each category (as measured in the Web 1T corpus). This allowed us to excluded words which we knew to have web-related senses that would dominate on the Web 1T data.

The accuracy obtained was 44%, which is substantially lower than for the named entity unigram experiments (maximum 78%). However, the variation in performance across the categories was extremely high, as demonstrated in Table 12. Some categories, such as cognition are extremely difficult.

This demonstrates that MEB is very good at extracting certain kinds of lexical semantic knowledge – primarily for categories that are very well defined, with frequent terms that appear in fairly constrained or idiomatic contexts, for example animals and food. For these categories, MEB performed just as well on common nouns as it did on many of the proper noun named entity categories.

Conclusions

We have presented two novel applications of Mutual Exclusion Bootstrapping (MEB): extracting bigram named entities and common nouns from WordNet. We confirmed that MEB is sensitive to the geometry of the context window surrounding the extracted terms. As expected, a larger context leads to higher accuracy, but interestingly, this is almost entirely due to extra context on the left of the target term. Overall, this makes bigram and longer n-gram extraction more difficult on fixed-sized window data, such as the Web 1T corpus.

Surprisingly, we discovered that MEB is relatively insensitive to the order the semantic classes are presented and to noise in the possible terms themselves.

We applied MEB to common nouns using 25 semantic classes defined by the WordNet lexical files. We performed automatic evaluation using a closed vocabulary and found that MEB performed well on classes with narrower definitions such as animal, but poorly on classes such as cognition. This is partly due to the concrete categories having more coherent contexts. We found that increasing the number of seed terms improved the accuracy, even with poor quality seed terms.

We now plan to experiment with loosening the mutual exclusion assumption to allow for some overlap between categories. There are many possibilities for improving the performance of MEB on common nouns – here we have presented only a preliminary analysis of the WordNet results. We also plan to experiment with text other than the Web 1T corpus so that we can test whether allowing wider contexts will further improve performance.

The experiments we have presented in this paper have demonstrated that MEB is an efficient and accurate method of extracting semantic classes over both unigram and bigram named entities. We have also demonstrated its potential for extracting semantic classes from WordNet for common nouns.

Acknowledgements

Both authors were funded on this work under ARC Discovery grants DP0453131 and DP0665973. We would like to thank the anonymous reviewers for their useful feedback.

References

Eugene Agichtein, Eleazar Eskin, and Luis Gravano. 2000. Combining strategies for extracting relations from text collections. Technical Report CUCS-006-00, Department of Computer Science, Columbia University, New York, March.

Thorsten Brants and Alex Franz. 2006. Web 1T 5-gram version 1. Technical Report LDC2006T13, Linguistic Data Consortium.

James R. Curran, Tara Murphy, and Bernhard Scholz. 2007. Minimising semantic drift with mutual exclusion bootstrapping. In *Proceedings of the Conference of the Pacific Association for Computational Linguistics*, pages 172–180, Melbourne, Australia, 19–21 September.

James R. Curran. 2004. *From Distributional to Semantic Similarity*. Ph.D. thesis, University of Edinburgh, Edinburgh, UK.

Cristiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. The MIT Press, Cambridge, MA USA.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th international conference on Computational Linguistics*, pages 539–545, Nantes, France, 23–28 July.

Chris Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.

Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 474–479, Orlando, FL USA, 18–22 July.

Ellen Riloff and Jessica 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, Providence, 1–2 August.

Ellen Riloff. 1996. Automatically generating extraction patterns from untagged text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 1044–1049.

Brian Roark and Eugene Charniak. 1998. Noun-phrase co-occurrence statistic for semi-automatic semantic lexicon construction. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th annual meeting of the Association for Computational Linguistics*, pages 1110–1116, Montréal, Québec, Canada, 10–14 August.

Neel Sundaresan and Jeonghee Yi. 2000. Mining the web for relations. In *Proceedings of the 9th International World Wide Web Conference*, Amsterdam, Netherlands, 15–19 May.

Ralph Weischedel and Ada Brunstein. 2005. BBN pronoun coreference and entity type corpus. Technical Report LDC2005T33, Linguistic Data Consortium.