Reducing semantic drift with bagging and distributional similarity

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

Abstract

Iterative bootstrapping algorithms are typically compared using a single set of hand-picked seeds. However, we demonstrate that performance varies greatly depending on these seeds, and favourable seeds for one algorithm can perform very poorly with others, making comparisons unreliable. We exploit this wide variation with bagging, sampling from automatically extracted seeds to reduce semantic drift.

However, semantic drift still occurs in later iterations. We propose an integrated distributional similarity filter to identify and censor potential semantic drifts, ensuring over 10% higher precision when extracting large semantic lexicons.

1 Introduction

Iterative bootstrapping algorithms have been proposed to extract semantic lexicons for NLP tasks with limited linguistic resources. Bootstrapping was initially proposed by Riloff and Jones (1999), and has since been successfully applied to extracting general semantic lexicons (Riloff and Jones, 1999; Thelen and Riloff, 2002), biomedical entities (Yu and Agichtein, 2003), facts (Paşca et al., 2006), and coreference data (Yang and Su, 2007).

Bootstrapping approaches are attractive because they are domain and language independent, require minimal linguistic pre-processing and can be applied to raw text, and are efficient enough for tera-scale extraction (Paşca et al., 2006).

Bootstrapping is minimally supervised, as it is initialised with a small number of seed instances of the information to extract. For semantic lexicons, these seeds are terms from the category of interest. The seeds identify contextual patterns that express a particular semantic category, which in turn recognise new terms (Riloff and Jones, 1999).

Unfortunately, *semantic drift* often occurs when ambiguous or erroneous terms and/or patterns are introduced into and then dominate the iterative process (Curran et al., 2007).

Bootstrapping algorithms are typically compared using only a single set of hand-picked seeds. We first show that different seeds cause these algorithms to generate diverse lexicons which vary greatly in precision. This makes evaluation unreliable – seeds which perform well on one algorithm can perform surprisingly poorly on another. In fact, random gold-standard seeds often outperform seeds carefully chosen by domain experts.

Our second contribution exploits this diversity we have identified. We present an unsupervised bagging algorithm which samples from the extracted lexicon rather than relying on existing gazetteers or hand-selected seeds. Each sample is then fed back as seeds to the bootstrapper and the results combined using voting. This both improves the precision of the lexicon and the robustness of the algorithms to the choice of initial seeds.

Unfortunately, semantic drift still dominates in later iterations, since erroneous extracted terms and/or patterns eventually shift the category's direction. Our third contribution focuses on detecting and censoring the terms introduced by semantic drift. We integrate a distributional similarity filter directly into WMEB (McIntosh and Curran, 2008). This filter judges whether a new term is more similar to the earlier or most recently extracted terms, a sign of potential semantic drift.

We demonstrate these methods for extracting biomedical semantic lexicons using two bootstrapping algorithms. Our unsupervised bagging approach outperforms carefully hand-picked seeds by $\sim 10\%$ in later iterations. Our distributional similarity filter gives a similar performance improvement. This allows us to produce large lexicons accurately and efficiently for domain-specific language processing.

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 396–404,
Suntec, Singapore, 2-7 August 2009. ©2009 ACL and AFNLP

2 Background

Hearst (1992) exploited patterns for information extraction, to acquire *is-a* relations using manually devised patterns like *such Z as X and/or Y* where *X* and *Y* are hyponyms of *Z*. Riloff and Jones (1999) extended this with an automated bootstrapping algorithm, *Multi-level Bootstrapping* (MLB), which iteratively extracts semantic lexicons from text.

In MLB, bootstrapping alternates between two stages: pattern extraction and selection, and term extraction and selection. MB is seeded with a small set of user selected *seed* terms. These seeds are used to identify contextual patterns they appear in, which in turn identify new lexicon entries. This process is repeated with the new lexicon terms identifying new patterns. In each iteration, the top-n candidates are selected, based on a metric scoring their membership in the category and suitability for extracting additional terms and patterns.

Bootstrapping eventually extracts polysemous terms and patterns which weakly constrain the semantic class, causing the lexicon's meaning to shift, called *semantic drift* by Curran et al. (2007). For example, female firstnames may drift into flowers when *Iris* and *Rose* are extracted. Many variations on bootstrapping have been developed to reduce semantic drift.[1]

One approach is to extract multiple semantic categories simultaneously, where the individual bootstrapping instances compete with one another in an attempt to actively direct the categories away from each other. Multi-category algorithms outperform MLB (Thelen and Riloff, 2002), and we focus on these algorithms in our experiments.

In BASILISK, MEB, and WMEB, each competing category iterates simultaneously between the term and pattern extraction and selection stages. These algorithms differ in how terms and patterns selected by multiple categories are handled, and their scoring metrics. In BASILISK (Thelen and Riloff, 2002), candidate terms are ranked highly if they have strong evidence for a category and little or no evidence for other categories. This typically favours less frequent terms, as they will match far fewer patterns and are thus more likely to belong to one category. Patterns are selected similarly, however patterns may also be selected by different categories in later iterations.

Curran et al. (2007) introduced *Mutual Exclusion Bootstrapping* (MEB) which forces stricter boundaries between the competing categories than BASILISK. In MEB, the key assumptions are that terms only belong to a category and that patterns only extract terms of a single category. Semantic drift is reduced by eliminating patterns that collide with multiple categories in an iteration and by ignoring colliding candidate terms (for the current iteration). This excludes generic patterns that can occur frequently with multiple categories, and reduces the chance of assigning ambiguous terms to their less dominant sense.

2.1 Weighted MEB

The scoring of candidate terms and patterns in MEB is naïve. Candidates which 1) match the most input instances; and 2) have the potential to generate the most new candidates, are preferred (Curran et al., 2007). This second criterion aims to increase recall. However, the selected instances are highly likely to introduce drift.

Our *Weighted* MEB algorithm (McIntosh and Curran, 2008), extends MEB by incorporating term and pattern weighting, and a cumulative pattern pool. WMEB uses the χ^2 statistic to identify patterns and terms that are strongly associated with the growing lexicon terms and their patterns respectively. The terms and patterns are then ranked first by the number of input instances they match (as in MEB), but then by their weighted score.

In MEB and BASILISK[2], the top-k patterns for each iteration are used to extract new candidate terms. As the lexicons grow, general patterns can drift into the top-k and as a result the earlier precise patterns lose their extracting influence. In WMEB, the pattern pool accumulates all top-k patterns from previous iterations, to ensure previous patterns can contribute.

2.2 Distributional Similarity

Distributional similarity has been used to extract semantic lexicons (Grefenstette, 1994), based on the *distributional hypothesis* that semantically similar words appear in similar contexts (Harris, 1954). Words are represented by context vectors, and words are considered similar if their context vectors are similar.

Patterns and distributional methods have been combined previously. Pantel and Ravichandran

[1] Komachi et al. (2008) used graph-based algorithms to reduce semantic drift for Word Sense Disambiguation.

[2] In BASILISK, k is increased by one in each iteration, to ensure at least one new pattern is introduced.

TYPE (#)	MEDLINE
Terms	1 347 002
Contexts	4 090 412
5-grams	72 796 760
Unfiltered tokens	6 642 802 776

Table 1: Filtered 5-gram dataset statistics.

(2004) used lexical-syntactic patterns to label clusters of distributionally similar terms. Mirkin et al. (2006) used 11 patterns, and the distributional similarity score of each pair of terms, to construct features for lexical entailment. Paşca et al. (2006) used distributional similarity to find similar terms for verifying the names in date-of-birth facts for their tera-scale bootstrapping system.

2.3 Selecting seeds

For the majority of bootstrapping tasks, there is little or no guidance on how to select seeds which will generate the most accurate lexicons. Most previous works used seeds selected based on a user's or domain expert's intuition (Curran et al., 2007), which may then have to meet a frequency criterion (Riloff et al., 2003).

Eisner and Karakos (2005) focus on this issue by considering an approach called *strapping* for word sense disambiguation. In strapping, semi-supervised bootstrapping instances are used to train a meta-classifier, which given a bootstrapping instance can predict the usefulness (*fertility*) of its seeds. The most fertile seeds can then be used in place of hand-picked seeds.

The design of a strapping algorithm is more complex than that of a supervised learner (Eisner and Karakos, 2005), and it is unclear how well strapping will generalise to other bootstrapping tasks. In our work, we build upon bootstrapping using unsupervised approaches.

3 Experimental setup

In our experiments we consider the task of extracting biomedical semantic lexicons from raw text using BASILISK and WMEB.

3.1 Data

We compared the performance of BASILISK and WMEB using 5-grams (t_1, t_2, t_3, t_4, t_5) from raw MEDLINE abstracts[3]. In our experiments, the candidate terms are the middle tokens (t_3), and the patterns are a tuple of the surrounding tokens (t_1, t_2, t_4, t_5). Unlike Riloff and Jones (1999) and Yangarber (2003), we do not use syntactic knowledge, as we aim to take a language independent approach.

CAT	DESCRIPTION
ANTI	Antibodies: Immunoglobulin molecules that react with a specific antigen that induced its synthesis *MAb IgG IgM rituximab infliximab* (κ_1:0.89, κ_2:1.0)
CELL	Cells: A morphological or functional form of a cell *RBC HUVEC BAEC VSMC SMC* (κ_1:0.91, κ_2:1.0)
CLNE	Cell lines: A population of cells that are totally derived from a single common ancestor cell *PC12 CHO HeLa Jurkat COS* (κ_1:0.93, κ_2: 1.0)
DISE	Diseases: A definite pathological process that affects humans, animals and or plants *asthma hepatitis tuberculosis HIV malaria* (κ_1:0.98, κ_2:1.0)
DRUG	Drugs: A pharmaceutical preparation *acetylcholine carbachol heparin penicillin tetracyclin* (κ_1:0.86, κ_2:0.99)
FUNC	Molecular functions and processes *kinase ligase acetyltransferase helicase binding* (κ_1:0.87, κ_2:0.99)
MUTN	Mutations: Gene and protein mutations, and mutants *Leiden C677T C282Y 35delG null* (κ_1:0.89, κ_2:1.0)
PROT	Proteins and genes *p53 actin collagen albumin IL-6* (κ_1:0.99, κ_2:1.0)
SIGN	Signs and symptoms of diseases *anemia hypertension hyperglycemia fever cough* (κ_1:0.96, κ_2:0.99)
TUMR	Tumors: Types of tumors *lymphoma sarcoma melanoma neuroblastoma osteosarcoma* (κ_1:0.89, κ_2:0.95)

Table 2: The MEDLINE semantic categories.

The 5-grams were extracted from the MEDLINE abstracts following McIntosh and Curran (2008). The abstracts were tokenised and split into sentences using bio-specific NLP tools (Grover et al., 2006). The 5-grams were filtered to remove patterns appearing with less than 7 terms[4]. The statistics of the resulting dataset are shown in Table 1.

3.2 Semantic Categories

The semantic categories we extract from MEDLINE are shown in Table 2. These are a subset of the TREC Genomics 2007 entities (Hersh et al., 2007). Categories which are predominately multi-term entities, e.g. *Pathways* and *Toxicities*, were excluded.[5] *Genes* and *Proteins* were merged into PROT as they have a high degree of metonymy, particularly out of context. The *Cell or Tissue Type* category was split into two fine grained classes, CELL and CLNE (*cell line*).

[3] The set contains all MEDLINE abstracts available up to Oct 2007 (16 140 000 abstracts).

[4] This frequency was selected as it resulted in the largest number of patterns and terms loadable by BASILISK

[5] Note that polysemous terms in these categories may be correctly extracted by another category. For example, all *Pathways* also belong to FUNC.

The five hand-picked seeds used for each category are shown in italics in Table 2. These were carefully chosen based on the evaluators' intuition, and are as unambiguous as possible with respect to the other categories.

We also utilised terms in *stop categories* which are known to cause semantic drift in specific classes. These extra categories bound the lexical space and reduce ambiguity (Yangarber, 2003; Curran et al., 2007). We used four stop categories introduced in McIntosh and Curran (2008): AMINO ACID, ANIMAL, BODY and ORGANISM.

3.3 Lexicon evaluation

The evaluation involves manually inspecting each extracted term and judging whether it was a member of the semantic class. This manual evaluation is extremely time consuming and is necessary due to the limited coverage of biomedical resources. To make later evaluations more efficient, all evaluators' decisions for each category are cached.

Unfamiliar terms were checked using online resources including MEDLINE, Medical Subject Headings (MeSH), Wikipedia. Each ambiguous term was counted as correct if it was classified into one of its correct categories, such as *lymphoma* which is a TUMR and DISE. If a term was unambiguously part of a multi-word term we considered it correct. Abbreviations, acronyms and typographical variations were included. We also considered obvious spelling mistakes to be correct, such as *nuetrophils* instead of *neutrophils* (a type of CELL). Non-specific modifiers are marked as incorrect, for example, *gastrointestinal* may be incorrectly extracted for TUMR, as part of the entity *gastrointestinal carcinoma*. However, the modifier may also be used for DISE (*gastrointestinal infection*) and CELL.

The terms were evaluated by two domain experts. Inter-annotator agreement was measured on the top-100 terms extracted by BASILISK and WMEB with the hand-picked seeds for each category. All disagreements were discussed, and the kappa scores, before (κ_1) and after (κ_2) the discussions, are shown in Table 2. Each score is above 0.8 which reflects an agreement strength of "almost perfect" (Landis and Koch, 1977).

For comparing the accuracy of the systems we evaluated the precision of samples of the lexicons extracted for each category. We report average precision over the 10 semantic categories on the 1-200, 401-600 and 801-1000 term samples, and over the first 1000 terms. In each algorithm, each category is initialised with 5 seed terms, and the number of patterns, k, is set to 5. In each iteration, 5 lexicon terms are extracted by each category. Each algorithm is run for 200 iterations.

4 Seed diversity

The first step in bootstrapping is to select a set of seeds by hand. These *hand-picked* seeds are typically chosen by a domain expert who selects a reasonably unambiguous representative sample of the category with high coverage by introspection.

To improve the seeds, the frequency of the potential seeds in the corpora is often considered, on the assumption that highly frequent seeds are better (Thelen and Riloff, 2002). Unfortunately, these seeds may be too general and extract many non-specific patterns. Another approach is to identify seeds using hyponym patterns like, * is a [NAMED ENTITY] (Meij and Katrenko, 2007).

This leads us to our first investigation of seed variability and the methodology used to compare bootstrapping algorithms. Typically algorithms are compared using one set of hand-picked seeds for each category (Pennacchiotti and Pantel, 2006; McIntosh and Curran, 2008). This approach does not provide a fair comparison or any detailed analysis of the algorithms under investigation. As we shall see, it is possible that the seeds achieve the maximum precision for one algorithm and the minimum for another, and thus the single comparison is inappropriate. Even evaluating on multiple categories does not ensure the robustness of the evaluation. Secondly, it provides no insight into the sensitivity of an algorithm to different seeds.

4.1 Analysis with random gold seeds

Our initial analysis investigated the sensitivity and variability of the lexicons generated using different seeds. We instantiated each algorithm 10 times with different random gold seeds (S_{gold}) for each category. We randomly sample S_{gold} from two sets of correct terms extracted from the evaluation cache. UNION: the correct terms extracted by BASILISK and WMEB; and UNIQUE: the correct terms uniquely identified by only one algorithm. The degree of ambiguity of each seed is unknown and term frequency is not considered during the random selection.

Firstly, we investigated the variability of the

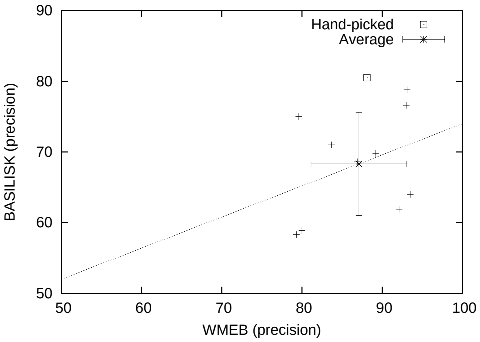

Figure 1: Performance relationship between WMEB and BASILISK on $\mathrm{S_{gold}}$ UNION

extracted lexicons using UNION. Each extracted lexicon was compared with the other 9 lexicons for each category and the term overlap calculated. For the top 100 terms, BASILISK had an overlap of 18% and WMEB 44%. For the top 500 terms, BASILISK had an overlap of 39% and WMEB 47%. Clearly BASILISK is far more sensitive to the choice of seeds – this also makes the cache a lot less valuable for the manual evaluation of BASILISK. These results match our annotators' intuition that BASILISK retrieved far more of the esoteric, rare and misspelt results. The overlap between algorithms was even worse: 6.3% for the top 100 terms and 9.1% for the top 500 terms.

The plot in Figure 1 shows the variation in precision between WMEB and BASILISK with the 10 seed sets from UNION. Precision is measured on the first 100 terms and averaged over the 10 categories. The $\mathrm{S_{hand}}$ is marked with a square, as well as each algorithms' average precision with 1 standard deviation (S.D.) error bars. The axes start at 50% precision. Visually, the scatter is quite obvious and the S.D. quite large. Note that on our $\mathrm{S_{hand}}$ evaluation, BASILISK performed significantly better than average.

We applied a linear regression analysis to identify any correlation between the algorithm's performances. The resulting regression line is shown in Figure 1. The regression analysis identified no correlation between WMEB and BASILISK ($\mathrm{R}^2 = 0.13$). It is almost impossible to predict the performance of an algorithm with a given set of seeds from another's performance, and thus comparisons using only one seed set are unreliable.

Table 3 summarises the results on $\mathrm{S_{gold}}$, including the minimum and maximum averages over the 10 categories. At only 100 terms, lexicon

$\mathrm{S_{gold}}$	$\mathrm{S_{hand}}$	Avg.	Min.	Max.	S.D.
UNION					
BASILISK	80.5	68.3	58.3	78.8	7.31
WMEB	88.1	87.1	79.3	93.5	5.97
UNIQUE					
BASILISK	80.5	67.1	56.7	83.5	9.75
WMEB	88.1	91.6	82.4	95.4	3.71

Table 3: Variation in precision with random gold seed sets

variations are already obvious. As noted above, $\mathrm{S_{hand}}$ on BASILISK performed better than average, whereas WMEB $\mathrm{S_{gold}}$ UNIQUE performed significantly better on average than $\mathrm{S_{hand}}$. This clearly indicates the difficulty of picking the best seeds for an algorithm, and that comparing algorithms with only one set has the potential to penalise an algorithm. These results do show that WMEB is significantly better than BASILISK.

In the UNIQUE experiments, we hypothesized that each algorithm would perform well on its own set, but BASILISK performs significantly worse than WMEB, with a S.D. greater than 9.7. BASILISK's poor performance may be a direct result of it preferring low frequency terms, which are unlikely to be good seeds.

These experiments have identified previously unreported performance variations of these systems and their sensitivity to different seeds. The standard evaluation paradigm, using one set of hand-picked seeds over a few categories, does not provide a robust and informative basis for comparing bootstrapping algorithms.

5 Supervised Bagging

While the wide variation we reported in the previous section is an impediment to reliable evaluation, it presents an opportunity to improve the performance of bootstrapping algorithms. In the next section, we present a novel unsupervised bagging approach to reducing semantic drift. In this section, we consider the standard bagging approach introduced by Breiman (1996). Bagging was used by Ng and Cardie (2003) to create committees of classifiers for labelling unseen data for retraining.

Here, a bootstrapping algorithm is instantiated $n = 50$ times with random seed sets selected from the UNION evaluation cache. This generates n new lexicons $L_1, L_2, \ldots, L_n$ for each category. The next phase involves aggregating the predictions in L_{1-n} to form the final lexicon for each category, using a weighted voting function.

	1-200	401-600	801-1000	1-1000
$\mathrm{S_{hand}}$				
BASILISK	76.3	67.8	58.3	66.7
WMEB	90.3	82.3	62.0	78.6
$\mathrm{S_{gold}}$ BAG				
BASILISK	84.2	80.2	58.2	78.2
WMEB	95.1	79.7	65.0	78.6

Table 4: Bagging with 50 gold seed sets

Our weighting function is based on two related hypotheses of terms in highly accurate lexicons: 1) the more category lexicons in L_{1-n} a term appears in, the more likely the term is a member of the category; 2) terms ranked higher in lexicons are more reliable category members. Firstly, we rank the aggregated terms by the number of lexicons they appear in, and to break ties, we take the term that was extracted in the earliest iteration across the lexicons.

5.1 Supervised results

Table 4 compares the average precisions of the lexicons for BASILISK and WMEB using just the hand-picked seeds ($\mathrm{S_{hand}}$) and 50 sample supervised bagging ($\mathrm{S_{gold}}$ BAG).

Bagging with samples from $\mathrm{S_{gold}}$ successfully increased the performance of both BASILISK and WMEB in the top 200 terms. While the improvement continued for BASILISK in later sections, it had a more variable effect for WMEB. Overall, BASILISK gets the greater improvement in performance (a 12% gain), almost reaching the performance of WMEB across the top 1000 terms, while WMEB's performance is the same for both $\mathrm{S_{hand}}$ and $\mathrm{S_{gold}}$ BAG. We believe the greater variability in BASILISK meant it benefited from bagging with gold seeds.

6 Unsupervised bagging

A significant problem for supervised bagging approaches is that they require a larger set of gold-standard seed terms to sample from – either an existing gazetteer or a large hand-picked set. In our case, we used the evaluation cache which took considerable time to accumulate. This saddles the major application of bootstrapping, the quick construction of accurate semantic lexicons, with a chicken-and-egg problem.

However, we propose a novel solution – sampling from the terms extracted with the hand-picked seeds ($\mathrm{L_{hand}}$). WMEB already has very high precision for the top extracted terms (88.1% for the top 100 terms) and may provide an acceptable source of seed terms. This approach now only requires the original 50 hand-picked seed terms across the 10 categories, rather than the 2100 terms used above. The process now uses two rounds of bootstrapping: first to create $\mathrm{L_{hand}}$ to sample from and then another round with the 50 sets of randomly unsupervised seeds, $\mathrm{S_{rand}}$.

BAGGING	1-200	401-600	801-1000	1-1000
Top-100				
BASILISK	72.3	63.5	58.8	65.1
WMEB	90.2	78.5	66.3	78.5
Top-200				
BASILISK	70.7	60.7	45.5	59.8
WMEB	91.0	78.4	62.2	77.0
Top-500				
BASILISK	63.5	60.5	45.4	56.3
WMEB	92.5	80.9	59.1	77.2
PDF-500				
BASILISK	69.6	68.3	49.6	62.3
WMEB	92.9	80.7	72.1	81.0

Table 5: Bagging with 50 unsupervised seed sets

The next decision is how to sample $\mathrm{S_{rand}}$ from $\mathrm{L_{hand}}$. One approach is to use uniform random sampling from restricted sections of $\mathrm{L_{hand}}$. We performed random sampling from the top 100, 200 and 500 terms of $\mathrm{L_{hand}}$. The seeds from the smaller samples will have higher precision, but less diversity.

In a truly unsupervised approach, it is impossible to know if and when semantic drift occurs and thus using arbitrary cut-offs can reduce the diversity of the selected seeds. To increase diversity we also sampled from the top n=500 using a probability density function (PDF) using rejection sampling, where r is the rank of the term in $\mathrm{L_{hand}}$:

$$\mathrm{PDF}(r) \quad = \quad \frac{\sum_{i=r}^{n} i^{-1}}{\sum_{i=1}^{n}\sum_{j=i}^{n} j^{-1}} \qquad (1)$$

6.1 Unsupervised results

Table 5 shows the average precision of the lexicons after bagging on the unsupervised seeds, sampled from the top 100 – 500 terms from $\mathrm{L_{hand}}$. Using the top 100 seed sample is much less effective than $\mathrm{S_{gold}}$ BAG for BASILISK but nearly as effective for WMEB. As the sample size increases, WMEB steadily improves with the increasing variability, however BASILISK is more effective when the more precise seeds are sampled from higher ranking terms in the lexicons.

Sampling with PDF-500 results in more accurate lexicons over the first 1000 terms than the other

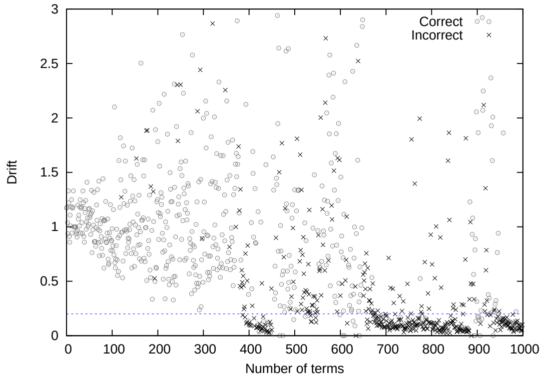

Figure 2: Semantic drift in CELL (n=20, m=20)

sampling methods for WMEB. In particular, WMEB is more accurate with the unsupervised seeds than the $\mathrm{S_{gold}}$ and $\mathrm{S_{hand}}$ (81.0% vs 78.6% and 78.6%). WMEB benefits from the larger variability introduced by the more diverse sets of seeds, and the greater variability available out-weighs the potential noise from incorrect seeds. The PDF-500 distribution allows some variability whilst still preferring the most reliable unsupervised seeds. In the critical later iterations, WMEB PDF-500 improves over supervised bagging ($\mathrm{S_{gold}}$ BAG) by 7% and the original hand-picked seeds ($\mathrm{S_{hand}}$) by 10%.

7 Detecting semantic drift

As shown above, semantic drift still dominates the later iterations of bootstrapping even after bagging. In this section, we propose distributional similarity measurements over the extracted lexicon to detect semantic drift during the bootstrapping process. Our hypothesis is that semantic drift has occurred when a candidate term is more similar to recently added terms than to the seed and high precision terms added in the earlier iterations. We experiment with a range of values of both.

Given a growing lexicon of size N, L_N, let $L_{1...n}$ correspond to the first n terms extracted into L, and $L_{(N-m)...N}$ correspond to the last m terms added to L_N. In an iteration, let t be the next candidate term to be added to the lexicon.

We calculate the average distributional similarity (sim) of t with all terms in $L_{1...n}$ and those in $L_{(N-m)...N}$ and call the ratio the *drift* for term t:

$$\mathrm{drift}(t,n,m) \quad = \quad \frac{\mathrm{sim}(L_{1...n},t)}{\mathrm{sim}(L_{(N-m)...N},t)} \qquad (2)$$

Smaller values of $\mathrm{drift}(t,n,m)$ correspond to the current term moving further away from the first terms. A $\mathrm{drift}(t,n,m)$ of 0.2 corresponds to a 20% difference in average similarity between $L_{1...n}$ and $L_{(N-m)...N}$ for term t.

Drift can be used as a post-processing step to filter terms that are a possible consequence of drift. However, our main proposal is to incorporate the drift measure directly within the WMEB bootstrapping algorithm, to detect and then prevent drift occuring. In each iteration, the set of candidate terms to be added to the lexicon are scored and ranked for their suitability. We now additionally determine the drift of each candidate term before it is added to the lexicon. If the term's drift is below a specified threshold, it is discarded from the extraction process. If the term has zero similarity with the last m terms, but is similar to at least one of the first n terms, the term is selected. Preventing the drifted term from entering the lexicon during the bootstrapping process, has a flow on effect as it will not be able to extract additional divergent patterns which would lead to accelerated drift.

For calculating drift we use the distributional similarity approach described in Curran (2004). We extracted window-based features from the filtered 5-grams to form context vectors for each term. We used the standard t-test weight and weighted Jaccard measure functions (Curran, 2004). This system produces a distributional score for each pair of terms presented by the bootstrapping system.

7.1 Drift detection results

To evaluate our semantic drift detection we incorporate our process in WMEB. Candidate terms are still weighted in WMEB using the χ^2 statistic as described in (McIntosh and Curran, 2008). Many of the MEDLINE categories suffer from semantic drift in WMEB in the later stages. Figure 2 shows the distribution of correct and incorrect terms appearing in the CELL lexicon extracted using $\mathrm{S_{hand}}$ with the term's ranks plotted against their drift scores. Firstly, it is evident that incorrect terms begin to dominate in later iterations. Encouragingly, there is a trend where low values of drift correspond to incorrect terms being added. Drift also occurs in ANTI and MUTN, with an average precision at 801-1000 terms of 41.5% and 33.0% respectively.

We utilise drift in two ways with WMEB; as a post-processing filter (WMEB+POST) and internally during the term selection phase (WMEB+DIST). Table 6 shows the performance

	1-200	401-600	801-1000	1000
WMEB	90.3	82.3	62.0	78.6
WMEB+POST				
n:20 m:5	90.3	82.3	62.1	78.6
n:20 m:20	90.3	81.5	62.0	76.9
n:100 m:5	90.2	82.3	62.1	78.6
n:100 m:20	90.3	82.1	62.1	78.1
WMEB+DIST				
n:20 m:5	90.8	79.7	72.1	80.2
n:20 m:20	90.6	80.1	76.3	81.4
n:100 m:5	90.5	82.0	79.3	82.8
n:100 m:20	90.5	81.5	77.5	81.9

Table 6: Semantic drift detection results

of drift detection with WMEB, using $\mathrm{S}_{\mathrm{hand}}$. We use a drift threshold of 0.2 which was selected empirically. A higher value substantially reduced the lexicons' size, while a lower value resulted in little improvements. We experimented with various sizes of initial terms $L_{1...n}$ (n=20, n=100) and $L_{(N-m)...N}$ (m=5, m=20).

There is little performance variation observed in the various WMEB+POST experiments. Overall, WMEB+POST was outperformed slightly by WMEB. The post-filtering removed many incorrect terms, but did not address the underlying drift problem. This only allowed additional incorrect terms to enter the top 1000, resulting in no appreciable difference.

Slight variations in precision are obtained using WMEB+DIST in the first 600 terms, but noticeable gains are achieved in the 801-1000 range. This is not surprising as drift in many categories does not start until later (cf. Figure 2).

With respect to the drift parameters n and m, we found values of n below 20 to be inadequate. We experimented initially with n=5 terms, but this is equivalent to comparing the new candidate terms to the initial seeds. Setting m to 5 was also less useful than a larger sample, unless n was also large. The best performance gain of 4.2% overall for 1000 terms and 17.3% at 801-1000 terms was obtained using n=100 and m=5. In different phases of WMEB+DIST we reduce semantic drift significantly. In particular, at 801-1000, ANTI increase by 46% to 87.5% and MUTN by 59% to 92.0%.

For our final experiments, we report the performance of our best performing WMEB+DIST system (n=100 m=5) using the 10 random GOLD seed sets from section 4.1, in Table 7. On average WMEB+DIST performs above WMEB, especially in the later iterations where the difference is 6.3%.

	S_{hand}	Avg.	Min.	Max.	S.D.
1-200					
WMEB	90.3	82.2	73.3	91.5	6.43
WMEB+DIST	90.7	84.8	78.0	91.0	4.61
401-600					
WMEB	82.3	66.8	61.4	74.5	4.67
WMEB+DIST	82.0	73.1	65.2	79.3	4.52

Table 7: Final accuracy with drift detection

8 Conclusion

In this paper, we have proposed unsupervised bagging and integrated distributional similarity to minimise the problem of semantic drift in iterative bootstrapping algorithms, particularly when extracting large semantic lexicons.

There are a number of avenues that require further examination. Firstly, we would like to take our two-round unsupervised bagging further by performing another iteration of sampling and then bootstrapping, to see if we can get a further improvement. Secondly, we also intend to experiment with machine learning methods for identifying the correct cutoff for the drift score. Finally, we intend to combine the bagging and distributional approaches to further improve the lexicons.

Our initial analysis demonstrated that the output and accuracy of bootstrapping systems can be very sensitive to the choice of seed terms and therefore robust evaluation requires results averaged across randomised seed sets. We exploited this variability to create both supervised and unsupervised bagging algorithms. The latter requires no more seeds than the original algorithm but performs significantly better and more reliably in later iterations. Finally, we incorporated distributional similarity measurements directly into WMEB which detect and censor terms which could lead to semantic drift. This approach significantly outperformed standard WMEB, with a 17.3% improvement over the last 200 terms extracted (801-1000). The result is an efficient, reliable and accurate system for extracting large-scale semantic lexicons.

Acknowledgments

We would like to thank Dr Cassie Thornley, our second evaluator who also helped with the evaluation guidelines; and the anonymous reviewers for their helpful feedback. This work was supported by the CSIRO ICT Centre and the Australian Research Council under Discovery project DP0665973.

References

Leo Breiman. 1996. Bagging predictors. *Machine Learning*, 26(2):123–140.

James R. Curran, Tara Murphy, and Bernhard Scholz. 2007. Minimising semantic drift with mutual exclusion bootstrapping. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 172–180, Melbourne, Australia.

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

Jason Eisner and Damianos Karakos. 2005. Bootstrapping without the boot. In *Proceedings of the Conference on Human Language Technology and Conference on Empirical Methods in Natural Language Processing*, pages 395–402, Vancouver, British Columbia, Canada.

Gregory Grefenstette. 1994. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, USA.

Claire Grover, Michael Matthews, and Richard Tobin. 2006. Tools to address the interdependence between tokenisation and standoff annotation. In *Proceedings of the Multi-dimensional Markup in Natural Language Processing Workshop*, Trento, Italy.

Zellig Harris. 1954. Distributional structure. *Word*, 10(2/3):146–162.

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.

William Hersh, Aaron M. Cohen, Lynn Ruslen, and Phoebe M. Roberts. 2007. TREC 2007 Genomics Track Overview. In *Proceedings of the 16th Text REtrieval Conference*, Gaithersburg, MD, USA.

Mamoru Komachi, Taku Kudo, Masashi Shimbo, and Yuji Matsumoto. 2008. Graph-based analysis of semantic drift in Espresso-like bootstrapping algorithms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1011–1020, Honolulu, USA.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement in categorical data. *Biometrics*, 33(1):159–174.

Tara McIntosh and James R. Curran. 2008. Weighted mutual exclusion bootstrapping for domain independent lexicon and template acquisition. In *Proceedings of the Australasian Language Technology Association Workshop*, pages 97–105, Hobart, Australia.

Edgar Meij and Sophia Katrenko. 2007. Bootstrapping language associated with biomedical entities. The AID group at TREC Genomics 2007. In *Proceedings of The 16th Text REtrieval Conference*, Gaithersburg, MD, USA.

Shachar Mirkin, Ido Dagan, and Maayan Geffet. 2006. Integrating pattern-based and distributional similarity methods for lexical entailment acquistion. In *Proceedings of the 21st International Conference on Computational Linguisitics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 579–586, Sydney, Australia.

Vincent Ng and Claire Cardie. 2003. Weakly supervised natural language learning without redundant views. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 94–101, Edmonton, USA.

Marius Paşca, Dekang Lin, Jeffrey Bigham, Andrei Lifchits, and Alpa Jain. 2006. Names and similarities on the web: Fact extraction in the fast lane. In *Proceedings of the 21st International Conference on Computational Linguisitics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 809–816, Sydney, Australia.

Patrick Pantel and Deepak Ravichandran. 2004. Automatically labelling semantic classes. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 321–328, Boston, MA, USA.

Marco Pennacchiotti and Patrick Pantel. 2006. A bootstrapping algorithm for automatically harvesting semantic relations. In *Proceedings of Inference in Computational Semantics (ICoS-06)*, pages 87–96, Buxton, England.

Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th National Conference on Artificial Intelligence and the 11th Innovative Applications of Artificial Intelligence Conference*, pages 474–479, Orlando, FL, USA.

Ellen Riloff, Janyce Wiebe, and Theresa Wilson. 2003. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 25–32.

Michael Thelen and Ellen Riloff. 2002. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 214–221, Philadelphia, USA.

Xiaofeng Yang and Jian Su. 2007. Coreference resolution using semantic relatedness information from automatically discovered patterns. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 528–535, Prague, Czech Republic.

Roman Yangarber. 2003. Counter-training in discovery of semantic patterns. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 343–350, Sapporo, Japan.

Hong Yu and Eugene Agichtein. 2003. Extracting synonymous gene and protein terms from biological literature. *Bioinformatics*, 19(1):i340–i349.