Weighted Mutual Exclusion Bootstrapping for Domain Independent Lexicon and Template Acquisition

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

Abstract

We present the *Weighted Mutual Exclusion Bootstrapping* (WMEB) algorithm for simultaneously extracting precise semantic lexicons and templates for multiple categories. WMEB is capable of extracting larger lexicons with higher precision than previous techniques, successfully reducing semantic drift by incorporating new weighting functions and a cumulative template pool while still enforcing mutual exclusion between the categories.

We compare WMEB and two state-of-the-art approaches on the Web 1T corpus and two large biomedical literature collections. WMEB is more efficient and scalable, and we demonstrate that it significantly outperforms the other approaches on the noisy web corpus and biomedical text.

1 Introduction

Automatically acquiring semantic lexicons and templates from raw text is essential for overcoming the knowledge bottleneck in many natural language processing tasks, e.g. question answering (Ravichandran and Hovy, 2002). These tasks typically involve identifying named entity (NE) classes which are not found in annotated corpora and thus supervised NE recognition models are not always available. This issue becomes even more evident in new domains, such as biomedicine, where new semantic categories are often poorly represented in linguistic resources, if at all (Hersh et al., 2007).

There are two common approaches to extract semantic lexicons: distributional similarity and template-based bootstrapping . In *template-based bootstrapping* algorithms, templates that express a particular semantic type are used to recognise new terms, and in turn these new terms help identify new templates iteratively (Riloff and Jones, 1999). These algorithms are attractive as they are domain and language independent, require minimal linguistic preprocessing, are relatively efficient, and can be applied to raw text.

Unfortunately, *semantic drift* often occurs when ambiguous or erroneous terms or patterns are introduced into the lexicon or set of templates. Curran et al. (2007) developed Mutual Exclusion Bootstrapping (MEB) to reduce semantic drift by forcing semantic classes to be mutually exclusive.

We introduce a new algorithm, Weighted Mutual Exclusion Bootstrapping (WMEB), that automatically acquires multiple semantic lexicons and their templates simultaneously. It extends on the Curran et al. (2007) assumption of mutual exclusion between categories by incorporating a novel cumulative template pool and new term and template weighting functions.

We compare WMEB against two state-of-the-art mutual bootstrapping algorithms, MEB (Curran et al., 2007) and BASILISK (Thelen and Riloff, 2002). We have evaluated the terms and templates these algorithms extract under a range of conditions from three raw text collections: noisy web text, biomedical abstracts, and full-text articles.

We demonstrate that WMEB outperforms these existing algorithms in extracting precise lexicons and templates from all three datasets. WMEB is significantly less susceptible to semantic drift and so can produce large lexicons accurately and efficiently across multiple domains.

2 Background

Hearst (1992) pioneered the use of templates for information extraction, focussing on acquiring *is-a* relations using manually devised templates like *such W as X, ..., Y and/or Z* where *X, ..., Y, Z* are hyponyms of *W*. Various automated template-based bootstrapping algorithms have since been developed to iteratively build semantic lexicons from texts. Riloff and Shepherd (1997) proposed *Iterative Bootstrapping* (IB) where seed instances of a semantic category are used to identify related terms that frequently co-occur.

In *Mutual Bootstrapping* (MB) (Riloff and Jones, 1999) seed instances of a desired type are used to infer new templates, which in turn identify new lexicon entries. This process is repeated with the new terms identifying new templates. In each iteration, new terms and templates are selected based on a metric scoring their suitability for extracting additional templates and terms for the category. Unfortunately, if a term with multiple senses or a template which weakly constrains the semantic class is selected, *semantic drift* of the lexicon and templates occurs – the semantic class drifts into another category (Curran et al., 2007).

Extracting multiple semantic categories simultaneously has been proposed to reduce semantic drift. The bootstrapping instances compete with one another in an attempt to actively direct the categories away from each other (Thelen and Riloff, 2002; Yangarber et al., 2002; Curran et al., 2007). This strategy is similar to the one sense per discourse assumption (Yarowsky, 1995).

In BASILISK (Thelen and Riloff, 2002), candidate terms for a category are ranked highly if they have strong evidence for the category and little or no evidence for another. It is possible for an ambiguous term to be assigned to the less dominant sense, and in turn less precise templates will be selected, causing semantic drift. Drift may also be introduced as templates can be selected by different categories in different iterations.

NOMEN (Yangarber et al., 2002) was developed to extract generalized names such as diseases and drugs, with no capitalisation cues. NOMEN, like BASILISK, identifies semantic category lexicons in parallel, however NOMEN extracts the left and right contexts of terms independently and generalises the contexts.

Curran et al. (2007) introduced the algorithm *Mutual Exclusion Bootstrapping* (MEB) which more actively defines the semantic boundaries of the lexicons extracted simultaneously. In MEB, the categories compete for both terms and templates. Semantic drift is reduced in two ways: by eliminating templates that collide with two or more categories in an iteration (from all subsequent iterations), and by ignoring colliding candidate terms (for an iteration). This effectively excludes general templates that can occur frequently with multiple categories, and reduces the chance of assigning ambiguous terms to their less dominant sense.

The scoring metric for candidate terms and templates in MEB is simple and naïve. Terms and templates 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 criteria aims to increase recall, however the selected instances are highly likely to introduce drift. We introduce a new weighting scheme to effectively overcome this.

Template-based bootstrapping algorithms have also been used in various Information Extraction (IE) tasks. Agichtein and Gravano (2000) developed the SNOWBALL system to identify the locations of companies, and Yu and Agichtein (2003) applied SNOWBALL to extract synonymous gene and protein terms. Pantel and Pennacchiotti (2006) used bootstrapping to identify numerous semantic relationships, such as *is-a* and *part-of* relationships. They incorporate the *pointwise mutual information* (MI) measure between the templates and instances to determine template reliability, as well as exploiting generic templates and the Web for filtering incorrect instances. We evaluate the effectiveness of MI as a weighting function for selecting terms and templates in WMEB.

In the biomedical domain, there is an increased interest in automatically extracting lexicons of biomedical entities such as *antibodies* and *mutations*, and the templates which extract such terms. This is primarily due to the lack, and scope, of annotated resources, and the introduction of new semantic categories which are severely underrepresented in corpora and lexicons. Meij and Ka-

trenko (2007) applied MB to identify biomedical entities and their templates, which were both then used to find potential answer sentences for the TREC Genomics Track task (Hersh et al., 2007). The accuracy of their extraction process was not evaluated, however their Information Retrieval system had performance gains in unambiguous and common entity types, where little semantic drift is likely to occur.

3 Weighted MEB (WMEB) Algorithm

Our algorithm, *Weighted Mutual Exclusion Bootstrapping* (WMEB), extends MEB described in Curran et al. (2007). MEB is a minimally supervised, mutual bootstrapping algorithm which reduces semantic drift by extracting multiple semantic categories with individual bootstrapping instances in parallel, and by forcing the categories to be mutually exclusive (Figure 1). In MEB, the templates describe the context of a term (two terms to the left and right). Each MEB instance iterates simultaneously between two stages: template extraction and selection, and term extraction and selection. The key assumption of MEB is that terms only have a single sense and that templates only extract terms of a single sense. This is forced by excluding terms and templates from all categories if in one iteration they are selected by more than one category.

In this section we describe the architecture of WMEB. WMEB employs a new weighting scheme, which identifies candidate templates and terms that are strongly associated with the lexicon terms and their templates respectively. In WMEB, we also introduce the concept of a cumulative template pool. These techniques reduce the semantic drift in WMEB more effectively than in MEB.

3.1 System Architecture

WMEB takes as input a set of manually labelled seed terms for each category. Each category's seed set forms it's initial lexicon.

Template Extraction and Selection

For each term in the category lexicon, WMEB extracts all candidate templates the term matches. To enforce mutually exclusive templates, candidate templates identified by multiple categories are excluded from the candidate set and all subsequent iterations. The remaining candidates are then ranked according to their *reliability* measure and their *relevance weight* (see Section 3.2).

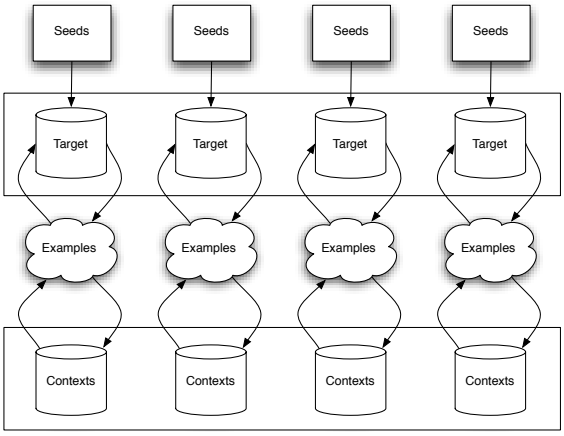

Figure 1: MEB architecture.

After manually inspecting the templates selected by MEB and BASILISK, we introduced the *cumulative template pool* (pool) in WMEB. In MEB (Curran et al., 2007) and BASILISK (Thelen and Riloff, 2002), the top-k[1] templates for each iteration are used to extract new candidate terms. We observed that as the lexicons grow, more general templates can drift into the top-k. This was also noted by Jones et al. (1999). As a result the earlier precise templates lose their influence.

WMEB successfully overcomes this by accumulating all selected templates from the current and all previous iterations in the pool, ensuring previous templates can contribute. The templates in the pool have equal weight in all iterations.

In WMEB, the top-k templates are selected for addition to the pool. If all top-k templates are already in the pool, then the next available top template is added. This ensures at least one new template is added in each iteration.

Term Extraction and Selection

For each template in a category's pool, all available candidate terms matching the templates are identified. Like the candidate templates, terms which are extracted by multiple categories are also excluded. A colliding term will collide in all consecutive iterations due to the cumulative pool and thus WMEB creates a stricter term boundary between categories than MEB. The candidate

[1] BASILISK also adds an additional template in each iteration, i.e. k is increased by one in each iteration.

terms are ranked with respect to their *reliability* and *relevance weight*, and the top-k terms are added to the category's lexicon.

3.2 Term and Template Weighting

In MEB, candidate terms and templates are ranked according to their *reliability* measure and ties are broken using the *productivity* measure. The *reliability* of a term for a given category, is the number of input templates in an iteration that can extract the term. The *productivity* of a term is the number of potentially new templates it may add in the next iteration. These measures are symmetrical for both terms and templates. More reliable instances would theoretically have higher precision, while high productive instances will have a high recall. Unfortunately, high productive instances could potentially introduce drift.

In WMEB we replace the productivity measure with a new *relevance weight*. We have investigated scoring metrics which prefer terms and templates that are highly associated with their input instances, including: the chi-squared (χ^2) statistic and three variations of the pointwise mutual information (MI) measure (Manning and Schutze, 1999, Chapter 5). Each of these estimates the strength of the co-occurance of a term and a template. They do not give the likelihood of the instance being a member of a semantic category.

The first variation of MI we investigate is MI^2, which scales the probability of the term (t) and template (c) pair to ensure more frequent combinations have a greater weight.

$$\text{MI}^2(t,c) = \log_2 \frac{p(t,c)^2}{p(t)p(c)}$$

Each of the probabilities are calculated directly from the relative frequencies without smoothing. The scores are set to 0 if their observed frequencies are less than 5, as these estimates are sensitive to low frequencies.

The other variation of MI function we utilise is truncated MI (MIT), and is defined as:

$$\text{MIT}(t,c) = \begin{cases} \text{MI}^2(t,c) & : \quad \text{MI}(t,c) > 0 \\ 0 & : \quad \text{MI}(t,c) \leq 0 \end{cases}$$

The overall *relevance weight* for a term or template is the sum of the scores of the pairs, where score corresponds to one of the scoring metrics, and C is the set of templates matching term t, and T is the set of terms matching template c.

$$\text{weight}(t) = \sum_{c \in C} \text{score}(t,c)$$

$$\text{weight}(c) = \sum_{t \in T} \text{score}(c,t)$$

The terms and templates are ordered by their *reliability*, and ties are broken by their *relevance weight*. WMEB is much more efficient than BASILISK using these weighting scores – for all possible term and template pairs, the scores can be pre-calculated when the data is loaded, whereas in BASILISK, the scoring metric is more computationally expensive. In BASILISK, each individual calculation is dependent on the current state of the bootstrapping process, and therefore scores cannot be pre-calculated.

TYPE (#)	MEDLINE	TREC	Web1T
Terms	1 347 002	1 478 119	568 202
Templates	4 090 412	8 720 839	10 568 219
5-grams	72 796 760	63 425 523	42 704 392
Orig. tokens	6 642 802 776	3 479 412 905	$\sim$ 1 trillion

Table 1: Filtered 5-gram dataset statistics.

4 Experimental Setting

4.1 Data

We evaluated the performance of BASILISK, MEB and WMEB using 5-grams from three raw text resources: the Google Web 1T corpus (Brants and Franz, 2006), MEDLINE abstracts[2] and the TREC Genomics Track 2007 full-text articles (Hersh et al., 2007). In our experiments, the *term* is the middle token of each 5-gram and the *template* is the two tokens on either side. Unlike Riloff and Jones (1999) and Yangarber (2003), we do not use syntactic knowledge. Although we only extract unigrams, each algorithm can identify multi-term entities (Murphy and Curran, 2007).

The Web 1T 5-grams were filtered by removing templates appearing with only one term and templates containing numbers. All 5-gram contexts with a non-titlecase term were also filtered as we are extracting proper nouns.

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

CAT	DESCRIPTION
FEM	Person: female first name
	Mary Patricia Linda Barbara Elizabeth
MALE	Person: male first name
	James John Robert Michael William
LAST	Person: last name
	Smith Johnson Williams Jones Brown
TTL	Honorific title
	General President Director King Doctor
NORP	Nationality, Religion, Political (adjectival)
	American European French British Western
FOG	Facilities and Organisations
	Ford Microsoft Sony Disneyland Google
PLCE	Place: Geo-political entities and locations
	Africa America Washington London Pacific
DAT	Reference to a date or period
	January May December October June
LANG	Any named language
	English Chinese Japanese Spanish Russian

Table 2: Web 1T semantic categories and seeds.

Limited preprocessing was required to extract the 5-grams from MEDLINE and TREC. The TREC documents were converted from HTML to raw text, and both collections were tokenised using bio-specific NLP tools (Grover et al., 2006). We did not exclude lowercase terms or templates containing numbers. Templates appearing with less than 7 (MEDLINE) or 3 (TREC) terms were removed. These frequencies were selected to permit the largest number of templates and terms loadable by BASILISK[3], to allow a fair comparison.

The size of the resulting datasets are shown in Table 1. Note that, Web 1T has far fewer terms but many more templates than the biomedical sets, and TREC articles result in more templates than MEDLINE for a similar number of terms.

4.2 Semantic Categories & Stop Categories

In the Web 1T experiments, we are extracting proper-noun NE and their templates. We use the categories from Curran et al. (2007) which are a subset of those in the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein, 2005), and are shown in Table 2.

In the MEDLINE and TREC experiments we considered the TREC Genomics 2007 entities with a few modifications (Hersh et al., 2007). We excluded the categories *Toxicities*, *Pathways* and *Biological Substances*, which are predominately multi-term entities, and the category *Strains* due to the difficulty for biologists to distinguish between strains and organisms. We combined the categories *Genes* and *Proteins* into PROT as there is a very high degree of metonymy between these, particularly once out of context. We were also interested in the fine grain distinction between types of *cells* and *cell lines*, so we split the *Cell or Tissue Type* category into CELL and CLNE entities.

[3] BASILISK requires n times more memory to store the term and template frequency counts than MEB and WMEB, where n is the number of categories.

CAT	DESCRIPTION
ANTI	Antibodies: Immunoglobulin molecules
	MAb IgG IgM rituximab infliximab
CELL	Cells: A morphological or functional form of a cell
	RBC HUVEC BAEC VSMC SMC
CLNE	Cell lines: Cell clones grown in tissue culture
	PC12 CHO HeLa Jurkat COS
DISE	Diseases: pathological process affecting organisms
	asthma hepatitis tuberculosis HIV malaria
DRUG	Drugs: A pharmaceutical preparation
	acetylcholine carbachol heparin penicillin tetracyclin
FUNC	Molecular functions and processes
	kinase ligase acetyltransferase helicase binding
MUTN	Gene and protein mutations, and mutants
	Leiden C677T C28Y L5178Y S100B (MEDLINE)
	T47D S100B K44A F442A G93A (TREC)
PROT	Proteins and genes
	p53 actin collagen albumin IL-6
SIGN	Signs and symptoms of diseases
	anemia hypertension hyperglycemia fever cough
TUMR	Tumor types
	lymphoma sarcoma melanoma neuroblastoma osteosarcoma

Table 3: Biomedical semantic categories and seeds.

Five seed terms (as non-ambiguous as possible) were selected for each category based on the evaluators' knowledge of them and their high frequency counts in the collections, and are shown in Table 2 and 3. Separate MUTN seeds for MEDLINE and TREC were used as some high frequency MUTN terms in MEDLINE do not appear in TREC.

As in Curran et al. (2007) and Yangarber (2003), we used additional *stop categories*, which are extracted as well but then discarded. Stop categories help constrain the categories of interest by creating extra boundaries against semantic drift. For the Web 1T experiments we used the stop categories described in Curran et al. (2007) – ADDRESS, BODY, CHEMICAL, COLOUR, DRINK, FOOD, JEWEL and WEB terms. In the biomedical experiments we introduced four stop categories – AMINO ACID, ANIMAL, BODY and ORGANISM.

4.3 Evaluation

Our evaluation process involved manually inspecting each extracted term and judging whether it was a member of the semantic class. The biomedical terms were evaluated by a domain expert. Unfamiliar terms were checked using online resources including MEDLINE, Medical Subject Headings (MeSH), Wikipedia and Google.

Each ambiguous term was counted as correct if it was classified into one of its correct categories. If a single term was unambiguously part of a multi-word term we considered it correct. Modifiers such as *cultured* in *cultured lymphocytes* and *chronic* in *chronic arthritis* were marked as incorrect.

For comparing the accuracy of the systems we evaluated the precision of the first *n* selected terms for each category. In some experiments we report the average precision over each category (Av(*n*)).

Our evaluation also includes judging the quality of the templates extracted. This is the first empirical evaluation of the templates identified by bootstrapping algorithms. We inspected the first 100 templates extracted for each category, and classified them into three groups. Templates where the context is semantically coherent with terms only from the assigned category are considered to accurately define the category and are classified as *true matches* (TM). Templates where another category's term could also be inserted were designated as *possible matches* (PM). For example, DRUG matches: *pharmacokinetics of* X *in patients* and *mechanism of* X *action* ., however the latter is a PM as it also matches PROT. Templates like *compared with* X *for the* and *Value of* X *in the* are *false matches* as they do not provide any contextual information for a specific entity.

5 Results

All of our experiments use the stop categories mentioned in §4.3, unless otherwise stated. The maximum number of terms and templates (top-k) which can be added in each iteration is 5.

Our first experiment investigates the weighting functions for WMEB on the Web 1T and MEDLINE data (Table 4). For the Web 1T data, all of the measures are approximately equal at 400 extracted terms. On MEDLINE, χ^2 outperforms the other measures and is more consistent. In Table 4 we also see the first difference between the two domains. The MEDLINE data scores are significantly higher, with little semantic drift down to 400 terms. For the remaining WMEB experiments we use the χ^2 weighting function.

	Web 1T				MEDLINE			
	100	200	300	400	100	200	300	400
χ^2	78.5	74.2	69.1	65.2	87.1	89.5	89.0	89.4
MI	76.0	70.7	67.4	65.0	84.7	87.0	86.8	87.2
MI^2	80.7	72.7	68.1	64.7	84.3	82.8	82.3	80.7
MIT	79.3	74.4	69.1	65.5	86.1	84.4	84.3	83.8

Table 4: Results comparing WMEB scoring functions.

5.1 Terms

Table 5 and 6 summarise the comparison of BASILISK, MEB and WMEB, on the Web 1T and MEDLINE data respectively. The category analysis is measured on the top 100 terms. BASILISK outperforms MEB on both datasets, whereas WMEB performs similarly to BASILISK on the Web 1T data. For the MEDLINE data, WMEB outperforms both BASILISK and MEB.

Each algorithm will stop extracting terms in a category if all candidate terms are exhausted. Templates may also become exhausted. Thus, each algorithm may be penalised when we evaluate past their stopping points. We have provided adjusted scores in brackets to take this into account. After adjustment, WMEB and BASILISK significantly outperform MEB, and WMEB is more accurate than BASILISK.

It is clear that some categories are much easier than others to extract, e.g. LAST and CELL, while others are quite difficult, e.g. NORP and FUNC. For many categories there is a wide variation across the algorithms' performance, e.g. NORP and TUMR.

The stop categories' seeds were optimised for MEB. When the stop categories are introduced BASILISK gains very little compared to MEB and WMEB. In BASILISK-NOSTOP categories rarely drift into unspecified categories, however they do drift into similar semantic categories of interest, e.g. TUMR drifts into CLNE and vice-versa, in early iterations. This is because BASILISK weakly defines the semantic boundaries. In WMEB, this rarely occurs as the boundaries are strict. We find DISE drifts into BODY and thus a significant per-

CAT	NO STOP			STOP		
	BAS	MEB	WMEB	BAS	MEB	WMEB
FEM	98	100	100	99	100	100
MALE	97	100	100	97	100	98
LAST	100	100	100	100	100	100
TTL	39	9	46	37	31	53
NORP	67	17	41	64	22	40
FOG	90	98	98	90	95	98
PLCE	98	100	94	96	100	98
DATE	35	47	24	38	57	23
LANG	97	90	96	95	93	97
Av(100)	**80.2**	73.4	77.7	**79.5**	77.6	78.6
	(84.7)		(**86.1**)	(84.3)		(**87.1**)
Av(200)	73.5	68.7	**71.0**	72.2	73.2	**74.2**
	(**84.5**)		(80.8)	(82.4)		(**85.4**)

Table 5: Results comparing Web 1T terms.

CAT	NO STOP			STOP		
	BAS	MEB	WMEB	BAS	MEB	WMEB
ANTI	47	95	98	49	92	96
CELL	95	95	98	95	98	100
CLNE	91	93	96	81	100	100
DISE	77	33	49	82	39	76
DRUG	67	77	92	69	92	100
FUNC	73	60	71	73	61	81
MUTN	88	87	87	88	63	81
PROT	99	99	100	99	100	99
SIGN	96	55	67	97	95	99
TUMR	51	33	39	51	23	39
Av(100)	78.5	72.7	**79.7**	78.4	76.3	**87.1**
Av(200)	75.0	66.1	78.6	74.7	70.1	89.5
Av(300)	72.3	60.2	78.3	72.1	64.8	89.0
Av(400)	70.0	56.3	**77.4**	71.0	60.8	**89.4**

Table 6: Results comparing MEDLINE terms.

formance gain is achieved with the BODY stop category.

The remainder of the analysis will be performed on the biomedical data. In Table 7, we can observe the degree of drift which occurs in each algorithm on MEDLINE for a given number of extracted terms. For example, the 101–200 row gives the accuracy of the 101–200th extracted terms. WMEB performs fairly consistently in each range, whereas MEB degrades quickly and BASILISK to a lesser extent. The terms extracted by WMEB in later stages are more accurate than the first 100 extracted terms identified by BASILISK and MEB.

In practice, it is unlikely we would only have 5 seed terms. Thus, we investigated the impact of using 100 seeds as input for each algorithm (Table 8). Only BASILISK improved with these large

RANGE	BAS	MEB	WMEB
0–100	78.4	76.3	87.1
101–200	71.0	63.8	91.8
201–300	66.9	54.3	88.0
301–400	67.6	48.7	90.7
401–500	69.5	49.7	83.5

Table 7: Drift results on MEDLINE.

AV(N)	BAS	MEB	WMEB
50	82.4	62.4	72.0
100	83.4	57.0	71.3
200	**82.7**	53.9	72.2
5 seeds: 200	78.4	**76.3**	**89.5**

Table 8: Results comparing 100 seeds on MEDLINE.

seed sets, however it did not outperform WMEB with only 5 input seeds. WMEB and MEB do not gain from these additional seeds as they severely limit the search space by introducing many more colliding templates in the early iterations.

5.2 Templates

Previous work has not evaluated the quality of the templates extracted, which is crucial for tasks like question answering that will utilise the templates. Our evaluation compares the first 100 templates identified by each algorithm. Table 9 shows the distribution of *true* and *possible matches*.

Although each algorithm performs well on CELL and CLNE, the templates for these are predominately PM. This is due to the difficulty of disambiguating CELL from CLNE. Categories which are hard to identify have far more partial and false matches. For example, the majority of TUMR templates can also semantically identify BODY. WMEB still performs well on categories with few TM, in particular SIGN (99% with 54 PM) . This is a result of mutual exclusion which forces those templates to a single category.

5.3 Other experiments

BASILISK is noticeably less efficient than MEB (14 times slower on Web 1T, 5 times on MEDLINE) and WMEB (10 times slower on Web 1T, 4 times on MEDLINE). BASILISK cannot precalculate the scoring metrics as they are dependent on the state of the bootstrapping process.

Table 10 shows the effectiveness of WMEB's individual components on MEDLINE. Here MEB is

CAT	BAS		MEB		WMEB	
	TM	PM	TM	PM	TM	PM
ANTI	63	8	97	0	**100**	0
CELL	2	**98**	1	**68**	2	**84**
CLNE	1	99	79	21	78	22
DISE	80	15	5	81	95	5
DRUG	80	17	82	16	78	17
FUNC	62	33	10	42	49	50
MUTN	3	27	3	26	9	91
PROT	98	1	54	0	99	0
SIGN	93	6	90	5	12	**54**
TUMR	4	94	0	81	2	67
Av(100)	48.6	39.8	42.1	34.0	**52.4**	39.0

Table 9: Results comparing MEDLINE templates

	100	200	500
MEB	76.3	70.0	58.6
WMEB-pool	83.2	81.7	77.8
WMEB-weight	82.3	79.5	76.4
WMEB	87.1	89.5	88.2

Table 10: Effect of WMEB weights and pool.

the baseline with no pool or weighting. WMEB-pool corresponds to WMEB with weighting and without the cumulative pool, and WMEB-weight corresponds to WMEB with the pool and no weighting. The weighting is extremely effective with an approximate 7% performance gain over MEB. The pool also noticeably improved performance, especially in the later iterations where it is needed most. These two components combine effectively together to significantly outperform MEB and BASILISK.

Our last evaluation is performed on the final test-set – the TREC Genomics full-text articles. We compare the performance of each algorithm on the TREC and MEDLINE collections (Table 11). WMEB performs consistently better than BASILISK and MEB, however each has a significant performance drop on TREC. This is due to the variation in language use. In abstracts, the content is more dense and precise, and thus contexts are likely to be less noisy. Full-text articles also contain less cohesive sections. In fact, ANTI with the largest performance drop for each algorithm (WMEB 95% MEDLINE, 30% TREC) extracted templates from the methods section, identifying companies that provide ANTI.

ALG	MEDLINE		TREC	
	Av(100)	Av(200)	Av(100)	Av(200)
BAS	78.4	74.7	63.0	57.9
MEB	76.3	70.1	55.2	49.6
WMEB	87.1	**89.5**	67.8	**66.0**

Table 11: Results comparing MEDLINE and TREC.

6 Conclusions

In this paper, we have proposed *Weighted Mutual Exclusion Bootstrapping* (WMEB), for efficient extraction of high precision lexicons, and the templates that identify them, from raw text. WMEB extracts the terms and templates of multiple categories simultaneously, based on the assumption of mutual exclusion. WMEB extends on MEB by incorporating more sophisticated scoring of terms and templates based on association strength and a cumulative template pool to keep templates active in the extraction process.

As a result, WMEB is significantly more effective at reducing semantic drift than MEB, which uses a simple weighting function, and BASILISK, which does not strictly enforce mutual exclusion between categories. We have evaluated these algorithms using a variety of semantic categories on three different raw text collections. We show that WMEB extracts more reliable large semantic lexicons than MEB and BASILISK (even with far fewer seeds). WMEB is more robust within the biomedical domain, which has an immediate need for these tools.

In the future, we plan to further investigate the mutual exclusion assumption, and whether it can be weakened to increase recall without suffering semantic drift, and how it interacts with the term and template scoring functions.

Our results demonstrate that WMEB can accurately and efficiently extract terms and templates in both general web text and domain-specific biomedical literature, and so will be useful in a wide range of NLP applications.

Acknowledgements

We would like to thank the anonymous reviewers and members of the LTRG at the University of Sydney, for their feedback. This work was supported by the CSIRO ICT Centre and ARC Discovery grant DP0665973.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the 5th ACM International Conference on Digital Libraries*, pages 85–94.

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

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.

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.

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.

Rosie Jones, Andrew Mccallum, Kamal Nigam, and Ellen Riloff. 1999. Bootstrapping for text learning tasks. In *Proceedings of the IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications*.

Christopher Manning and Hinrich Schutze. 1999. *Foundations of Statistical Natural Language Processing*. MIT.

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.

Tara Murphy and James R. Curran. 2007. Experiments in mutual exclusion bootstrapping. In *Proceedings of the Australasian Language Technology Workshop (ALTW)*, pages 66–74.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the Conference on Computational Linguistics and the 46th Annual Meeting of the Association for Computational Linguistics*, pages 113–120, Sydney, Australia.

Deepak Ravichandran and Eduard Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 41–47, Philadelphia, USA.

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 and Jessica Shepherd. 1997. A corpus-based approach for building semantic lexicons. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, pages 117–124, Providence, RI, USA.

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.

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

Roman Yangarber, Winston Lin, and Ralph Grishman. 2002. Unsupervised learning of generalized names. In *Proceedings of the 19th International Conference on Computational linguistics (COLING)*, pages 1135–1141, San Francisco.

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.

David Yarowsky. 1995. Unsupervised word sense disambiguiation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196.

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