Bootstrapping a Semantic Lexicon on Verb Similarities

Shaheen Syed[1], Marco Spruit[1] and Melania Borit[2]
[1]*Department of Information and Computer Sciences, Utrecht University, Utrecht, The Netherlands*
[2]*Norwegian College of Fishery Science, University of Tromsø, Tromsø, Norway*
{s.a.s.syed, m.r.spruit}@uu.nl, melania.borit@uit.no

Keywords: Semantic Lexicon, Bootstrapping, Extraction Patterns, Web Mining.

Abstract: We present a bootstrapping algorithm to create a semantic lexicon from a list of seed words and a corpus that was mined from the web. We exploit extraction patterns to bootstrap the lexicon and use collocation statistics to dynamically score new lexicon entries. Extraction patterns are subsequently scored by calculating the conditional probability in relation to a non-related text corpus. We find that verbs that are highly domain related achieved the highest accuracy and collocation statistics affect the accuracy positively and negatively during the bootstrapping runs.

1 INTRODUCTION

Semantic lexicons for specific domains are increasingly becoming important in many Natural Language Processing (NLP) tasks such as information extraction, word sense disambiguation, anaphora resolution, speech recognition, and discourse processing. Online lexical resources such as WordNet (Miller et al., 1990) and Cyc (Lenat, 1995) are useful for generic domains but often fall short for domains that include specific terms, jargon, acronyms and other lexical variations. Handcrafting domain-specific lexicons can be time-consuming and costly. Various techniques have been developed to automatically create semantic lexicons, such as lexicon induction, lexicon learning, lexicon bootstrapping, lexical acquisition, hyponym learning, and web-based information extraction.

We define a semantic lexicon as a dictionary of hyponym words that share the same hypernym. For example, *cat*, *dog* or *cow* are all hyponym words that share the hypernym `ANIMAL`. Likewise, the words *red*, *green*, and *blue* are semantically related by the hypernym `COLOR`. A semantic lexicon differs from an ontology or taxonomy as it does not describe the formal representation of shared conceptualizations or information about concepts and their instances, nor does it provide a strict hierarchy of classes.

Several attempts have been made to automatically create semantic lexicons from text corpora by utilizing semantic relations in conjunctions (*dogs and cats and cows*), lists (*dogs, cats, cows*), appositives (*labrador retriever, a dog*) and compound nouns (*dairy cow*) (Riloff and Shepherd, 1997; Roark and Charniak, 1998; Phillips and Riloff, 2002; Widdows and Dorow, 2002). Others have used extraction patterns (*Colombia* was divided, *the country* was divided) (Thelen and Riloff, 2002; Igo and Riloff, 2009), instance/concept ("is-a") relationships (Pantel and Ravichandran, 2004), coordination patterns (Ziering et al., 2013a), multilingual symbiosis (Ziering et al., 2013b), or combinations thereof (Qadir and Riloff, 2012). Lexical learning from informal text, such as social media, has also been performed (Qadir et al., 2015).

Learning semantic lexicons is often based on existing corpora that may not be available for all domains. Furthermore, little research on lexicon learning has been performed on web text from informative sites, forums, blogs, and comment sections that contain content written by a variety of people, thus containing different writing idiosyncracies. We incorporate web mining techniques to build our own domain corpus and combine the BASILISK (Thelen and Riloff, 2002) bootstrapping algorithm and the Pointwise Mutual Information (PMI) scoring metric proposed by (Igo and Riloff, 2009). We based the hyponym relationships on the extraction pattern context of verb stem similarities and used a probability scoring metric to extract the most suitable verbs. Our aim is to use existing web content to create a semantic lexicon and subsequently use extraction patterns to find semantically related words. Extraction patterns, and more specifically the verbs within these patterns, are

Syed, S., Spruit, M. and Borit, M.
Bootstrapping a Semantic Lexicon on Verb Similarities.
In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 1: KDIR*, pages 189-196
ISBN: 978-989-758-203-5

scored against a non-related text corpus to explore if verbs that occur more frequently in domain text are more likely to be accompanied by semantically related nouns.

The remainder of this paper is structured as follows: First, we present previous work in semantic lexicon learning for hypernym-hyponym relationships. Second, we detail our bootstrapping algorithm, the extraction of the text corpus, the creation of extraction patterns, and the scoring method. Third, we analyze the semantic lexicon and the accuracy of our algorithm during each successive run of the bootstrapping process. We then conclude our work and provide new research directives.

2 PREVIOUS WORK

(Riloff and Shepherd, 1997) used noun co-occurrence statistics to bootstrap a semantic lexicon from raw data. Their bootstrapping algorithm scored conjunctions, lists, appositives and nominal compounds to find category words within a context window. It is one of the first attempts to build a semantic lexicon and uses human intervention to review the words and select the best ones for the final dictionary. (Roark and Charniak, 1998) applied a similar technique but use a different definition for noun co-occurrence and scoring candidate words. (Riloff and Shepherd, 1997) ranked and selected candidate words based on the ratio of noun co-occurrences in the seed list to the total frequency of the noun in the corpus while (Roark and Charniak, 1998) used log-likelihood statistics (Dunning, 1993) for final ranking.

(Widdows and Dorow, 2002) used graph models of the British National Corpus for lexical acquisition. They focused on the relationships between nouns when they occurred as part of a list. The nodes represent nouns and are linked to each other when they conjunct with either *and* or *or*. The edges are weighted by the frequency of the co-occurrence. Their algorithm mitigates the infection of bad words entering the candidate word list by looking at type frequency rather than token frequency.

(Phillips and Riloff, 2002) automatically created a semantic lexicon by looking at strong syntactic heuristics. They distinguish between two types of lexicons, (1) proper noun phrase and (2) common nouns, by utilizing the syntactic relationships between them. Syntactic structures are defined by appositives, compound nouns, and ”is-a” clauses—identity clauses with a main verb of *to be*. Statistical filtering is applied to avoid inaccurate syntactic structures and deterioration of the lexicon entries. (Pantel and Ravichandran, 2004) also looked at ”is-a” relationships by looking at concept signatures and applying a top-down approach. Their method first uses co-occurrence statistics for semantic classes to then find the most appropriate hyponym relationship.

(Thelen and Riloff, 2002) proposed a weakly supervised bootstrapping algorithm (BASILISK) to learn semantic lexicons by looking at extraction patterns. They used the AutoSlog (Riloff, 1996) extraction pattern learner and a list of manually selected seed words to build semantic lexicons for the MUC-4 proceedings, a terrorism domain related corpus. Extraction patterns capture role relationships and are used to find noun phrases with similar semantic meaning as a result of syntax and lexical semantics. This is explained by their example verb *robbed*. The subject of the verb robbed often indicates the perpetrator while the direct object of the verb robbed could indicate the victim or target. To avoid inaccurate words entering the lexicon and to increase the overall accuracy they learned multiple categories simultaneously. (Igo and Riloff, 2009) used BASILISK and applied co-occurrence statistics for hypernym en seed word collocation by computing a variation of Pointwise Mutual Information (PMI). They used web statistics to re-rank the words after the bootstrapping process was finished.

(Qadir and Riloff, 2012) used pattern-based dictionary induction, contextual semantic tagging, and coreference resolution in an ensemble approach. They combined the three techniques in a single bootstrapping process and added lexicon entries if words occur in at least two of the three methods. Since each of them exploit independent sources, their ensemble method improves precision in the early stages of the bootstrapping process, in which semantic drift (Curran et al., 2007) can decrease the overall accuracy substantially. (Ziering et al., 2013b) have also employed an ensemble method and used linguistic variations between multiple languages to reduce semantic drift.

We combine the BASILISK bootstrapping algorithm (Thelen and Riloff, 2002) and co-occurrence statistics proposed by (Igo and Riloff, 2009) to dynamically score each word before it enters the lexicon. We compare BASILISK's scoring metric *AvgLog* in contrast to the PMI metric at each bootstrap run rather than using PMI scores to re-rank the lexicon after the bootstrapping process has finished.

3 LEXICON BOOTSTRAPPING

We use a list of seed words to initiate the bootstrapping process and use this set of seed words to create a highly related text corpus by mining the web. Web mining was applied because domain specific corpora are not always readily available. Before bootstrapping begins, we use the linguistic expressions of extraction patterns (Riloff, 1996) and group noun phrases when they occur with the same stemmed verb. The stemmed verbs are then scored by a probability score with respect to a non-related text corpus. Verbs that are domain specific, that is, they infrequently occur in general text are given a higher score. We then use BASILISK's bootstrapping algorithm and a PMI scoring metric before adding new words to the lexicon. The PMI score is a measure of association between words. For this, we use web count statistics between hyponym and hypernym words to calculate collocation statistics and utilize these scores to only allow the strongest hyponyms to enter the lexicon, thus decreasing wrong lexicon entries during the bootstrapping process. A schematic overview of the bootstrapping process is displayed in Figure 1 and Algorithm 1. We will discuss the bootstrapping process in detail in subsequent sections.

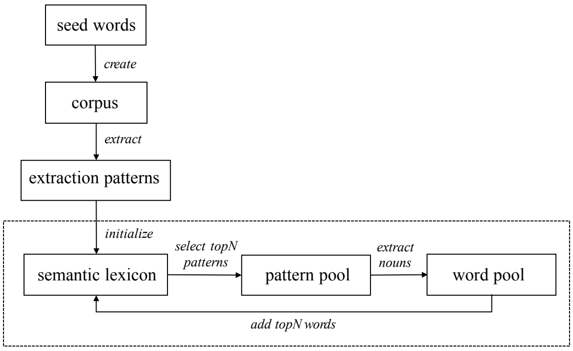

Figure 1: Schematic overview of bootstrapping process.

Algorithm 1: Bootstrap Lexicon.

```
$lexicon \leftarrow seedwords$
$corpus \leftarrow TopNSearches(seedwords)$
$patterns \leftarrow Patterns(corpus, 0.5 \leq p \leq 1.0)$
for $i = 0$ to $i < m$ do
    $patternpool \leftarrow Score(patterns, topN + i)$
    $words \leftarrow GetNouns(patternpool)$
    $lexicon \leftarrow ScoreWord(words, topN) \notin lexicon$
    $i++$
end for
return $lexicon$
```

3.1 Domain and Seed Words

We choose to build a single related lexicon for the *fisheries* domain. The fisheries domain includes a multitude of knowledge production approaches, from mono- to transdisciplinary. Biologists, oceanographers, mathematicians, computer scientists, anthropologists, sociologists, political scientists, economists and researchers from many other more disciplines contribute to the fisheries body of knowledge, together with non-academic participants, such as decision makers and stakeholders. Due to these diverse contributions of specialized language from a multitude of knowledge production approaches, the fisheries domain is characterized by a large body of words. For the same reason, this body of words is extremely rich in concepts. However, sometimes the concepts use different words to refer to the same abstraction (e.g. fishermen and fishers) and sometimes, even though they are using the same words, they are referring to different abstractions (e.g. fisher behavior may mean something for an economist and something else for an anthropologist). In addition, the fisheries domain has a high frequency of compound words (e.g. fisheries management, fishing method), in order to differentiate it from other resource management domains, such as forestry, for example. The definition of a hyponym-hypernym relation is therefore based on a more abstract level and also includes transitive relationships. For example, if x is a hyponym of y, and y is a hyponym of z, then x is a hyponym of z.

The seed words were chosen by experts from the Arctic University of Norway where they were asked for a list of 10 nouns or compound nouns that best cover the domain. The list contains the phrases *fishery ecosystem, fisheries management, fisheries policy, fishing methods, fishing gear, fishing area, fish, fish species, fishermen, fish supply chain.*

3.2 Building the Corpus

The fisheries corpus was created by extracting web text from a Google search for each of the seed words and extracting the content of the first 30 URLs. We enclosed the 10 search terms with quotations marks to force Google into an exact match. We found that the first 30 searches provided sufficient text for the corpus and still contained search related content. After mining the pages we started an extensive data cleaning process. Besides cleaning HTML, JavaScript, and CSS tags, we needed to clean text from e.g. headers, footers, labels, sidebars that entered the corpus. We removed content such as *"click here"*, *"all rights reserved"*, *"copyright"*. Finally, we scored the mined text fragments with an en-US and en-GB lookup dictionary D as:

$$S_{text_i} = \frac{\sum_{i=1}^{V_i} w_i}{V_i} \qquad (1)$$

Where V_i is the vocabulary size of $text_i$, and w_i a word match such that $w_i \in D$. We found $S_{text_i} \geq 0.3$ sufficient to clean the corpus even further as it removes non-English and improper formatted text. For example, phrases like *"Not logged inTalkContributionsCreate accountLog in"* bear no meaning and serves navigational purposes only when rendered by the web browser, yet it is extracted from the HTML content. Figure 2 shows the dispersion plot for the seed words unigrams after the cleaning phase.

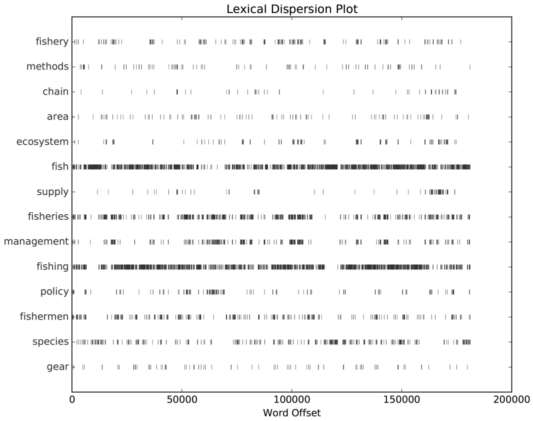

Figure 2: The lexical dispersion plot shows how often a word, displayed on the y-axis, occurs in the cleaned corpus. The x-axis displays the position within the corpus from beginning to end.

3.3 Chunking

We used the Punkt sentence tokenizer and a part-of-speech (POS) tagger to create individual sentences with their grammatical properties. We then used a shallow parser with regular expressions (RE) on POS tags to extract noun phrases (NP) when they occur as a direct object or subject. The verb that precedes or follows the NP is used to group related NPs when they share the same stem.

The RE grammar for the parser is defined as (1) `(<VB.*><DT>?<JJ>*<NN>+)` for direct object noun phrases and (2) `(<DT>?<JJ>*<NN>+<VB.*>+)` as subject noun phrases. (1) extracts a verb in any tense (`<VB.*>`), followed by an optional determiner (`<DT>`), followed by 0 or more adjectives (`<JJ>*`) and ending with at least one noun (`<NN>+`). Figure 3 shows the parse tree of the phrase *manage European fish and protect the marine environment*. We extract two patterns *manage European fish* and *protect the marine environment*.

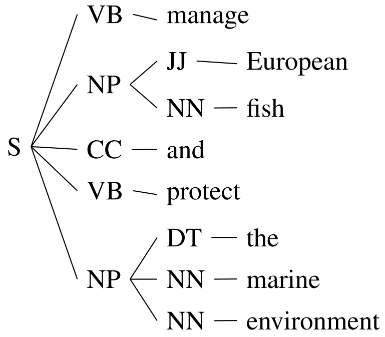

Figure 3: Parse tree for the noun phrase *European fish* with its accompanying transitive verb *manage* and noun phrase *the marine environment* with transitive verb *protect*. Both NPs occur as a direct object.

(2) extracts patterns starting with an optional determiner (`<DT>`), followed by 0 or more adjectives (`<JJ>*`), followed by at least one noun (`<NN>+`) and ending in any verb tense (`<VB.*>`). Figure 4 shows the parse tree for the phrase *the northern fisherman catches* in which the NP occurs as a subject.

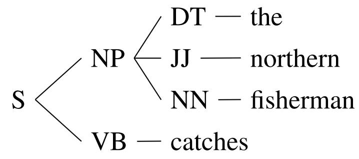

Figure 4: Parse tree for the noun phrase *the northern fisherman* with its accompanying transitive verb *catches*. The NP occurs as a subject.

3.4 Scoring Verbs

The verb tenses that precede an NP as a direct object, or follow the NP as a subject are stemmed with the Porter algorithm (Porter, 1980) to subsequently group NPs when creating the extraction patterns. Stemming groups verb tenses into the same stem—not necessarily the root of the verb. To score stemmed verbs we used a non-related text corpus containing rural information. The idea behind scoring the domain related verbs is to limit the extraction patterns to verbs that are more frequently found in the *fisheries* domain. For example, the verb *create* is often found in all sorts of domains but the verb *catch* not. The noun phrases that are linked to the verb *catch* are more likely to contain semantically related words.

Let D_1 be the set of stemmed verbs from the seed word related corpus (Section 3.2) and D_2 be the set of stemmed verbs from the non-related corpus (rural corpus) and that the vocabulary size $|V|_{D_1} \approx |V|_{D_2}$. Now let L be the combined set of stemmed verbs such that for every $l \in L$ we have that $l \in D_1 \cup D_2$. Let N_{l,D_1} denote the number of instances l contain in D_1. We estimate the probability that stemmed verb l contained within the seed word related set D_1:

$$P(l|D_1) = \frac{N_{l,D_1}}{N_{l,D_1} + N_{l,D_2}} \tag{2}$$

For example, the verb supervise, which includes supervised and supervising, gets stemmed into supervis. A frequency of D_1=12 and D_2=3 results in a score of 0.8. The distribution of scores is shown in Figure 5 and shows the number of verbs with their estimated probability. For example, there are 50 verbs with a score $\rho \geq 0.9$ which are highly domain related, such as *prohibit, catch, rescue, exploit, breath, deplete, fish*.

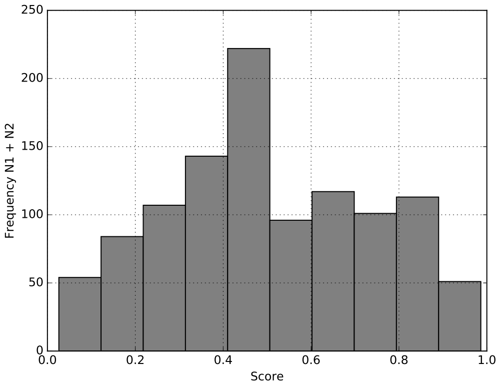

Figure 5: Frequency of stemmed verbs and estimated probability.

3.5 Verb Extraction Pattern

We selected verbs in which $\rho \geq 0.5$ and group them in steps of 0.1 such that $0.9 \leq \rho \leq 1.0$, $0.8 \leq \rho \leq 1.0$, ... , $0.5 \leq \rho \leq 1.0$ so that we analyze the top 10%, 20%, ... , 50% of verbs. To create the extraction pattern, any noun <NN> or compound noun <NN>+ extracted by the chunker is grouped together if they share the same stemmed verb in either subject and direct object cases. For example, Table 1 shows some phrases for the root verb regulate which gets stemmed into regul and root verb catch which has an identical stem. The nouns *industry*, *seafood trade*, *quota*, and *legislation* are grouped together into an extraction pattern because they all share the same stem regulate.

For every NP, we have extracted the noun or compound noun. We have not restricted ourselves to the head noun as compound nouns are generally more informative for the fisheries domain. For example, *marine science* was accepted as a fisheries related word but *science* not. An overview of some extraction patterns are listed below:

- {industry, seafood trade, quota, legislation}
- {fish, marine life, deep-dwelling, fish, fisherman}
- {catch, conservation, sustainability, growth, fishing, participation, tuna, management approach}
- {oxygen, air, fish, equipment, right}
- {sediment resuspension, world, bycatch, potential, fishing technique, hunger}

Table 1: Example phrases where the transitive verb of a phrase gets stemmed into the same stem.

stem	verb	noun	phrase
regul	regulated	industry	... regulated industry ...
regul	regulated	seafood trade	... regulated seafood trade ...
regul	regulated	quota	... quota regulated ...
regul	regulates	legislation	... legislation regulates ...
catch	catching	fish	... catching wild fish ...
catch	catching	marine life	... catching marine life ...
catch	catch	deep-dwelling fish	... catch deep-dwelling fish ...
catch	catches	fisherman	... the fisherman catches ...

We found 468 different extraction patterns in our corpus. An overview of the distribution is shown in Figure 6 (y-axis logarithmically scaled).

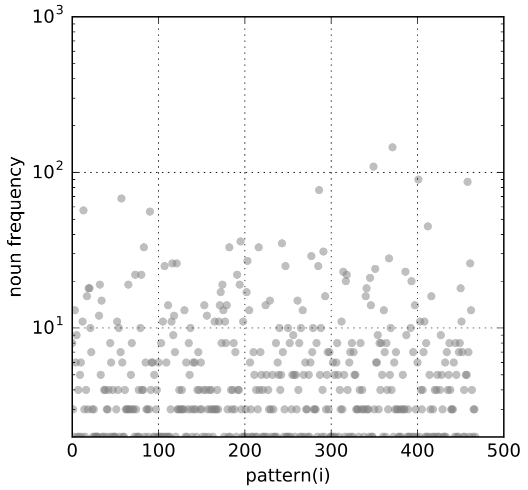

Figure 6: The frequency of nouns in an extraction pattern. The x-axis shows the number of extraction patterns found in our corpus (468). The y-axis shows the number of nouns grouped within that pattern.

3.6 Bootstrapping

The bootstrapping process starts with a list of 10 seed words (Section 3.1). Next, all the extraction patterns are scored by calculating the *RlogF* score (Riloff, 1996):

$$RlogF(pattern_i) = \begin{cases} \frac{F_i}{N_i} * log_2(F_i) & \text{if } F_i \geq 1 \\ -1 & \text{if } F_i = 0 \end{cases} \quad (3)$$

Where F_i is the number of lexicon words found in $pattern_i$ and N_i the total number of nouns in $pattern_i$. Extraction patterns that contain nouns that are already part of the lexicon will get a higher score. The first iteration selects the top N patterns which are then placed into a pattern pool. We used N=20 for the first iteration and incremented it by 1 every next run to allow new patterns to enter the process.

The next step is to score all the nouns that are part of the pattern pool. We evaluated two scoring metrics: (1) BASILISK's *AvgLog*, (2) PMI that uses search counts from the Bing search engine. The *AvgLog* score is defined as:

$$AvgLog(word_i) = \frac{\sum_{j=1}^{P_i} log_2(F_j + 1)}{P_i} \quad (4)$$

(1) *AvgLog* uses all the patterns to score the nouns found in the top N patterns. P_i is the number of patterns in which $word_i$ occurs and F_j the number of lexicon words found in pattern j. The nouns that are part of the pattern pool are given a higher score, thus being more semantically related, when they also occur in other extraction patterns with a high number of lexicon word matches.

(2) is based on hypernym collocation statistics proposed by (Igo and Riloff, 2009). We implement the PMI scoring metric within the bootstrapping process and dynamically calculate collocation statistics before adding new words in the lexicon. We hypothesize that lexicon words that occur more often in collocation with its domain are more likely to be semantically related. We use the number of hits between a lexicon word (hyponym) and its hypernym word by utilizing the *NEAR* operator from the Microsoft's BING search engine. We choose collocation range of 10 and define the PMI score as:

$$PMI_{x,y} = log \frac{\rho_{x,y}}{\rho_x \rho_y} \quad (5)$$

$$PMI_{x,y} = log(N) + log \frac{N_{x,y}}{N_x N_y} \quad (6)$$

$$PMI_{x,y} = log \frac{N_{x,y}}{N_x N_y} \quad (7)$$

$PMI_{x,y}$ is the Pointwise Mutual Information that lexicon word x occurs with hypernym y, where $\rho_{x,y}$ is the probability that x and y occur together on the web, ρ_x the probability that x occurs on the web, and ρ_y the probability that y occurs on the web. We would have to calculate probabilities such as ρ_x by dividing the number of x by the total number of web pages N and can rewrite $PMI_{x,y}$ by incorporating N. However, N is not known and can be omitted because it will be the same for each lexicon word. We can rewrite $PMI_{x,y}$ again by taking the log of the number of hits from the collocation statistics and dividing it by statistics of their individual parts.

Each noun that was part of the extraction pattern was given a score and the top-N nouns were selected to enter the lexicon. We added the noun with the highest score (N=1) to the lexicon and repeated the bootstrapping process.

4 EVALUATION

We evaluated the lexicon entries by a gold standard dictionary. Domain experts have labeled every noun or compound noun that was found in the extraction patterns. A value of 1 was assigned if the word was related to the fisheries domain and 0 if it had no relationship. We did not distinguish between highly relevant domain words and less relevant words. For example, the word *marine science* is highly relevant and has often a direct link to the domain, but the word *conservation* in itself can be ambiguous. It could be related to e.g. conserving artwork or to prevent depletion of natural resources. However, a word was considered to be correct if any sense of the word is semantically related. Furthermore, unknown words were manually looked up for their meaning. For example, *plecoglossus altivelis*, *oncorhynchus keta*, *leucosternon*, *peach anthias* and *khaki grunter* are types of fish unknown to the annotators, yet are semantically related.

We have run the bootstrapping algorithm until the lexicon contained 100 words. We repeated the process for the top 10% to top 50% scored verbs as discussed in Section 3.5 and compared the *AvgLog* (S_b) and PMI (S_{pmi}) scoring metric. Examples of lexicon entries are: *invertebrate seafood, ground fish, shellfish, demersal fish, mackerel, carp, crabs, deepwater shrimp, school, life, squid, hake, fisherman, cod, tuna, conservation reference size, trout, quota, sea, shrimp, mortality, freshwater, trawl, salmon, tenkara, snapper, method, license, attractor, pocket water, fee, vessel license, style fly, artisan, plastic worm, freshwater fly, saltwater, jack mackerel, trawler.*

Figure 7 shows the accuracy (percentage of correct lexicon words) for S_b when learning 100 semantically related words. The lines represent the verb probability groupings (e.g. top 10%, 20%). The accuracy degrades when incorrect words enter the lexicon and starts to contribute in learning more incorrect words. The accuracy converges to around 70% for all probability groupings. Figure 8 shows the accuracy for S_{pmi}. It shows that using web statistics for hyponym-hypernym words affects the accuracy positively and negatively, yet still converges to around 70% after 100 words are learned. The use of S_{pmi} affects the top 30% verbs substantially, outperforming S_b up to learning 90 words. However, S_{pmi} negatively affects the accuracy when looking at the top 50% verbs compared to S_b.

Scoring verbs before creating extraction patterns causes differences in accuracy up to a lexicon size of 90. Verbs that occur more often in domain related text, such as discussed in Section 3.4, essentially benefit the accuracy of the lexicon up to a certain size

Table 2: Lexicon accuracy when learning upto a 100 words. Accuracy scores are given for the top 10%, ... , 50% verbs for both S_b (scoring new entries with BASILISK's *AvgLog*), and S_{pmi} (scoring new entries with PMI).

Lexicon entries	$0.5 \leq \rho \leq 1.0$		$0.6 \leq \rho \leq 1.0$		$0.7 \leq \rho \leq 1.0$		$0.8 \leq \rho \leq 1.0$		$0.9 \leq \rho \leq 1.0$	
	S_b	S_{pmi}	S_b	S_{pmi}	S_b	S_{pmi}	S_b	S_{pmi}	S_b	S_{pmi}
20	0.8	0.75	0.8	0.75	0.8	0.9	0.8	0.8	0.9	0.95
30	0.73	0.67	0.67	0.67	0.73	0.87	0.73	0.73	0.8	0.8
40	0.68	0.65	0.68	0.6	0.68	0.78	0.7	0.68	0.8	0.82
50	0.74	0.68	0.64	0.68	0.64	0.7	0.66	0.66	0.8	0.8
60	0.72	0.68	0.68	0.68	0.68	0.73	0.67	0.68	0.75	0.77
70	0.69	0.67	0.67	0.67	0.67	0.71	0.67	0.66	0.71	0.76
80	0.68	0.69	0.68	0.68	0.68	0.71	0.68	0.64	0.71	0.75
90	0.69	0.7	0.69	0.7	0.69	0.69	0.68	0.68	0.69	0.72
100	0.7	0.69	0.69	0.68	0.7	0.7	0.69	0.68	0.66	0.68

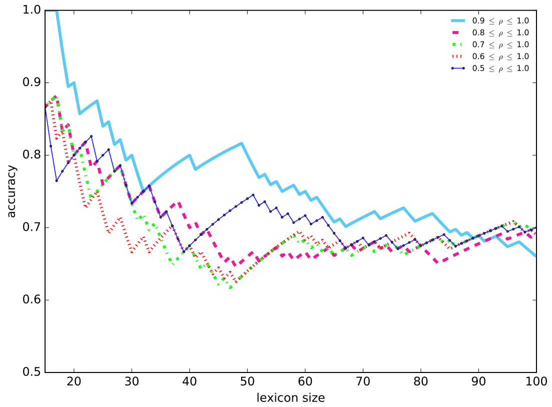

Figure 7: Graph that shows the accuracy when learning 100 words for the top 10% to top 50% verbs when scoring candidate words with BASILISK's *AvgLog* (S_b).

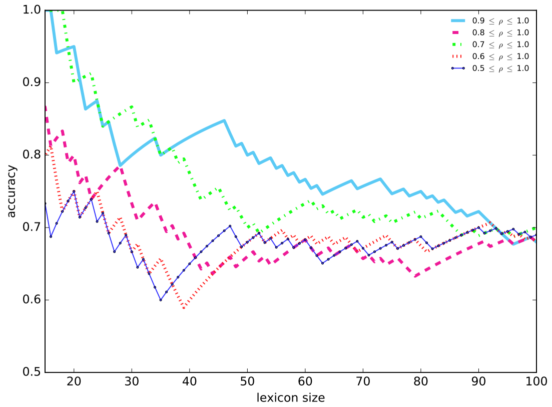

Figure 8: Graph that shows the accuracy when learning 100 words for the top 10% to top 50% verbs when scoring candidate words with Pointwise Mutual Information (S_{pmi}).

yet have limited effect on large lexicons. We would, however, have expected that the top 10% verbs outperform the top 20% and so on. This does not seem to hold for both S_b and S_{pmi}. For example, when using S_b and learning 50 words, the top 50% verbs achieved a higher accuracy (0.74), compared to the top 20% (0.66). Similarly, when learning 40 words and using S_{pmi}, the top 20% verbs achieved lower accuracy (0.68) than the top 30% verbs (0.78). The top 10% verbs for S_b and S_{pmi} achieve the highest accuracy in nearly all stages of the bootstrapping process. Small lexicons would benefit from selecting only the top 10% verbs to create extraction patterns, achieving an accuracy of around 0.8 when learning 50 words. An overview of all accuracy scores is given in Table 2.

5 CONCLUSION

In this paper, we presented a bootstrapping algorithm based on BASILISK and a highly related corpus that was created by mining web pages. We have created the corpus by utilizing the same set of seed words that initially was used to start the bootstrapping process. We used extraction patterns to group noun phrases with similar semantic meaning by grouping them when they share the same stemmed verb. We scored the extraction patterns with a non-related text corpus and calculated accuracy scores for the top 10%, 20%, 30%, 40% and 50%. Next to using BASILISK original scoring metric, we used a PMI score by looking at hyponym-hypernym collocation statistics.

We found varied results between the scored extraction patterns. Patterns that were created by looking at strong verbs that most often occur in domain related text, and less frequent in general (non-related) text, created a higher accuracy lexicon when looking at the top 10% scored verbs while other top scores showed mixed results. The use of collocation statistics by utilizing the NEAR operator of Microsoft's Bing search engine provided better accuracy for a number of scores but simultaneously caused a degrade in the accuracy for other verb scores.

The achieved accuracy covers web text for the fisheries domain and more research is needed concerning the generalizability into other domains and other forms of text, such as scientific literature and other technical language. Furtermore, research is needed to explain why accuracy varies between verb scores and why collocation statistics work better in some cases. Finally, research is also necessary when scoring the verbs against a non-related text corpus to see which types or genres of non-related domain corpora affects the domain under study.

ACKNOWLEDGEMENTS

This research was funded by the project SAF21 - Social science aspects of fisheries for the 21^{st} Century. SAF21 is a project financed under the EU Horizon 2020 Marie Skłodowska-Curie (MSC) ITN - ETN programme (project 642080).

REFERENCES

Curran, J. R., Murphy, T., and Scholz, B. (2007). Minimising semantic drift with mutual exclusion bootstrapping. In *In Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 172–180.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Igo, S. P. and Riloff, E. (2009). Corpus-based semantic lexicon induction with web-based corroboration. In *Proceedings of the Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics*, UMSLLS '09, pages 18–26, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lenat, D. B. (1995). Cyc: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):33–38.

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to WordNet: an online lexical database. *International Journal of Lexicography*, 3(4):235–244.

Pantel, P. and Ravichandran, D. (2004). Automatically labeling semantic classes. In *HLT-NAACL*, pages 321–328.

Phillips, W. and Riloff, E. (2002). Exploiting strong syntactic heuristics and co-training to learn semantic lexicons. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 125–132, Stroudsburg, PA, USA. Association for Computational Linguistics.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137.

Qadir, A., Mendes, P. N., Gruhl, D., and Lewis, N. (2015). Semantic lexicon induction from twitter with pattern relatedness and flexible term length. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2432–2439. AAAI Press.

Qadir, A. and Riloff, E. (2012). Ensemble-based semantic lexicon induction for semantic tagging. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 199–208, Stroudsburg, PA, USA. Association for Computational Linguistics.

Riloff, E. (1996). Automatically generating extraction patterns from untagged text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, AAAI'96, pages 1044–1049. AAAI Press.

Riloff, E. and Shepherd, J. (1997). A corpus-based approach for building semantic lexicons. In *In Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124.

Roark, B. and Charniak, E. (1998). Noun-phrase cooccurrence statistics for semiautomatic semantic lexicon construction. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 2*, COLING '98, pages 1110–1116, Stroudsburg, PA, USA. Association for Computational Linguistics.

Thelen, M. and Riloff, E. (2002). A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 214–221, Stroudsburg, PA, USA. Association for Computational Linguistics.

Widdows, D. and Dorow, B. (2002). A graph model for unsupervised lexical acquisition. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*, COLING '02, pages 1–7, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ziering, P., van der Plas, L., and Schütze, H. (2013a). Bootstrapping semantic lexicons for technical domains. In *IJCNLP*, pages 1321–1329. Asian Federation of Natural Language Processing / ACL.

Ziering, P., van der Plas, L., and Schütze, H. (2013b). Multilingual lexicon bootstrapping - improving a lexicon induction system using a parallel corpus. In *IJCNLP*, pages 844–848. Asian Federation of Natural Language Processing / ACL.