Comparing the Quality of Focused Crawlers and of the Translation Resources Obtained from them

B. R. Laranjeira♣, V. P. Moreira♣, A. Villavicencio♣, C. Ramisch♡, M. J. Finatto♢
♣Institute of Informatics, Federal University of Rio Grande do Sul (Brazil)
{bruno.rezendelaranjeira,viviane,avillavicencio}@inf.ufrgs.br
♡Aix Marseille Université, CNRS, LIF UMR 7279, 13288, Marseille (France)
carlos.ramisch@lif.univ-mrs.fr
♢Institute of Language and Linguistics, Federal University of Rio Grande do Sul (Brazil)
mfinatto@terra.com.br

Abstract

Comparable corpora have been used as an alternative for parallel corpora as resources for computational tasks that involve domain-specific natural language processing. One way to gather documents related to a specific topic of interest is to traverse a portion of the web graph in a targeted way, using focused crawling algorithms. In this paper, we compare several focused crawling algorithms using them to collect comparable corpora on a specific domain. Then, we compare the evaluation of the focused crawling algorithms to the performance of linguistic processes executed after training with the corresponding generated corpora. Also, we propose a novel approach for focused crawling, exploiting the expressive power of multiword expressions.

Keywords: Focused Crawling, Comparable Corpora, Machine Translation

1. Introduction

Traditionally, a parallel corpus is used to train a statistical machine translation (SMT) system. Parallel corpora, however, are scarce resources. They are usually restricted to government (EuroParl and Hansards) and religious texts (Bible). This scarcity becomes critical when we deal with domain specific translation, which requires knowledge about terms belonging to the domain of interest. For example, names of diseases are not likely to be found in these standard available parallel corpora. Comparable corpora, defined as collections of texts in two or more languages on the same domain but which are not translations of one another, on the other hand, are more abundant and thus, represent a more practically viable alternative for many tasks and domains.

There are basically two ways for collecting comparable corpora automatically. One way is to send carefully crafted queries to a *search engine* (such as Google or Yahoo) and retrieve the resulting pages. This is a potentially efficient alternative, but it is also highly dependent on the search engine and vulnerable, since the search engine can modify its services API or start charging fees for automatic accesses.

Another option is to use web crawlers (Liu, 2009), which are extremely valuable tools for collecting web documents. They start by retrieving and analysing the contents pointed by a set of seed URLs given as input. After the analysis, the crawler extracts and stores what the final application might consider useful, which may be pictures, videos, structural information or, in most cases, the raw textual content. The hyperlinks found are also extracted and inserted into a *queue*. Then, the URLs in this queue are followed, extracting the contents and the links from the pages pointed by them. This process continues recursively, until some stopping criteria is reached. This criteria may be, for example, a time limit for the crawler to run or the maximum size of the collection gathered, in number of pages or its logical file size (Liu, 2009; Chakrabarti, 2002; Baeza-Yates and Ribeiro-Neto, 1999).

Although crawlers are mainly used to build search engine indexes, it is also possible to customize them to follow the page links in a directed way in a process known as *focused crawling* (Chakrabarti, 2002). Focused crawlers organize URLs in a queue, known as *frontier*, that prioritizes pages that are more likely to be relevant. They are usually evaluated by their average *precision* (average cosine similarity between the collected pages and a user-defined set of terms which describe the domain) and by their *harvest rate* (ratio between the number of pages whose cosine similarity with the set of terms describing the domain is greater than a threshold and the total number of collected pages). However, since these measures depend highly on the quality of the set of terms used to guide the crawling process, they may not reflect the quality of the linguistic resources (e.g. comparable corpora) derived from the crawled pages.

The focus of this paper is on evaluating the relationship between the crawling strategy and the resulting translations obtained by an SMT system based on the collected comparable corpus. We ran experiments with several focused crawling algorithms, with varying sets of seed URLs, collecting corpora in English and Portuguese for the *dermatology* domain. Then, we used the resulting corpus to adapt a general-purpose SMT system to the specific domain and translate typical domain-specific sentences. We evaluated the quality of the crawling algorithms by calculating *intrinsic* metrics such as average precision. In addition, we calculated the BLEU score for the translations obtained by the adapted SMT system using each corpus. This can be seen as an *extrinsic* metric for the quality of the crawling process. The results indicate that the quality of seed URLs have a strong impact on the quality of the translations.

The remainder of this paper is structured as follows: Section 2. discusses some related work and emphasizes how

this one differs from them. Sections 3. and 4. explain the corpus crawling methods tested. The experimental setup is described in Section 5.. Results are shown in Section 6.. Finally, Section 7. discusses the results and presents our ongoing work.

2. Related Work

In the NLP community, the web has been exploited to build very large text bases. These monolingual resources are useful, for instance, for building corpus-driven models that require large amounts of data (Baroni and Lenci, 2010; Kiela and Clark, 2013). The WaCky repository[1] provides English, German, French, and Italian versions of freely available web corpora, containing around 2 billion words each. Baroni et al. (2009) describe the construction methodology of the WaCky corpora, based on search engines. They send random dictionary words as seed queries to the engine, and download the retrieved pages. Then, they apply text extraction and cleaning tools, in order to filter out spurious and non-textual content, keeping only raw sentences. As a result, a huge body of general-purpose texts is collected. BootCat (Baroni and Bernardini, 2004) is a popular tool that can be used for building corpora using web search engines.

Granada et al. (2012) adapt this technique to acquire domain-specific comparable corpora. They exploit the labels of concepts in multilingual ontologies as keywords for a search engine. The documents pointed by the top-10 retrieved URLs are then collected, keeping only HTML pages and discarding results like PDF and images. The collected documents are cleaned, using heuristics to identify and remove dates, URLs, e-mail addresses and boilerplate content (menus, headers). Finally, the texts are linguistically processed: words are lemmatized, part-of-speech tagged and parsed.

Talvensaari et al. (2008) use focused web crawlers for acquiring comparable corpora in order to train domain-specific MT systems. First, queries with keywords from the target domain are manually submitted to a search engine. The most frequent terms in the retrieved documents are used to construct more queries, whose results are scored according to their frequency and rank in the search result. The seed URLs for the crawling are the ones with the highest scores in the websites whose pages had the largest sum of scores. The score given to each page in the frontier is measured by the ratio of relevant words in the anchor text where the link was found in the pointer page and in the set of pages belonging to the same host as the pointer page. A word is considered relevant if it was among the most frequent words in the documents retrieved by the set of manually constructed queries. The documents collected by the focused crawler were aligned at the paragraph level and then used for adapting a generic SMT system to the genomics domain. Experiments were conducted in English, German, and Spanish, and results showed improvements compared to a MT system trained with a larger but generic corpus.

[1] http://wacky.sslmit.unibo.it/doku.php?id=corpora

Comparable corpora are invaluable resources for domain-specific multilingual NLP tasks and applications, when parallel data is scarce or nonexistent. For instance, the work of Daille (2012) addresses the acquisition of monolingual and bilingual domain-specific terms from comparable corpora. They extract monolingual candidate terms from the monolingual parts of a bilingual comparable corpus and use distributional information to obtain candidate translations for a source term. Monolingual candidates are extracted using syntactic patterns involving nouns, adjectives and multiword nominal sequences. Translations are obtained by aligning a source and a target term, assuming that they tend to occur in similar contexts. Our work also uses comparable corpora for performing domain-specific translation, but we focus on whole sentences instead of single terms.

Unlike related approaches (Granada et al., 2012; Baroni et al., 2009), our work does not rely on the use of search engines. We compare corpora built using unsupervised focused crawlers that walk the web graph following links. Like Talvensaari et al. (2008), we also use the corpora collected by focused crawlers to adapt SMT systems to a specific domain. However, our goal is to compare the quality of several focused crawling algorithms and the quality of the respective domain-specific SMT systems, rather than evaluating whether they are useful for the task. We assess whether there is a correlation between the intrinsic quality metrics of focused crawlers and the extrinsic quality metrics of automatic translations.

3. Crawling for Comparable Corpora

Our methodology is divided into two main steps: *crawling* and *training an SMT system*. For the crawling stage, we developed an easily extensible and customizable crawler. To define a new crawling strategy, we only needed to specify the behavior of the URL queue, according to the following crawling algorithms:

Universal Crawler (UC) This is a standard breadth first search. It does not employ any strategy to constrain or guide the crawling process. Any link found in any page receives the same priority level. Thus, this is the simplest crawling algorithm, because it just follows the links as they are found, in FIFO order.

Best-N-First (BFS) This algorithm sorts the links according to the relevance of their parent pages. The relevance of a page is measured by the cosine similarity between a vector $\vec{p}$ representing the page and another vector $\vec{e}$ that represents an example document. The provided example document must contain the terms that are considered relevant for the topic. Every position of these vectors corresponds to a word, and its value is the frequency of occurrence of the word in the document – the *Term Frequency* (TF), multiplied by its *Inverse Document Frequency* (IDF) in the collection. The intuition for using IDF is that rare terms (*e.g.*, fibroblast) are more descriptive for a document than common terms (*e.g.*, person). IDF estimates the rarity of a term t in a collection C by taking the log of the ratio between the number of documents in the collection, and the number of documents containing that

term:

$$IDF(t, C) = \log \frac{\|C\|}{\|\{c \in C, t \in c\}\|} \quad (1)$$

This is a standard vector-space representation of documents, called the *Term Frequency - Inverse Document Frequency* (TF-IDF) model. In the original Best-First algorithm, only the page at the head of the queue is downloaded in each iteration. We adopt a variation in which every iteration starts by downloading the N pages with highest priority from the queue (Liu, 2009). Only after the N-th page is downloaded, we extract the links from these pages, weight their priority and insert them into the queue. If we refer to the Best-100-First, for example, it means that $N = 100$ and, therefore, 100 URLs are taken from the queue at the beginning of every iteration.

N-gram Based BFS (NBFS) This strategy is similar to BFS. The main difference is that the granularity of the textual unit goes from single words to n-grams. Hence, the definition of relevance is slightly adapted to consider n-grams, instead of words. The components of the vectors for calculating the cosine similarity are now representations of n-grams. The values of the vector components are the product of how many times the n-gram can be found in the document and its IDF in the collection.

MWE Focused Crawler (MWEBFS) This approach is based on the $NBFS$. We have adapted the original TF-IDF model to add a third factor,– the *MWE Factor*, abbreviation for *Multiword Expression Factor*. The MWE Factor is only applicable when the vector components contain frequency information about n-grams, because it is based on the strength of association between the n-gram component words. Currently, we use the normalized *Pointwise Mutual Information* (PMI) of the n-gram as its MWE Factor. As future work, we would like to test other popular association measures, like log likelihood, Dice coefficient, or combinations of two or more scores (Manning and Schütze, 1999). High values of PMI mean that the n-gram component words are highly associated with each other. A large part of specialised terminology is composed of multiword terms. Thus, the idea of adding the MWE Factor to the TF-IDF model is to explore the expressive power of multiword expressions.

Shark Search (SS) The Shark Search algorithm, proposed by Hersovici et al. (1998), overcomes a problem faced by other focused crawlers, that are not able to distinguish links within the same document. Intuitively, a link found in the body of a page and among terms related to the topic of interest is more likely to be relevant than one found in the copyright footer, or in the advertisements section. Hence, the SS algorithm adds a locality score to the scores inherited from the parent page, in order to weight links within the same page. Locality information weights the text of the anchor of the link, the text around the anchor and the text of the whole page. We use this algorithm with the best parameters, according to the experiments conducted in the original paper, which eliminate the inherited score, leaving only the locality information. The text of the anchor was weighted by 0.8 and the text around it, by 0.2.

N-Gram Based SS (NSS) The NSS algorithm is an n-gram variant of SS. Its goal is to use coarse grained textual units, transforming the components of the vectors representing both, the text of the anchor and the text around it, into frequency information about n-grams and not about single words, as $NBFS$ also does. The rest of the algorithm and the parameters are kept just like those used in SS.

HMM Based Crawler (HMMC) This strategy is inspired on the work of Liu et al. (2006). They propose a focused crawler that learns how to find relevant pages by observing browsing sessions of expert users seeking for relevant reading material. While browsing, the user must mark what pages he/she considers relevant. After the sessions have been monitored, all accessed pages by the users are used as input to train a XMeans clustering model (Pelleg et al., 2000). The accessed pages are used to build a graph, in order to determine the distance, in links, between all non-relevant pages and any relevant one. With the clusterer and the information held in the graph, a *Hidden Markov Model* (HMM) is built. During the crawl, when a page is visited, its corresponding document is classified by the clustering model, by assigning it the class of its cluster. The hidden state of a page is its distance to a relevant page and the Viterbi algorithm (Forney Jr, 1973) is applied to determine the probability of the page pointed by the URL to be in every possible queue position, unlike the original paper that uses the forward algorithm. The position of the page in the queue is computed by its probability to be in a low-numbered state. For instance, if the probabilities of page u belonging to the states 0 and 1 are 0.2 and 0.3, and the probabilities calculated for page v are 0.2 and 0.4, then v will have higher priority than u. If the probabilities of both being at state 1 were also equal, the next criterion would be the probability of them being at state 2, and so on.

4. Training the SMT System

We adapted a generic phrase-based SMT system, trained on general-purpose parallel corpora, to the dermatology domain. Therefore, we used a target language model built from the specialized documents collected by every focused crawler algorithm. All training data (parallel and crawled corpora) was uniformly prepared to be given as input to the SMT training pipeline. Preprocessing involved sentence splitting, tokenization and sentence pruning by length. We observed that very short sentences represented, mainly, titles or menus and, thus, we removed them. Very long sentences, in turn, were, most times, multiple sentences concatenated together and, therefore, were also removed. Then, the result of the preprocessing steps was used as

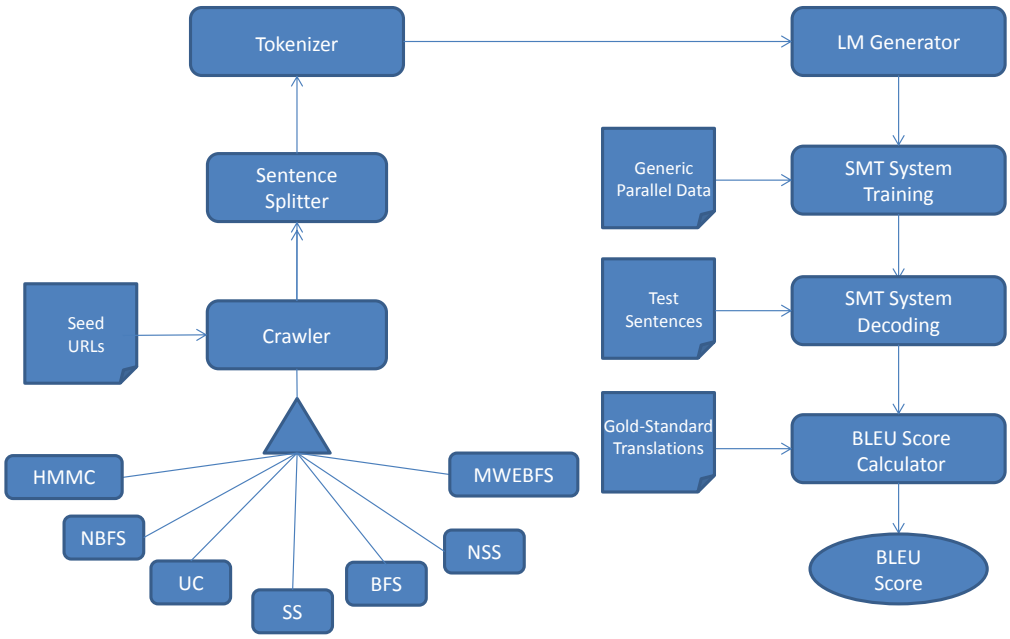

Figure 1: Summary of the evaluation methodology: the output of each focused crawling algorithm is preprocessed and fed into a language model generator. The specialized language models are used in a generic SMT system to translate test sentences, which are compared with gold-standard translations to calculate each system's BLEU.

monolingual training data for building a probabilistic n-gram language model. This model was used to adapt the SMT system, which scores the translation hypotheses by combining the generic phrase table probabilities with the specialised target n-gram probabilities. Finally, we used the adapted SMT systems to translate the same set of test sentences. Translation quality was measured by comparing the produced translations to gold-standard references and calculating the BLEU score for each system (Papineni et al., 2002).

The whole process, from focused crawling to automatic translation, is summarized in Figure 1.

5. Experimental Setup

We conducted our experiments in the domain of *dermatology*. Four human translators, experts in this domain, provided gold-standard sentence translations from English to Portuguese, which were used to calculate the BLEU score.

We defined two sets of seed pages to start the crawling process. One set, referred to as the *good seeds*, was composed of pages that were relevant to the dermatology domain. The other one contained pages from the education and sports domains, which were not related to the topic of interest, and was called the set of *bad seeds*. Table 5. details both URL sets we have used for the experiments. All URLs point to pages in Portuguese, because it was the target language of the translation experiments we conducted.

The crawlers were guided by a set of 35 terms extracted from pages describing the topic of dermatology, ranked by their TF-IDF values. The terms were also manually judged on whether they belong to the domain of interest. The set of terms contained words such as *cryosurgery* and *dermatopathologists*.

We used texts extracted from newspapers to estimate the values of IDF and PMI, in order to obtain more reliable counts. For English, we used texts from the *Los Angeles Times*[2], from 1994, and the *Glasgow Herald*[3], from 1995. For Portuguese, we used texts from *Folha de São Paulo*[4], from 1994 to 1995. To compute frequency and association scores, we used *Text-NSP* (Banerjee and Pedersen, 2003).

For every crawling algorithm, we used $N = 100$, consuming 100 URLs from the queue in every iteration. We varied the *NBFS*, *MWEBFS* and *NSS* algorithms by ranging the granularity of the textual unit from 2-grams to 3-grams. 20000 pages were collected using each of the focused crawling algorithms detailed in Section 3.

For the translation experiments, we trained a standard English→Portuguese phrase-based SMT system with Moses (Koehn et al., 2007), using parallel texts from the European Parliament and the JRC-Acquis Multilingual Parallel Corpus [5].

To generate the Portuguese language models, we preprocessed the corpora collected by the crawlers using the *Natural Language Toolkit*[6] for splitting sentences. The lower bound used for pruning sentences based on their lengths (in number of words) was 3 words. Based on the work of Granada et al. (2012), which analyzed the average sentence lengths for English, French, and Portuguese, we defined the upper bound as 22. These strict thresholds help ensuring that we discard menus, list items and wrongly split sentences.

The resulting sentences were used as input to IRSTLM (Federico et al., 2008), which generated a trigram language model for the corpus collected by each of the fo-

[2] http://www.latimes.com/
[3] http://www.heraldscotland.com/
[4] http://www.folha.uol.com.br/
[5] Both available at http://www.linguateca.pt/
[6] http://www.nltk.org/

Table 1: Lists of seed URLs used for focused crawling in the experiments.

Seed Set	URL
Good seeds	http://www.virtual.epm.br/cursos/dermabas/frame.htm http://www.dermatologia.net/novo/base/index.shtml http://pt.wikipedia.org/wiki/Dermatologia http://www.sbd.org.br/ http://www.anaisdedermatologia.org.br/public/default.aspx http://protetoresdapele.org.br/ http://www.enago.com.br/journal/Anais-Brasileiros-de-Dermatologia-402/ http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0365-05962010000100008
Bad seeds	http://www.gazetaesportiva.net/ http://globoesporte.globo.com/ http://esporte.uol.com.br/ http://www.educacao.sp.gov.br/ http://www.mec.gov.br/ http://www.estadao.com.br/educacao/

cused crawlers. Then, the language models were used to adapt Moses to the dermatology domain, generating a set of SMT systems. The idea is that the translation hypotheses generated by the generic phrase table can be combined by the log-linear model with domain-specific information from the language model. Thus, translations related to the domain will be preferred and receive higher scores.

Finally, each adapter SMT system was used to translate to Portuguese a set of sentences written in English. The output translations were evaluated using BLEU, compared to the gold-standard translations provided by the human experts. The results are shown in Section 6.

6. Results

Figure 3 summarizes the average precision for the crawlers obtained using good and bad seeds. The suffixes 2 and 3 in the name of some algorithms represent the length of the n-gram used. The best results were obtained by *MWEBFS-2*, using good seeds and by *NBFS-2*, with bad seeds, where good seeds in general outperformed the bad ones and had a slower decrease in precision with the increase in the number of pages.

BLEU scores calculated for the SMT systems are shown in Table 2. The best results are shown in bold. The table details partial scores with n-gram sizes from 1 to 4 and the final BLEU score in the last column. Similar to what was observed with the average precision, as expected, the use of good seeds tends to outperform the use of bad seeds.

A potential correlation between an algorithm with good average precision generating better SMT performance was not confirmed, since algorithms with low average precision obtained good BLEU scores, and vice-versa. For instance, the algorithms that obtained best average precision were *MWEBFS-2* and *NBFS-2*, for good and bad seeds. However, in terms of BLEU scores, they were ranked respectively sixth and fourth out of ten strategies.

7. Conclusion

In this work, we experimented with several focused crawling algorithms for gathering comparable corpora on a specific topic. We have also proposed a new approach for focused crawling, which takes into account the expressive power of multiword expressions. The quality of the focused crawlers was evaluated by their average cosine similarity with the set of terms that define the domain of interest. Texts collected by them were used to generate language models to adapt generic SMT systems to the target domain. Comparing their average cosine similarity with the quality of the SMT systems, we observed that the quality of the translations seems to depend more on the quality of the seed pages than on the focused crawling algorithm used.

Currently, we are working on using distributional measures for detecting domain-specific cross-language synonyms. These synonyms will be used to improve quality of SMT by providing domain-specific translation examples as additional training data or phrase table entries. As an upper bound for evaluating the quality of the synonyms, we intend to use the titles of all cross-language links from the Wikipedia articles that are under in the dermatology category, and train a SMT system using these examples.

For future work, we intend to apply improved techniques for removing uninteresting content from the HTML files, keeping only the text from the pages. We also plan to perform a larger scale evaluation of terms and expressions from these corpora, and in terms of their relevance to the domain of interest. We shall also expand our experiments by adding more languages, domains, and focused crawlers (such as the one proposed by Liu et al. (2004)).

Acknowledgements

We would like to thank the support of projects CAPES-COFECUB 707/11, CNPq 312184/2012-3, 482520/2012-4, and 478979/2012-6.

8. References

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley.

Banerjee, S. and Pedersen, T. (2003). The design, implementation, and use of the Ngram Statistic Package. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 370–381, Mexico City, February.

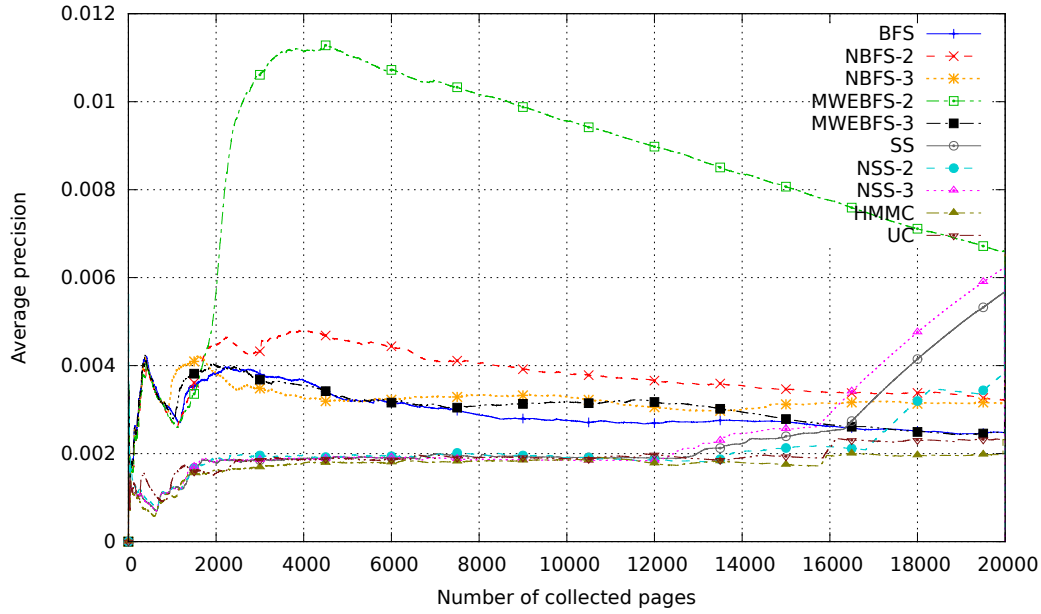

Figure 2: Average precision of the crawling algorithms using *good seeds*, as the number of downloaded pages increases.

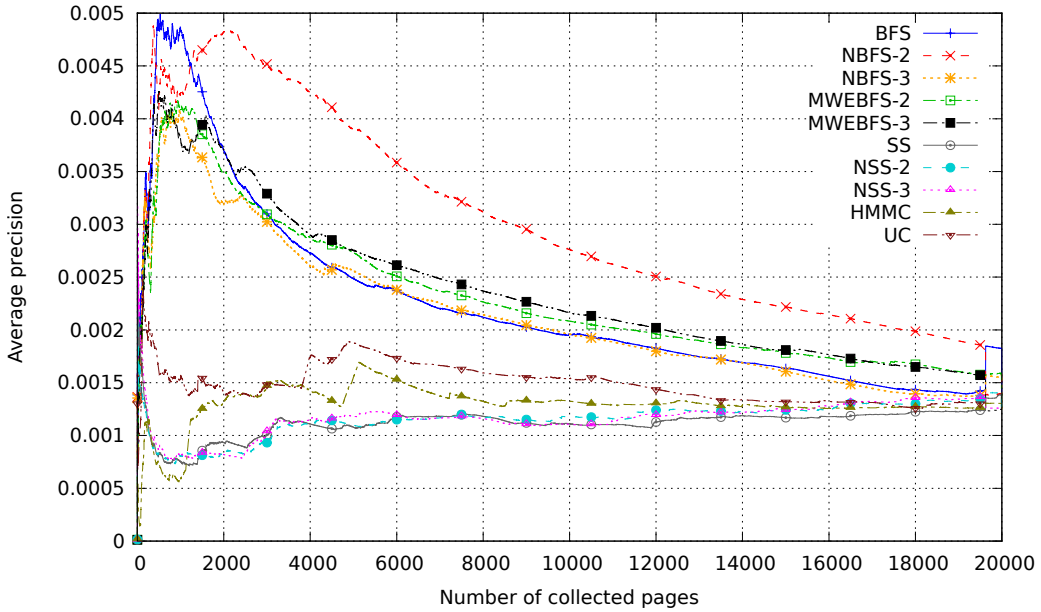

Figure 3: Average precision of the crawling algorithms using *bad seeds*, as the number of downloaded pages increases.

Baroni, M. and Bernardini, S. (2004). Bootcat: Bootstrapping corpora and terms from the web. In *Proceedings of LREC*, volume 4, pages 1313–1316.

Baroni, M. and Lenci, A. (2010). Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Chakrabarti, S. (2002). *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann.

Daille, B. (2012). Building bilingual terminologies from comparable corpora: The ttc termsuite. In *Proceedings of the 5th Workshop on Building and Using Comparable Corpora with special topic "Language Resources for Machine Translation in Less-Resourced Languages and Domains", co-located with LREC 2012*, Istanbul, Turkey.

Federico, M., Bertoldi, N., and Cettolo, M. (2008). Irstlm: an open source toolkit for handling large scale language models. In *Interspeech*, pages 1618–1621.

Forney Jr, G. D. (1973). The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Granada, R., Lopes, L., Ramisch, C., Trojahn, C., Vieira, R., and Villavicencio, A. (2012). A comparable corpus based on aligned multilingual ontologies. In *Proceedings of the First Workshop on Multilingual Modeling*,

Table 2: BLEU scores using good and bad seeds

Good Seeds					
	1-Gram	2-Gram	3-Gram	4-Gram	BLEU
BFS	64.5224	29.1139	13.5632	7.5758	20.1983
NBFS-2	**68.6654**	**31.7992**	**16.1731**	**9.5000**	**23.2883**
NBFS-3	64.9524	29.0123	13.8702	7.5980	20.5155
MWEBFS-2	64.3426	29.1577	14.3868	7.5325	20.2430
MWEBFS-3	64.6035	29.0795	13.2118	6.0000	18.7897
SS	65.5378	29.5896	13.6792	6.4935	19.3057
NSS-2	66.9291	29.8507	14.4186	7.6726	20.5602
NSS-3	66.7992	31.2500	15.0588	7.7720	21.1544
HMMC	67.0659	28.5714	12.0567	5.7292	18.0731
UC	65.0485	31.0924	15.5606	7.7889	21.5285
Bad Seeds					
	1-Gram	2-Gram	3-Gram	4-Gram	BLEU
BFS	64.0244	23.6203	11.3527	5.0667	16.0595
NBFS-2	61.8750	23.5828	12.4378	6.6116	17.4842
NBFS-3	64.1237	23.7668	12.5307	6.5217	17.5141
MWEBFS-2	61.8661	22.9075	11.3253	5.3191	16.3475
MWEBFS-3	61.4604	21.8062	9.8795	5.3191	15.5166
SS	63.2780	25.2822	12.3762	6.5753	17.8490
NSS-2	63.0165	25.1685	11.8227	5.7221	17.0813
NSS-3	62.6305	25.4545	13.7157	7.1823	18.5818
HMMC	**65.7315**	**27.3913**	**14.0143**	**7.5916**	**19.8186**
UC	63.0165	25.1685	11.8227	5.7221	17.0813

pages 25–31. Association for Computational Linguistics.

Hersovici, M., Jacovi, M., Maarek, Y., Pelleg, D., Shtalhaim, M., and Ur, S. (1998). The shark-search algorithm. an application: tailored web site mapping. *Computer Networks and ISDN Systems*, 30(1):317–326.

Kiela, D. and Clark, S. (2013). Detecting compositionality of multi-word expressions using nearest neighbours in vector space models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1427–1432, Seattle, Washington, USA, October. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. Association for Computational Linguistics.

Liu, H., Milios, E., and Janssen, J. (2004). Probabilistic models for focused web crawling. In *Proceedings of the 6th annual ACM international workshop on Web information and data management*, pages 16–22. ACM.

Liu, H., Janssen, J., and Milios, E. (2006). Using hmm to learn user browsing patterns for focused web crawling. *Data & Knowledge Engineering*, 59(2):270–291.

Liu, B. (2009). *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)*. Springer.

Manning, C. D. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. The MIT Press.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Pelleg, D., Moore, A., et al. (2000). X-means: Extending k-means with efficient estimation of the number of clusters. In *Proceedings of the seventeenth international conference on machine learning*, volume 1, pages 727–734. San Francisco.

Talvensaari, T., Pirkola, A., Järvelin, K., Juhola, M., and Laurikkala, J. (2008). Focused web crawling in the acquisition of comparable corpora. *Information Retrieval*, 11(5):427–445.