Domain-Specific Corpus Expansion with Focused Webcrawling

Steffen Remus and **Chris Biemann**
FG Language Technology
Computer Science Department
Technische Universität Darmstadt, Darmstadt, Germany
{remus,biem}@cs.tu-darmstadt.de

Abstract

This work presents a straightforward method for extending or creating in-domain web corpora by focused webcrawling. The focused webcrawler uses statistical N-gram language models to estimate the relatedness of documents and weblinks and needs as input only N-grams or plain texts of a predefined domain and seed URLs as starting points. Two experiments demonstrate that our focused crawler is able to stay focused in domain and language. The first experiment shows that the crawler stays in a focused domain, the second experiment demonstrates that language models trained on focused crawls obtain better perplexity scores on in-domain corpora. We distribute the focused crawler as open source software.

Keywords: web crawling, focused, topical, in-domain, web-corpora, language model, perplexity

1. Introduction

With increasing power of computational resources and algorithms to efficiently process more and more data in less time, the demand for larger text collections grows and the web as a huge and dynamic resource is nowadays the main source of any kind of data. The *WaCKy*[1] corpora (Baroni et al., 2009) or the *COW*[2] project (Schäfer and Bildhauer, 2012) are just some examples for language-processing-oriented web corpora. *ClueWeb* (Callan et al., 2009) or Amazons *common crawl* (Common Crawl Foundation, 2011) provide unprocessed html data from web crawls, which are further refined e.g. in (Pomikálek et al., 2012; Buck et al., 2014).

However, the data is largely collected without notions of topical interest. If an interest in a particular topic exists, corpora have to undergo extensive document filtering with simple and/or complex text classification methods. This leads to a lot of downloaded data being discarded with lots of computational resources being unnecessarily wasted.

One approach to work around these issues is to use the *BootCat* method (Baroni and Bernardini, 2004), which collects, based on keyword lists, web documents by sending combinations of keywords to a search engine provider. Here, one particular disadvantage is the use of a search engine provider as a black box, which makes it dependent on *a*) the general availability of the service, and *b*) ranking of the results based on the provider's (possibly subjective) choice (Kilgarriff, 2007).

This paper introduces a simple tool for *focused crawling*, which makes efficient use of computational resources as it downloads mainly websites of interest, i.e. those belonging to a certain topic. The domain of interest is defined by a statistical N-gram language model or sample texts that can be used to create a language model from.

2. Focused Crawling

The term 'focused crawling' (Chakrabarti et al., 1999) also known as 'topical crawling' (Menczer et al., 2004) refers to the process of crawling the web in a guided way with a focus on a specific topic. The task is to decide which link to follow and which not, or in which order to follow links before actually downloading the content of their respective destination, all of which happens during crawling time (Chakrabarti et al., 1999). This can either be seen as a classification task (McCallum et al., 1999; Chakrabarti et al., 1999; Medelyan et al., 2006) with binary decisions (yes or no), or as a ranking problem with a priority queue.

Our approach differs from existing approaches in that we employ a *language model* and *perplexity* as a measure of relevance for a particular web page, whereas other approaches use features such as individual components of the URL itself, e.g. server, path, query strings, etc., the surrounding context of the extracted hyperlink, text of the parent webpage, include lexical resources like *WordNet* (Fellbaum, 1998), and many more (Safran et al., 2012). One major advantage of the proposed methodology is the deliberate omission of negative instances for modeling.

This being said, the method does not need positively and negatively labeled data (Blum and Mitchell, 1998), neither needs the focus to be predefined as a certain category in a predefined taxonomy (Chakrabarti et al., 1999), and it also does not require manually constructed lexical resources for feature extraction (Safran et al., 2012), but only operates on an initially provided corpus of plain texts, which serves as domain definition[3].

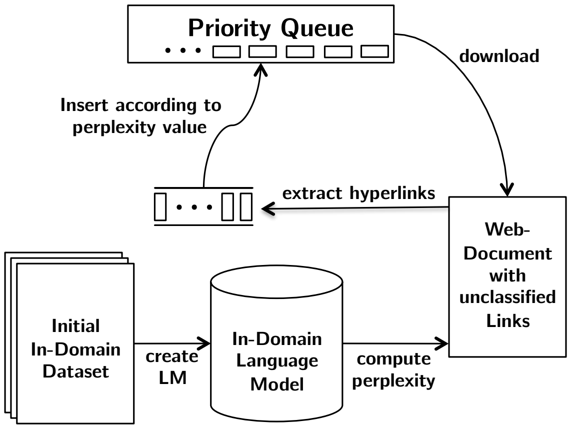
3. Methodology

Web pages of a certain genre or domain use a certain vocabulary (Biber, 1995), and these web pages, in turn, link to other web pages of the same genre or domain. This oversimplifying assumption is typically exploited in focused crawling (Chakrabarti et al., 1999; Menczer et al., 2004;

[1] Web-as-Corpus Kool Yinitiative http://wacky.sslmit.unibo.it/

[2] Corpora from the web, http://corporafromtheweb.org/

[3] E.g., in one of our experiments we use one Wikipedia article as domain definition.



Priority Queue
download
Insert according to perplexity value
extract hyperlinks
Web-Document with unclassified Links
Initial In-Domain Dataset
create LM
In-Domain Language Model
compute perplexity

Figure 1: Schematic overview of our focused crawling process.

Safran et al., 2012; Medelyan et al., 2006). Our approach on focused crawling relies on statistical *language models* combined with *perplexity* as a measure of relatedness. The priority of an extracted weblink is determined by the perplexity value of its parent document.

3.1. Modeling Domain Specificity

In this work, we employ statistical N-gram language models. Specifically, we use Kneser and Ney (1995)'s method, which constitutes the state of the art in word-based language modeling. In our experiments we used a 5-gram model[4]. Also, we filtered numbers and punctuation characters and considered only sequences with a minimum length of five words.

3.2. Perplexity Measure

Perplexity, often used to evaluate language model performance, is defined as $PP(X) = 2^{H(X)}$, where

$$H(X) = -\frac{1}{|X|} \sum_{x \in X} \log_2 p(x) \qquad (1)$$

is the cross entropy and $p(x)$ refers to the probability of a particular N-gram in the set of test N-grams X. The *boilerpipe* toolkit[5] (Kohlschütter et al., 2010) is used for *html stripping* and *boilerplate* removal.

URLs are prioritized based on the perplexity value of the parent document, i.e. the document it was extracted from. We maintain a priority-queue-like structure of URLs which have been collected so far and process this queue in increasing[6] order. URLs, which were extracted from documents that consent with the language model are thus considered for download before those with less consensus. Figure 1 shows a schematic overview of this basic crawling procedure.

3.3. System Architecture

Following Baroni et al. (2009), Schäfer and Bildhauer (2012), and Callan et al. (2009)[7] we use *Heritrix*[8] (Mohr et al., 2004) as the base crawler since it provides a well-established and sophisticated crawling framework and is extensible due to its modular design.

Heritrix's architecture follows suggestions by Manning et al. (2008) and uses one queue per server; the set of all queues is called *frontier*. Our priority scheme includes URL and server queues such that the priority of a server queue is determined by the highest priority of a member URL. Additionally, a bucketing strategy is introduced, which splits the perplexity range into three buckets called HIGH, MEDIUM, and NORMAL. URLs in higher prioritized buckets will be downloaded before others, even if this delays the overall crawling process[9]. These boundaries have to be assigned by the user and can be determined by running short test crawls.

4. Experiments

We conduct two experiments:

1. focused vs. non-focused small scale, single threaded crawls, limited to seed websites, and
2. focused vs. non-focused large scale, parallelized crawl for the German educational domain.

Our notion of "non-focused" is the default breadth-first-search-like crawling strategy in Heritrix.

For the first experiment we defined four seed URLs, each coming from the same language as one other and the same domain as one other, but not both. We chose *cats* and *technology* as domains and English and German as languages. The crawls are limited to the websites defined by the seed URLs, which comprise websites from a technical topic and a cat related topic. Maximally 100 documents are collected in a non-focused setting for reference purposes and in the following focused settings: We created two language models, one from the Wikipedia article for *cat*[10] and one for *Hauskatze*[11] (eng. *domestic cat*) and initialized two focused crawls for each language model. We then count the number of documents that were downloaded from each of the website servers. Our hypothesis is that the focused crawls download more documents from the websites that correspond to their domain definition.

For the second experiment we conducted a larger crawl and collected roughly 500GB of html data on the *German educational domain* in a non-focused and in a focused setting. The initial domain defining corpus is provided by

[4] 5-grams were chosen because personal experience in preliminary experiments yielded better results than other N-gram models.

[5] `https://code.google.com/p/boilerpipe`

[6] A lower perplexity score is preferable over a higher value.

[7] For ClueWeb12 (`http://www.lemurproject.org/clueweb12.php/`).

[8] `http://crawler.archive.org` developed and used by the Internet Archive Project: `https://archive.org`

[9] Using this strategy it is also possible to dynamically change the behavior of the crawl during runtime, e.g. by adapting threshold values for current needs.

[10] `http://en.wikipedia.org/w/index.php?title=Cat&oldid=651849595`

[11] `http://de.wikipedia.org/w/index.php?title=Hauskatze&oldid=139331448`

LM	catchannel.com	techcrunch.com	meine-katze.de	heise.de
–	27	25	25	23
(en) Cat	93	2	3	2
(de) Hauskatze	1	1	97	1

Table 1: Number of downloaded web pages for a non-focused crawl and two focused crawls based on an English and German language model for the domain “cats”. The crawls were bound to English and German technical and cat-related seed websites. Crawls were limited to 100 documents in total.

Nam et al. (2014) and its size is around 800K unique sentences. We split the corpus into two equally-sized *training* and *test* sets, where we use the training set for initialization of the language model and the test set for testing the crawler’s performance after the crawl has ended. The language of the original data is mainly German but contains small amounts of other languages. The distribution of languages in the corpus is listed in Table 2.

We then built different language models using the training set plus the cleaned plain text data. This is done in intervals, after collecting certain amounts of data during the crawl. At each interval the resulting language models are evaluated using the test set. Further, we took care of sentence deduplification, such that each sentence occurs only once for training the individual language models.

For evaluation, we calculate perplexity of the language model trained on the aggregated corpora on the test set. Our hypothesis is that a more focused crawl lets the perplexity value decline faster. We manually selected about 20 seed URLs, which refer to webpages related to the German educational domain. The maximum perplexity score achievable for the language model on the training data is around 10^7, which happens if only unknown words occur in a document. We discard links from documents with a perplexity larger than 10^5. Note that the crawl is not bound by any other limitation than perplexity scores. Hence, web pages are collected from arbitrary top-level domains, which is another advantage over corpora created with top-level-domain crawling (Goldhahn et al., 2014).

5. Results

Results of the first experiment, shown in Table 1, indicate that the focused crawler is able to focus on the specified topic. The non-focused crawl collects documents from all four websites in equal quantities as expected. The focused crawls, on the other hand, mainly download documents from servers that correspond to their domain definition. E.g. the focused crawl based on the English Wikipedia article for cat downloads most documents from `catchannel.com`, a website containing cat related content in English.

Results for the second, large-scale experiment are explained in the following sections.

Language	$train$	$test$	f	nf
de	96.81	96.83	92.31	15.51
fr	0.57	0.56	0.62	3.09
en	0.00	0.00	4.50	73.19
nl	0.02	0.02	0.22	0.55
es	0.00	0.00	0.26	1.58
it	0.01	0.01	0.32	1.16
$other$	2.59	2.58	1.78	4.91

Table 2: Distribution of languages in % in the train and test set as well in the focused (f) and non-focused (nf) crawl.

5.1. Adapting to Language

Because the crawler is generally un-bound, we collect URLs from a variety of top-level domains and also documents containing texts from different languages, something which obviously happens to a much larger extent in the non-focused setting (cf. Table 2).

English is the main interfering language since it is also the prevalent language used in the web. The focused crawl still roughly collects equal proportions of languages as in the training corpus.

5.2. Adapting to Domain

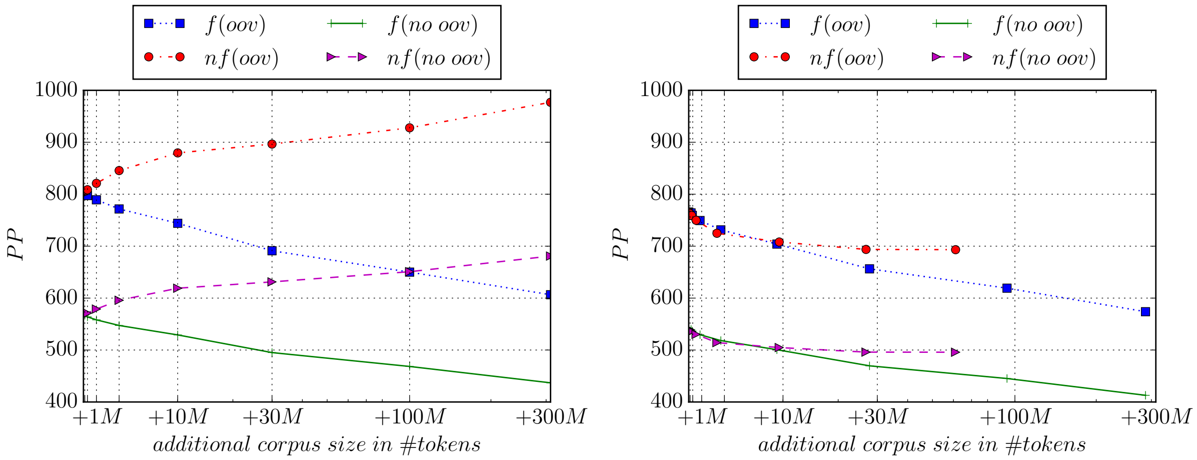
As described in Section 4. we built separate language models at intervals, i.e. after a certain amount of data was collected. The perplexity values on the held-out test set decreases with increasing corpus size in the focused crawl and increases with increasing corpus size in the non-focused crawl (cf. Figure 2a). This is also due to the fact of incorporating a significant amount of non-target languages in the non-focused crawl.

Since German is the prevalent language in our test corpus with $\sim$96%, we re-evaluated test set perplexity for the same chunks of data by selecting only German documents[12]. Figure 2b shows that the focused crawler is able to harvest language relevant documents throughout the crawl. When only considering German documents from the crawled data, the focused crawl yields consistently lower perplexity values and the difference increases as the crawl progresses. However, while more data is collected, the larger becomes the fractional amount of relevant / German vs. irrelevant / non-German data. That is, after downloading 300M tokens, the unfocused crawl’s usable German data amounts to 61M tokens and the focused crawl’s yields 277M tokens, which increases harvest by a factor of over 4.5.

6. Conclusion & Future Work

In this work, we presented a straightforward solution for corpus expansion as a focused web crawling approach using N-gram language models and perplexity as means for assessing the relevance of a web page and its outgoing web links. Experiments revealed that the methodology is able to improve model performance by using more domain specific

[12] We use JLani from the ASV-toolbox (Biemann et al., 2008) for automatic language identification.



(a)

(b)

size downloaded	100K	300K	1M	3M	10M	30M	100M	300M
size (de) f	87,105	267,371	867,686	2,661,758	9,122,831	27,783,058	93,799,389	277,303,711
size (de) nf	34,250	95,167	216,432	557,515	2,305,689	9,452,318	26,817,917	61,122,895

(c)

Figure 2: Perplexity on test set by crawl size for German educational data (a) and crawls filtered for German (b) comparing focused(f) and non-focused(nf) crawling. Perplexity is measured on the test set, where *out of vocabulary* words on the basis of the train set are considered (oov) or removed ($no\ oov$). The corpus size is given in terms of the number of additional tokens for the training set. Table (c) shows the absolute number of tokens of German data vs. downloaded data for both crawls.

data provided by the focused crawl. The software is made available as open source application[13].

Acknowledgments

This work has been supported by the German Institute for Educational Research (DIPF) under the KDSL program.

References

Baroni, M. and Bernardini, S. (2004). Bootcat: Bootstrapping corpora and terms from the web. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 1313–1316, Lisbon, Portugal. European Language Resources Association (ELRA).

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.

Biber, D. (1995). *Dimensions of Register Variation: A Cross-Linguistic Comparison*. Cambridge University Press, Cambridge.

Biemann, C., Quasthoff, U., Heyer, G., and Holz, F. (2008). Asv toolbox: a modular collection of language exploration tools. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, pages 1760–1767, Marrakech, Morocco.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory (COLT)*, pages 92–100, Madison, Wisconsin, USA.

Buck, C., Heafield, K., and van Ooyen, B. (2014). N-gram counts and language models from the Common Crawl. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 3579–3584, Reykjavik, Iceland.

Callan, J., Hoy, M., Yoo, C., and Zhao, L. (2009). The ClueWeb09 dataset. http://lemurproject.org/about.php.

Chakrabarti, S., van den Berg, M., and Dom, B. (1999). Focused crawling: a new approach to topic-specific web resource discovery. *Computer Networks*, 31(11–16):1623–1640.

Common Crawl Foundation. (2011). Common Crawl Dataset. http://www.commoncrawl.org/.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. The MIT Press.

Goldhahn, D., Remus, S., Quasthoff, U., and Biemann, C. (2014). Top-level Domain Crawling for Producing Comprehensive Monolingual Corpora from the Web. In *Proceedings of the LREC-14 workshop on Challenges in the Management of Large Corpora (CMLC-2)*, pages 10–14, Reykjavik, Iceland.

Kilgarriff, A. (2007). Googleology is bad science. *Computational Linguististics (CL)*, 33(1):147–151.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 181–184.

[13] `https://tudarmstadt-lt.github.io/topicrawler/`

Kohlschütter, C., Fankhauser, P., and Nejdl, W. (2010). Boilerplate detection using shallow text features. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 441–450, New York, NY, USA.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

McCallum, A., Nigam, K., Rennie, J., and Seymore, K. (1999). Building domain-specific search engines with machine learning techniques. In *Proc. AAAI-99 Spring Symposium on Intelligent Agents in Cyberspace*.

Medelyan, O., Schulz, S., Paetzold, J., Poprat, M., and Markó, K. (2006). Language specific and topic focused web crawling. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 865–868, Genoa, Italy.

Menczer, F., Pant, G., and Srinivasan, P. (2004). Topical web crawlers: Evaluating adaptive algorithms. *ACM Transactions Internet Technology (TOIT)*, 4(4):378–419.

Mohr, G., Kimpton, M., Stack, M., and Ranitovic, I. (2004). Introduction to heritrix, an archival quality web crawler. In *Proceedings of the 4th International Web Archiving Workshop IWAW'04*, pages 1–15, Bath, UK.

Nam, J., Kirschner, C., Ma, Z., Erbs, N., Neumann, S., Oelke, D., Remus, S., Biemann, C., Eckle-Kohler, J., Fürnkranz, J., Gurevych, I., Rittberger, M., and Weihe, K. (2014). Knowledge discovery in scientific literature. In *Proceedings of the 12th Konferenz zur Verarbeitung natürlicher Sprache (KONVENS 2014)*, pages 66–76, Hildesheim, Germany.

Pomikálek, J., Jakubíček, M., and Rychlý, P. (2012). Building a 70 billion word corpus of English from ClueWeb. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC)*, pages 502–506, Istanbul, Turkey.

Safran, M. S., Althagafi, A., and Che, D. (2012). Improving relevance prediction for focused web crawlers. In *2014 IEEE/ACIS 13th International Conference on Computer and Information Science (ICIS)*, pages 161–166. IEEE Computer Society.

Schäfer, R. and Bildhauer, F. (2012). Building large corpora from the web using a new efficient tool chain. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*, pages 486–493, Istanbul, Turkey.