Dirt Cheap Web-Scale Parallel Text from the Common Crawl

Jason R. Smith[1,2] jsmith@cs.jhu.edu
Herve Saint-Amand[3] herve@saintamh.org
Magdalena Plamada[4] plamada@cl.uzh.ch
Philipp Koehn[3] pkoehn@inf.ed.ac.uk
Chris Callison-Burch[1,2,5] ccb@cs.jhu.edu *
Adam Lopez[1,2] alopez@cs.jhu.edu

[1]Department of Computer Science, Johns Hopkins University
[2]Human Language Technology Center of Excellence, Johns Hopkins University
[3]School of Informatics, University of Edinburgh
[4]Institute of Computational Linguistics, University of Zurich
[5]Computer and Information Science Department, University of Pennsylvania

Abstract

Parallel text is the fuel that drives modern machine translation systems. The Web is a comprehensive source of preexisting parallel text, but crawling the entire web is impossible for all but the largest companies. We bring web-scale parallel text to the masses by mining the Common Crawl, a public Web crawl hosted on Amazon's Elastic Cloud. Starting from nothing more than a set of common two-letter language codes, our open-source extension of the STRAND algorithm mined 32 terabytes of the crawl in just under a day, at a cost of about $500. Our large-scale experiment uncovers large amounts of parallel text in dozens of language pairs across a variety of domains and genres, some previously unavailable in curated datasets. Even with minimal cleaning and filtering, the resulting data boosts translation performance across the board for five different language pairs in the news domain, and on open domain test sets we see improvements of up to 5 BLEU. We make our code and data available for other researchers seeking to mine this rich new data resource.[1]

1 Introduction

A key bottleneck in porting statistical machine translation (SMT) technology to new languages and domains is the lack of readily available parallel corpora beyond curated datasets. For a handful of language pairs, large amounts of parallel data are readily available, ordering in the hundreds of millions of words for Chinese-English and Arabic-English, and in tens of millions of words for many European languages (Koehn, 2005). In each case, much of this data consists of government and news text. However, for most language pairs and domains there is little to no curated parallel data available. Hence discovery of parallel data is an important first step for translation between most of the world's languages.

The Web is an important source of parallel text. Many websites are available in multiple languages, and unlike other potential sources—such as multilingual news feeds (Munteanu and Marcu, 2005) or Wikipedia (Smith et al., 2010)—it is common to find document pairs that are direct translations of one another. This natural parallelism simplifies the mining task, since few resources or existing corpora are needed at the outset to bootstrap the extraction process.

Parallel text mining from the Web was originally explored by individuals or small groups of academic researchers using search engines (Nie et al., 1999; Chen and Nie, 2000; Resnik, 1999; Resnik and Smith, 2003). However, anything more sophisticated generally requires direct access to web-crawled documents themselves along with the computing power to process them. For most researchers, this is prohibitively expensive. As a consequence, web-mined parallel text has become the exclusive purview of large companies with the computational resources to crawl, store, and process the entire Web.

To put web-mined parallel text back in the hands of individual researchers, we mine parallel text from the Common Crawl, a regularly updated 81-terabyte snapshot of the public internet hosted

*This research was conducted while Chris Callison-Burch was at Johns Hopkins University.

[1]github.com/jrs026/CommonCrawlMiner

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 1374–1383,
Sofia, Bulgaria, August 4-9 2013.

on Amazon's Elastic Cloud (EC2) service.[2] Using the Common Crawl completely removes the bottleneck of web crawling, and makes it possible to run algorithms on a substantial portion of the web at very low cost. Starting from nothing other than a set of language codes, our extension of the STRAND algorithm (Resnik and Smith, 2003) identifies potentially parallel documents using cues from URLs and document content (§2). We conduct an extensive empirical exploration of the web-mined data, demonstrating coverage in a wide variety of languages and domains (§3). Even without extensive pre-processing, the data improves translation performance on strong baseline news translation systems in five different language pairs (§4). On general domain and speech translation tasks where test conditions substantially differ from standard government and news training text, web-mined training data improves performance substantially, resulting in improvements of up to 1.5 BLEU on standard test sets, and 5 BLEU on test sets outside of the news domain.

2 Mining the Common Crawl

The Common Crawl corpus is hosted on Amazon's Simple Storage Service (S3). It can be downloaded to a local cluster, but the transfer cost is prohibitive at roughly 10 cents per gigabyte, making the total over $8000 for the full dataset.[3] However, it is unnecessary to obtain a copy of the data since it can be accessed freely from Amazon's Elastic Compute Cloud (EC2) or Elastic MapReduce (EMR) services. In our pipeline, we perform the first step of identifying candidate document pairs using Amazon EMR, download the resulting document pairs, and perform the remaining steps on our local cluster. We chose EMR because our candidate matching strategy fit naturally into the Map-Reduce framework (Dean and Ghemawat, 2004).

Our system is based on the STRAND algorithm (Resnik and Smith, 2003):

1. *Candidate pair selection:* Retrieve candidate document pairs from the CommonCrawl corpus.
2. *Structural Filtering:*
   (a) Convert the HTML of each document into a sequence of start tags, end tags, and text chunks.
   (b) Align the linearized HTML of candidate document pairs.
   (c) Decide whether to accept or reject each pair based on features of the alignment.
3. *Segmentation:* For each text chunk, perform sentence and word segmentation.
4. *Sentence Alignment:* For each aligned pair of text chunks, perform the sentence alignment method of Gale and Church (1993).
5. *Sentence Filtering:* Remove sentences that appear to be boilerplate.

Candidate Pair Selection We adopt a strategy similar to that of Resnik and Smith (2003) for finding candidate parallel documents, adapted to the parallel architecture of Map-Reduce.

The *mapper* operates on each website entry in the CommonCrawl data. It scans the URL string for some indicator of its language. Specifically, we check for:

1. Two/three letter language codes (ISO-639).
2. Language names in English and in the language of origin.

If either is present in a URL and surrounded by non-alphanumeric characters, the URL is identified as a potential match and the mapper outputs a key value pair in which the key is the original URL with the matching string replaced by $\star$, and the value is the original URL, language name, and full HTML of the page. For example, if we encounter the URL `www.website.com/fr/`, we output the following.

- Key: `www.website.com/*/`
- Value: `www.website.com/fr/`, French, (full website entry)

The *reducer* then receives all websites mapped to the same "language independent" URL. If two or more websites are associated with the same key, the reducer will output all associated values, as long as they are not in the same language, as determined by the language identifier in the URL.

This URL-based matching is a simple and inexpensive solution to the problem of finding candidate document pairs. The mapper will discard

[2] `commoncrawl.org`
[3] `http://aws.amazon.com/s3/pricing/`

most, and neither the mapper nor the reducer do anything with the HTML of the documents aside from reading and writing them. This approach is very simple and likely misses many good potential candidates, but has the advantage that it requires no information other than a set of language codes, and runs in time roughly linear in the size of the dataset.

Structural Filtering A major component of the STRAND system is the alignment of HTML documents. This alignment is used to determine which document pairs are actually parallel, and if they are, to align pairs of text blocks within the documents.

The first step of structural filtering is to linearize the HTML. This means converting its DOM tree into a sequence of start tags, end tags, and chunks of text. Some tags (those usually found within text, such as "font" and "a") are ignored during this step. Next, the tag/chunk sequences are aligned using dynamic programming. The objective of the alignment is to maximize the number of matching items.

Given this alignment, Resnik and Smith (2003) define a small set of features which indicate the alignment quality. They annotated a set of document pairs as parallel or non-parallel, and trained a classifier on this data. We also annotated 101 Spanish-English document pairs in this way and trained a maximum entropy classifier. However, even when using the best performing subset of features, the classifier only performed as well as a naive classifier which labeled every document pair as parallel, in both accuracy and F1. For this reason, we excluded the classifier from our pipeline. The strong performance of the naive baseline was likely due to the unbalanced nature of the annotated data— 80% of the document pairs that we annotated were parallel.

Segmentation The text chunks from the previous step may contain several sentences, so before the sentence alignment step we must perform sentence segmentation. We use the Punkt sentence splitter from NLTK (Loper and Bird, 2002) to perform both sentence and word segmentation on each text chunk.

Sentence Alignment For each aligned text chunk pair, we perform sentence alignment using the algorithm of Gale and Church (1993).

Sentence Filtering Since we do not perform any boilerplate removal in earlier steps, there are many sentence pairs produced by the pipeline which contain menu items or other bits of text which are not useful to an SMT system. We avoid performing any complex boilerplate removal and only remove segment pairs where either the source and target text are identical, or where the source or target segments appear more than once in the extracted corpus.

3 Analysis of the Common Crawl Data

We ran our algorithm on the 2009-2010 version of the crawl, consisting of 32.3 terabytes of data. Since the full dataset is hosted on EC2, the only cost to us is CPU time charged by Amazon, which came to a total of about $400, and data storage/transfer costs for our output, which came to roughly $100. For practical reasons we split the run into seven subsets, on which the full algorithm was run independently. This is different from running a single Map-Reduce job over the entire dataset, since websites in different subsets of the data cannot be matched. However, since the data is stored as it is crawled, it is likely that matching websites will be found in the same split of the data. Table 1 shows the amount of raw parallel data obtained for a large selection of language pairs.

As far as we know, ours is the first system built to mine parallel text from the Common Crawl. Since the resource is new, we wanted to understand the quantity, quality, and type of data that we are likely to obtain from it. To this end, we conducted a number of experiments to measure these features. Since our mining heuristics are very simple, these results can be construed as a lower bound on what is actually possible.

3.1 Recall Estimates

Our first question is about recall: of all the possible parallel text that is actually available on the Web, how much does our algorithm actually find in the Common Crawl? Although this question is difficult to answer precisely, we can estimate an answer by comparing our mined URLs against a large collection of previously mined URLs that were found using targeted techniques: those in the French-English Gigaword corpus (Callison-Burch et al., 2011).

We found that 45% of the URL pairs would

	French	German	Spanish	Russian	Japanese	Chinese
Segments	10.2M	7.50M	5.67M	3.58M	1.70M	1.42M
Source Tokens	128M	79.9M	71.5M	34.7M	9.91M	8.14M
Target Tokens	118M	87.5M	67.6M	36.7M	19.1M	14.8M
	Arabic	**Bulgarian**	**Czech**	**Korean**	**Tamil**	**Urdu**
Segments	1.21M	909K	848K	756K	116K	52.1K
Source Tokens	13.1M	8.48M	7.42M	6.56M	1.01M	734K
Target Tokens	13.5M	8.61M	8.20M	7.58M	996K	685K
	Bengali	**Farsi**	**Telugu**	**Somali**	**Kannada**	**Pashto**
Segments	59.9K	44.2K	50.6K	52.6K	34.5K	28.0K
Source Tokens	573K	477K	336K	318K	305K	208K
Target Tokens	537K	459K	358K	325K	297K	218K

Table 1: The amount of parallel data mined from CommonCrawl for each language paired with English. Source tokens are counts of the foreign language tokens, and target tokens are counts of the English language tokens.

have been discovered by our heuristics, though we actually only find 3.6% of these URLs in our output.[4] If we had included "f" and "e" as identifiers for French and English respectively, coverage of the URL pairs would increase to 74%. However, we chose not to include single letter identifiers in our experiments due to the high number of false positives they generated in preliminary experiments.

3.2 Precision Estimates

Since our algorithms rely on cues that are mostly external to the contents of the extracted data and have no knowledge of actual languages, we wanted to evaluate the precision of our algorithm: how much of the mined data actually consists of parallel sentences?

To measure this, we conducted a manual analysis of 200 randomly selected sentence pairs for each of three language pairs. The texts are heterogeneous, covering several topical domains like tourism, advertising, technical specifications, finances, e-commerce and medicine. For German-English, 78% of the extracted data represent perfect translations, 4% are paraphrases of each other (convey a similar meaning, but cannot be used for SMT training) and 18% represent misalignments. Furthermore, 22% of the true positives are potentially machine translations (judging by the quality), whereas in 13% of the cases one of the sentences contains additional content not expressed in the other. As for the false positives, 13.5% of them have either the source or target sentence in the wrong language, and the remaining ones representing failures in the alignment process. Across three languages, our inspection revealed that around 80% of randomly sampled data appeared to contain good translations (Table 2). Although this analysis suggests that language identification and SMT output detection (Venugopal et al., 2011) may be useful additions to the pipeline, we regard this as reasonably high precision for our simple algorithm.

Language	Precision
Spanish	82%
French	81%
German	78%

Table 2: Manual evaluation of precision (by sentence pair) on the extracted parallel data for Spanish, French, and German (paired with English).

In addition to the manual evaluation of precision, we applied language identification to our extracted parallel data for several additional languages. We used the "langid.py" tool (Lui and Baldwin, 2012) at the segment level, and report the percentage of sentence pairs where both sentences were recognized as the correct language. Table 3 shows our results. Comparing against our manual evaluation from Table 2, it appears that many sentence pairs are being incorrectly judged as non-parallel. This is likely because language identification tends to perform poorly on short segments.

[4]The difference is likely due to the coverage of the CommonCrawl corpus.

French	German	Spanish	Arabic
63%	61%	58%	51%
Chinese	Japanese	Korean	Czech
50%	48%	48%	47%
Russian	Urdu	Bengali	Tamil
44%	31%	14%	12%
Kannada	Telugu	Kurdish	
12%	6.3%	2.9%	

Table 3: Automatic evaluation of precision through language identification for several languages paired with English.

3.3 Domain Name and Topic Analysis

Although the above measures tell us something about how well our algorithms perform in aggregate for specific language pairs, we also wondered about the actual contents of the data. A major difficulty in applying SMT even on languages for which we have significant quantities of parallel text is that most of that parallel text is in the news and government domains. When applied to other genres, such systems are notoriously brittle. What kind of genres are represented in the Common Crawl data?

We first looked at the domain names which contributed the most data. Table 4 gives the top five domains by the number of tokens. The top two domain names are related to travel, and they account for about 10% of the total data.

We also applied Latent Dirichlet Allocation (LDA; Blei et al., 2003) to learn a distribution over latent topics in the extracted data, as this is a popular exploratory data analysis method. In LDA a topic is a unigram distribution over words, and each document is modeled as a distribution over topics. To create a set of documents from the extracted CommonCrawl data, we took the English side of the extracted parallel segments for each URL in the Spanish-English portion of the data. This gave us a total of $444,022$ documents. In our first experiment, we used the MALLET toolkit (McCallum, 2002) to generate 20 topics, which are shown in Table 5.

Some of the topics that LDA finds correspond closely with specific domains, such as topics 1 (`blingee.com`) and 2 (`opensubtitles.org`). Several of the topics correspond to the travel domain. Foreign stop words appear in a few of the topics. Since our system does not include any language identification, this is not surprising.[5] However it does suggest an avenue for possible improvement.

In our second LDA experiment, we compared our extracted CommonCrawl data with Europarl. We created a set of documents from both CommonCrawl and Europarl, and again used MALLET to generate 100 topics for this data.[6] We then labeled each document by its most likely topic (as determined by that topic's mixture weights), and counted the number of documents from Europarl and CommonCrawl for which each topic was most prominent. While this is very rough, it gives some idea of where each topic is coming from. Table 6 shows a sample of these topics.

In addition to exploring topics in the datasets, we also performed additional intrinsic evaluation at the domain level, choosing top domains for three language pairs. We specifically classified sentence pairs as useful or boilerplate (Table 7). Among our observations, we find that commercial websites tend to contain less boilerplate material than encyclopedic websites, and that the ratios tend to be similar across languages in the same domain.

	FR	**ES**	**DE**
`www.booking.com`	52%	71%	52%
`www.hotel.info`	34%	44%	-
`memory-alpha.org`	34%	25%	55%

Table 7: Percentage of useful (non-boilerplate) sentences found by domain and language pair. `hotel.info` was not found in our German-English data.

4 Machine Translation Experiments

For our SMT experiments, we use the Moses toolkit (Koehn et al., 2007). In these experiments, a baseline system is trained on an existing parallel corpus, and the experimental system is trained on the baseline corpus plus the mined parallel data. In all experiments we include the target side of the mined parallel data in the language model, in order to distinguish whether results are due to influences from parallel or monolingual data.

[5] We used MALLET's stop word removal, but that is only for English.

[6] Documents were created from Europarl by taking "SPEAKER" tags as document boundaries, giving us 208,431 documents total.

Genre	Domain	Pages	Segments	Source Tokens	Target Tokens
	Total	444K	5.67M	71.5M	67.5M
travel	www.booking.com	13.4K	424K	5.23M	5.14M
travel	www.hotel.info	9.05K	156K	1.93M	2.13M
government	www.fao.org	2.47K	60.4K	1.07M	896K
religious	scriptures.lds.org	7.04K	47.2K	889K	960K
political	www.amnesty.org	4.83K	38.1K	641K	548K

Table 4: The top five domains from the Spanish-English portion of the data. The domains are ranked by the combined number of source and target tokens.

Index	Most Likely Tokens
1	glitter graphics profile comments share love size girl friends happy blingee cute anime twilight sexy emo
2	subtitles online web users files rar movies prg akas dwls xvid dvdrip avi results download eng cd movie
3	miles hotels city search hotel home page list overview select tokyo discount destinations china japan
4	english language students details skype american university school languages words england british college
5	translation japanese english chinese dictionary french german spanish korean russian italian dutch
6	products services ni system power high software design technology control national applications industry
7	en de el instructions amd hyper riv saab kfreebsd poland user fr pln org wikimedia pl commons fran norway
8	information service travel services contact number time account card site credit company business terms
9	people time life day good years work make god give lot long world book today great year end things
10	show km map hotels de hotel beach spain san italy resort del mexico rome portugal home santa berlin la
11	rotary international world club korea foundation district business year global hong kong president ri
12	hotel reviews stay guest rooms service facilities room smoking submitted customers desk score united hour
13	free site blog views video download page google web nero internet http search news links category tv
14	casino game games play domaine ago days music online poker free video film sports golf live world tags bet
15	water food attribution health mango japan massage medical body baby natural yen commons traditional
16	file system windows server linux installation user files set debian version support program install type
17	united kingdom states america house london street park road city inn paris york st france home canada
18	km show map hotels hotel featured search station museum amsterdam airport centre home city rue germany
19	hotel room location staff good breakfast rooms friendly nice clean great excellent comfortable helpful
20	de la en le el hotel es het del und die il est der les des das du para

Table 5: A list of 20 topics generated using the MALLET toolkit (McCallum, 2002) and their most likely tokens.

4.1 News Domain Translation

Our first set of experiments are based on systems built for the 2012 Workshop on Statistical Machine Translation (WMT) (Callison-Burch et al., 2012) using all available parallel and monolingual data for that task, aside from the French-English Gigaword. In these experiments, we use 5-gram language models when the target language is English or German, and 4-gram language models for French and Spanish. We tune model weights using minimum error rate training (MERT; Och, 2003) on the WMT 2008 test data. The results are given in Table 8. For all language pairs and both test sets (WMT 2011 and WMT 2012), we show an improvement of around 0.5 BLEU.

We also included the French-English Gigaword in separate experiments given in Table 9, and Table 10 compares the sizes of the datasets used. These results show that even on top of a different, larger parallel corpus mined from the web, adding CommonCrawl data still yields an improvement.

4.2 Open Domain Translation

A substantial appeal of web-mined parallel data is that it might be suitable to translation of domains other than news, and our topic modeling analysis (§3.3) suggested that this might indeed be the case. We therefore performed an additional set of experiments for Spanish-English, but we include test sets from outside the news domain.

Europarl	CommonCrawl	Most Likely Tokens
9	2975	hair body skin products water massage treatment natural oil weight acid plant
2	4383	river mountain tour park tours de day chile valley ski argentina national peru la
8	10377	ford mercury dealer lincoln amsterdam site call responsible affiliates displayed
7048	675	market services european competition small public companies sector internal
9159	1359	time president people fact make case problem clear good put made years situation
13053	849	commission council european parliament member president states mr agreement
1660	5611	international rights human amnesty government death police court number torture
1617	4577	education training people cultural school students culture young information

Table 6: A sample of topics along with the number of Europarl and CommonCrawl documents where they are the most likely topic in the mixture. We include topics that are mostly found in Europarl or CommonCrawl, and some that are somewhat prominent in both.

WMT 11	FR-EN	EN-FR	ES-EN	EN-ES	EN-DE
Baseline	30.46	29.96	30.79	32.41	16.12
+Web Data	30.92	30.51	31.05	32.89	16.74
WMT 12	**FR-EN**	**EN-FR**	**ES-EN**	**EN-ES**	**EN-DE**
Baseline	29.25	27.92	32.80	32.83	16.61
+Web Data	29.82	28.22	33.39	33.41	17.30

Table 8: BLEU scores for several language pairs before and after adding the mined parallel data to systems trained on data from WMT data.

WMT 11	FR-EN	EN-FR
Baseline	30.96	30.69
+Web Data	31.24	31.17
WMT 12	**FR-EN**	**EN-FR**
Baseline	29.88	28.50
+Web Data	30.08	28.76

Table 9: BLEU scores for French-English and English-French before and after adding the mined parallel data to systems trained on data from WMT data including the French-English Gigaword (Callison-Burch et al., 2011).

Corpus	EN-FR	EN-ES	EN-DE
News Commentary	2.99M	3.43M	3.39M
Europarl	50.3M	49.2M	47.9M
United Nations	316M	281M	-
FR-EN Gigaword	668M	-	-
CommonCrawl	121M	68.8M	88.4M

Table 10: The size (in English tokens) of the training corpora used in the SMT experiments from Tables 8 and 9 for each language pair.

For these experiments, we also include training data mined from Wikipedia using a simplified version of the sentence aligner described by Smith et al. (2010), in order to determine how the effect of such data compares with the effect of web-mined data. The baseline system was trained using only the Europarl corpus (Koehn, 2005) as parallel data, and all experiments use the same language model trained on the target sides of Europarl, the English side of all linked Spanish-English Wikipedia articles, and the English side of the mined CommonCrawl data. We use a 5-gram language model and tune using MERT (Och, 2003) on the WMT 2009 test set.

Unfortunately, it is difficult to obtain meaningful results on some open domain test sets such as the Wikipedia dataset used by Smith et al. (2010). Wikipedia copied across the public internet, and we did not have a simple way to filter such data from our mined datasets.

We therefore considered two tests that were less likely to be problematic. The Tatoeba corpus (Tiedemann, 2009) is a collection of example sentences translated into many languages by volunteers. The front page of `tatoeba.org` was discovered by our URL matching heuristics, but we excluded any sentence pairs that were found in the CommonCrawl data from this test set.

The second dataset is a set of crowdsourced translation of Spanish speech transcriptions from the Spanish Fisher corpus.[7] As part of a research effort on cross-lingual speech applications, we obtained English translations of the data using Amazon Mechanical Turk, following a protocol similar to one described by Zaidan and Callison-Burch (2011): we provided clear instructions, employed several quality control measures, and obtained redundant translations of the complete dataset (Lopez et al., 2013). The advantage of this data for our open domain translation test is twofold. First, the Fisher dataset consists of conversations in various Spanish dialects on a wide variety of prompted topics. Second, because we obtained the translations ourselves, we could be absolutely assured that they did not appear in some form anywhere on the Web, making it an ideal blind test.

	WMT10	Tatoeba	Fisher
Europarl	89/72/46/20	94/75/45/18	87/69/39/13
+Wiki	92/78/52/24	96/80/50/21	91/75/44/15
+Web	96/82/56/27	99/88/58/26	96/83/51/19
+Both	96/84/58/29	99/89/60/27	96/83/52/20

Table 11: n-gram coverage percentages (up to 4-grams) of the source side of our test sets given our different parallel training corpora computed at the type level.

	WMT10	Tatoeba	Fisher
Europarl	27.21	36.13	46.32
+Wiki	28.03	37.82	49.34
+Web	28.50	41.07	51.13
+Both	28.74	41.12	52.23

Table 12: BLEU scores for Spanish-English before and after adding the mined parallel data to a baseline Europarl system.

We used 1000 sentences from each of the Tatoeba and Fisher datasets as test. For comparison, we also test on the WMT 2010 test set (Callison-Burch et al., 2010). Following Munteanu and Marcu (2005), we show the n-gram coverage of each corpus (percentage of n-grams from the test corpus which are also found in the training corpora) in Table 11. Table 12 gives end-to-end results, which show a strong improvement on the WMT test set (1.5 BLEU), and larger improvements on Tatoeba and Fisher (almost 5 BLEU).

[7] Linguistic Data Consortium LDC2010T04.

5 Discussion

Web-mined parallel texts have been an exclusive resource of large companies for several years. However, when web-mined parallel text is available to everyone at little or no cost, there will be much greater potential for groundbreaking research to come from all corners. With the advent of public services such as Amazon Web Services and the Common Crawl, this may soon be a reality. As we have shown, it is possible to obtain parallel text for many language pairs in a variety of domains very cheaply and quickly, and in sufficient quantity and quality to improve statistical machine translation systems. However, our effort has merely scratched the surface of what is possible with this resource. We will make our code and data available so that others can build on these results.

Because our system is so simple, we believe that our results represent lower bounds on the gains that should be expected in performance of systems previously trained only on curated datasets. There are many possible means through which the system could be improved, including more sophisticated techniques for identifying matching URLs, better alignment, better language identification, better filtering of data, and better exploitation of resulting cross-domain datasets. Many of the components of our pipeline were basic, leaving considerable room for improvement. For example, the URL matching strategy could easily be improved for a given language pair by spending a little time crafting regular expressions tailored to some major websites. Callison-Burch et al. (2011) gathered almost 1 trillion tokens of French-English parallel data this way. Another strategy for mining parallel webpage pairs is to scan the HTML for links to the same page in another language (Nie et al., 1999).

Other, more sophisticated techniques may also be possible. Uszkoreit et al. (2010), for example, translated all non-English webpages into English using an existing translation system and used near-duplicate detection methods to find candidate parallel document pairs. Ture and Lin (2012) had a similar approach for finding parallel Wikipedia documents by using near-duplicate detection, though they did not need to apply a full translation system to all non-English documents.

Instead, they represented documents in bag-of-words vector space, and projected non-English document vectors into the English vector space using the translation probabilities of a word alignment model. By comparison, one appeal of our simple approach is that it requires only a table of language codes. However, with this system in place, we could obtain enough parallel data to bootstrap these more sophisticated approaches.

It is also compelling to consider ways in which web-mined data obtained from scratch could be used to bootstrap other mining approaches. For example, Smith et al. (2010) mine parallel sentences from *comparable* documents in Wikipedia, demonstrating substantial gains on open domain translation. However, their approach required seed parallel data to learn models used in a classifier. We imagine a two-step process, first obtaining parallel data from the web, followed by comparable data from sources such as Wikipedia using models bootstrapped from the web-mined data. Such a process could be used to build translation systems for new language pairs in a very short period of time, hence fulfilling one of the original promises of SMT.

Acknowledgements

Thanks to Ann Irvine, Jonathan Weese, and our anonymous reviewers from NAACL and ACL for comments on previous drafts. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement 288487 (MosesCore). This research was partially funded by the Johns Hopkins University Human Language Technology Center of Excellence, and by gifts from Google and Microsoft.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Kay Peterson, Mark Przybocki, and Omar F. Zaidan. 2010. Findings of the 2010 joint workshop on statistical machine translation and metrics for machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 17–53. Association for Computational Linguistics.

Chris Callison-Burch, Philipp Koehn, Christof Monz, and Omar F. Zaidan. 2011. Findings of the 2011 workshop on statistical machine translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, pages 22–64. Association for Computational Linguistics.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June. Association for Computational Linguistics.

Jiang Chen and Jian-Yun Nie. 2000. Parallel web text mining for cross-language ir. In *IN IN PROC. OF RIAO*, pages 62–77.

J. Dean and S. Ghemawat. 2004. Mapreduce: simplified data processing on large clusters. In *Proceedings of the 6th conference on Symposium on Opearting Systems Design & Implementation-Volume 6*, pages 10–10. USENIX Association.

William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Comput. Linguist.*, 19:75–102, March.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 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*, ACL '07, pages 177–180. Association for Computational Linguistics.

P. Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5.

Edward Loper and Steven Bird. 2002. Nltk: the natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics - Volume 1*, ETMTNLP '02, pages 63–70. Association for Computational Linguistics.

Adam Lopez, Matt Post, and Chris Callison-Burch. 2013. Parallel speech, transcription, and translation: The Fisher and Callhome Spanish-English speech translation corpora. Technical Report 11, Johns Hopkins University Human Language Technology Center of Excellence.

Marco Lui and Timothy Baldwin. 2012. langid.py: an off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, ACL '12, pages 25–30. Association for Computational Linguistics.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving Machine Translation Performance by Exploiting Non-Parallel Corpora. *Comput. Linguist.*, 31:477–504, December.

Jian-Yun Nie, Michel Simard, Pierre Isabelle, and Richard Durand. 1999. Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the web. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pages 74–81, New York, NY, USA. ACM.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *acl*, pages 160–167, Sapporo, Japan.

P. Resnik and N. A Smith. 2003. The web as a parallel corpus. *Computational Linguistics*, 29(3):349–380.

Philip Resnik. 1999. Mining the web for bilingual text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 527–534. Association for Computational Linguistics.

Jason R. Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting Parallel Sentences from Comparable Corpora using Document Level Alignment. In *NAACL 2010*.

Jörg Tiedemann. 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia, Borovets, Bulgaria.

Ferhan Ture and Jimmy Lin. 2012. Why not grab a free lunch? mining large corpora for parallel sentences to improve translation modeling. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 626–630, Montréal, Canada, June. Association for Computational Linguistics.

Jakob Uszkoreit, Jay M. Ponte, Ashok C. Popat, and Moshe Dubiner. 2010. Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1101–1109. Association for Computational Linguistics.

Ashish Venugopal, Jakob Uszkoreit, David Talbot, Franz J. Och, and Juri Ganitkevitch. 2011. Watermarking the outputs of structured prediction with an application in statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1363–1372. Association for Computational Linguistics.

Omar F. Zaidan and Chris Callison-Burch. 2011. Crowdsourcing translation: Professional quality from non-professionals. In *Proc. of ACL*.