Domain Adaptation of Statistical Machine Translation using Web-Crawled Resources: A Case Study

Pavel Pecina[1], **Antonio Toral**[2], **Vassilis Papavassiliou**[3], **Prokopis Prokopidis**[3], **Josef van Genabith**[2]

[1]Faculty of Mathematics and Physics
Charles University in Prague
Czech Republic
pecina@ufal.mff.cuni.cz

[2]School of Computing
Dublin City Universiy
Dublin 9, Ireland
{atoral,josef}@computing.dcu.ie

[3]Institute for Language and
Speech Processing, Athena RIC
Athens, Greece
{vpapa,prokopis}@ilsp.gr

Abstract

We tackle the problem of domain adaptation of Statistical Machine Translation by exploiting domain-specific data acquired by domain-focused web-crawling. We design and evaluate a procedure for automatic acquisition of monolingual and parallel data and their exploitation for training, tuning, and testing in a phrase-based Statistical Machine Translation system. We present a strategy for using such resources depending on their availability and quantity supported by results of a large-scale evaluation on the domains of Natural Environment and Labour Legislation and two language pairs: English–French, English--Greek. The average observed increase of BLEU is substantial at 49.5% relative.

1 Introduction

Recent advances of Statistical Machine Translation (SMT) have improved Machine Translation (MT) quality to such an extent that it can be successfully used in industrial processes (Flournoy and Duran, 2009). However, this mostly happens in very specific domains for which ample training data is available (Wu et al., 2008). Using in-domain[1] data for training has a substantial effect on the final translation quality: SMT, as any other machine-learning application, is not guaranteed to perform optimally if the data for training and testing are not identically (and independently) distributed, which is often the case in practice. The main problem is usually vocabulary coverage: specific domain texts typically contain vocabulary that is not likely to be found in texts from other domains (Banerjee et al., 2010). Other problems can be caused by divergence in style or genre where the difference is not only in lexis but also in grammar.

[1]In this work, in-domain always refers to the domain of test data.

In order to achieve optimal performance, an SMT system should be trained on data from the same domain, genre, and style as it is applied to. For many domains, though, in-domain data of a size sufficient to train a full system is hard to find. Recent experiments have shown that even small amounts of such data can be used to adapt a system to the domain of interest (Koehn et al., 2007).

In this work, we present a strategy for automatic web-crawling and cleaning of domain-specific data. Further, our exhaustive experiments, carried out for the Natural Environment (*env*) and Labour Legislation (*lab*) domains and English–French (*EN–FR*) and English–Greek (*EN–EL*) language pairs (in both directions), demonstrate how the crawled data improves SMT quality.

After an overview of related work, we discuss the possibility of adapting a general-domain SMT system by using various types of in-domain data. Then, we present our web-crawling procedure followed by a description of a series of experiments exploiting the data we acquired. Finally, we report on the results and conclude with recommendations for similar attempts to domain adaptation in SMT.

2 Related work and state of the art

2.1 Domain-focused web crawling

A key challenge for a focused crawler that aspires to build domain-specific web collections is the prioritisation of the links to follow. Several algorithms have been exploited for selecting the most promising links. The Best-First algorithm (Cho et al., 1998) sorts the links with respect to their relevance scores and selects a predefined amount of them as the seeds for the next crawling cycle. Menczer and Belew (2000) proposed an adaptive population of agents, called InfoSpiders, and searched for pages relevant to a domain using evolving query vectors and Neural Networks to decide which links to follow. Hybrid models and modifications of these crawling strategies have

language pair (L1–L2)	*dom*	*set*	*source*	*sentence pairs*	*L1 tokens*	*/ vocabulary*	*L2 tokens*	*/ vocabulary*
English–French	*gen*	train	Europarl 5	1,725,096	47,956,886	73,645	53,262,628	103,436
		dev	WPT 2005	2,000	58,655	5,734	67,295	6,913
		test	WPT 2005	2,000	57,951	5,649	66,200	6,876
English–Greek	*gen*	train	Europarl 5	964,242	27,446,726	61,497	27,537,853	173,435
		dev	WPT 2005	2,000	58,655	5,734	63,349	9,191
		test	WPT 2005	2,000	57,951	5,649	62,332	9,037

Table 1: Detailed statistics of the general-domain data sets obtained from the Europarl corpus and the WPT 2005 workshop.

also been proposed (Gao et al., 2010) with the aim of reaching relevant pages rapidly.

Apart from the crawling algorithm, classification of web content as relevant to a domain or not also affects the acquisition of domain-specific resources, on the assumption that relevant pages are more likely to contain links to more pages in the same domain. Qi and Davison (2009) review features and algorithms used in web page classification. In most of the algorithms reviewed, on-page features (i.e. textual content and HTML tags) are used to construct a corresponding feature vector and then, several machine-learning approaches, such as SVMs, Decision Trees, and Neural Networks, are employed (Yu et al., 2004).

Considering the Web as a parallel corpus, Resnik and Smith (2003) proposed the STRAND system, in which they used Altavista to search for multilingual websites and examined the similarity of the HTML structures of the fetched web pages in order to identify pairs of potentially parallel pages. Similarly, Esplà-Gomis and Forcada (2010) proposed Bitextor, a system that exploits shallow features (file size, text length, tag structure, and list of numbers in a web page) to mine parallel documents from multilingual web sites. Besides structure similarity, other systems either filter fetched web pages by keeping only those containing language markers in their URLs (Désilets et al., 2008), or employ a predefined bilingual wordlist (Chen et al., 2004), or a naive aligner (Zhang et al., 2006) in order to estimate the content similarity of candidate parallel web pages.

2.2 Domain adaptation in SMT

The first attempt towards domain adaptation in SMT was made by Langlais (2002) who integrated in-domain lexicons into the translation model. Eck et al. (2004) presented a language model adaptation technique applying an information retrieval approach based on selecting similar sentences from available training data. Hildebrand et al. (2005) applied the same approach on the translation model. Wu et al. (2005) proposed an alignment adaptation approach to improve domain-specific word alignment. Munteanu and Marcu (2005) automatically extracted in-domain bilingual sentence pairs from large comparable (non-parallel) corpora to enlarge the in-domain bilingual corpus. Koehn and Schroeder (2007) integrated in-domain and out-of-domain language models as log-linear features in the Moses (Koehn et al., 2007) phrase-based SMT system with multiple decoding paths for combining multiple domain translation tables. Nakov (2008) combined in-domain translation and reordering models with out-of-domain models into Moses. Finch and Sumita (2008) employed a probabilistic mixture model combining two models for questions and declarative sentences with a general model. They used a probabilistic classifier to determine a vector of probability representing class membership.

In general, all approaches to domain adaptation of SMT depend on the availability of domain-specific data. If the data is available, it can be directly used to improve components of the MT system. Otherwise, it can be extracted from a pool of texts from different domains or even from the web, which is also the case in our work.

3 Resources and their acquisition

In this section, we review the existing resources we used for training the general-domain systems and present the acquisition procedures of in-domain data used for domain adaptation of these systems.

3.1 Existing general domain data

For the baseline, a general-domain system, we exploited the widely used data provided for the SMT workshops (WPT 2005 – WMT 2010): the Europarl parallel corpus (Koehn, 2005) as training data for translation and language models, and WPT 2005 development and test sets as development and test data for general-domain parameter optimization and testing, respectively (Table 1). Europarl is extracted from the European Parliament proceedings and for practical reasons we consider this corpus to contain general-domain texts.

		initial phase				main phase						
language	*dom*	*sites*	*pages stored /*	*sampled /*	*acc (%)*	*sites*	*pages visited*	*/ stored*	*(Δ%)*	*/ dedup*	*(Δ%)*	*t (h)*
English	*env*	146	505	224	92.9	3,181	90,240	34,572	*38.3*	28,071	*18.8*	47
	lab	150	461	215	91.6	1,614	121,895	22,281	*18.3*	15,197	*31.8*	50
French	*env*	106	543	232	95.7	2,016	160,059	35,488	*22.2*	23,514	*33.7*	67
	lab	64	839	268	98.1	1,404	186,748	45,660	*27.2*	26,675	*41.6*	72
Greek	*env*	112	524	227	97.4	1,104	113,737	31,524	*27.7*	16,073	*49.0*	48
	lab	117	481	219	88.1	660	97,847	19,474	*19.9*	7,124	*63.4*	38
Average					94.0				*25.6*		*39.7*	

Table 2: Statistics from the initial (focused on domain-classification accuracy estimation) and main phases of crawling monolingual data: *stored* refers to the *visited* pages classified as in-domain, *dedup* refers to pages after near-duplicate removal, *time* is the total duration (in hours), *acc* is accuracy estimated on the *sampled* pages, Δ refers to reduction w.r.t. *pages visited*.

language	*dom*	*paragraphs all*	*/ clean*	*(Δ%)*	*/ unique*	*(Δ%)*	*sentences*	*tokens*	*vocabulary*
English	*env*	5,841,059	1,088,660	*18.6*	693,971	*11.9*	1,700,436	44,853,229	225,650
	lab	3,447,451	896,369	*26.0*	609,696	*17.7*	1,407,448	43,726,781	136,678
French	*env*	4,440,033	1,069,889	*24.1*	666,553	*15.0*	1,235,107	42,780,009	246,177
	lab	5,623,427	1,382,420	*24.6*	822,201	*14.6*	1,232,707	46,992,912	180,628
Greek	*env*	3,023,295	672,763	*22.3*	352,017	*11.6*	655,353	20,253,160	324,544
	lab	2,176,571	521,109	*23.9*	284,872	*13.1*	521,358	15,583,737	273,602
Average				*23.3*		*14.0*			

Table 3: Statistics from the cleaning stage of the monolingual data acquisition procedure and of the final data set: *clean* refers to paragraphs classified as non-boilerplate, *unique* to those kept after duplicate removal, Δ to reduction w.r.t. *paragraphs all*.

3.2 Web-crawling for monolingual data

To acquire monolingual in-domain corpora used in improving language models, we enhanced a workflow described in Pecina et al. (2011). Considering the small size of crawled data in that work (repeated here as col. 3–6 in Table 2), we implemented a focused monolingual crawler that adopts a distributed computing architecture based on Bixo (2011), an open source web mining toolkit. Moreover, an out-link relevance score l was calculated as: $l = p/N + \sum_{i=1}^{M} n_i \cdot w_i$, where p is the relevance score of its source page as in Pecina et al. (2011), N is the amount of links originating from the source page, M is the number of entries in a domain definition consisting of relevant terms extracted from Eurovoc[2], n_i denotes the number of occurrences of the i-th term in the surrounding text and w_i is the weight of the i-th term. Further processing steps include boilerplate detection and language identification at paragraph level. These enhancements resulted in acquiring much more in-domain data (col. 8 in Table 2). In addition, the evolutions of the crawls were satisfactory since the ratio of pages classified as in-domain with the visited ones is 25.6% on average (col. 9 in Table 2).

Then, near-duplicates were removed by employing the deduplication strategy included in the Nutch framework[3]. The relatively high percentages of documents removed (col. 13 in Table 2) are in accordance with Baroni et al.'s (2009) observation that during building of the Wacky corpora the amount of documents was reduced by more than 50% after deduplication. Another observation is that the percentages of duplicates for the *lab* domain are much higher than the ones for *env*. This can be explained by the fact that *lab* web pages are mainly legal documents or press releases replicated on many websites.

Final processing of the monolingual data (see Table 3) concerned the exclusion of paragraphs annotated as not in the targeted language or as boilerplate, which reduced their total amount to 23.3% on average (col. 5). Removal of duplicate paragraphs then reduced their total number to 14.0% on average (col. 7). However, most of the removed paragraphs were very short chunks of text (such as navigation links). In terms of tokens, the reduction is only to 50.6%. The last three columns in Table 3 refer to the final monolingual data sets used for training language models. For *EN* and *FR*, we acquired about 45 million tokens for each domain; for *EL*, which is less frequent on the web, we obtained only about 15–20 million tokens.

3.3 Web-crawling for parallel data

Some steps involved in parallel data acquisition (including language identification and cleaning) were discussed in the previous subsection as a part of the monolingual data acquisition. To guide the focused bilingual crawler we used sets of bilin-

[2]http://eurovoc.europa.eu/

[3]http://nutch.apache.org

language pair	*dom*	*sites*	*docs*	*sentences all*	*/ paired*	*(Δ%)*	*/ good*	*(Δ%)*	*/ unique*	*(Δ%)*	*/ sampled*	*/ corrected*
English–French	*env*	6	559	19,042	14,881	*78.1*	14,079	*73.9*	13,840	*72.7*	3,600	3,392
	lab	4	900	35,870	31,541	*87.9*	27,601	*76.9*	23,861	*66.5*	3,600	3,411
English–Greek	*env*	14	288	17,033	14,846	*87.2*	14,028	*82.4*	13,253	*77.8*	3,600	3,000
	lab	7	203	13,169	11,006	*83.6*	9,904	*75.2*	9,764	*74.1*	2,700	2,506
Average						*84.2*		*77.1*		*72.8*		

Table 4: Statistics from the parallel data acquisition: document pairs (*docs*), source sentences (*sentences all*), aligned sentence pairs (*paired*), those of sufficient translation quality (*good*); after duplicate removal (*unique*); sentences randomly selected for manual correction (*sampled*) and those really corrected (*corrected*). Δ always refers to percentages w.r.t. the previous step.

gual topic definitions. In order to construct the list of seed URLs we selected web pages that were collected during the monolingual crawls and originated from in-domain multilingual web sites. Since it is likely that these multilingual sites contain parallel documents, we initialize the crawler with these seed URLs and force the crawler to follow only links internal to these sites. After downloading in-domain pages from the selected web sites, we employed Bitextor to identify pairs of documents that could be considered parallel.

3.4 Parallel sentence extraction

After identification of parallel documents, the next steps aimed at extraction of parallel sentences. For each document pair free of boilerplate paragraphs, we applied these steps: sentence splitting and tokenization by the Europarl tools, and sentence alignment by Hunalign (Varga et al., 2005). Hunalign implements a heuristic, language--independent method for identification of parallel sentences in parallel texts which can be improved by providing an external bilingual dictionary of word forms. Without having such dictionaries for *EN–FR* and *EN–EL* at hand, we realign data in these languages from Europarl by Hunalign and used the dictionaries produced by this tool.

For each sentence pair identified as parallel, Hunalign provides a confidence score which reflects the level of parallelness. We manually investigated a sample of sentence pairs extracted by Hunalign from the pool data (about 50 sentence pairs for each language pair and domain), by relying on the judgement of native speakers, and estimated that sentence pairs with a score above 0.4 are of a good translation quality. We kept sentence pairs with 1:1 alignment only (one sentence on each side) and removed those with scores below this threshold. Finally, we also removed duplicate sentence pairs.

The statistics from the parallel data acquisition procedure are given in Table 4. On average, 84.2% of the source sentences extracted from the parallel documents were aligned in the 1:1 fashion (col. 7), 10% of them were removed due to low translation quality, and after discarding duplicate sentences pairs we acquired 72.8% of the original source sentences aligned to their target sides (col. 11).

The translation quality of the parallel sentences obtained by the procedure described above is not guaranteed in any sense. Tuning the procedure and focusing on high-quality translations is possible but leads to a trade-off between quality and quantity. For translation model training, high translation quality of the data is not as essential as for testing. Bad phrase pairs can be removed from the translation tables based on their low translation probabilities. However, a development set containing sentence pairs which are not good translations of each other might lead to sub-optimal values of model weights which would harm system performance. If such sentence pairs are used in the test set, the evaluation would clearly be unreliable.

In order to create reliable test and development sets for each language pair and domain, we performed the following low-cost procedure. From the data obtained by the steps described in the previous section, we selected a random sample of 3,600 sentence pairs (2,700 for *EN–EL* in the *lab* domain, for which less data was available) and asked native speakers to check and correct them. The task consisted of checking that the sentence pairs belonged to the right domain, the sentences within a sentence pair were equivalent in terms of content, and the translation quality was adequate and (if needed) correcting it. The goal was to obtain at least 3,000 correct sentence pairs for each domain and language pair; thus the correctors did not have to correct every sentence pair. They were allowed to skip (remove) misaligned sentence pairs and asked to remove those sentence pairs that were obviously from a very different domain (despite being correct translations). The number of corrected sentences is in the last column of Table 4.

According to the human judgements (see Table 5), 53–72% of sentence pairs were accurate translations, 22–34% needed only minor corrections, 1–

category	*EN–EL / env*	*EN–FR / lab*
1. perfect translation	53.49	72.23
2. minor corrections done	34.15	21.99
3. major corrections needed	3.00	0.33
4. misaligned sentence pair	5.09	1.58
5. wrong domain	4.28	3.86

Table 5: Results (%) of the manual correction of parallel data.

3% would require major corrections (which was not necessary, as the accurate sentence pairs together with those requiring minor corrections were enough to reach our goal of at least 3,000 sentence pairs in most cases), 2–5% of sentence pairs were misaligned and would have had to be translated completely, and about 4% were from a different domain (despite being correct translations).

Further, we selected 2,000 pairs from the corrected sentences for the test set and left the remaining part for the development set. The parallel sentences which were not selected for corrections were used as training sets. See further statistics in Table 6. The correctors confirmed that the manual corrections were about 5–10 times faster than translating the sentences from scratch, so this can be viewed as low-cost method for acquiring in-domain test and development sets for SMT.

4 Domain adaptation experiments

In this section, we present experiments that exploit all the acquired in-domain data in eight different evaluation scenarios involving two domains (*env*, *lab*) and two language pairs (*EN–FR*, *EN–EL*) in both directions. Our primary evaluation measure is BLEU (Papineni et al., 2002). For detailed analysis we also present NIST (Doddington, 2002) and METEOR (Banerjee and Lavie, 2005) in Table 8.

4.1 System description

Our MT system is based on Moses (Koehn et al., 2007). For training the baseline system, training data is tokenized and lowercased using the Europarl tools. The original (non-lowercased) target sides of the parallel data are kept for training the Moses recaser. The lowercased versions of the target sides are used for training an interpolated 5--gram language model with Kneser-Ney discounting using the SRILM toolkit (Stolcke, 2002). Translation models are trained on the relevant parts of the Europarl corpus, lowercased and filtered on sentence level; we kept all sentence pairs having less than 100 words on each side and with length ratio within the interval $\langle 0.11, 9.0 \rangle$. The maximum

pair	*dom*	*set*	*sents*	*L1 tokens*	*/ voc*	*L2 tokens*	*/ voc*
English–French	*env*	train	10,240	300,760	10,963	362,899	14,209
		dev	1,392	41,382	4,660	49,657	5,542
		test	2,000	58,865	5,483	70,740	6,617
	lab	train	20,261	709,893	12,746	836,634	17,139
		dev	1,411	52,156	4,478	61,191	5,535
		test	2,000	71,688	5,277	84,397	6,630
English–Greek	*env*	train	9,653	240,822	10,932	267,742	20,185
		dev	1,000	27,865	3,586	30,510	5,467
		test	2,000	58,073	4,893	63,551	8,229
	lab	train	7,064	233,145	7,136	244,396	14,456
		dev	506	15,129	2,227	16,089	3,333
		test	2,000	62,953	4,022	66,770	7,056

Table 6: Details of the in-domain parallel data sets obtained by web-crawling and manual correction: sentence pairs (*sents*), source (*L1*) and target (*L2*) tokens and vocabulary size (*voc*).

length of aligned phrases is set to 7 and the reordering models are generated using parameters: *distance*, *orientation-bidirectional-fe*. The model parameters are optimized by Minimum Error Rate Training (Och, 2003, MERT) on development sets.

For decoding, test sentences are tokenized, lowercased, and translated by the tuned system. Letter casing is then reconstructed by the recaser and extra blank spaces in the tokenized text are removed in order to produce human-readable text.

4.2 Using out-of-domain test data

A number of previous experiments (Wu et al., 2008; Banerjee et al., 2010, e.g.) showed significant degradation of translation quality if an SMT system was applied to out-of-domain data. In order to verify this observation we trained and tuned our system on general-domain data and compared its performance on test sets from general (*gen*) and specific (*env*, *lab*) domains (the results are referred to as *vX* and *v0* in Table 7, respectively). The average decrease in BLEU is 44.3%: while on general--domain test sets we observe scores in the interval 42.24–57.00, the scores on the specific-domain test sets are in the range 20.20–31.79. This is presumably caused by the divergence of training and test data: the out-of-vocabulary (OOV) rate increased from 0.25% to 0.90% (see col. 4 and 16 in Table 7).

4.3 Using in-domain development data

Optimization of parameters of the SMT log-linear models is known to have a big influence on the performance. The first step towards domain adaptation of a general-domain system it to use in--domain development data. Such data usually comprises of a small set of parallel sentences which are repeatedly translated while the model parameters are adjusted towards their optimal val-

direction	*dom*	*vX*	*/ OOV*	*dom*	*v0*	*/ OOV*	*v1*	*/ Δ%*	*v2*	*/ Δ%*	*v3*	*/ Δ%*	*v4*	*/ Δ%*	*/ OOV*
English–Fench	*gen*	49.12	0.11	*env*	28.03	0.98	35.81	*27.8*	39.23	*40.0*	40.53	*44.6*	40.72	*45.3*	0.65
				lab	22.26	0.85	30.84	*35.6*	34.00	*52.7*	39.55	*77.7*	39.35	*76.8*	0.48
Fench–English	*gen*	57.00	0.11	*env*	31.79	0.81	39.04	*22.5*	40.57	*27.6*	42.23	*32.8*	42.17	*32.7*	0.54
				lab	27.00	0.68	33.52	*23.7*	38.07	*41.0*	44.14	*63.5*	43.85	*62.4*	0.38
English–Greek	*gen*	42.24	0.22	*env*	20.20	1.15	26.18	*29.1*	32.06	*58.7*	33.83	*67.5*	34.50	*70.8*	0.82
				lab	22.92	0.47	28.79	*25.7*	33.59	*46.6*	33.54	*46.3*	33.71	*47.1*	0.40
Greek–English	*gen*	44.15	*0.56*	*env*	29.23	*1.53*	34.15	*16.8*	36.93	*26.3*	39.13	*33.9*	39.18	*34.0*	1.20
				lab	31.71	0.69	37.55	*18.4*	40.17	*26.7*	40.44	*27.5*	40.33	*27.2*	0.62
Average			*0.25*			*0.90*		*25.5*		*40.0*		*49.2*		*49.5*	*0.64*

Table 7: BLEU scores from domain adaptation of the baseline general-domain systems (*v0*) by exploiting: corrected devel. data (*v1*), monolingual training data (*v2*), parallel training data (*v3*), both monolingual and parallel training data (*v4*); *vX* refers to the baseline systems applied to general-domain test sets, *OOV* to out-of-vocabulary rates, Δ to relative improvement over *v0*.

ues. The minimum number of development sentences is not strictly given. The only requirement is that the optimization procedure (MERT in our case) must converge, which might not happen if the set is too small. By using the parallel data acquisition procedure (see Section 3.2), we acquired development sets (506–1,411 sentence pairs in each) which proved to be very beneficial: compared to the baseline systems trained and tuned on general-domain data only (*v0*), systems trained on general-domain data and tuned on in-domain data (*v1*) improved BLEU scores by 25.5% on average. Taking into account that the development sets contain only several hundreds of parallel sentences each, such improvement is remarkable (compare columns *v0* and *v1* in Table 7).

4.4 Adding in-domain monolingual data

Improving an SMT system by adding in-domain monolingual training data cannot reduce the relatively high OOV rate observed when general-domain systems were applied on test sets from specific domains. However, such data can improve the language models and contribute to better estimations of probabilities of n-grams consisting of known words. To verify this hypothesis, we trained systems (*v2*) on general-domain parallel training data, in-domain development data, and a concatenation of general-domain and in-domain monolingual data described in Section 3.2.1 (comprising 15–45 million words). Compared to the systems *v1*, the BLEU scores were improved by additional 14.5% absolute on average. In comparison with the baseline systems *v0*, the total increase of BLEU is 40.0% on average. The most substantial improvement over the system *v1* is achieved for translations to Greek (23.0% for *env*, 16.2% for *lab*) despite the smallest size of the monolingual data acquired for this language (Table 3) which is probably due to the complex Greek morphology.

4.5 Adding in-domain parallel training data

Parallel data is essential for building translation models of SMT systems. While a good language model can improve an SMT system by preferring better translation options in given contexts, it has no effect if the translation model offers no translation at all, which is the case for OOV words. In the next experiment, we use in-domain parallel training data acquired as described in Section 3.2.3 (7–20 thousand sentence pairs). First, we trained systems (*v3*) on a concatenation of general-domain and in-domain parallel training data, in-domain development data, and a general-domain monolingual data only which outperformed the previous systems (*v2*) by additional 9.2% absolute on average (49.2% over the baseline). In some scenarios, the overall improvement was above 70%.

To provide a complete picture we also trained fully adapted systems (*v4*) using both general--domain and in-domain sets of parallel and monolingual data and tuned on the corrected in-domain development sets. In most scenarios the difference of results of these systems compared to systems *v3* are not statistically significant (p=0.05). The average relative improvement over the baseline (*v0*) is 49.5%, which is almost identical to 49.2% from the previous experiment (*v3*). In practice, this means that using additional monolingual in-domain data on top of the in-domain parallel data has no effect on the translation quality. Although additional experiments would verify whether larger monolingual data could bring any additional improvement or not, it seems that parallel data is more important.

5 Conclusions

We presented two methods for the acquisition of domain-specific monolingual and parallel data from the web. They employ existing open-source tools for normalization, language identification,

		Natural Environment								*Labour Legislation*							
	sys	*BLEU*	*Δ%*	*NIST*	*Δ%*	*MET*	*Δ%*	*WER*	*Δ%*	*BLEU*	*Δ%*	*NIST*	*Δ%*	*MET*	*Δ%*	*WER*	*Δ%*
English-French	*v0*	28.03	*0.0*	7.03	*0.0*	63.32	*0.0*	63.70	*0.0*	22.26	*0.0*	6.27	*0.0*	56.73	*0.0*	69.93	*0.0*
	v1	35.81	*27.7*	8.10	*15.2*	68.44	*8.0*	53.78	*-15.5*	30.84	*38.5*	7.42	*18.3*	62.94	*10.9*	57.99	*-17.0*
	v2	39.23	*39.9*	8.43	*19.9*	70.35	*11.1*	51.34	*-19.4*	34.00	*52.7*	7.68	*22.4*	65.56	*15.5*	57.06	*-18.4*
	v3	40.53	*44.6*	8.61	*22.4*	71.10	*12.2*	50.04	*-21.4*	39.55	*77.6*	8.37	*33.4*	69.82	*23.0*	52.04	*-25.5*
	v4	40.72	*45.2*	8.63	*22.7*	71.23	*12.4*	49.92	*-21.6*	39.35	*76.7*	8.34	*33.0*	69.79	*23.0*	52.29	*-25.2*
French-English	*v0*	31.79	*0.0*	7.77	*0.0*	66.25	*0.0*	57.09	*0.0*	27.00	*0.0*	7.07	*0.0*	59.90	*0.0*	61.57	*0.0*
	v1	39.04	*22.8*	8.75	*12.6*	69.17	*4.4*	48.26	*-15.4*	33.52	*24.1*	7.98	*12.8*	63.70	*6.3*	53.39	*-13.2*
	v2	40.57	*27.6*	8.90	*14.5*	70.23	*6.0*	47.19	*-17.3*	38.07	*41.0*	8.47	*19.8*	66.88	*11.6*	50.35	*-18.2*
	v3	42.23	*32.8*	9.09	*16.9*	71.40	*7.7*	46.07	*-19.3*	44.14	*63.4*	9.22	*30.4*	71.24	*18.9*	45.49	*-26.1*
	v4	42.17	*32.6*	9.09	*16.9*	71.32	*7.6*	46.05	*-19.3*	43.85	*62.4*	9.17	*29.7*	71.07	*18.6*	45.81	*-25.6*
English-Greek	*v0*	20.20	*0.0*	5.73	*0.0*	82.81	*0.0*	67.83	*0.0*	22.92	*0.0*	5.93	*0.0*	87.27	*0.0*	65.88	*0.0*
	v1	26.18	*29.6*	6.57	*14.6*	84.19	*1.6*	60.80	*-10.3*	28.79	*25.6*	6.80	*14.6*	87.91	*0.7*	58.20	*-11.6*
	v2	32.06	*58.7*	7.24	*26.3*	84.52	*2.0*	56.68	*-16.4*	33.59	*46.5*	7.36	*24.1*	88.34	*1.2*	54.71	*-16.9*
	v3	33.83	*67.4*	7.63	*33.1*	86.10	*3.9*	53.47	*-21.1*	33.54	*46.3*	7.34	*23.7*	89.55	*2.6*	54.68	*-17.0*
	v4	34.50	*70.7*	7.57	*32.1*	85.91	*3.7*	54.16	*-20.1*	33.71	*47.0*	7.34	*23.7*	89.42	*2.4*	54.71	*-16.9*
Greek-English	*v0*	29.23	*0.0*	7.50	*0.0*	60.57	*0.0*	54.69	*0.0*	31.71	*0.0*	7.76	*0.0*	62.42	*0.0*	52.34	*0.0*
	v1	34.16	*16.8*	8.01	*6.8*	64.98	*7.2*	51.15	*-6.4*	37.55	*18.4*	8.28	*6.7*	67.36	*7.9*	49.02	*-6.3*
	v2	36.93	*26.3*	8.27	*10.2*	66.60	*9.9*	49.40	*-9.6*	40.17	*26.6*	8.58	*10.5*	68.67	*10.0*	47.03	*-10.1*
	v3	39.13	*33.8*	8.55	*14.0*	68.24	*12.6*	47.94	*-12.3*	40.44	*27.5*	8.61	*10.9*	68.91	*10.4*	46.78	*-10.6*
	v4	39.18	*34.0*	8.54	*13.8*	68.19	*12.5*	47.94	*-12.3*	40.33	*27.1*	8.60	*10.8*	68.83	*10.2*	47.00	*-10.2*

Table 8: Complete results of the domain adaptation experiments. With the exception of NIST, all scores are percentages; MET denotes METEOR, system identifiers refer to those in Table 7, and Δ to relative improvement over the baseline systems *v0*.

cleaning, deduplication, and parallel sentence extraction. These methods were applied to acquire monolingual and parallel data for two language pairs and two domains with only minimal manual intervention (domain definitions and seed URLs).

The acquired resources were then successfully used to adapt general-domain SMT systems to the new domains. The average relative improvement of BLEU achieved in eight scenarios was a substantial 49.5%. Based on our experiments we made the following observations: even small amounts of in-domain parallel data is more important for translation quality than large amounts of in-domain monolingual data. As few as 500–1,000 sentence pairs can be used as development data with expected 25% relative improvement of BLEU. Additional parallel data can be used to improve translation models: 7,000–20,000 sentences pairs in our experiments increased BLEU by other 25% relative on average. If such data is not available, a general-domain system can benefit from using additional in-domain monolingual data, however quite large amounts (tens of million words) are necessary to obtain a moderate improvement.

Acknowledgments

This research was supported by the EU FP7 project PANACEA (contract no. 7FP-ITC-248064) and by the Czech Science Foundation (grant no. P103/12/G084). We thank Victoria Arranz, Olivier Hamon, and Khalid Choukri for their help with manual correction of the *EN–FR* data; Maria Giagkou and Voula Giouli for construction of the domain definitions and correction of the *EN–EL* data.

References

Banerjee, S. and A. Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp 65–72, Ann Arbor, Michigan.

Banerjee, P., J. Du, B. Li, S. Naskar, A. Way, and J. van Genabith. 2010. Combining Multi-Domain Statistical Machine Translation Models using Automatic Classifiers. In *The Ninth Conference of the Association for MT in the Americas*, pp 141–150.

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

Bixo. 2011. Web mining toolkit. http://openbixo.org/.

Chen, J., R. Chau, and C.-H. Yeh. 2004. Discovering parallel text from the World Wide Web. In *Proc. of the 2nd workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation*, volume 32, pp 157–161, Darlinghurst, Australia.

Cho, J., H. Garcia-Molina, and L. Page. 1998. Efficient crawling through URL ordering. *Comput. Netw. ISDN Syst.*, 30:161–172.

Désilets, A., B. Farley, M. Stojanovic, and G. Patenaude. 2008. WeBiText: Building Large Heterogeneous Translation Memories from Parallel Web Content. In *Proc. of Translating and the Computer (30)*, London, UK.

Doddington, G. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proc. of the second international conference on Human Language Technology Research*, pp 138–145, San Diego, California.

Eck, M., S. Vogel, and A. Waibel. 2004. Language Model Adaptation for Statistical Machine Translation based on Information Retrieval. In *International Conference on Language Resources and Evaluation*, Lisbon, Portugal.

Esplà-Gomis, M. and M. L. Forcada. 2010. Combining Content-Based and URL-Based Heuristics to Harvest Aligned Bitexts from Multilingual Sites with Bitextor. *The Prague Bulletin of Mathemathical Linguistics*, 93:77–86.

Finch, A. and E. Sumita. 2008. Dynamic model interpolation for statistical machine translation. In *Proc. of the Third Workshop on Statistical Machine Translation*, pp 208–215, Columbus, Ohio, USA.

Flournoy, R. and C. Duran. 2009. Machine translation and document localization at Adobe: from pilot to production. In *MT Summit XII: proc. of the twelfth Machine Translation Summit*, pp 425–428.

Gao, Z., Y. Du, L. Yi, Y. Yang, and Q. Peng. 2010. Focused Web Crawling Based on Incremental Learning. *Journal of Comp. Information Systems*, 6:9–16.

Hildebrand, A. S., M. Eck, S. Vogel, and A. Waibel. 2005. Adaptation of the Translation Model for Statistical Machine Translation based on Information Retrieval. In *Proc. of the 10th Annual Conference of the European Association for Machine Translation*, pp 133–142, Budapest, Hungary.

Hua, W., W. Haifeng, and L. Zhanyi. 2005. Alignment model adaptation for domain-specific word alignment. In *43rd Annual Meeting on Association for Computational Linguistics*, pp 467–474, Ann Arbor, Michigan, USA.

Koehn, P. and J. Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proc. of the Second Workshop on Statistical Machine Translation*, pp 224–227, Prague, Czech Rep.

Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proc. of the 45th Annual Meeting of the ACL on Interactive Poster and Demo Sessions*, pp 177–180, Prague, Czech Rep.

Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proc.: the tenth Machine Translation Summit*, pp 79–86, Phuket, Thailand.

Kohlschütter, C., P. Fankhauser, and W. Nejdl. 2010. Boilerplate detection using shallow text features. In *Proc. of the 3rd ACM International Conference on Web Search and Data Mining*, pp 441–450, NY.

Langlais, P. 2002. Improving a general-purpose Statistical Translation Engine by terminological lexicons. In *COLING-02 on COMPUTERM 2002: second international workshop on computational terminology - Volume 14*, pp 1–7, Taipei, Taiwan.

Menczer, F. and R. K. Belew. 2000. Adaptive Retrieval Agents: Internalizing Local Contextand Scaling up to the Web. *Machine Learning*, 39:203–242.

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

Nakov, P. 2008. Improving English-Spanish statistical machine translation: experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing. In *Proc. of the Third Workshop on Statistical Machine Translation*, pp 147–150, Columbus, USA.

Och, F. J. 2003. Minimum error rate training in statistical machine translation. In *41st Annual Meeting on Association for Computational Linguistics*, pp 160–167, Sapporo, Japan.

Papineni, K., S. Roukos, T. Ward, and W.-J. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *40th Annual Meeting on Association for Computational Linguistics*, pp 311–318, Philadelphia, USA.

Pecina, P., A. Toral, A. Way, V. Papavassiliou, P. Prokopidis, and M. Giagkou. 2011. Towards Using Web-Crawled Data for Domain Adaptation in Statistical Machine Translation. In *Proc. of the 15th Annual Conference of the European Associtation for Machine Translation*, pp 297–304, Leuven, Belgium.

Qi, X. and B. D. Davison. 2009. Web page classification: Features and algorithms. *ACM Computing Surveys*, 41:12:1–12:31.

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

Stolcke, A. 2002. SRILM-an extensible language modeling toolkit. In *Proc. of International Conference on Spoken Language Processing*, pp 257–286, Denver, Colorado, USA.

Varga, D., L. Németh, P. Halácsy, A. Kornai, V. Trón, and V. Nagy. 2005. Parallel corpora for medium density languages. In *Recent Advances in Natural Language Processing*, pp 590–596.

Wu, H., H. Wang, and C. Zong. 2008. Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proc. of the 22nd International Conference on Computational Linguistics - Volume 1*, pp 993–1000.

Yu, H., J. Han, and K. C.-C. Chang. 2004. PEBL: Web Page Classification without Negative Examples. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):70–81.

Zhang, Y., K. Wu, J. Gao, and P. Vines. 2006. Automatic Acquisition of Chinese-English Parallel Corpus from the Web. In *Proc. of the 28th European Conference on Information Retrieval*, pp 420–431.