Simple and Effective Parameter Tuning for Domain Adaptation of Statistical Machine Translation

Pavel Pecina[1], Antonio Toral[2], Josef van Genabith[2]
(1) Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
(2) Centre for Next Generation Localisation, School of Computing,
Dublin City University, Ireland
pecina@ufal.mff.cuni.cz, atoral@computing.dcu.ie, josef@computing.dcu.ie

ABSTRACT
Current state-of-the-art Statistical Machine Translation systems are based on log-linear models that combine a set of feature functions to score translation hypotheses during decoding. The models are parametrized by a vector of weights usually optimized on a set of sentences and their reference translations, called development data. In this paper, we explore a (common and industry relevant) scenario where a system trained and tuned on general domain data needs to be adapted to a specific domain for which no or only very limited in-domain bilingual data is available. It turns out that systems can be adapted successfully by re-tuning model parameters using surprisingly small amounts of parallel in-domain data, by cross-tuning or no tuning at all. We show in detail how and why this is effective, compare the approaches and effort involved. We also study the effect of hyperparameters (such as maximum phrase length and development data size) and their optimal values in this scenario.

TITLE AND ABSTRACT IN CZECH

Jednoduchá a efektivní optimalizace parametrů pro doménovou adaptaci statistického strojového překladu

Současné systémy statistického strojového překladu jsou založeny na logarotmicko-lineárních modelech, které ve fázi dekódování kombinují příznakové funkce pro hodnocení překladových hypotéz. Tyto modely jsou parametrizovány vektorem vah, které se optimalizují na tzv. vývojových datech, což je množina vět a jejich referenčních překladů. V článku se zabýváme (častou a pro průmyslové nasazení relevantní) situací, kdy je třeba překladový systém natrénovaný na datech z obecné domény adaptovat na nějakou specifickou doménu, pro kterou jsou k dispozici paralelní data jen ve velice omezeném (či žádném) množství. Ukazujeme, že takové systémy mohou být vhodně adaptovány pomocí optimalizace parametrů, a to za použité jen překvapivě malého množství paralelních doménově-specifických dat, či tzv. křížovou optimalizací, nebo bez použití optimalizace vůbec. Toto pozorování důkladně analyzujeme, porovnáváme použité přístupy a jejich celkovou náročnost. Dále se zabýváme analýzou hyperparametrů (např. maximální délkou frází a velikostí vývojových dat) a jejich optimalizací.

KEYWORDS: machine translation, domain adaptation, parameter optimization.

KEYWORDS IN CZECH: strojový překlad, doménová adaptace, optimalizace parametrů.

1 Introduction

Statistical Machine Translation (SMT) is an instance of a machine learning application and, in general, will work best if the data for training and testing are drawn from the same distribution (i.e. domain, genre, and style). In practice, however, it is often difficult to obtain sufficient amounts of in-domain data (in particular parallel data required for translation and distortion models) to train a well performing system for a specific domain.

Recently, Pecina et al. (2011) showed that just using in-domain development data for parameter tuning improves output quality of a Phrase-Based SMT (PB-SMT) system trained on general--domain data but applied to a specific domain. Although further additional improvements can be realized by using in-domain parallel and/or monolingual training data, parameter tuning on in-domain data requires only a relatively small set of parallel sentences, which is often easier to obtain. They report on a series of experiments carried out on the domains of Natural Environment (*env*) and Labour Legislation (*lab*) and two language pairs: English–French and English–Greek (both directions) and observe a substantial average relative improvement of 25% in terms of BLEU (Papineni et al., 2002) when switching from general-domain to in-domain tuning.

In this paper, we corroborate the results reported by Pecina et al. (2011), carrying out similar experiments on the domain of medical texts (*med*). In contrast to earlier work, we explain the improvements brought about by specific domain tuning by analysing the results in detail. In a nutshell: domain tuning for matching-domain training, tuning and test data results in feature vectors that trust (often long) translation table entries, while tuning with and for specific domains (while using generic training data) allows the MT system to stitch together translations from smaller bits and pieces with significantly more reordering, effectively undoing or "de--tuning" any previous optimizations. In a sense, this is natural: substantial divergence between test and training data means that in particular long and potentially high quality phrase pairs obtained in training may no longer be applicable to the test data and that this divergence can only be bridged by smaller translation units and more flexible recombination. Furthermore, our findings show that in the general-domain training and specific-domain test scenario, approaches that do not perform any parameter tuning (at all) or that tune on other specific development sets may in fact fare better than tuning on general-domain data.

In addition, there is a question of how much specific-domain tuning data is in fact required to "de-tune" a general domain system to a specific domain. Finally, given the fact that a general--domain system can only use limited length translation units when translating specific-domain data, we explore limited length training and decoding.

After a brief overview of the log-linear model including its parameter optimization and an overview of the state-of-the-art in domain adaptation for SMT, we describe our experiments, present the results, the analysis, explore the resulting research questions with additional experiments, and conclude.

2 Phrase-Based Statistical Machine Translation

In PB-SMT, implemented e.g. in Moses (Koehn et al., 2007), an input sentence is segmented into sequences of consecutive words, which are called phrases. Each phrase is then translated into a target language phrase, which may be reordered with other translated phrases to produce the output.

Formally, the model is based on the noisy channel model. The translation **e** of an input sentence **f** is searched for by maximizing the translation probability $p(\mathbf{e}|\mathbf{f})$ formulated as a log-linear combination of a set of feature functions h_i and their weights λ_i:

$$p(\mathbf{e}|\mathbf{f}) = \prod_i^n h_i(\mathbf{e}, \mathbf{f})^{\lambda_i}$$

Typically, the components include features of the following models: *phrase translation model*, which ensures that the source and target phrases are good translations of each other (e.g. direct and inverse phrase translation probability, direct and indirect lexical weighting, and phrase penalty), *language model*, which ensures that the translations are fluent, *reordering* (*distortion*) *model*, which allows to reorder phrases in the input sentences (e.g. distance-based and lexicalized reordering) and *word penalty*, which prevents the translations from being too long or too short. These models are trained on either parallel or monolingual training data.

The weights of the log-linear combination influence overall translation quality; however, the optimal setting depends on the translation direction and data. A common solution to optimise weights is to use Minimum Error Rate Training (MERT), proposed by Och (2003), which automatically searches for the values that minimize a given error measure (or maximize a given translation quality measure) on a development set of parallel sentences. Theoretically, any automatic measure can be used for this purpose; however, the most commonly used is BLEU (Papineni et al., 2002). The search algorithm is a type of coordinate ascent: considering *n*-best translation hypotheses for each input sentence, it updates the feature weight most likely promising to improve the objective and iterates until convergence. The error surface is highly non-convex and as the algorithm cannot explore the whole parameter space, it may converge to a local maximum; in practise, it often produces good results (Bertoldi et al., 2009).

3 Domain adaptation in Statistical Machine Translation

Domain-adaptation is a very active research topic within the area of SMT. Three main topics can be identified: (i) combination of in-domain and out-of-domain resources for training, (ii) training data selection and (iii) acquisition of specific-domain data. Below we briefly review a selection of relevant work that falls into these topics.

The first attempt to perform domain adaptation was carried out by Langlais (2002), who integrated in-domain lexicons in the translation model. Koehn and Schroeder (2007) integrate in-domain and out-of-domain language models as log-linear features in Moses. Nakov (2008) combines in-domain translation and reordering models with out-of-domain models. Finch and Sumita (2008) use a probabilistic mixture model combining two models for questions and declarative sentences with a general model.

Training data selection is another approach to domain-adaptation. The assumption is that a general-domain corpus, if sufficiently broad, includes sentences that resemble the target domain. Eck et al. (2004) present a technique for adapting the language model by selecting similar sentences from available training data. Hildebrand et al. (2005) extended this approach to the translation model. Foster et al. (2010) weigh phrase pairs from out-of-domain corpora according to their relevance to the target domain.

Munteanu and Marcu (2005) extract in-domain sentence pairs from comparable corpora. Daumé III and Jagarlamudi (2011) attempt to reduce out-of-vocabulary terms when targeting a specific domain by mining their translations from comparable corpora. Bertoldi et al. (2009) rely on large in-domain monolingual data to create synthetic parallel corpora. Pecina et al.

languages (L1-L2)	*dom*	*set*	*sentences*	*L1 tokens*	*/ voc*	*L2 tokens*	*/ voc*
English–French	*gen*	train	1,725,096	47,956,886	73,645	53,262,628	103,436
		dev	2,000	58,655	5,734	67,295	6,913
		test	2,000	57,951	5,649	66,200	6,876
	env	dev	1,392	41,382	4,660	49,657	5,542
		test	2,000	58,865	5,483	70,740	6,617
	lab	dev	1,411	52,156	4,478	61,191	5,535
		test	2,000	71,688	5,277	84,397	6,630
	med	dev	1,064	16,807	3,484	18,932	4,865
		test	2,000	31,725	5,268	34,884	7,331
English–Greek	*gen*	train	964,242	27,446,726	61,497	27,537,853	173,435
		dev	2,000	58,655	5,734	63,349	9,191
		test	2,000	57,951	5,649	62,332	9,037
	env	dev	1,000	27,865	3,586	30,510	5,467
		test	2,000	58,073	4,893	63,551	8,229
	lab	dev	506	15,129	2,227	16,089	3,333
		test	2,000	62,953	4,022	66,770	7,056
	med	dev	1,064	16,807	3,484	20,625	3,893
		test	2,000	31,725	5,268	38,614	5,754

Table 1: Statistics of the training, development and test data sets from the domains used in the experiments including the number of sentence pairs, tokens, and vocabulary size (*voc*).

(2011) exploit automatically web-crawled in-domain resources for parameter optimization and improving language models. (Pecina et al., 2012) extend the work by using the web-crawled resources to also improve translation models.

4 Experimental setup

Our experimental setup follows and extends the one used in Pecina et al. (2011). In addition to the two evaluation domains (*env*, *lab*) used in that work, and in order to corroborate their earlier findings, we also carry out experiments on medical domain data (*med*).

4.1 Data

Our general-domain system is trained on the Europarl parallel corpus (Koehn, 2005, v5) extracted from the proceedings of the European Parliament and for the purposes of this work considered to contain general-domain texts (it covers a very broad range of topics and it is to a considerable extent spoken language). The general-domain development and test data used for parameter optimization and testing, respectively, are adopted from the WPT 2005[1] machine translation shared task. These sets were extracted from the same source as Europarl and contain 2,000 sentence pairs each.

The specific-domain development and test data for the *env* and *lab* domains were acquired by domain-focused web-crawling within the PANACEA project[2] and are available from the ELRA catalogue [3] under reference numbers ELRA-W0057 and ELRA-W0058. The entire acquisition procedure is described in detail in Pecina et al. (2011). The test sets consist of 2,000 sentence pairs each and the amount of sentence pairs in the development sets varies from 506 to 2,000.

[1] http://www.statmt.org/wpt05/
[2] http://www.panacea-lr.eu/
[3] http://catalog.elra.info/

dev	*test*	*English–French*		*French–English*		*English–Greek*		*Greek–English*	
gen	*gen*	49.12	*0.00*	57.00	*0.00*	42.24	*0.00*	44.15	*0.00*
	env	28.03	*−42.94*	31.79	*−44.23*	20.20	*−52.18*	29.23	*−33.79*
	lab	22.26	*−54.68*	27.00	*−52.63*	22.92	*−45.74*	31.71	*−28.18*
	med	12.32	*−74.92*	15.33	*−73.11*	8.96	*−78.79*	14.79	*−66.50*
average			*−57.51*		*−56.65*		*−58.90*		*−42.82*

Table 2: Results (in BLEU) of the systems tuned on general-domain and tested on the specific domains (*env*, *lab*, *med*) compared with the results on the general domain (*gen*); the figures in italics indicate the relative change (in percentage).

The *med* development and test data were extracted from the EMEA parallel corpus of texts from the European Medicines Agency, distributed as a part of the OPUS corpus (Tiedemann, 2009). A set of 3,500 parallel sentences in English, French, and Greek was randomly sampled from the sentence-aligned corpus data and manually checked for translation quality. Correct sentences were left untouched, sentences with minor errors were corrected, and those which required major corrections or were misaligned were discarded completely. We aimed at acquiring at least 3,000 correct sentence pairs: 2,000 for the test sets and the rest for the development sets. Finally, the test and development sets contained 2,000 and 1,064 sentence pairs respectively. All data sets used in our experiments contain one reference translation. Statistics are given in Table 1.

4.2 System description

Our MT system is based on the Moses PB-SMT system (Koehn et al., 2007). For training, all data sets are tokenized and lowercased using the Europarl tools. The original (non-lowercased) target side of the parallel data is kept for training the Moses recaser. The lowercased versions of the target side 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 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 length for aligned phrases is set to 7 and the reordering models are generated using the following parameters: *distance, orientation-bidirectional-fe*. The resulting system combines 14 feature functions, listed below. The corresponding parameters are optimized on the development sets by MERT.

1. distance reordering score
2-7. lexicalised reordering scores
8. language model score
9. inverse phrase translation probability
10. inverse lexical weighting
11. direct phrase translation probability
12. direct lexical weighting
13. phrase penalty
14. word penalty

For decoding, test sentences are tokenized and lowercased. After translation, letter casing is reconstructed by the recaser and extra blank spaces are removed in order to produce human--readable text.

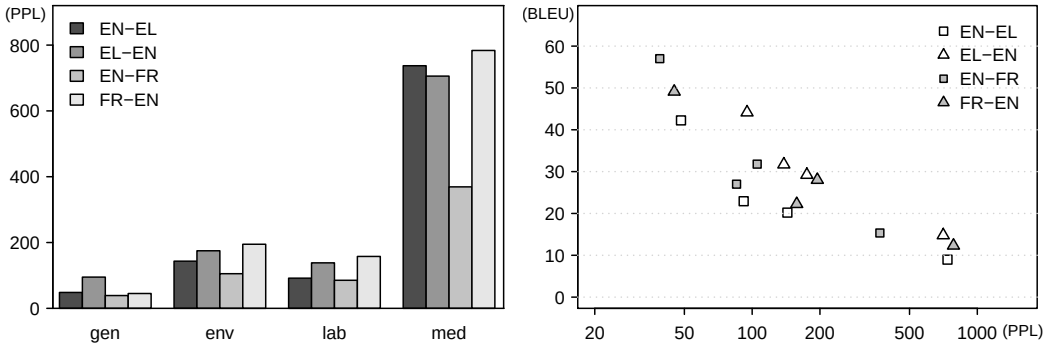


Figure 1: Perplexity (PPL) of the source side of the test sets given the language models trained on the source side of the training sets (left). Perplexity of the source side of the test data versus BLEU scores of the corresponding systems tuned on general-domain development data (right).

5 Experiments

Translation quality in our experiments is automatically evaluated using BLEU (Papineni et al., 2002) and all BLEU scores are reported as percentages.

5.1 Baseline system performance

Performance of the baseline system trained and tuned on the general-domain data and tested on the same domain varies from 42.24 to 57.00 (row 1 in Table 2). Applying the baseline general--domain system on the specific-domain data leads to significant degradation of translation quality (Banerjee et al., 2010; Wu et al., 2008). Pecina et al. (2011) reported an average decrease of 44.3% when the general-domain system was applied to the *env* and *lab* domains (see rows 2–3 in Table 2). Our experiments on the *med* domain show even more pronounced decrease: e.g. in case of the English–French translation, BLEU drops from 49.12 to 12.32; for English–Greek the change is from 42.24 to 8.96; other translation directions produce similar results. The average decrease for all directions on the *med* domain is 73.33% relative – the domain divergence between the training and test data from this domain is even more pronounced than in the case of the other two domains. The average decrease taken over all translation directions and all the domains is 53.97% relative.

5.2 Measuring domain divergence

From the results presented above, it is evident that the translation quality of a particular test set depends on the extent to which its domain differs from the domain of the training data. Quality is maximal when the domains match and decreases when the test data diverges from the training data. To quantify this observation, we measure cross perplexity of the test data given the training data. For each domain and translation direction, a language model of the same order as the maximum phrase length (7) used in the SMT systems is trained on the source side of the training data and applied to the source side of the test data. The results are presented in Figure 1 (left).

As expected, the perplexity of the general domain test sets is the lowest. It ranges from 40 to 90 depending on the language. In case of the *env* and *lab* domains, perplexity is slightly higher: on the *env* data it ranges from 100 to 190 and on the *lab* data from 80 to 160. Not surprisingly,

test	*dev*	*English–French*		*French–English*		*English–Greek*		*Greek–English*	
env	*gen*	28.03	*0.00*	31.79	*0.00*	20.20	*0.00*	29.23	*0.00*
	env	35.81	*+27.76*	39.04	*+22.81*	26.18	*+29.60*	34.16	*+16.87*
lab	*gen*	22.26	*0.00*	27.00	*0.00*	22.92	*0.00*	31.71	*0.00*
	lab	30.84	*+38.54*	33.52	*+24.15*	28.79	*+25.61*	37.55	*+18.42*
med	*gen*	12.32	*0.00*	15.33	*0.00*	8.96	*0.00*	14.79	*0.00*
	med	18.47	*+49.92*	24.42	*+59.30*	14.57	*+62.61*	18.10	*+22.38*
average			*+38.74*		*+35.42*		*+39.28*		*+19.22*

Table 3: The effect (measured by BLEU) of general-domain (*gen*) and in-domain (*env*, *lab*, *med*) tuning. The figures in italics indicate relative improvement (in percentage) obtained from using domain-specifc development data for tuning.

the perplexity scores obtained on the *med* domain are substantially higher; for most language directions they exceed 700. The only exception is the French–English test set, for which the score is as low as 370. This higher drop is consistent across the other domains (compare the yellow bars with other language pairs in Figure 1, left) and in line with the higher decrease for this domain in terms of BLEU (see Section 5.1).

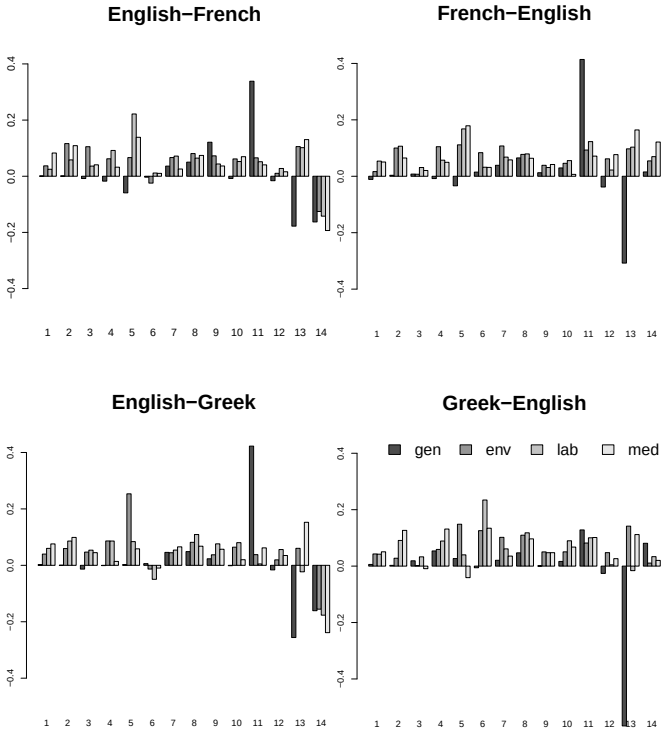
To complete the picture, we directly compare the perplexity scores with the translation quality measured by BLEU and provide a plot in Figure 1 (right). It is quite obvious that the perplexity scores on the logarithmic X axis (PPL) are highly correlated (inversely) with the BLEU scores on the Y axis. Higher perplexity indicates lower translation quality. This finding is in line with previous research on translation confidence estimation (Specia et al., 2011; He et al., 2010).

5.3 Parameter tuning on specific-domain development data

The baseline systems trained and tuned on general-domain data perform much worse on specific domains. Pecina et al. (2011) reported that a surprisingly significant amount of loss can be recovered by tuning on in-domain development data. The average relative improvement measured on the *env* and *lab* domains reported in this work was 25.5%. Our results, including those on the *med* domain, confirm the previous findings (see Table 3). The average relative improvement of BLEU e.g. in English–French translation is 38.74%. Similar improvements are obtained on French–English and English–Greek. Slightly lower improvements were achieved on Greek–English, 19.22% on average. The overall average increase of BLEU is 33.16% relative. Given that the development sets contain only several hundred sentence pairs each, such improvement is remarkable.

5.4 Analysis of model parameters

The only component that changes when the system is tuned on in-domain data are the weights of the feature functions in the log-linear model optimized by MERT. The reordering, language, and translation models all remain untouched (trained on general-domain data). Recall that the parameter space searched through by MERT is large and the error surface highly non-convex, therefore the resulting weight vectors might not be globally optimal and there might be other (i.e. different) weight vectors which perform equally well or even better. For this reason, the actual parameter values are not usually investigated. However, our experiments (Figure 2, left) show that the parameter values and their changes observed when switching from general--domain to specific-domain tuning are in fact highly consistent, indicating interesting trends.



EN–FR	*gen*	*env*	*lab*	*med*
gen	–	0.25	0.12	0.15
env	0.25	–	0.82	0.89
lab	0.12	0.82	–	0.89
med	0.15	0.89	0.89	–

FR–EN	*gen*	*env*	*lab*	*med*
gen	–	0.10	0.15	−0.12
env	0.10	–	0.92	0.86
lab	0.15	0.92	–	0.92
med	−0.12	0.86	0.92	–

EN–EL	*gen*	*env*	*lab*	*med*
gen	–	0.17	0.28	0.18
env	0.17	–	0.81	0.73
lab	0.28	0.81	–	0.76
med	0.18	0.73	0.76	–

EL–EN	*gen*	*env*	*lab*	*med*
gen	–	−0.29	0.19	−0.20
env	−0.29	–	0.72	0.71
lab	0.19	0.72	–	0.81
med	−0.20	0.71	0.81	–

Figure 2: Visualization of model weights of the four systems in the twelve evaluation scenarios; the black bars refer to model weights of the systems tuned on general-domain (*gen*) development sets while the grey bars refer to the model weights of the systems tuned on specific-domain development sets (*env*, *lab*, *med*) (left). Cosine similarity of the system feature vectors (right).

First, we analyse parameters of the systems tuned on the general-domain data (black bars):

1. The high weights assigned to h_{11} (*direct phrase translation probability*) indicate that the phrase pairs in the systems' translation tables apply well to the development data which are from the same domain as the training data; a high reward is given to translation hypotheses consisting of phrases with high translation probability (i.e. good general--domain translations).
2. The low negative weights assigned to h_{13} (*phrase penalty*) imply that the systems prefer hypotheses consisting of fewer but longer phrases.
3. Reordering in the hypotheses is not rewarded (weights of the reordering models h_1–h_7 are assigned values around zero). In some cases (e.g. for English–French), reordering is even slightly penalized (the weights of h_1–h_7 are negative).
4. The weight of h_{14} (*word penalty*) is negative for translations from English and slightly positive for translations to English. This reflects the fact that translation from English prefers shorter hypotheses and translation to English prefers longer hypotheses.

test	*dev*	*EN–FR*	*FR–EN*	*EN–EL*	*EL–EN*	*Average*
gen	*gen*	4.37	3.46	3.76	2.35	3.49
env	*gen*	3.00	2.49	2.69	2.18	2.59
	def	2.33	2.12	2.12	2.03	2.15
	env	2.16	1.77	2.17	1.54	1.91
lab	*gen*	2.82	2.45	2.97	2.43	2.67
	def	2.24	2.09	2.30	2.21	2.21
	lab	2.05	1.83	2.46	2.30	2.16
med	*gen*	2.00	1.71	1.74	1.43	1.72
	def	1.62	1.52	1.47	1.41	1.51
	med	1.54	1.20	1.38	1.21	1.33

Table 4: Average phrase lengths in translations of all test sets (in all directions) by systems tuned on general (*gen*) and specific domains (*env*, *lab*, *med*) and with the default weights (*def*).

Now, we compare these findings with the systems tuned on the specific domains (grey bars).

1. The weights of h_{11} (*direct phrase translation probability*) decrease rapidly, in some scenarios this weight is very close to zero. The translation tables do not provide enough good quality translations for the specific domains and the best translations of the development sentences consist of phrases with varying translation probabilities.
2. Hypotheses consisting of few (and long) phrases are not rewarded anymore (weights of h_{13} are higher); in most cases they are penalized and hypotheses consisting of more (and short) phrases are allowed or even preferred.
3. In almost all cases the reordering feature weights (features h_1–h_7) increased substantially and for specific-domain data the model significantly prefers hypotheses with altered word order (which is consistent with the two preceding observations).
4. Language model weights (h_8) do not change substantially, its importance remains similar on general-domain and specific-domain data.

These findings are highly consistent across domains and language pairs. The weight vectors of the systems tuned on specific-domain data are quite similar but differ substantially from the parameters obtained by tuning on general-domain. This observation can be quantified by measuring cosine similarity (see Figure 2, right) as proposed by Hopkins and May (2011). Lower scores, as in the first rows/columns of each table, indicate low similarity of the vectors – specific-domain tuned weights differ a lot from the general-domain tuned ones; and vice versa – specific-domain tuned parameters are quite similar when compared to each other.

5.5 Analysis of phrase-length distribution

From the analysis presented above, we conclude that a PB-SMT system tuned on data from the same domain as the training data strongly prefers to construct translations consisting of long phrases. Such phrases are usually of good translation quality (local mistakes of word alignment disappear), fluent (formed by consecutive sequences of words), and recurrent (frequent in data from the same domain); therefore they form good translations of the input sentences and are preferred during decoding. This is, of course, a positive behaviour when the system translates sentences from the same domain. However, if this is not the case and the input sentences contain no or very few longer phrases from the translation tables, the system is not able to construct good translations from shorter phrases.

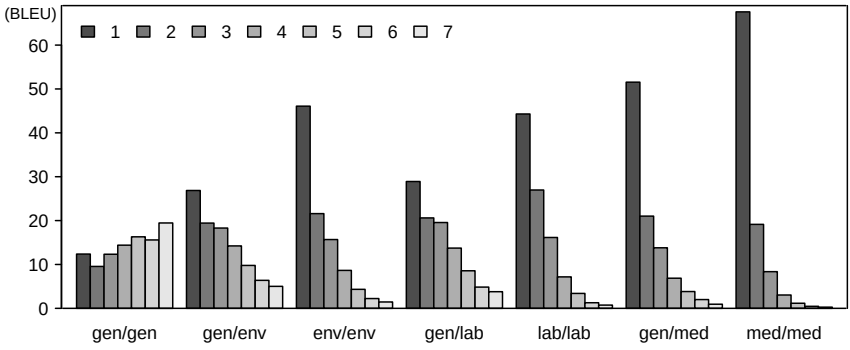


Figure 3: Phrase-length distribution in English–French translations by systems tuned and tested on various combinations of general (*gen*) and specific (*env*, *lab*, *med*) domains.

To support this hypothesis we analyse the phrase length distribution actually seen in the translation of the test sets. The average phrase lengths estimated for various combinations of tuning and test domains and all language pairs are shown in Table 4. The highest values are observed for translations of general-domain test sets by systems tuned on the same domain: 3.49 on average across all language pairs. The scores for systems trained on general and tuned and tested on specific-domain data are significantly lower and range from 1.21 to 3.00, depending on the domain and language pair. Figure 3 presents complete phrase-length distribution in English–French translations by systems tuned and tested on various combinations of general and specific domains. Generally, a higher divergence of the test domain from the training domain leads to shorter phrases being used in translation. However, when the systems tuned on general-domain are applied to specific domains, the average phrase lengths are consistently longer than for specific-domain tuning. The systems are tuned to prefer long phrases (see Table 4) but the translation quality is lower (see Table 3). This situation can be interpreted as overtraining, the model overfits the training (and tuning) data and on different data fails to form the best possible translations (given the translation, reordering, and language models).

5.6 Overfitting reduction

The optimal solution in case of such overfitting is to employ a sufficient amount of specific--domain development data, effectively tuning the system to using shorter phrases (see Figure 3). However, if such tuning data is not available (which is quite a realistic scenario in many applications) we explore the following alternatives: simply side-step parameter tuning (no tuning at all), or tune on a different domain, or use smaller amounts of development data, or reduce the maximum phrase length in decoding. All these methods work surprisingly well and are discussed in the following subsections.

5.6.1 No parameter tuning

Essentially, there are two options how to set the weight vectors without tuning. Either we can use the default weights set by Moses ($h_{1...7} = 0.3$, $h_8 = 0.5$, $h_{9...13} = 0.2$, $h_{14} = -1$) or a flat vector ($h_{1...14} = 1$). We explored both options and the results are given in Table 5 (see the rows denoted *def* and *flat*, respectively, in the development data column). In all scenarios, both options outperform the systems trained and tuned on general-domain data. In some cases (e.g. English-Greek translations in all the specific domains), the results are very close

test	*dev*	*English–French*		*French–English*		*English–Greek*		*Greek–English*	
env	*gen*	28.03	*0.00*	31.79	*0.00*	20.20	*0.00*	29.23	*0.00*
	env	35.81	*+27.76*	**39.04**	***+22.81***	**26.18**	***+29.60***	**34.16**	***+16.87***
	lab	**36.16**	***+29.00***	38.78	*+21.99*	**26.13**	***+29.36***	33.85	*+15.81*
	med	32.40	*+15.59*	36.89	*+16.04*	24.89	*+23.22*	**34.01**	***+16.35***
	def	34.94	*+24.65*	34.05	*+7.11*	**26.09**	***+29.16***	31.33	*+7.18*
	flat	32.22	*+14.95*	37.66	*+18.46*	21.91	*+8.47*	32.84	*+12.35*
lab	*gen*	22.26	*0.00*	27.00	*0.00*	22.92	*0.00*	31.71	*0.00*
	env	30.13	*+35.35*	**33.21**	***+23.00***	28.36	*+23.73*	**37.57**	***+18.48***
	lab	**30.84**	***+38.54***	**33.52**	***+24.15***	**28.79**	***+25.61***	**37.55**	***+18.42***
	med	27.04	*+21.47*	30.77	*+13.96*	26.85	*+17.15*	**37.52**	***+18.32***
	def	29.26	*+31.45*	29.73	*+10.11*	28.48	*+24.26*	34.95	*+10.22*
	flat	27.16	*+22.01*	32.24	*+19.41*	25.13	*+9.64*	35.79	*+12.87*
med	*gen*	12.32	*0.00*	15.33	*0.00*	8.96	*0.00*	14.79	*0.00*
	env	**18.74**	***+52.11***	23.75	*+54.92*	13.89	*+55.02*	**17.88**	***+20.89***
	lab	**18.91**	***+53.49***	23.73	*+54.79*	13.69	*+52.79*	17.62	*+19.13*
	med	18.47	*+49.92*	**24.42**	***+59.30***	**14.57**	***+62.61***	**18.10**	***+22.38***
	def	18.20	+47.73	21.15	*+37.96*	13.82	*+54.24*	16.70	*+12.91*
	flat	17.06	*+38.47*	23.02	*+50.16*	11.99	*+33.82*	17.71	*+19.74*

Table 5: Translation quality (in BLEU) of the general-domain systems tuned and tested on various domains. The figures in italics indicate relative improvement (in percentage) over the system tuned on general domain. The figures in bold denote the best performing combination for each test domain and translation direction and those which are not significantly different (Koehn, 2004, $p = 0.05$).

to those of systems tuned on specific-domain data. The overall average relative improvement of the systems with default parameters over the systems tuned on general domain is 24.75% (compare with 33.16% obtained from specific-domain tuning). The average phrase length in translations produced by such systems falls between the scores of general-domain-tuned and specific-domain-tuned systems (see rows with *def* in the development data column in Table 4). The systems with the flat weight vectors achieve an average relative improvement of 21.70%. However, they outperform the systems with the default parameters always when the translation direction is to English; the systems with the default parameters are better when translating from English.

5.6.2 Cross-domain tuning

It seems that the problem of the overfitted general-domain models and their poor performance on specific domains can be reduced by "diverting" the systems away from the general domain they are tuned to translate – but not necessarily towards a particular specific domain. To analyse this hypothesis we perform "cross-domain" tuning, i.e. tuning on specific domains different from the test domains. The results are shown in Table 5 (see the rows where the test and development domain do not match). In all scenarios the cross-domain tuned system performs better than the un-tuned ones. In a few cases the systems tuned on a cross domain perform even better than the in-domain ones: e.g. the EN–FR system tuned on the *lab* domain and tested on the *env* and *med* domains or the EL–EN system tuned on the *env* domain and tested on the *lab* domain, however, in most such cases the improvement is not statistically significant. The overall average relative gain over the systems tuned on general domain is 27.62% (compare with 24.75% obtained from no tuning and 33.16% from in-domain tuning).

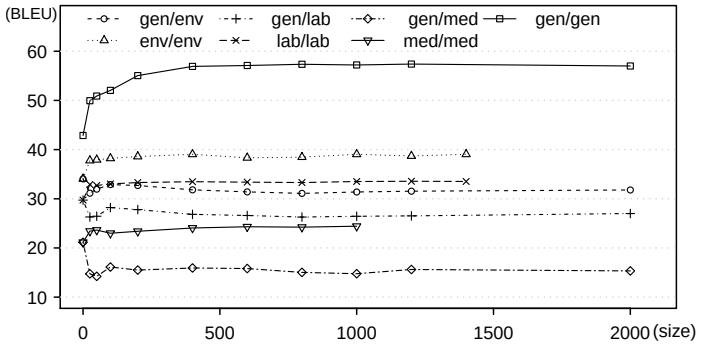


Figure 4: Translation quality (BLEU) of French–English systems tuned on data of varying size.

Similar results were observed also on mixtures of two domains (e.g. tuned on *lab+env* and tested on *med*). In general, we can conclude that cross-domain tuning is a reasonable solution when no in-domain development data is available (and the domains differ in a similar way).

5.6.3 Tuning on small development data

In the previous two scenarios we did not use any specific-domain development data for tuning, but were able to get very close to the performance of the systems tuned on a specific domain. Specific-domain parallel data is scarce, for many domains not available at all and must be prepared by manual translation of monolingual in-domain sentences. We investigate how much development data is needed. The only technical requirement is that MERT, the parameter optimization method, must converge in a reasonable number of iterations. For this reason, typical development sets contain about 1,000 – 2,000 sentence pairs (compare e.g. the size of development sets provided for the WMT[4] translation shared tasks). We vary the amount of sentences in our development sets, tune the systems, test their performance on the test sets and plot learning curves to capture the dependency of translation quality (in terms of BLEU) against gradually increasing the size of development data.

The general shapes of the curves are consistent across all translations (and domains) and thus we provide the curves for the English–French translation direction only (see Figure 4). Increasing the size of development sets is beneficial only in case the domains of development and test data are the same. The curve of the system tuned and tested on the general domain reaches a plateau for about 500 sentence pairs. In case of in-domain tuning for specific domains, the plateau is reached much earlier. Usually, as few as 100–200 sentence pairs are enough to get optimal results. This is encouraging, as tuning on specific-domains yields best results and fortunately requires only very limited amounts of bilingual data (and expense). Development sets of more than 400–600 sentences pairs do not improve translation quality at all and make the tuning process take longer. The systems tuned on the general domain and tested on specific domain do not benefit from the development data at all. The relatively high BLEU scores achieved with no tuning (zero development data size) decrease with increasing size of the development sets.

[4] http://www.statmt.org/wmt12/

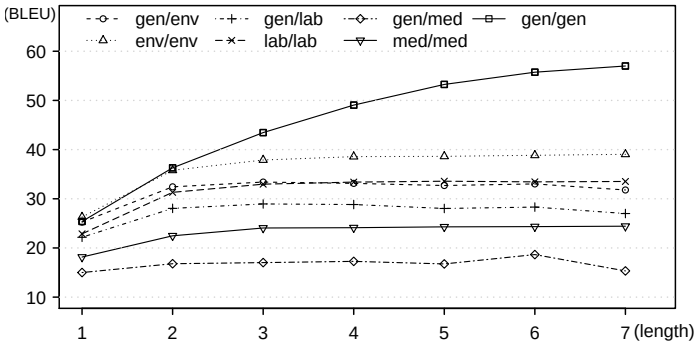


Figure 5: Translation quality (BLEU) of French–English systems with varying max phrase length.

5.6.4 Limiting phrase length

In the last experiment presented in this paper we limit the maximum phrase length allowed during training and decoding and study how system performance changes. The systems tuned on general-domain prefer longer phrases which, however, do not occur frequently in the specific--domain test sets. Our baseline systems, trained and tuned on general domain with maximum phrase length set to seven, translate general-domain test sets with an average phrase length of 3.49 (see Table 4). However, for the systems tuned and tested on in-domain data, this score is as low as 1.80. Figure 5 illustrates how the translation quality changes when the maximum phrase length varies from one to seven. The only case when longer phrases improve translation quality is for the systems trained, tuned and tested on the same (general) domain. In all other cases, the results for phrases up to three words long are as good as for longer phrases. If the domain of the test data does not match the domain of training and tuning data, the maximum phrase length set to three is enough in all scenarios. Longer phrases lead to degradation of translation quality and increase time for training and decoding, as well as memory requirements for building and storing the translation models. A similar result was reported already by Koehn et al. (2003). They observed that limiting the maximum length of a phrase to only three words achieved top performance. However, current state-of-the-art SMT systems usually benefit from longer phrases than three (see e.g. the top curve in Figure 5 which refers to a general-domain system applied to a general-domain test set), and our result applies only to scenarios where the training and test domains do not match; in that case setting the maximum phrase length to three is sufficient.

6 Conclusions

In this work, we have analysed domain adaptation of PB-SMT by tuning parameters of the underlying log-linear model. We confirmed the observation from previous research that systems trained and tuned on general domain perform poorly on specific domains. This finding is not very surprising, but the amount of loss and the fact that it is observed consistently in many evaluation scenarios was unexpected. We found that perplexity of the source side of the test data given the source side of the training nicely correlates with the translation quality.

Further, we confirmed that tuning the systems trained on general domain on specific target domain data recovers a (often) spectacular amount of the loss. We carried out a detailed analysis of the model parameters and phrase length distribution in translations of the test data and found that a system trained and tuned on general domain strongly prefers long and few

phrases in the output translations and therefore underperforms on specific domains where such phrases do not occur frequently. By contrast, the same systems tuned on specific-domain data form output translations from shorter phrases, allow more reordering and perform significantly and consistently better on specific domain data.

We investigated possible solutions for (common) scenarios when no or very little in-domain data is available for parameter tuning. Skipping tuning, i.e. using the default model parameters, performs surprisingly well and always outperforms systems tuned on general domain. Based on this observation, this should be preferred over general domain tuning if the test domain differs substantially. Cross-domain tuning on a different set also offers a good solution when no in-domain development data is available, especially when the domains differ in a similar way (e.g. measured by perplexity). This step has the effect of disassembling the original general-domain system towards shorter phrases and it does not matter much which different development set to use.

The analysis of learning curves of the tuning process showed that in-domain tuning of the general-domain systems requires about 100–200 sentence pairs to achieve decent translation quality (in terms of BLEU, the gain obtained from tuning on more data was negligible). We also experimented with limiting the maximum phrase length of decoding. The results showed that setting this parameter to three is sufficient for translating data from specific domains; longer phrases in this case do not improve translation quality and increase computational requirements of the translation systems. The last two results (limiting phrase length and using sufficient amounts of development data) have efficiency implications of paramount importance in industrial application scenarios.

Acknowledgments

This research was supported by the the Czech Science Foundation (grant no. P103/12/G084), by EU FP7 projects PANACEA (contract no. 248064) and Khresmoi (contract no. 257528), and by Science Foundation Ireland (grant no. 07/CE/I1142) as part of the Centre for Next Generation Localisation (http://www.cngl.ie/) at Dublin City University.

References

Banerjee, P., Du, J., Li, B., Naskar, S., Way, A., and van Genabith, J. (2010). Combining Multi-Domain Statistical Machine Translation Models using Automatic Classifiers. In *9th Conference of the Association for MT in Americas*, pages 141–150, Denver, Colorado, USA.

Bertoldi, N., Haddow, B., and Fouet, J.-B. (2009). Improved minimum error rate training in moses. *Prague Bulletin of Mathematical Linguistics*, No. 91:7–16.

Callison-Burch, C., Osborne, M., and Koehn, P. (2006). Re-evaluating the role of bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 249–256, Trento, Italy.

Daumé III, H. and Jagarlamudi, J. (2011). Domain adaptation for machine translation by mining unseen words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics and Human Language Technologies, Short Papers*, pages 407–412, Portland, Oregon, USA.

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

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

Foster, G., Goutte, C., and Kuhn, R. (2010). Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 451–459, Cambridge, MA.

He, Y., Ma, Y., Roturier, J., Way, A., and van Genabith, J. (2010). Improving the Post-Editing Experience Using Translation Recommendation: A User Study. In *Proceedings of the Ninth Conference of the Association for Machine Translation in the Americas*, pages 247–256, Denver, Colorado, USA.

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

Hopkins, M. and May, J. (2011). Tuning as ranking. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1352–1362, Edinburgh, United Kingdom.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In Lin, D. and Wu, D., editors, *Proceedings of EMNLP 2004*, pages 388–395, Barcelona, Spain.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings of the Tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand.

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

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 48–54, Edmonton, Canada.

Koehn, P. and Schroeder, J. (2007). Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 224–227, Prague, Czech Republic.

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*, pages 1–7, Taipei, Taiwan.

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

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

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

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

Pecina, P., Toral, A., Papavassiliou, V., Prokopidis, P., and van Genabith, J. (2012). Domain adaptation of statistical machine translation using web-crawled resources: a case study. In Cettolo, M., Federico, M., Specia, L., and Way, A., editors, *EAMT 2012: Proceedings of the 16th Annual Conference of the European Association for Machine Translation*, pages 145–152, Trento, Italy.

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

Specia, L., Hajlaoui, N., Hallett, C., and Aziz, W. (2011). Predicting machine translation adequacy. In *Proceedings of the Machine Translation Summit XIII*, Xiamen, China.

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

Tiedemann, J. (2009). News from OPUS – A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In Nicolov, N., Angelova, G., and Mitkov, R., editors, *Recent Advances in Natural Language Processing V*, volume 309 of *Current Issues in Linguistic Theory*, pages 227–248. John Benjamins, Amsterdam & Philadelphia.

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