Issues in Incremental Adaptation of Statistical MT from Human Post-edits

Mauro Cettolo † **Christophe Servan** ‡ **Nicola Bertoldi** †

Marcello Federico † **Loïc Barrault** ‡ **Holger Schwenk** ‡

† FBK, Fondazione Bruno Kessler
38123 Povo, Trento, Italy
LastName@fbk.eu

‡ LIUM, University of Le Mans
72085 Le Mans cedex 9, France
FirstName.LastName@lium.univ-lemans.fr

Abstract

This work investigates a crucial aspect for the integration of MT technology into a CAT environment, that is the ability of MT systems to adapt from the user feedback. In particular, we consider the scenario of an MT system tuned for a specific translation project that after each day of work adapts from the post-edited translations created by the user. We apply and compare different state-of-the-art adaptation methods on post-edited translations generated by two professionals during two days of work with a CAT tool embedding MT suggestions. Both translators worked at the same legal document from English into Italian and German, respectively. Although exactly the same amount of translations was available each day for each language, the application of the same adaptation methods resulted in quite different outcomes. This suggests that adaptation strategies should not be applied blindly, but rather taking into account language specific issues, such as data sparsity.

1 Introduction

In this work, we refer to the experimental framework set-up by the MateCat project,[1] which is developing a Web-based CAT tool for professional translators that will integrate new MT capabilities. Among them is what we named self-tuning MT, that is the automatic and incremental adaptation of the MT engine by exploiting user post-edits collected during the life of a translation project.[2]

The main contribution of the paper is to assess the effectiveness of popular SMT adaptation techniques in a real CAT framework, where the supervision is provided through post-edits from professional translators. The methods have been validated in laboratory tests on data collected in a two-day field test, which involved professionals for the translation of English documents to Italian and to German, in the legal domain. This domain represents a relevant sector in the translation industry and is suitable for exploiting SMT, since the information source is sufficiently homogeneous, the language is sufficiently complex, and there is sufficient multilingual data available to train and tune MT models.

The paper is organized as follows. Section 2 lists some of the related works. Section 3 introduces methods used for project adaptation. Section 4 briefly describes the conduct of the field test. Section 5 and Section 6, respectively, introduce the set-up and results of experiments. Section 7 concludes the paper with a discussion on the overall results.

2 Related Work

Our work deals with MT adaptation in general, and incremental adaptation more specifically.

Bertoldi et al. (2012) present an adaptation scenario where foreground translation and reordering models (TM) and language model (LM) of a phrase-based SMT system are incrementally trained on batches of fresh data and then paired to static background models. Similarly, the use of *local* and *global* models for incremental learning was previously proposed through a log-linear combination (Koehn and Schroeder, 2007), a mixture model (linear or log-linear) (Foster and Kuhn, 2007), the filling-up (Bisazza et al., 2011), or via ultraconservative updating (Liu et al., 2012).

[1] www.matecat.com

[2] By *translation project* we mean a set of homogeneous documents assigned to one or more translators.

Sharon O'Brien, Michel Simard and Lucia Specia (eds.)
Proceedings of MT Summit XIV Workshop on Post-editing Technology and Practice, Nice, September 2, 2013, p. 111–118.

Bach et al. (2009) investigate how a speech-to-speech translation system can adapt day-to-day from collected data on day one to improve performance on day two, similarly to us. However, the adaptation of the MT module involves only the LM and is performed on the MT outputs.

On standard machine translation tasks, Niehues and Waibel (2012) compare different approaches to adapt a SMT system towards a target domain using small amounts of parallel in-domain data, namely the backoff, the factored, and the already mentioned log-linear and fill-up techniques; the general outcome is that each of them is effective in improving non-adapted models and none is definitely better than each other, which is the best depending on how well the test data matches the in-domain training data.

This work deals with data selection as well, which is a problem widely investigated by the SMT community, see for example (Yasuda et al., 2008; Matsoukas et al., 2009; Foster et al., 2010; Axelrod et al., 2011). We apply a standard selection technique (Moore and Lewis, 2010), but in a quite different scenario where the task-specific data is extremely small and the generic corpus is actually close to the domain of the task.

3 Adaptation Methods

In this section we describe the techniques employed to adapt our SMT systems, namely data selection and translation, distortion and language model combination.

3.1 Data selection

It has been believed for a long time that just adding more training data always improves performance of a statistical model, e.g. a n-gram LM. However, this is in general true only if the new data is enough relevant to the task at hand, a condition which is rarely satisfied. The typical case is that of a narrow domain, for which a small task-specific text sample can result much more valuable than a very large generic text corpus, coming from sources that may be heterogeneous with respect to size, quality, domain, production period, etc.

The main idea of data selection is to try nevertheless to take advantage of the generic corpus, by picking out a subset of training data that is mostly relevant to the task of interest, which in our case is a specific translation project.

Similarly to (Servan et al., 2012), we first score a generic corpus against a LM trained on a seed of task-specific data, and compute the cross-entropy for each sentence. Then, the same generic corpus is scored against a LM trained on a random sample of itself. The sample size is roughly set equal to the seed corpus. From this point, the difference between task-specific cross-entropy and generic cross-entropy is computed for each sentence. Finally, sentences are sorted on the basis of this score. According the original paper (Gao and Zhang, 2002), this procedure leads to better selection than the simple perplexity sorting.

Now, the best splitting point of the sorted generic corpus has to be determined. The estimation is performed by minimizing the perplexity of a development set on growing percentages of the sorted corpus.

Moore and Lewis (2010) reported that the perplexity decreases when less, but more appropriate data is used (typically reaching a minimum with about 10 to 20% of the generic data). As a side effect, the models become considerably smaller which is an important aspect when deploying SMT systems in real applications.

Note that in our case the selection of parallel text was done by considering only one side of the parallel seed corpus, either the source or the target.

3.2 Adaptation of SMT models

Translation and distortion models: *Fill-up* is a technique for combining translation and distortion models estimated on corpora of different size and content. Initially proposed by Nakov (2008) and then refined by Bisazza et al. (2011), it merges the background phrase table with the foreground phrase table by adding only phrase pairs that do not appear in the foreground table. Only for the translation model, an additional indicator feature signals whether the phrase stems from the foreground or from the background phrase table. We chose the fill-up technique because it performs as good as other popular adaptation techniques (Niehues and Waibel, 2012) but with models that are more compact and easier to tune. It is worth noticing that the fill-up technique investigated by Niehues and Waibel (2012) slightly differs from the one described by Bisazza et al. (2011) in the way the candidate selection is performed.

We also apply a simplified version of the fill-

training	segments (M)	tokens (M) source	target
en→it	1.7	51.1	52.6
en→de	3.2	61.4	67.1

Table 1: Overall statistics on parallel data used for training purposes: number of segments and running words of source and target sides. Symbol M stands for 10^6.

up, called *backoff*, in which the indicator feature is discarded. Again, the backoff method proposed by Niehues and Waibel (2012) differs slightly in the way the scores of the phrase pairs stemming from the background phrase table are computed.

Language model: As concerns the LM adaptation, we employed the mixture of LMs which consists of the convex combination of one or more background LMs with a foreground LM. The method is available in the IRSTLM toolkit (Federico et al., 2008).

4 Field Test

For each language pair, the field test was organized over two days in which a document had to be translated by four translators. During the first day, for the translation of the first half of the document, translators received suggestions by the baseline MT engines described in Section 6; during the second day, MT suggestions for the second half of the document came from a system adapted to the text of the first day by means of one of the adaptation methods tested in our experiments (Section 6). Translators post-edited machine-generated translations for correcting mistakes and making them stylistically appropriate. The document was selected such that the size of its halves corresponds approximately to the daily productivity of professional translators, that is three to five thousand words.

A report on the field test including an analysis of the productivity of translators has already been published (Federico et al., 2012). Moreover, we performed a preliminary measure of the performance of MT outputs versus the post-edition of each translator. In both cases, pretty large inter-translator differences were observed. Since the limited number of subjects would have led to scores with large variances, we decided to choose one single representative translator per language pair, postponing analysis statistically more significant to forthcoming field tests involving more translators.

evaluation		segments	tokens source	target
en→it	D0	91	2,960	3,202
	D1	90	3,007	3,421
en→de	D0	86	2,960	2,712
	D1	89	3,007	2,999

Table 2: Overall statistics on test sets used in Day 0 and Day 1 of the field tests.

lng. pair	name	seed	for test on	%	tokens (M) src	trg
en→it	FGtgt	$D0_{\mathrm{tgt}}$	D1	10.1	5.1	5.3
	FGsrc	$D01_{\mathrm{src}}$	D1	9.8	5.1	5.2
	FGtgt	$D1_{\mathrm{tgt}}$	D0	10.1	5.1	5.3
	FGsrc	$D01_{\mathrm{src}}$	D0	9.8	5.1	5.2
en→de	FGtgt	$D0_{\mathrm{tgt}}$	D1	48.1	35.2	32.3
	FGsrc	$D01_{\mathrm{src}}$	D1	39.6	28.4	26.6
	FGtgt	$D1_{\mathrm{tgt}}$	D0	38.7	28.3	26.0
	FGsrc	$D01_{\mathrm{src}}$	D0	21.6	15.3	14.5

Table 3: Statistics of the selected parallel data.

5 Data

Training Data: Training data come from Version 3.0 of the JRC-Acquis collection (Steinberger et al., 2006). Refer to Table 1 for statistics on the actual corpora employed for training.

Evaluation Data: Concerning the evaluation, the document was taken from a motion for a European Parliament resolution published on the EUR-Lex platform in 2012. Statistics on the test documents translated during the field test are reported in Table 2; they refer to tokenized texts. Figures on the source side (English) refer to the texts the users are requested to translate; figures on the target side (Italian/German) refer to the text post-edited by the chosen translator (one for each language pair).[3]

The data selection described in Section 3.1 was applied to the training corpus. Table 3 provides the amount of data selected for each task. In our

[3] Although the document to translate is the same for the two language pairs, the segmentation differ due to a language-dependent automatic sentence alignment.

LM	en→it		en→de	
	D0	D1	D0	D1
	PP/OOV	PP/OOV	PP/OOV	PP/OOV
BG	97.5/0.31	93.2/0.27	209.9/1.91	172.9/1.40
FGtgt	73.3/0.54	72.2/0.67	181.4/1.91	147.4/1.43
FGsrc	69.4/0.73	67.6/0.70	166.1/1.95	136.8/1.43
Dn+BG	78.4/0.28	74.3/0.15	201.1/1.84	168.8/1.26
Dn+FGtgt	71.6/0.53	70.9/0.57	170.5/1.84	142.8/1.30
Dn+FGsrc	65.3/0.47	64.1/0.36	156.5/1.88	132.9/1.30
mix(Dn,FGtgt)	76.1/0.53	75.6/0.57	172.3/1.84	145.2/1.30
mix(Dn,FGsrc)	70.7/0.47	69.0/0.36	167.3/1.88	139.2/1.30
mix(Dn+FGtgt, BG)	80.8/0.28	78.1/0.15	185.5/1.84	167.8/1.26
mix(Dn+FGsrc, BG)	80.3/0.28	77.0/0.15	186.8/1.84	167.6/1.26
mix(Dn, FGtgt, BG)	66.8/0.28	64.7/0.15	169.3/1.84	146.3/1.26
mix(Dn, FGsrc, BG)	65.3/0.28	62.5/0.15	165.4/1.84	144.1/1.26

Table 4: Perplexity (PP) and out-of-vocabulary rate (OOV) of D0 and D1 on different 5gr LMs.

experiments, D0 and D1 alternatively played the role of development and test set. The seed for the selection was either the target side of the development set (Dn_{tgt}, n=0,1) or the concatenation of the source side of both the development and test set ($D01_{src}$); we name `FGtgt` and `FGsrc` the selected corpus and the models trained on it in the two cases.

The table also provides the percentage of data selected, computed with respect to the target side. The optimal splitting was performed by minimizing the perplexity of the target side of the development set.

6 Experiments

Lab test experiments have been performed on data sets described in Section 5. Performance are provided in terms of BLEU and TER, computed by means of the `MultEval` script implemented by Clark et al. (2011), and of GTM.[4] For statistical significance, p-values were calculated via approximate randomization for adapted systems with respect to the baselines and are reported in Tables 5 and 6 whenever not larger than 0.10.

The SMT systems have been built upon the open-source MT toolkit Moses (Koehn et al., 2007). The translation and the lexicalized reordering models are trained on the available parallel training data (Table 1); 5-gram LMs smoothed through the improved Kneser-Ney technique (Chen and Goodman, 1999) are estimated on the target side via the IRSTLM toolkit (Federico et al., 2008). The weights of the log-linear interpolation model have been optimized by means of the Margin Infused Relaxed Algorithm (MIRA) process (Hasler et al., 2011) provided within the Moses toolkit.

Various models have been built by means of the methods described in Section 3. Here the list of acronyms and corresponding meaning used in the rest of the paper. Note that whenever "data selected" is mentioned, we refer to the application of the procedure described in Section 3.1 with the training data playing the role of *generic corpus* and the portion of the document translated during either the first day (D0) or the second day (D1) that of *seed corpus*:

`BG`: background model, trained on the whole training data

`Dn+BG`: model trained on the concatenation of Dn (either D0 or D1) and training data

`FGtgt`: model trained on data selected using the target side of either D0 or D1 as seed corpus

`Dn+FGtgt`: model trained on the concatenation of the target side of either D0 or D1 and `FGtgt`

`FGsrc/Dn+FGsrc`: similar to `FGtgt/Dn+FGtgt`, but the selection is made using the concatenation of the source side of both D0 and D1 as seed corpus

[4] http://nlp.cs.nyu.edu/GTM

TM=BG	en→it						en→de					
	D0			D1			D0			D1		
LM	BLEU	TER	GTM	BLEU	TER	GTM	BLEU	TER	GTM	BLEU	TER	GTM
BG	47.9	34.9	73.6	46.8	34.7	74.3	32.2	58.4	60.1	37.4	49.1	65.5
Dn+BG	50.2$^{\blacktriangle}$	33.2$^{\blacktriangle}$	75.0	48.1$^{\triangle}$	34.0$^{\triangle}$	74.9	32.4$^{\diamond}$	57.1$^{\triangle}$	59.7	37.9$^{\diamond}$	48.6$^{\triangle}$	65.7
Dn+FGtgt	46.4	36.1	72.5	47.8	34.5	74.4	32.6$^{\diamond}$	56.5$^{\triangle}$	59.7	37.9$^{\diamond}$	48.1$^{\triangle}$	65.5
Dn+FGsrc	47.6	34.8	73.6	48.2	35.0	74.2	33.0$^{\diamond}$	56.2$^{\triangle}$	60.6	37.9$^{\diamond}$	47.9$^{\triangle}$	65.5
mix(Dn,FGtgt)	45.7$^{\diamond}$	35.2	73.5	46.8	34.7	74.3	33.0$^{\diamond}$	56.0$^{\triangle}$	60.3	36.1	49.0$^{\diamond}$	65.4
mix(Dn,FGsrc)	47.0	34.4	74.4	47.3	34.5	74.2	33.7$^{\triangle}$	55.4$^{\blacktriangle}$	60.8	36.2	48.8$^{\triangle}$	65.4
mix(Dn+FGtgt,BG)	49.8$^{\blacktriangle}$	33.0$^{\blacktriangle}$	75.3	47.7$^{\triangle}$	33.8$^{\triangle}$	75.0	33.4$^{\triangle}$	55.3$^{\blacktriangle}$	59.8	36.0	49.1	64.3
mix(Dn+FGsrc,BG)	48.8	33.6$^{\triangle}$	74.8	47.4	34.1$^{\diamond}$	74.7	33.8$^{\triangle}$	54.6$^{\blacktriangle}$	60.3	35.8	49.5	64.1
mix(Dn,FGtgt,BG)	47.6	34.0	74.3	48.5$^{\diamond}$	33.7	74.9	33.3$^{\diamond}$	56.4$^{\triangle}$	60.5	36.8	48.1$^{\triangle}$	66.2
mix(Dn,FGsrc,BG)	48.5	33.6	75.4	48.6$^{\diamond}$	33.2$^{\triangle}$	75.1	32.8$^{\diamond}$	56.9$^{\triangle}$	60.5	36.7	48.2$^{\triangle}$	66.0

Table 5: Performance on D0/D1 of systems with LMs adapted on D1/D0. Symbols $^{\blacktriangle}$, $^{\triangle}$ and $^{\diamond}$ near to BLEU and TER scores indicate that adapted models outperform BG with p-values not larger than 0.01, 0.05 and 0.10, respectively.

`mix()`: mixture of LMs (linear interpolation)

`fillup()`: fill-up of TMs

`backoff()`: backoff of TMs

It is worth noticing that using the source side of the test set for data selection (systems `FGsrc` and `Dn+FGsrc`) can be ambivalent. On the one hand the system can be penalized since its LM is estimated on the target side of the selected parallel data; on the other hand it can be rewarded since the seed corpus includes the actual text to translate, and hence the selected data could be more appropriate.

First of all, the quality of LMs was assessed in terms of perplexity (PP) and out-of-vocabulary (OOV) rate. Indeed, in the computation of PP of a text with respect to a given LM, the presence of OOV words is accounted by adding a fixed penalty for each OOV occurrence; nevertheless, we think useful to provide even explicit OOV values for the sake of completeness. Scores are provided in Table 4: they refer to D0 and to D1 and are computed on the baseline LM (`BG`) or on LMs adapted in various ways to the other portion of the field test document (D1 or D0).

In general, adapted models always improve the PP of the baseline LM, while the OOV decreases provided that the whole training text is also used to train the model. More specifically:

• data selection is effective: with reference to `FGtgt` and `FGsrc` rows, whatever the seed, the PP of Dn on the selected data always improves over the baseline, from 15% up to 30% depending on the target language and on the test set; of course, the OOV rate worsens because the lower amount of training data

• the selection on the concatenation of the source side of the development and evaluation sets is more effective than the selection made only on the target side of the development set: compare paired rows including `FGsrc` and `FGtgt`

• the linear interpolation of LMs gives contrasting results: from one side, `mix(Dn,FGsrc,BG)` allows the lowest PP on Italian and very competitive on German; from the other, when it is applied to `Dn` and `FGtgt/src` it fails with respect to the naive concatenation

• the use of the development set in LM training yields a significant improvement of the OOV rate, especially for D1 (from 0.27% to 0.15% for Italian, from 1.40% to 1.26% for German); at the same time, `Dn+BG` row shows a significant PP improvement over the baseline only for Italian: this means that D0 and D1 are alike for that language pair, less for English-German where consequently the adaptation could be more problematic.

6.1 MT results

MT experiments have been conducted either by varying the LM and keeping fixed the baseline TM (Table 5) and by consistently pairing the adapted models (Table 6). Baseline MT system uses `BG`

adapted LM/TM	en→it						en→de					
	D0			D1			D0			D1		
	BLEU	TER	GTM	BLEU	TER	GTM	BLEU	TER	GTM	BLEU	TER	GTM
BG	47.9	34.9	73.6	46.8	34.7	74.3	32.2	58.4	60.1	37.4	49.1	65.5
Dn+BG	49.9	32.9$^{\diamond}$	76.0	49.8$^{\blacktriangle}$	33.3$^{\diamond}$	75.8	32.6$^{\diamond}$	57.4$^{\triangle}$	60.0	37.6$^{\diamond}$	48.4$^{\triangle}$	65.6
Dn+FGtgt	45.5	35.8	72.9	49.3$^{\triangle}$	33.2	75.5	31.9	58.0$^{\diamond}$	59.6	38.1$^{\triangle}$	48.2$^{\triangle}$	65.9
Dn+FGsrc	48.3	33.7	74.3	49.6$^{\triangle}$	33.2	75.5	31.9	58.90	58.3	37.9$^{\triangle}$	48.4$^{\triangle}$	66.1
mix/fillup(Dn,FGtgt)	45.2	36.7	73.1	46.9	34.8	75.0	30.6	58.5	58.5	35.1	50.80	64.2
mix/fillup(Dn,FGsrc)	45.8	35.6	73.7	48.4	33.9	75.4	31.3	59.0	58.6	35.2	51.30	64.3
mix/fillup(Dn+FGtgt,BG)	50.3$^{\triangle}$	32.5$^{\blacktriangle}$	76.0	49.4$^{\blacktriangle}$	33.3$^{\triangle}$	75.6	31.6	58.4	59.4	38.5$^{\blacktriangle}$	47.8$^{\blacktriangle}$	65.7
mix/fillup(Dn+FGsrc,BG)	48.8	33.4$^{\diamond}$	75.0	49.5$^{\blacktriangle}$	32.3$^{\blacktriangle}$	76.1	31.1	59.0	58.9	37.9$^{\triangle}$	48.9$^{\triangle}$	65.7
mix/backoff(Dn+FGtgt,BG)	50.5$^{\blacktriangle}$	32.4$^{\blacktriangle}$	76.1	49.1$^{\triangle}$	33.4$^{\diamond}$	75.4	33.0$^{\triangle}$	56.7$^{\blacktriangle}$	60.4	38.9$^{\blacktriangle}$	48.1$^{\blacktriangle}$	66.0
mix/backoff(Dn+FGsrc,BG)	49.2	32.9$^{\triangle}$	75.1	49.1$^{\triangle}$	32.3$^{\blacktriangle}$	76.0	31.8	59.1	59.6	38.3$^{\triangle}$	48.4$^{\triangle}$	65.9
mix/fillup(Dn,FGtgt,BG)	46.6	35.5	73.7	49.0$^{\diamond}$	33.3	76.0	31.6	58.3	58.9	36.1	51.0	63.6
mix/fillup(Dn,FGsrc,BG)	48.1	34.4	75.2	49.4$^{\triangle}$	32.9$^{\diamond}$	75.8	30.3	59.7	58.8	36.5	50.6	64.3

Table 6: Performance on D0/D1 of systems with models adapted on D1/D0. Symbols $^{\blacktriangle}$, $^{\triangle}$ and $^{\diamond}$ near to BLEU and TER scores indicate that adapted models outperform BG with p-values not larger than 0.01, 0.05 and 0.10, respectively.

models; its results are replicated in the first row of the two tables for the sake of readability.

The first set of experiments aimed at isolating the contribution of adapted LMs, a fortiori since adaptation often involves just the LM. The symmetric experiments where only the TM is changed are less informative since the lack of LM support prevents improvements by the TM to emerge; therefore, they are not presented.

Adapted LMs: in many cases, Italian adapted LMs allows to outperform the performance of the baseline system, whereas no method yield significant improvements on both days in the English-German task; this should be due to the degree of similarity between D0 and D1: pretty high for English-Italian, quite low for English-German, as stated before in comments to Table 4.

For the English-Italian pair, in general the better the PP and OOV values are, the better the translation is. In fact, the low PP of LMs built over `Dn` and `FGtgt/src` only, does not yield good MT scores, because the high OOV rates. The naive concatenation of `Dn` and `BG` provides surprisingly good performance, matched only by the `mix(Dn+FGtgt/src,BG)` LMs; the interpolation of the three LMs, which gave the best PP, keeps its promise only on D1. Differently than PP, data selection on the source side of D0 and D1 not always overcomes that on the target side of the development set.

Concerning the English-German pair, the naive concatenation of `Dn` and `BG` does not improve nor hurt baseline performance. Again `mix(Dn+FGtgt/src,BG)` outperforms the baseline, but limited to D0. On D1 the only effective method is the concatenation of development and selected data (`Dn+FGtgt/src`).

A common outcome regards the unreliability of the `Dn` model: whenever it is combined with other LMs as it is (`mix(Dn,`$\cdots$`)`, effects are mostly negative compared to concatenation; this is due to the small amount of data used for its training (about 3,000 words).

Adapted Models: the different effectiveness of methods on the two language pairs observed by changing only the LM is confirmed when adapted LMs and TMs are consistently paired: most techniques are effective on English-Italian with the added value guaranteed by the reciprocal support of models, while controversial results characterize the English-German pair.

For the favorite pair, adapted systems significantly outperform the baseline provided that the whole training data is somehow used in building models. It deserves mentioning the excellent performance of the models built over the naive concatenation of the development and training data (`Dn+BG`). The best systems seem to

be those combining `(Dn+FGtgt,BG)` for D0, `(Dn+FGsrc,BG)` for D1: the fact that the references of D0 and D1 are post-edits and are used both for evaluation and for building adapted models could explain that apparently incoherent behavior. Again, the use of the model built on just `Dn` negatively affects performance.

On English-German task, the only good-performing technique on both days is the `mix/backoff(Dn+FGtgt,BG)`, while the naive concatenation `Dn+BG` slightly improves some scores and does not affect the others. Evidently, D0 and D1 are too different to allow models adapted on one of them to well represent also the other.

An outcome shared by the two tasks is that `backoff` is a bit more effective than `fillup`, probably due to the difficulty in properly setting the weight of the additional indicator feature of the latter method.

7 Discussion

The experiments with Italian and German translations, although performed on the same source texts and by applying the same adaptation methods, result in quite different outcomes.

We try now to summarize the main issues and to sketch possible explanations and directions we will investigate in the future to overcome them.

Data selection. The same amount of seed data (3,000 words) does not work equally for German and Italian. While for Italian, around 10% of the training data were selected by seeding with D0 and D1 texts, between 20%-48% were selected for German. Data selection relies on similarity scores computed using small language models estimated disregarding infrequent words (Moore and Lewis, 2010). Given its highly inflected language, it is likely that for German the large majority of project specific words in the seed are singletons, so that the corresponding LM looses most of its specificity. Alternative ways to explore, specifically for highly inflected languages such as German, could be to remove word inflection during data selection, e.g. by stemming words, and to work with low-order n-grams, e.g. 1-grams and 2-grams.

LM adaptation. All the tested LM adaptation methods provided improvements in terms of perplexity and OOV rate (Table 4), both on the Italian and German translations. Such improvements reflected for some adaptation methods in better MT scores (Table 5) for the English-Italian direction, but not always for the English-German direction. It is worth noticing that the simple concatenation of training and adaptation data performs better than more refined and probably too aggressive adaptation approaches. The inconsistent behavior of perplexity and translation scores for both translation directions can be explained by the fact that adapted LMs basically boost the probability of subsets of target words, that should likely occur in the reference test translations, thus giving better perplexity values and OOV rates. However, if the same target words are not reachable through the translation model, the advantage provided by the adapted LM vanishes. This mismatch becomes even more relevant when adapted translation models are employed, too.

TM adaptation. The use of only adapted LMs showed significant improvements to slight degradations in performance, according to the considered method, translation directions, and adaptation and testing sets. The addition of adapted translation models (Table 6) further widened the range of outcomes and, mostly important, does not show to be additive with language model adaptation. In fact, language model adaptation configurations that perform best do not seem to combine well with some translation model adaptation methods, especially for English-German. In fact, the most consistent behavior across all languages and data sets is shown by a specific configuration (`mix/backoff(Dn+FGtgt,BG)`), whereas very similar set-ups show mixed behaviors. A possible reason for this may be the overfitting of the TM on the adaptation data. In particular, as for each English word more German surface forms may correspond than for Italian, biasing the TM towards the observations of the adaptation data can hurt the overall quality of adapted models. Concerning data selection, the problem with German seems that the seed is not large enough to properly characterize the narrow domain of the document. Hence, in such a case, only soft adaptation methods appear adequate and safe.

As future work, we plan to investigate both on data selection in case of small seeds and on less

aggressive adaptation methods for inflected languages, such as biasing the translation model only at the lexical rather than phrase level and to generalize over different word inflections. Moreover, other field tests will be carried out in order to collect further post-edited translations.

Acknowledgments

This work was supported by the MateCAT project, which is funded by the EC under the 7^{th} Framework Programme.

References

Axelrod, A., X. He, and J. Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *EMNLP*, Edinburgh, UK.

Bach, N., R. Hsiao, M. Eck, P. Charoenpornsawat, S. Vogel, T. Schultz, I. Lane, A. Waibel, and A. W. Black. 2009. Incremental adaptation of speech-to-speech translation. In *NAACL HLT: Short Papers*, Boulder, US-CO.

Bertoldi, N., M. Cettolo, M. Federico, and C. Buck. 2012. Evaluating the learning curve of domain adaptive statistical machine translation systems. In *WMT*, Montréal, Canada.

Bisazza, A., N. Ruiz, and M. Federico. 2011. Fill-up versus interpolation methods for phrase-based SMT adaptation. In *IWSLT*, San Francisco, US-CA.

Chen, S. F. and J. Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, 4(13):359–393.

Clark, J., C. Dyer, A. Lavie, and N. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *ACL*, Portland, US-OR.

Federico, M., N. Bertoldi, and M. Cettolo. 2008. IRSTLM: an open source toolkit for handling large scale language models. In *Interspeech*, Melbourne, Australia.

Federico, M., A. Cattelan, and M. Trombetti. 2012. Measuring user productivity in machine translation enhanced computer assisted translation. In *AMTA*, Bellevue, US-WA.

Foster, G. and R. Kuhn. 2007. Mixture-model adaptation for SMT. In *WMT*, Prague, Czech Republic.

Foster, G., C. Goutte, and R. Kuhn. 2010. Discriminative instance weighting for domain adaptation in statistical machine translation. In *EMNLP*, Cambridge, US-MA.

Gao, J. and M. Zhang. 2002. Improving language model size reduction using better pruning criteria. In *ACL*, Philadelphia, US-PA.

Hasler, E., B. Haddow, and P. Koehn. 2011. Margin infused relaxed algorithm for Moses. *The Prague Bulletin of Mathematical Linguistics*, 96:69–78.

Koehn, P. and J. Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *WMT*, Prague, Czech Republic.

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 *ACL Companion Volume Proc. of the Demo and Poster Sessions*, Prague, Czech Republic.

Liu, L., H. Cao, T. Watanabe, T. Zhao, M. Yu, and C. Zhu. 2012. Locally training the log-linear model for SMT. In *EMNLP*, Jeju, Korea.

Matsoukas, S., A.-V. I. Rosti, and B. Zhang. 2009. Discriminative corpus weight estimation for machine translation. In *EMNLP*, Singapore.

Moore, R. C. and W. Lewis. 2010. Intelligent selection of language model training data. In *ACL Short Papers*, Uppsala, Sweden.

Nakov, P. 2008. Improving English-Spanish statistical machine translation: experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing. In *WMT*, Columbus, US-OH.

Niehues, J. and A. Waibel. 2012. Detailed analysis of different strategies for phrase table adaptation in SMT. In *AMTA*, San Diego, US-CA.

Servan, C., P. Lambert, A. Rousseau, H. Schwenk, and L. Barrault. 2012. LIUM's statistical machine translation systems for WMT 2012. In *WMT*, Montréal, Canada.

Steinberger, R., B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufi, and D. Varga. 2006. The JRC-Acquis: a multilingual aligned parallel corpus with 20+ languages. In *LREC*, Genoa, Italy.

Yasuda, K., R. Zhang, H. Yamamoto, and E. Sumita. 2008. Method of selecting training data to build a compact and efficient translation model. In *IJCNLP*, Hyderabad, India.