Deriving translation units using small additional corpora.

Carlos A. Henríquez Q.
TALP Research Centre
Barcelona, Spain
carlos.henriquez@upc.edu

José B. Mariño
TALP Research Centre
Barcelona, Spain
jose.marino@upc.edu

Rafael E. Banchs
Institute for Infocomm Research
Singapore
rembanchs@i2r.a-star.edu.sg

Abstract

We present a novel strategy to derive new translation units using an additional bilingual corpus and a previously trained SMT system. The units were used to adapt the SMT system. The derivation process can be applied when the additional corpus is very small compared with the original train corpus and it does not require to compute new word alignments using all corpora. The strategy is based in the Levenshtein Distance and its resulting path. We reported a statistically significant improvement, with a confidence level of 99%, when adapting an Ngram-based Catalan-Spanish system using an additional corpus that represents less than 0.5% of the original train corpus. The additional translation units were able to solve morphological and lexical errors and added previously unknown words to the vocabulary.

1 Introduction.

Statistical Machine Translation (SMT) systems are trained using parallel corpora. Therefore, once the system is trained and tuned, it is tightly coupled to the specific domain the train corpus belongs to. If later on we want to use additional bilingual corpora to improve or adapt our system, we could build additional translation models and interpolate them with the original one or we could join all the additional data with the original corpus and train a new system from scratch. However, those strategies often involve computing new word alignments considering all corpora together, which is a computational expensive task.

This study focuses on the use of additional bilingual corpora to adapt a previously trained SMT system, without the need to recompute word alignments. The proposed method utilizes the SMT system to translate the source side of the new corpus and then compares the translation output with its target side. This comparison allows the method to detect errors made during decoding and provide it at the same time with a possible solution, which is finally used to build additional translation units.

We have experimented with a Ngram-based SMT system (Mariño et al., 2006), translating from Catalan into Spanish and we have obtained a significant improvement in translation quality, adapting a state-of-the-art system trained with a corpus of more than four million sentences with an additional corpus of only 1.6 thousand sentences.

This document is organized as follows: Section 2 introduces us to the concept of Statistical Machine Translation, with an emphasis in Ngram-based SMT; Section 3 presents a description of the possible scenarios where the proposed strategy could be used, domain adaptation (subsection 3.1) and user feedback (subsection 3.2); Section 4 describes the experimental set-up, it details the baseline system in subsection 4.1 and the additional corpus in subsection 4.2, it also explains the main algorithm to derive, filter and interpolate the additional translation units with the baseline translation model (subsections 4.3, 4.4 and 4.5); finally, Section 5 presents and analyzes the results obtained with the new translation system while Section 6 summarize our findings.

2 Ngram-based Machine Translation.

The idea of Statistical Machine Translation (SMT) relies on the translation of a source language sentence f (usually referred as "French") into a tar-

get language sentence $\hat{e}$ (usually referred as "English"). Among all possible target language sentences e we choose the one with the highest score, as show in equation (1):

$$\hat{e} = \arg\max_e \left[\sum_{m=1}^{M} \lambda_m h_m(f,e) \right] \qquad (1)$$

This equation, called the log-linear model, is a variation of the source-channel approach to SMT (Brown et al., 1990). It was proposed by Och and Ney (2002) and allows using more than two models and to weight them independently.

Frequently used paradigms of SMT based on the log-linear model are Phrase-based SMT (Koehn et al., 2003), Hierarchical-based SMT (Chiang, 2007) and Ngram-based SMT (Mariño et al., 2006). In our experiments we used the Ngram-based approach.

The Ngram-based approach relies on the concept of tuple. A tuple is a bilingual unit with consecutive words both on the source and target side that is consistent with the word alignment. They must provide a unique monotonic segmentation of the sentence pair and they cannot be inside other tuple. This unique segmentation allows us to see the translation model as a language model, where the language is composed of tuples instead of words. That way, the context used in the translation model is bilingual and implicitly works as a language model with bilingual context as well. In fact, while a language model is required in phrase-based and hierarchical phrase-based systems, in Ngram-based systems it is considered just an additional feature.

This alternative approach to a translation model defines the probability as:

$$P(f,e) = \prod_{n=1}^{N} P\left((f,e)_n \mid (f,e)_{n-1}, \ldots, (f,e)_1\right)$$
$$(2)$$

where $(f,e)_n$ is the n-th tuple of hypothesis e for the source sentence f.

As additional features, we used:

- A Part-Of-Speech (POS) language model for the target side.

- A target word bonus model.

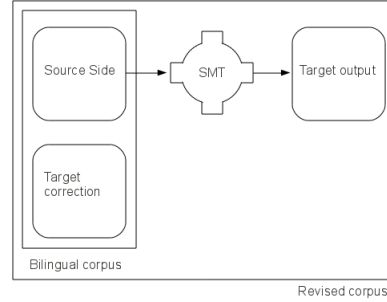We used the open source decoder MARIE (Crego et al., 2005) to build the different Ngram-based systems.

Figure 1: Revised corpus composition

3 Problem Statement.

Suppose we have a previously trained SMT system built with a large bilingual corpus (millions of sentences) that have an acceptable performance in the area it was designed for. We would like to adapt that system to scenarios it has not seen, for instance it was trained to translate Parliament sessions and we would like to translate news articles or tourist dialogues; another examples could be that we would like to renew its vocabulary coverage and writing style or that we would like to correct errors we have seen during translation.

In order to adapt our system we have available a small bilingual corpus (a few thousands sentences) specific to the problem we plan to solve. The idea is to use that corpus to generate translation units that will be added to our trained system without an alignment process that would involve the use of all the original parallel corpus.

Besides the additional bilingual corpus we also have the translation output of its source side computed with the system we want to adapt. Therefore our new data, named *revised corpus*, has actually three parts: The source side of the bilingual corpus, the target output (computed with the trained system) and the target correction (the target side of the bilingual corpus).

We present now two different cases that illustrates the scenario described before. Additionally, we can see a graphical description of the general case in Figure 1.

3.1 Domain adaptation.

Because SMT systems are tightly coupled to their corpus domain, they are prone to commit errors when they translate sentences that belong to a different domain. For instance, a SMT system trained with the Europarl Corpus (Koehn, 2005) may not translate movie reviews as expected.

Text corpora can be different in vocabulary, style or grammar and a method to adapt to different domains is preferred than building a whole new system for each domain we face. Moreover, it might be the only plausible solution if we have a big out-of-domain parallel corpus but a small in-domain corpus which, if used alone, would perform poorly.

Different methods have been studied to perform such adaptation, and they all require a small in-domain corpus whether it is bilingual or not, for the system to adapt. Some of them include: concatenate corpora and model interpolation (Koehn and Schroeder, 2007), using mono-lingual and cross-lingual information retrieval (Hildebrand et al., 2005; Xu et al., 2007; Snover et al., 2008), language model adaptation for difficult to translate phrases (Mohit et al., 2009), generating a synthetic corpus (Ueffing et al., 2007; Schwenk and Senellart, 2009) and finally post-editing approaches (Isabelle et al., 2007) combined with incremental training (Hardt and Elming, 2010).

The strategy proposed here assumes we have an out-of-domain system and a revised corpus that is composed of a small bilingual in-domain corpus and the translation of its source side computed with the system we like to adapt.

3.2 User feedback.

Similar to domain adaptation, user feedback is also a valid scenario for the proposed strategy. In this case, a previously trained SMT system is used to translate sentences provided by different users. Then, if the users consider it convenient, they can suggest a better translation than the one the system obtained. If we saved all those suggestions together with the input sentence and the translation output, eventually we would have an additional bilingual corpus with its corresponding system translation, which fits the definition of a revised corpus.

In spite of the frequent use of online machine translators, the users do not tend to send feedback to improve them and even when they do, it is hardly useful. Usually, the system offers the functionality of sending feedback without restrain. Therefore, the collect algorithms must confront with vicious feedbacks, orthographic errors, text unrelated with the original query, etc.

For that reason, to exploit user feedback we have to deal with two different problems: how to filter

Baseline Train	Catalan	Spanish
Number of sentences	4.6MM	
Running words	96.94MM	96.86MM
Vocabulary size	1.28MM	1.23MM

Tuning	Catalan	Spanish
Number of sentences	1,966	
Running words	46.76K	44.66K
Vocabulary size	9.1K	9.4K

Table 1: Baseline and tuning corpora.

user feedback so we keep only the valuable data; and how to use the selected data to adapt the machine translation system. This research addresses the second task.

4 Experimental set-up.

Without loss of generality we present the derivation process in the ambit of domain adaptation. The objective is to adapt an already tuned SMT system, trained with a corpus collected from old news, with a small additional corpus collected from more recent news. We do not plan to change the news domain but adapt it to modern times, adding new vocabulary and adapting the writing style.

4.1 Baseline system and corpus.

We started with the UPC's Catalan-Spanish system (named N-II), an Ngram-based SMT system which uses syntactic and morphological knowledge to improve its translations. A complete description of it and its translation quality can be found in Farrús (2009). It was built with news articles collected from the bilingual newspaper "El Peridico" during the period 2,000-2,007. Table 1 shows the statistics of this corpus.

It also includes a Part-Of-Speech (POS) 5-gram target language model, built with the POS version of the training corpus and a target word bonus model. The syntactic analysis was performed with Freeling (Padró et al., 2010), the translation model and the POS language model were built with SRILM (Stolcke, 2002).

N-II has an online version [1] available, which also includes an spell-checker, where the users can ask for translations and provide suggestions for better translation.

	Catalan	Spanish
Number of sentences	155K	
Running words	3,43M	3,43M
Vocabulary size	187K	184K

Table 2: Statistics of "El Peridico" corpus from 2,008.

Correction	Catalan	Spanish
Number of sentences	1,608	
Running words	34.67K	35.17K
Vocabulary size	11.02K	11.17K

Test	Catalan	Spanish
Number of sentences	2,048	
Running words	46.03K	46.00K
Vocabulary size	13.28K	13.39K

Table 3: Experimental corpora.

4.2 Revised corpus construction.

An additional corpus was also collected from the newspaper "El Peridico" but only with news from 2,008. It has a total of $155K$ sentences. A summary of its statistics can be seen in Table 2. We used N-II to obtain the translation output. For the purpose of these experiments, we only used two small subsets that were built taking samples without replacement, for training and testing. The first subset is called "Correction Corpus", it has $1.6K$ sentences and it was used to build the revised corpus, translating the Catalan side into Spanish with N-II. The second subset is the test corpus, it has $2,048$ sentences and it was used to measure the translation quality of the different systems. Table 3 outlines the statistics of both corpora.

4.3 Derivation process.

The derivation process is based on the comparison between the target output and the target correction of the revised corpus. This comparison is computed using the Levenshtein Distance.

The Levenshtein Distance is defined as the minimum number of edits needed to transform one string into the other; being insertion, deletion and substitution of a single character the allowed edit operations. The concept can be applied to full sentences as well, considering the sentence as the complete string and each word as a single character.

First we compute the Levenshtein distance between a target output sentence and its target correction. While we compute the distance we also keep track of its path in order to recover the minimum sequential steps to change the target output into its correction. Let p be a Levenshtein path from target output t_1 to correction t_2 and s, e, d, a the possible values of each step within the path, which stand for "replace a word", "do nothing", "delete a word" and "add a word" respectively.

Once we obtain p, we identify the longest substrings s_k that match one of the following regular expression (they are checked in order):

$$[sda] * s[sda] * \qquad (3)$$
$$e[da] + \qquad (4)$$
$$\hat{}[da] + e \qquad (5)$$

which represent a change zone.

Then, for each change zone s_k of p we extract the related words from t_1 and t_2 to build an *output-to-correction* translation unit. If the number of related words either in t_1 or t_2 is greater than 10, the change zone is not valid for the next step.

Finally, in order to obtain the unique monotonic segmentation of the source-correction sentence pair, we start from left to right using the original units; whenever we find a unit whose target words are involved in a valid s_k, we replace those target words for their corresponding correction words, according to the *output-to-correction* translation unit built previously. In case we find consecutive units whose target words are involved in a valid s_k, first we join the consecutive units to form a larger single unit and then we perform the replacement as explained before.

We can see a graphic example of the whole process in Figure 2. There, we have an output sentence provided by the system, *"con un largo fin de semana ya puede haber lo suficiente"* (a weekend may be enough), and a correction, *"un largo fin de semana ya puede bastar"*. With this sentence pair we computed the Levenshtein distance, which is 4, and the Levenshtein path $p = deeeeeeeddse$. The path indicates that the first, ninth and tenth output words must be deleted and the eleventh word must be replaced. According to the regular expressions (3), (4) and (5), this example gives us two different change zones, $s_1 = de$ at the beginng and $s_2 = dds$ at the end (section (i) in the figure); from s_1 we have the *output-to-correction* unit (*"con un", "un"*) and from s_2 we have (*"haber lo*
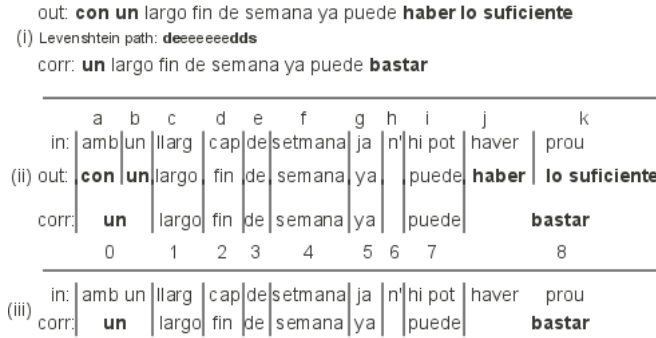
Figure 2: Example of the deriving process. (i) First we compute the Levenshtein path between output and correction (change zones and related words are in bold), (ii) then we segment the pair input-correction considering the original tuples c, d, e, f, g, h, i and change zones 0 and 8. (iii) Shows the final monotonic segmentation with the new tuples added.

suficiente","bastar"). Finally, to segment the sentence pair we started from left to right, joining the first two consecutive units, because their target output words were involved in s_1, and replacing their target output words for their corresponding correction words; then we used the next seven original units without change and we joined the last two original units, because of s_2, and replaced their target output words as well. The final monotonic segmentation can be seen in section (iii) in the figure.

4.4 Filter process.

Once we have the sentence segmentation, we apply a lexical filter which takes into account the lexical costs of the tuple (ignoring unknown words during computation), and set a threshold in the average of the lexical costs to remove all expensive units from the tuple vocabulary.

This filter is important because it deals with "new tuples" whose source and target side are not translations of each other, like *("s d' un any i mig" , "son dos vasos comunicantes")*.

Table 4 shows the vocabulary size of the different set of tuples: how many we had in the baseline system, how many were extracted from the revised corpus with and without the filter described above and finally, how many of those extracted tuples were not seen in the baseline system vocabulary.

4.5 Interpolation process.

With all problematic units removed from the vocabulary, we built the enhanced translation model following these steps:

1. We pruned the extracted tuples set, removing

System	Tuples
Baseline	$1.16MM$

Filter	Extracted	Unseen
None	$8,511$	$1,360$
Lexical	$8,307$	$1,097$

Table 4: Extracted and unseen tuples.

all units that had more than 10 words either in the source or the target side and leaving the 20 most frequent translation options for each tuple's source side.

2. We added the new remaining tuples to the vocabulary of baseline units.

3. We built a 3-gram translation model with interpolate estimates and modified Kneser-Ney discounting, considering the vocabulary defined in the previous step and using only the recently segmented corpus.

4. We interpolated the resulting translation model with the baseline translation model.

α	0.75	0.80	**0.85**	0.90	0.95
Dev	84.01	84.05	**84.12**	84.07	83.78

Table 5: α-values used to interpolate the translation models and the corresponding BLEU scores after tuning.

System	Corr.	Test	Conf.
Baseline	76.87	77.19	-
+Rev.tuples	85.96	77.23	87.24%
+Rev.tuples+lex.fil.	83.85	77.33	99.20%

Table 6: Systems tested and their BLEU scores.

The linear interpolation process followed the formula:

$$TM(n) = \alpha TM_{Base}(n) + (1 - \alpha)TM_{Add}(n) \tag{6}$$

where $TM(n)$ is the resulting translation model score for the n-gram n, TM_{Base} is the baseline translation model and TM_{Add} is the new translation model computed in the third step.

To determine the value of α we tuned the system considering five different values and kept the one that obtained the highest BLEU score. Table 5 shows the different BLEU scores obtained with the development set and $\alpha = 0.85$ as the best candidate.

5 Results and discussion.

We built two different systems and used $\alpha = 0.85$ for the interpolation. The results obtained over the correction and test corpora can be seen in Table 6. The second and third column correspond to the BLEU (Papineni et al., 2001) scores obtained by the different systems in the Correction and Test corpora, using only one reference. The fourth column gives the confidence level for test BLEU being higher than the baseline test BLEU.

Notice that all revised system performed better in the correction corpus, which is obvious because it is part of the revised corpus. Also, they all improved the baseline test score. What is interesting is that once we added the lexical filter, the correction BLEU decreased and the test BLEU increased. It means that the filter is helping the system generalize its learning.

Moreover, even though the revised system without filter is not significantly better than the baseline, we found that with the lexical filter we

System	Higher	Lower	Same
Baseline	122	154	1,772
+Rev.tuples+lex.fil.	154	122	1,772

Table 7: Sentence by sentence comparison of BLEU score

achieved a better performance, with a confidence level of 99%. The confidence levels were obtained using the "Pair Bootstrap Resampling" method described in Koehn (2004).

Besides the automatic test described before, we also compute the BLEU scores over the test set, sentence by sentence, with the baseline and the final system; then, we compared them to determine which sentences were better (or worse) in the final system and why. Table 7 shows these results. We can see that the final system was better in 154 sentences, the baseline system was better in 122 and that they got the same score in the remaining 1,772. We took a closer look at those 122 sentences and found that most of the final system outputs had used synonyms and paraphrases and that they were indeed valid although they were not used in the reference. On the other hand, we found some semantic, lexical and morphological errors solved among the 154 sentences where the final system had a better score.

Figure 3 shows a sample of both subsets, displaying first the baseline output and then the final system output. The first case corrected a word-by-word translation; it means "besides". The second and sixth pair are example of sentences with unknown words, "cirllic" and "Govern", that are solved with their correct translation by the final system, "cirlico" and "Gobierno Catalán"; their English translation are "Cyrillic" and "Catalan Government". The third one was the catalan word "drets" that has two meanings and the wrong one, "de pie", was chosen by the baseline; "de pie" stands for "stood (up)" while "derecho" means "right", like "human right". The fourth final system output corrected a morphological error, choosing the verb with the proper person and number. Pair number five presents two synonyms. The seventh pair adds a preposition that could also be omitted, as the baseline output did. Finally, the last pair presents two different ways of saying the same ("on the other hand") and they are equally valid even though the BLEU score is lower in the final system because the reference matches the baseline.

B:**por añadidura**, estos últimos años han coincidido (...)
F: **además**, estos últimos años han coincidido (...)

B:en **ciríl·lic** o chino
F: en **cirílico** o chino

B:destacó 3 puntos de actuación: universalizar **de pie**, hacer políticas (...)
F: destacó 3 puntos de actuación: universalizar **derechos**, hacer políticas (...)

B:(...) muchas de ellas no hace ni una semana **poder** enterrar a sus familiares (...)
F: (...) muchas de ellas no hace ni una semana **pudieron** enterrar a sus familiares (...)

B:(...) y tiene una pista de 100 metros de largo y 50 metros de **anchura**.
F: (...) y tiene una pista de 100 metros de largo y 50 metros de **ancho**.

B:(...) como en el caso de la gestión de el **Govern** (...)
F: (...) como en el caso de la gestión de el **Gobierno Catalán** (...)

B:la propuesta requiere una cuidadosa labor informativa (...)
F: la propuesta requiere **de** una cuidadosa labor informativa (...)

B:**por otro lado**, las poblaciones también están divididas, (...)
F: **por otra parte**, las poblaciones también están divididas, (...)

Figure 3: Output samples from the baseline and final systems. Every pair presents first the baseline output (labeled "B") and then the final system output (labeled "F"). The first four pairs are examples of a higher final BLEU, the last four pairs had a higher baseline BLEU.

6 Conclusions and further work.

We have presented a strategy to enhance a translation model with new and reinforced units using a revised corpus. A revised corpus was defined as a bilingual corpus together with the automatic translation of its source side. Therefore it is composed of a source side (coming from the bilingual corpus), a target output (coming from the translation system) and a target correction (coming from the bilingual corpus).

The strategy produces an adapted translation model with additional translation units and vocabulary, without the need of computing expensive word alignments or using the baseline corpus. Instead, it is based in the structure of the original translation units and the alignment provided by the comparison between the target output and the target correction.

This strategy consists in computing a sentence-by-sentence Levenshtein path, using the target output and the target correction. The Levenshtein path allows us to correct local errors found during decoding and to combine them with the source side to add additional tuples in the original translation model. At the same time, the method reinforces the original tuples that were correctly used during decoding in a specific context. We also defined a lexical filter that must be used to remove problematic units found during the extraction phase.

Results showed a statistical improvement with a confidence level of 99% in a state-of-the-art Catalan-to-Spanish Ngram-based SMT system. This was achieved using a bilingual corpus of $1.6K$ sentences, which represents less than 0.5% of the original corpus.

We plan to continue this line of research testing different language pairs and SMT paradigms. First, we will try an Spanish-to-English Ngram-based SMT system and then we will change it to a Phrase-based SMT systems. The Spanish-to-English experiments will explore the strategy for domain adaptation, using a big out-of-domain corpus to train the baseline translation model and a smaller in-domain bilingual corpus to derive the units from.

Acknowledgment.

References

Brown, Peter F., John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Rossin. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics*, 16:79–85.

Chiang, David. 2007. Hierarchical Phrase-Based Translation. *Computational Linguistics*, 33(2):201–228.

Crego, Josep M., Adrià de Gispert, and José B. Mariño. 2005. An Ngram-based Statistical Machine Translation Decoder. In *Proceedings of 9th European Conference on Speech Communication and Technology (Interspeech)*.

Farrús, Mireia, Marta R. Costa-jussà, Marc Poch, Adolfo Hernández, and José B. Mariño. 2009. Improving a catalan-spanish statistical translation system using morphosyntactic knowledge. In *Proceedings of European Association for Machine Translation 2009*.

Hardt, Daniel and Jakob Elming. 2010. Incremental re-training for post-editing SMT. In *AMTA 2010: the Ninth conference of the Association for Machine Translation in the Americas*.

Hildebrand, Almut Silja, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the Translation Model for Statistical Machine Translation based on Information Retrieval. In *EAMT 2005 Conference Proceedings*.

Isabelle, Pierre, Cyril Goutte, and Simard Michel. 2007. Domain Adaptation of MT Systems Through Automatic Post Editing. In *Machine Translation Summit XI*.

Koehn, Philipp and Josh Schroeder. 2007. Experiments in Domain Adaptation for Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *HLT-NAACL*, pages 48–54.

Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, volume 4, pages 388–395.

Koehn, Philipp. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Machine Translation Summit*.

Mariño, José B., Rafael E. Banchs, Josep M. Crego, Adrià de Gispert, Patrik Lambert, José A. R. Fonollosa, and Marta R. Costa-jussà. 2006. Ngram-based Machine Translation. *Computational Linguistics*, 32(4):527–549.

Mohit, Behrang, Frank Liberato, and Rebecca Hwa. 2009. Language Model Adaptation for Difficult to Translate Phrases. In *Proceedings of the 13th Annual Conference of the EAMT*.

Och, Franz Josef and Hermann Ney. 2002. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Padró, Lluís, Miquel Collado, Samuel Reese, Marina Lloberes, and Irene Castellón. 2010. FreeLing 2.1: Five Years of Open-Source Language Processing Tools. In *Proceedings of 7th Language Resources and Evaluation Conference (LREC 2010)*, La Valleta, Malta, May.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. IBM Research Report, RC22176, September.

Schwenk, Holger and Jean Senellart. 2009. Translation model adaptation for an Arabic/French news translation system by lightly-supervised training. In *MT Summit*.

Snover, Matthew, Bonnie Dorr, and Richard Schwartz. 2008. Language and Translation Model Adaptation using Comparable Corpora. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*.

Stolcke, A. 2002. Srilm - an extensible language modeling toolkit. In *International Conference on Spoken Language Processing*.

Ueffing, Nicola, Gholamreza Haffari, and Anoop Sarkar. 2007. Semi-supervised Model Adaptation for Statistical Machine Translation. *Machine Translation*, 21:77–94.

Xu, J., Y. Deng, Y. Gao, and Hermann Ney. 2007. Domain Dependent Statistical Machine Translation. In *Machine Translation Summit*, Copenhagen, Denmark, September.