Context Adaptation in Statistical Machine Translation Using Models with Exponentially Decaying Cache

Jörg Tiedemann
Department of Linguistics and Philology
Uppsala University, Uppsala/Sweden
jorg.tiedemann@lingfil.uu.se

Abstract

We report results from a domain adaptation task for statistical machine translation (SMT) using cache-based adaptive language and translation models. We apply an exponential decay factor and integrate the cache models in a standard phrase-based SMT decoder. Without the need for any domain-specific resources we obtain a 2.6% relative improvement on average in BLEU scores using our dynamic adaptation procedure.

1 Introduction

Most data-driven approaches to natural language processing (NLP) are subject to the well-known problem of lack of portability to new domains/genres. Usually there is a substantial drop in performance when testing on data from a domain different to the training data. Statistical machine translation is no exception. Despite its popularity, standard SMT approaches fail to provide a framework for general application across domains unless appropriate training data is available and used in parameter estimation and tuning.

The main problem is the general assumption of independent and identically distributed (i.i.d.) variables in machine learning approaches applied in the estimation of static global models. Recently, there has been quite some attention to the problem of domain switching in SMT (Zhao et al., 2004; Ueffing et al., 2007; Civera and Juan, 2007; Bertoldi and Federico, 2009) but ground breaking success is still missing. In this paper we report our findings in dynamic model adaptation using cache-based techniques when applying a standard model to the task of translating documents from a very different domain.

The remaining part of the paper is organized as follows: First, we will motivate the chosen approach by reviewing the general phenomenon of repetition and consistency in natural language text. Thereafter, we will briefly discuss the dynamic extensions to language and translation models applied in the experiments presented in the second last section followed by some final conclusions.

2 Motivation

Domain adaptation can be tackled in various ways. An obvious choice for empirical systems is to apply supervised techniques in case domain-specific training data is available. It has been shown that small(er) amounts of in-domain data are sufficient for such an approach (Koehn and Schroeder, 2007). However, this is not really a useful alternative for truly open-domain systems, which will be confronted with changing domains all the time including many new, previously unknown ones among them.

There are also some interesting approaches to dynamic domain adaptation mainly using flexible mixture models or techniques for the automatic selection of appropriate resources (Hildebrand et al., 2005; Foster and Kuhn, 2007; Finch and Sumita, 2008). Ideally, a system would adjust itself to the current context (and thus to the current domain) without the need of explicit topic mixtures. Therefore, we like to investigate techniques for general context adaptation and their use in out-of-domain translation.

There are two types of properties in natural language and translation that we like to explore. First of all, repetition is very common – much more than standard stochastic language models would predict. This is especially true for content words. See, for instance, the sample of a medical document shown in figure 1. Many content words are repeated in close context. Hence, appropriate language models should incorporate changing occurrence likelihoods to account for these very common repetitions. This is exactly what adaptive language models try to do (Bellegarda, 2004).

Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, ACL 2010, pages 8–15,
Uppsala, Sweden, 15 July 2010.

"They may also have **episodes** of depression . Abilify is used to treat moderate to severe **manic episodes** and to prevent **manic episodes** in patients who have responded to the **medicine** in the past . The solution for injection is used for the rapid control of agitation or disturbed behaviour when taking the **medicine** by mouth is not appropriate . The **medicine** can only be obtained with a prescription ."

Figure 1: A short example from a document from the European Medicines Agency (EMEA)

Another known fact about natural language is consistency which is also often ignored in statistical models. A main problem in most NLP applications is ambiguity. However, ambiguity is largely removed within specific domains and contexts in which ambiguous items have a well-defined and consistent meaning. This effect of "meaning consistency" also known as the principle of "one sense per discourse" has been applied in word sense disambiguation with quite some success (Gale et al., 1992). For machine translation this means that adapting to the local domain and sticking to consistent translation choices within a discourse seems to be better than using a global static model and context independent translations of sentences in isolation. For an illustration, look at the examples in figure 2 taken from translated movie subtitles. Interesting is not only the consistent meaning of "honey" within each discourse but also the consistent choice among equivalent translations (synonyms "älskling" och "gumman"). Here, the distinction between "honey" and "sweetheart" has been transferred to Swedish using consistent translations.

The 10 commandments	**Kerd ma lui**
To some land flowing with milk and **honey**! Till ett land fullt av mjölk och **honung**. I've never tasted **honey**. Jag har aldrig smakat **honung**. ...	Mari **honey** ... Mari, **gumman** ... **Sweetheart**, where are you going? **Älskling**, var ska du? ... Who was that, **honey**? Vem var det, **gumman**?

Figure 2: Consistency in subtitle translations

In summary: Repetition and consistency are very important when modeling natural language and translation. A proper translation engine should move away from translating sentences in isolation but should consider wider context to include these discourse phenomena. In the next section we discuss the cache-based models that we implemented to address this challenge.

3 Cache-based Models

The main idea behind cache-based language models (Kuhn and Mori, 1990) is to mix a large global (static) language model with a small local (dynamic) model estimated from recent items in the history of the input stream. It is common to use simple linear interpolations and fixed cache sizes k (100-5000 words) to achieve this: $P(w_n|history) = (1-\lambda)P_{n-gram}(w_n|history) + \lambda P_{cache}(w_n|history)$

Due to data sparseness one is usually restricted to simple cache models. However, unigram models are often sufficient and smoothing is not necessary due to the interpolation with the smoothed background model. From the language modeling literature we know that caching is an efficient way to reduce perplexity (usually leading to modest improvements on in-domain data and large improvements on out-of-domain data). Table 1 shows this effect yielding 53% reduction of perplexity on our out-of-domain data.

	different settings for λ			
cache	0.05	0.1	0.2	0.3
0	376.1	376.1	376.1	376.1
50	270.7	259.2	*256.4*	264.9
100	261.1	246.6	*239.2*	243.3
500	252.2	233.1	219.1	*217.0*
1000	240.6	218.0	199.2	*192.9*
2000	234.6	209.6	187.9	*179.1*
5000	235.3	209.1	185.8	**175.8**
10000	237.6	210.7	186.6	*176.1*
20000	239.9	212.5	187.7	*176.7*

Table 1: Perplexity of medical texts (EMEA) using a language model estimated on Europarl and a unigram cache component

Even though a simple unigram cache is quite effective it now requires a careful optimization of its size. In order to avoid the dependence on cache size and to account for recency a decaying factor can be introduced (Clarkson and Robinson, 1997):

$$P_{cache}(w_n|w_{n-k}..w_{n-1}) \approx \frac{1}{Z}\sum_{i=n-k}^{n-1} I(w_n = w_i)e^{-\alpha(n-i)}$$

Here, $I(A) = 1$ if A is true and 0 otherwise. Z is a normalizing constant. Figure 3 illustrates the effect of cache decay on our data yielding another significant reduction in perplexity (even though

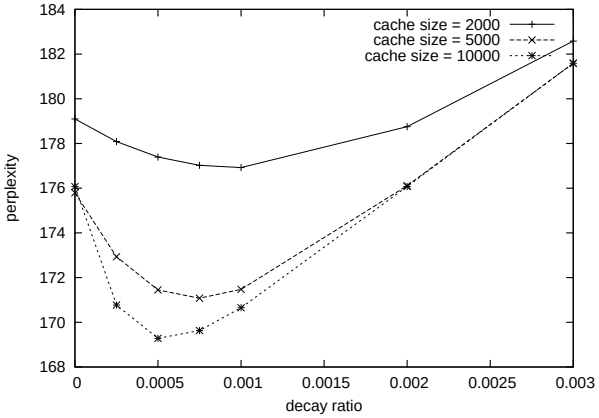

Figure 3: Out-of-domain perplexity using language models with decaying cache.

the improvement is much less impressive than the one obtained by introducing the cache).

The motivation of using these successful techniques in SMT is obvious. Language models play a crucial role in fluency ranking and a better fit to real data (supporting the tendency of repetition) should be preferred. This, of course, assumes correct translation decisions in the history in our SMT setting which will almost never be the case. Furthermore, simple cache models like the unigram model may wrongly push forward certain expressions without considering local context when using language models to discriminate between various translation candidates. Therefore, successfully applying these adaptive language models in SMT is surprisingly difficult (Raab, 2007) especially due to the risk of adding noise (leading to error propagation) and corrupting local dependencies.

In SMT another type of adaptation can be applied: cache-based adaptation of the translation model. Here, not only the repetition of content words is supported but also the consistency of translations as discussed earlier. This technique has already been tried in the context of interactive machine translation (Nepveu et al., 2004) in which cache features are introduced to adapt both the language model and the translation model. However, in their model they require an automatic alignment of words in the user edited translation and the source language input. In our experiments we investigate a close integration of the caching procedure into the decoding process of fully automatic translation. For this, we fill our cache with translation options used in the best (final) translation hypothesis of previous sentences. In our implementation of the translation model cache we use again a decaying factor in order to account for recency. For known source language items (f_n for which translation options exist in the cache) the following formula is used to compute the cache translation score:

$$\phi_{cache}(e_n|f_n) = \frac{\sum_{i=1}^{K} I(\langle e_n, f_n\rangle = \langle e_i, f_i\rangle) * e^{-\alpha i}}{\sum_{i=1}^{K} I(f_n = f_i)}$$

Unknown items receive a score of zero. This score is then used as an additional feature in the standard log-linear model of phrase-based SMT[1].

4 Experiments

Our experiments are focused on the unsupervised dynamic adaptation of language and translation models to a new domain using the cache-based mixture models as described above. We apply these techniques to a standard task of translating French to English using a model trained on the publicly available Europarl corpus (Koehn, 2005) using standard settings and tools such as the Moses toolkit (Koehn et al., 2007), GIZA++ (Och and Ney, 2003) and SRILM (Stolcke, 2002). The log-linear model is then tuned as usual with minimum error rate training (Och, 2003) on a separate development set coming from the same domain (Europarl). We modified SRILM to include a decaying cache model and implemented the phrase translation cache within the Moses decoder. Furthermore, we added the caching procedures and other features for testing the adaptive approach. Now we can simply switch the cache models on or off using additional command-line arguments when running Moses as usual.

4.1 Experimental Setup

For testing we chose to use documents from the medical domain coming from the EMEA corpus that is part of the freely available collection of parallel corpora OPUS[2] (Tiedemann, 2009). The reason for selecting this domain is that these documents include very consistent instructions and repetitive texts which ought to favor our caching techniques. Furthermore, they are very different

[1] Logarithmic values are used in the actual implementation which are floored to a low constant in case of zero ϕ scores.

[2] The OPUS corpus is available at this URL: `http://www.let.rug.nl/tiedeman/OPUS/`.

from the training data and, thus, domain adaptation is very important for proper translations. We randomly selected 102 pairs of documents with altogether 5,478 sentences. Sentences have an average length of about 19 tokens with a lot of variation among them. Documents are compiled from the European Public Assessment Reports (EPAR) which reflect scientific conclusions at the end of a centralized evaluation procedure for medical products. They include a lot of domain-specific terminology, short facts, lists and tables but also detailed textual descriptions of medicines and their use. The overall lowercased type/token ratio in the English part of our test collection is about 0.045 which indicates quite substantial repetitions in the text. This ratio is, however, much higher for individual documents.

In the experiment each document is processed individually in order to apply appropriate discourse breaks. The baseline score for applying a standard phrase-based SMT model yields an average score of 28.67 BLEU per document (28.60 per sentence) which is quite reasonable for an out-of-domain test. Intuitively, the baseline performance should be crucial for the adaptation. As discussed earlier the cache-based approach assumes correct history and better baseline performance should increase the chance of adding appropriate items to the cache.

4.2 Applying the LM Cache

In our first experiment we applied a decaying unigram cache in the language model. We performed a simple linear search on a separate development set for optimizing the interpolation weight which gave as a value of $\lambda = 0.001$. The size of the cache was set to 10,000 and the decay factor was set to $\alpha = 0.0005$ (according to our findings in figure 3). The results on our test data compared to the standard model are illustrated (with white boxes) in figure 4.

There is quite some variation in the effect of the cache LM on our test documents. The translations of most EMEA documents could be improved according to BLEU scores, some of them substantially, whereas others degraded slightly. Note that the documents differ in size and some of them are very short which makes it a bit difficult to interpret and directly compare these scores. On average the BLEU score is improved by 0.43 points per document and 0.39 points per sentence. This might

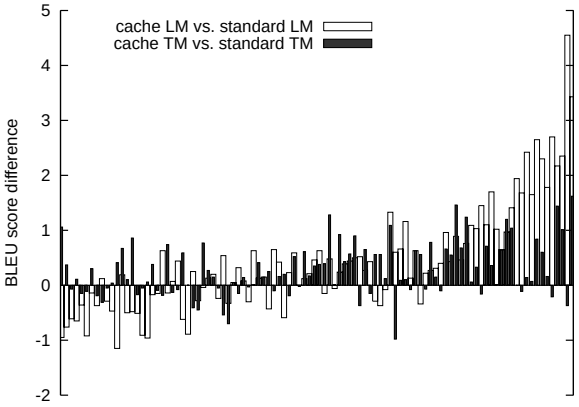

Figure 4: The differences in BLEU between a standard model and models with cache for 102 EMEA documents (sorted by overall BLEU score gain – see figure 5)

be not as impressive as we were hoping for after the tremendous perplexity reduction presented earlier. However, considering the simplicity of the approach that does not require any additional resources nor training it is still a valuable achievement.

4.3 Applying the TM Cache

In the next experiment we tested the effect of the TM cache on translation quality. Using our hypothesis of translation consistency we expected another gain on our test set. In order to reduce problems of noise we added two additional constraints: We only cache phrases that contain at least one word longer than 4 characters (a simplistic attempt to focus on content words rather than function words) and we only cache translation options for which the transition costs (of adding this option to the current hypothesis) in the global decoding model is larger than a given threshold (an attempt to use some notion of confidence for the current phrase pair; in our experiments we used a log score of -4). Using this setup and applying the phrase cache in decoding we obtained the results illustrated with filled boxes in the figure 4 above.

Again, we can observe a varied outcome but mostly improvements. The impact of the phrase translation cache (with a size of 5,000 items) is not as strong as for the language model cache which might be due to the rather conservative settings ($\lambda = 0.001$, $\alpha = 0.001$) and the fact that matching phrase pairs are less likely to appear than matching target words. On average the gain is about 0.275

BLEU points per document (0.26 per sentence).

4.4 Combining the Cache Models

Finally, we applied both types of cache in one common system using the same settings from the individual runs. The differences to the baseline model are shown in figure 5.

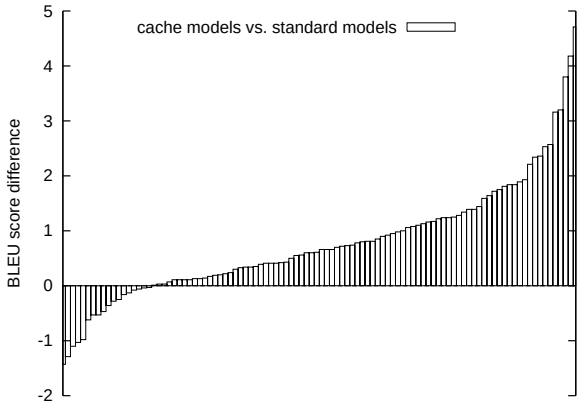

Figure 5: The BLEU score differences between a standard model and a model with cache for both TM and LM (sorted by BLEU score gain).

In most cases, applying the two types of cache together has a positive effect on the final BLEU score. Now, we see only a few documents with a drop in translation performance. On average the gain has increased to about 0.78 BLEU points per document (0.74 per sentence) which is about 2.7% relative improvement compared to the baseline (2.6% per sentence).

5 Discussion

Our experiments seem to suggest that caching could be a way to improve translation quality on a new domain. However, the differences are small and the assumption that previous translation hypotheses are good enough to be cached is risky. One obvious question is if the approach is robust enough to be helpful in general. If that is the the case we should also see positive effects on in-domain data where a cache model could adjust to topical shifts within that domain. In order to test this ability we ran an experiment with the 2006 test data from the workshop on statistical machine translation (Koehn and Monz, 2006) using the same models and settings as above. This resulted in the following scores (lowercased BLEU):

$BLEU_{baseline}$ = 32.46 (65.0/38.3/25.4/17.6, BP=0.999)
$BLEU_{cache}$ = 31.91 (65.1/38.1/25.1/17.3, BP=0.991)

Clearly, the cache models failed on this test even though the difference between the two runs is not large. There is a slight improvement in unigram matches (first value in brackets) but a drop on larger n-gram scores and also a stronger brevity penalty (BP). This could be an effect of the simplicity of the LM cache (a simple unigram model) which may improve the choice of individual lexical items but without respecting contextual dependencies.

One difference is that the in-domain data was translated in one step without clearing the cache at topical shifts. EMEA documents were translated one by one with empty caches at the beginning. It is now the question if proper initialization is essential and if there is a correlation between document length and the effect of caching. How much data is actually needed to take advantage of cached items and is there a point where a positive effect degrades because of topical shifts within the document? Let us, therefore, have a look at the relation between document length and BLEU score gain in our test collection (figure 6).

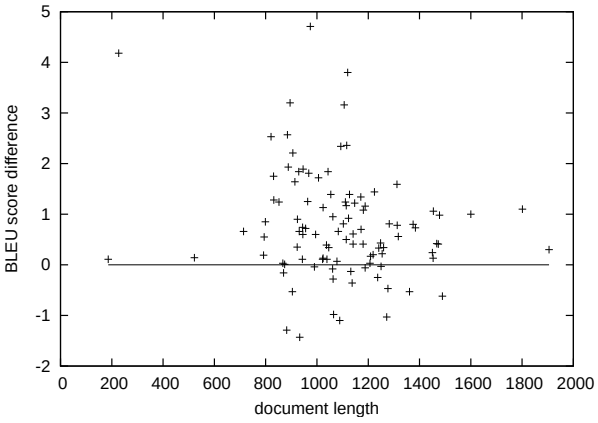

Figure 6: Correlation between document lengths (in number of tokens) and BLEU score gains with caching.

Concluding from this figure there does not seem to be any correlation. The length of the document does not seem to influence the outcome. What else could be the reason for the different behaviour among our test documents? One possibility is the quality of baseline translations assuming that better performance increases the chance of caching correct translation hypotheses. Figure 7 plots the BLEU score gains in comparison with the baseline scores.

Again, no immediate correlation can be seen.

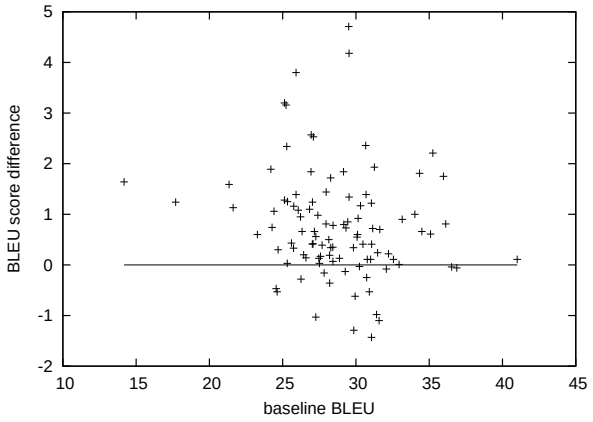

Figure 7: Correlation between baseline BLEU scores and BLEU score gains with caching

The baseline performance does not seem to give any clues for a possible success of caching. This comes as a surprise as our intuitions suggested that good baseline performance should be essential for the adaptive approach.

Another reason for their success should be the amount of repetition (especially among content words) in the documents to be translated. An indication for this can be given by type/token ratios assuming that documents with lower ratios contain a larger amount of repetitive text. Figure 8 plots the type/token ratios of all test documents in comparison with the BLEU score gains obtained with caching.

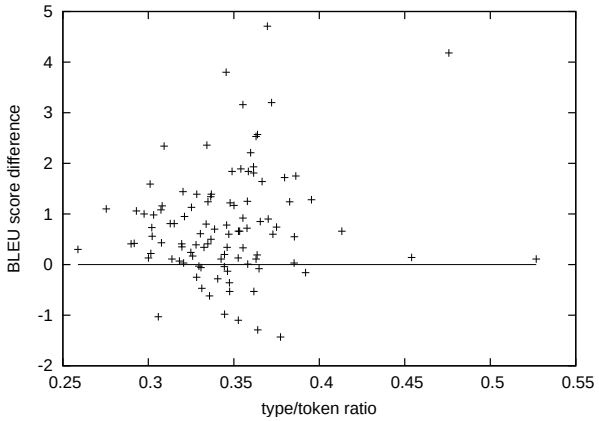

Figure 8: Correlation between type/token ratios and BLEU score gains with caching

Once again there does not seem to be any obvious correlation. So far we could not identify any particular property of documents that might help to reliably predict the success of caching. The answer is probably a combination of various factors. Further experiments are needed to see the effect on different data sets and document types.

Note that some results may also be an artifact of the automatic evaluation metrics applied. Qualitative evaluations using manual inspection could probably reveal important aspects of the caching approach. However, tracing changes caused by caching is rather difficult due to the interaction with other factors in the global decoding process. Some typical cases may still be identified. Figure 9 shows an example of a translation that has been improved in the cached model by making the translation more consistent (this is from a document that actually got a lower BLEU score in the end with caching).

baseline: report (evaluation of european public epar)
vivanza
in the short epar public
this document is a summary of the european public to evaluation report (epar) .

cache: report european public assessment (epar)
vivanza
epar to sum up the public
this document is a summary of the european public assessment report (epar) .

reference: european public assessment report (epar)
vivanza
epar summary for the public
this document is a summary of the european public assessment report (epar) .

Figure 9: A translation improved by caching.

Other improvements may not be recognized by automatic evaluation metrics and certain acceptable differences may be penalized. Look, for instance, at the examples in figure 10.

This is, of course, not a general claim that cache-based translations are more effected by this problem than, for example, the baseline system. However, this could be a direction for further investigations to quantify these issues.

6 Conclusions

In this paper we presented adaptive language and translation models that use an exponentially decaying cache. We applied these models to a domain adaptation task translating medical documents with a standard model trained on Europarl. On average the dynamic adaptation approach led to a gain of about 2.6% relative BLEU points per sentence. The main advantage of this approach is that it does not require any domain-specific train-

baseline:	the medication is issued on orders .
cache:	the medication is issued on **prescription-only** .
reference:	the medicine can **only** be obtained with a **prescription** .
baseline:	benefix **is a powder** keg , and a solvent to dissolve the injection for .
cache:	benefix **consists of a powder** and a solvent to dissolve the injection for .
reference:	benefix **is a powder** and solvent that are mixed together for injection .
baseline:	the principle of active benefix is the nonacog alfa (ix coagulation factor of recombinant) which favours the coagulation blood .
cache:	the principle of benefix is the nonacog alfa (ix coagulation factor of recombinant) which favours the coagulation blood .
reference:	benefix contains the active ingredient nonacog alfa (recombinant coagulation factor ix , which helps blood to clot) .
baseline:	**in any case** , it is benefix used ?
cache:	**in which case** it is benefix used ?
reference:	**what is** benefix used for ?
baseline:	benefix is used for the treatment and prevention of saignements among **patients with haemophilia b** (a disorder hémorragique hereditary due to a deficiency in factor ix) .
cache:	benefix is used for the treatment and prevention of saignements among **patients suffering haemophilia b** (a disorder hémorragique hereditary due to a lack factor in ix) .
reference:	benefix is used for the treatment and prevention of bleeding in **patients with haemophilia b** (an inherited bleeding disorder caused by lack of factor ix) .
baseline:	benefix can be used for adults and children **over** 6 years .
cache:	benefix can be used for adults and children **of more than** 6 years
reference:	benfix can be used in adults and children **over the age of** 6.

Figure 10: Examples translations with and without caching.

ing, tuning (assuming that interpolation weights and other cache parameters can be fixed after some initial experiments) nor the incorporation of any other in-domain resources. Cache based adaptation can directly be applied to any new domain and similar gains should be possible. However, a general conclusion cannot be drawn from our initial results presented in this paper. Further experiments are required to verify these findings and to explore the potentials of cache-based techniques. The main obstacle is the invalid assumption that initial translations are correct. The success of the entire method crucially depends on this assumption. Error propagation and the reinforcement of wrong decisions is the largest risk. Therefore, strategies to reduce noise in the cache are important and can still be improved using better selection criteria. A possible strategy could be to identify simple cases in a first run that can be used to reliably fill the cache and to use the full cache model on the entire text in a second run. Another idea for improvement is to attach weights to cache entries according to the translation costs assigned by the model. These weights could easily be incorporated into the cache scores returned for matching items. In future, we would like to explore these ideas and also possibilities to combine cache models with other types of adaptation techniques.

References

Jerome R. Bellegarda. 2004. Statistical language model adaptation: review and perspectives. *Speech Communication*, 42:93–108.

Nicola Bertoldi and Marcello Federico. 2009. Domain adaptation for statistical machine translation with monolingual resources. In *StatMT '09: Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 182–189, Morristown, NJ, USA. Association for Computational Linguistics.

Jorge Civera and Alfons Juan. 2007. Domain adaptation in statistical machine translation with mixture modelling. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

P.R. Clarkson and A. J. Robinson. 1997. Language model adaptation using mixtures and an exponentially decaying cache. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 799–802, Munich, Germany.

Andrew Finch and Eiichiro Sumita. 2008. Dynamic model interpolation for statistical machine translation. In *StatMT '08: Proceedings of the Third Workshop on Statistical Machine Translation*, pages 208–215, Morristown, NJ, USA. Association for Computational Linguistics.

George Foster and Roland Kuhn. 2007. Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 128–135, Prague, Czech Republic. Association for Computational Linguistics.

William A. Gale, Kenneth W. Church, and David Yarowsky. 1992. One sense per discourse. In *HLT '91: Proceedings of the workshop on Speech and Natural Language*, pages 233–237, Morristown, NJ, USA. Association for Computational Linguistics.

Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of the 10th Conference of the European Association for Machine Translation (EAMT)*, pages 133–142, Budapest.

Philipp Koehn and Christof Monz, editors. 2006. *Proceedings on the Workshop on Statistical Machine Translation*. Association for Computational Linguistics, New York City, June.

Philipp Koehn and Josh Schroeder. 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. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *ACL '07: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180, Morristown, NJ, USA. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit (MT Summit X)*.

Roland Kuhn and Renato De Mori. 1990. A cache-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6):570–583.

Laurent Nepveu, Lapalme, Guy, Langlais, Philippe, and George Foster. 2004. Adaptive Language and Translation Models for Interactive Machine Translation. In *Proceedings of the 9th Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 190–197, Barcelona, Spain.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

Martin Raab. 2007. *Language Modeling for Machine Translation*. VDM Verlag, Saarbrücken, Germany.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of the 7th international conference on spoken language processing (ICSLP 2002)*, pages 901–904, Denver, CO, USA.

Jörg Tiedemann. 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia.

Nicola Ueffing, Gholamreza Haffari, and Anoop Sarkar. 2007. Semi-supervised model adaptation for statistical machine translation. *Machine Translation*, 21(2):77–94.

Bing Zhao, Matthias Eck, and Stephan Vogel. 2004. Language model adaptation for statistical machine translation with structured query models. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 411, Morristown, NJ, USA. Association for Computational Linguistics.