Alignment Templates: the RWTH SMT System

Oliver Bender, Richard Zens, Evgeny Matusov, Hermann Ney

Chair of Computer Science VI
RWTH Aachen University
D-52056 Aachen, Germany
{bender,zens,matusov,ney}@cs.rwth-aachen.de

Abstract

In this paper, we describe the RWTH statistical machine translation (SMT) system which is based on log-linear model combination. All knowledge sources are treated as feature functions which depend on the source language sentence, the target language sentence and possible hidden variables. The main feature of our approach are the *alignment templates* which take shallow phrase structures into account: a phrase level alignment between phrases and a word level alignment between single words within the phrases. Thereby, we directly consider word contexts and local reorderings. In order to incorporate additional models (the IBM-1 statistical lexicon model, a word deletion model, and higher order language models), we perform n-best list rescoring. Participating in the International Workshop on Spoken Language Translation (IWSLT 2004), we evaluate our system on the *Basic Travel Expression Corpus* (BTEC) Chinese-to-English and Japanese-to-English tasks.

1. Introduction

The goal of machine translation is the translation of a text given in some source language into a target language. We are given a source string $f_1^J = f_1...f_j...f_J$, which is to be translated into a target string $e_1^I = e_1...e_i...e_I$. Among all possible target strings, we will choose the string with the highest probability:

$$\begin{aligned} \hat{e}_1^I &= \operatorname*{argmax}_{e_1^I} \left\{ Pr(e_1^I|f_1^J) \right\} \\ &= \operatorname*{argmax}_{e_1^I} \left\{ Pr(e_1^I) \cdot Pr(f_1^J|e_1^I) \right\} \end{aligned} \quad (1)$$

The decomposition into two knowledge sources in Equation 1 is known as the source-channel approach to statistical machine translation [1]. It allows an independent modeling of target language model $Pr(e_1^I)$ and translation model $Pr(f_1^J|e_1^I)$. The target language model describes the well-formedness of the target language sentence. The translation model links the source language sentence to the target language sentence. It can be further decomposed into alignment and lexicon model.

An alternative to the classical source-channel approach is the direct modeling of the posterior probability $Pr(e_1^I|f_1^J)$. Using a log-linear model [2], we obtain:

$$\begin{aligned} Pr(e_1^I|f_1^J) &= p_{\lambda_1^M}(e_1^I|f_1^J) \\ &= \frac{\exp \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J)}{\sum_{e'^I_1} \exp \sum_{m=1}^{M} \lambda_m h_m(e'^I_1, f_1^J)} \end{aligned}$$

The h_m denote the feature functions. As a decision rule, we obtain:

$$\hat{e}_1^I = \operatorname*{argmax}_{e_1^I} \left\{ \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J) \right\} \quad (2)$$

This approach is a generalization of the source-channel approach. It has the advantage that additional models or feature functions can be easily integrated into the overall system. The overall architecture of the log-linear model combination is summarized in Figure 1.

The argmax operation denotes the search problem, i.e. the generation of the output sentence in the target language. We have to maximize over all possible target language sentences.

In a way similar to [3], we train the model scaling factors λ_1^M with respect to the final translation quality measured by some error criterion, e.g. the NIST score [4], the BLEU score [5] or the word error rate (WER) [6].

The remainder of the paper is organized as follows: in section 2, we will outline the RWTH statistical machine translation system which introduces the alignment templates [7, 2]. We will describe the training and search procedure of our approach. For the Japanese-English task, we will show that reordering constraints improve translation quality compared to an unconstrained search. We will describe the additional features we integrate into our system. Section 3 will present experimental details and will show the translation results obtained for the Chinese-to-English and Japanese-to-English evaluation tasks. Finally, section 4 will conclude.

2. The RWTH SMT System

A general deficiency of single-word based approaches is that contextual information is not taken into account because they are only able to model correspondences between single words. A countermeasure is to consider word phrases rather than single words as the basis for the translation models. In other words, a whole group of adjacent words in the source sentence may be aligned with a whole group of adjacent words in the target language. As a result the context of words has a greater influence and local reorderings can be learned implicitly..

2.1. Word level alignments

The main feature of our translation model are the *alignment templates*. An alignment templates z is a triple $(\tilde{f}, \tilde{E}, \tilde{A})$ which describes the alignment $\tilde{A}$ between a source class sequence $\tilde{F}$ and a target class sequence $\tilde{E}$. The classes used in $\tilde{F}$ and $\tilde{E}$ are automatically trained bilingual classes using the method described in [8]. The use of classes instead of words themselves has the advantage of a better generalization. E.g., if a class "town" is used in both source and target language and alignment templates are learned for special towns, it is possible to generalize these alignment templates to all towns.

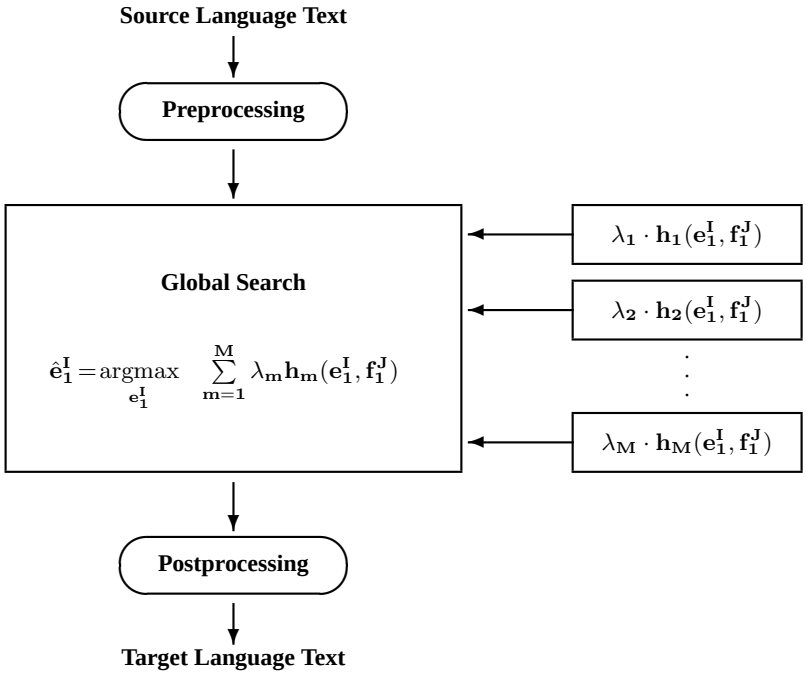

Figure 1: Architecture of the translation approach based on log-linear model combination.

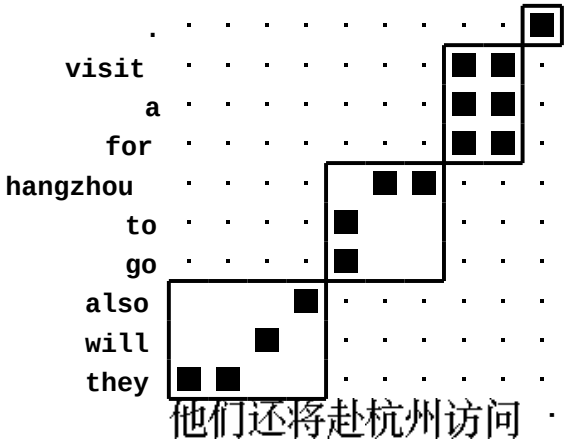

Figure 2: Example of a word aligned sentence pair and some possible alignment templates.

An alignment template is applicable to a sequence of source words if the alignment template classes and the classes of the source words are equal, and it constrains the target words to correspond to the target class sequence. For the selection of words from classes we use a statistical model for $p(\tilde{f}|z, \tilde{e})$ based on the lexicon probabilities of a statistical lexicon $p(f|e)$.

Figure 2 shows an example of a word aligned sentence pair. The word alignment is represented with the black boxes. The figure also includes some of the possible alignment templates, represented as the larger, unfilled rectangles. Note that the extraction algorithm would extract many more alignment templates from this sentence pair. In this example, the system input was the sequence of Chinese characters without any word segmentation.

2.2. Phrase level alignments

In order to describe the phrase level alignments in a formal way, we first decompose both the source sentence f_1^J and the target sentence e_1^I into a sequence of phrases ($k = 1, \ldots, K$). For the alignment a_1^K between the word phrases, we obtain the following equation:

$$\begin{aligned} Pr(f_1^J|e_1^I) &= \sum_{a_1^K} Pr(a_1^K, f_1^J|e_1^I) \\ &= \sum_{a_1^K} Pr(a_1^K|e_1^I) \cdot Pr(f_1^J|a_1^K, e_1^I) \end{aligned}$$

Further, we introduce the alignment templates as hidden variables for the translation of the K phrases:

$$\begin{aligned} Pr(f_1^J|e_1^I) &= \sum_{a_1^K, z_1^K} Pr(a_1^K|e_1^I) \cdot \\ & \quad Pr(z_1^K|a_1^K, e_1^I) \cdot Pr(f_1^J|z_1^K, a_1^K, e_1^I) \end{aligned} \quad (3)$$

Hence, we obtain three different probability distributions: the phrase alignment probability $Pr(a_1^K|e_1^I)$, the probability to apply an alignment template $Pr(z_1^K|a_1^K, e_1^I)$, and the phrase translation probability $Pr(f_1^J|z_1^K, a_1^K, e_1^I)$. The phrase translation probability is discussed in section 2.1. For a detailed description of modeling, training and search, see [7].

2.3. Feature functions

To use the three component models of Equation 3 in a log-linear approach, we define three different feature functions taking the logarithm for each component of the translation model instead of one feature function for the whole translation model $p(f_1^J|e_1^I)$. The feature functions have then not only a dependence on f_1^J and e_1^I but also on z_1^K, a_1^K. Yet, we are not limited to train only the alignment model scaling factors, the RWTH SMT system consists of the following base models:

- a phrase translation model,
- a phrase alignment model,
- a word translation model,
- a word-based trigram language model,
- a class-based five-gram language model, and
- a word penalty model.

These features allow a straightforward integration into the used dynamic programming search algorithm [7]. In addition, we extract n-best candidate translations using A^* search [9] and perform rescoring, for which we make use of the following extended models:

- the IBM-1 lexicon model as suggested by [10],
- a deletion model: for each source word, we check wether there exists a target translation with a probability higher than a given threshold. If not, this word is considered as deletion and the feature simply counts the number of deletions,
- additional language models: applying the SRI Language Modeling Toolkit [11], we train n-gram language models of increasing order.

We combine these different features in a log-linear model [2].

2.4. Optimization of model scaling factors

As training criterion, we use the maximum class posterior probability criterion:

$$\hat{\lambda}_1^M = \operatorname*{argmax}_{\lambda_1^M} \left\{ \sum_{s=1}^{S} \log p_{\lambda_1^M}(\mathbf{e}_s|\mathbf{f}_s) \right\} \quad (4)$$

on a parallel training corpus of sentence pairs $(\mathbf{f}_s, \mathbf{e}_s), s = 1, \ldots, S$. This criterion allows for only one reference translation, but for our tasks there exist multiple reference translations. Hence, we change the criterion to allow R_s reference translations $\mathbf{e}_{s,1}, \ldots, \mathbf{e}_{s,R_s}$ for the sentence $\mathbf{e}_s$:

$$\hat{\lambda}_1^M = \operatorname*{argmax}_{\lambda_1^M} \left\{ \sum_{s=1}^{S} \frac{1}{R_s} \sum_{r=1}^{R_s} \log p_{\lambda_1^M}(\mathbf{e}_{s,r}|\mathbf{f}_s) \right\}$$

We use this optimization criterion instead of the optimization criterion shown in Equation 4.

The model scaling factors are optimized on the development corpus with respect to the NIST score in a way similar to [3]. We use the downhill simplex algorithm from [12]. We do not perform the optimization on n-best lists but we retranslate the whole development corpus for each iteration of the optimization algorithm. In the experiments, the downhill simplex algorithm converged after about 200 iterations. This method has the advantage that it is not limited to the model scaling factors as the method described in [3].

2.5. Search

The base models described in section 2.3 are integrated into the used dynamic programming search algorithm [7]. Instead of Equation 1, we use the following search criterion:

$$\hat{e}_1^I = \operatorname*{argmax}_{e_1^I} \left\{ p(e_1^I) \cdot p(e_1^I|f_1^J) \right\} \quad (5)$$

This simplifies the search process as shown in [2]. As experiments have shown this approximation does not affect the quality of translation results.

The memory requirements for the alignment templates approach are quite large. To reduce these requirements for offline experiments, we apply a special method that works as follows. For each observed source word group (length typically two to twelve) in the test data we check wether the same word group has occurred in the training data. If yes, we calculate an alignment template model for this specific word group. In other words, we compute alignment templates models only for those words that occur in the test data.

Subsequently the actual translation process begins. It has a search organization along the positions of the target language string. During the search, we produce partial hypotheses which are extended by appending one target word. The set of all partial hypotheses can be structured as a graph with a source node representing the sentence start, leaf nodes representing full translations and intermediate nodes representing partial hypotheses. We recombine partial hypotheses which we do not have to distinguish by neither language model nor translation model. We also use beam-search in order to handle the huge search space.

Furthermore, we compute n-best lists [9] and rescore the candidate translations with the additional models described in section 2.3.

2.6. Reordering constraints

Within the alignment templates, the reordering is learned in training and kept fix during the search process. There are no constraints on the reorderings within the alignment templates.

Although unconstrained reordering looks perfect from a theoretical point of view, we found in [13] that constrained reordering shows better performance at least for the Japanese-to-English task. We used constraints based on *inversion transduction grammars* (ITG) [14, 15]. Here, we interpret the input sentence as a sequence of blocks. In the beginning, each alignment template is a block of its own. Then, the reordering process can be interpreted as follows: we select two consecutive blocks and merge them to a single block by choosing between two options: either keep the target phrases in monotone order or invert the order. This idea is illustrated in Figure 3. The dark boxes represent the two blocks to be merged. Once two blocks are merged, they are treated as a single block and they can be only merged further as a whole. It is not allowed to merge one of the sub-blocks again.

3. Translation Results

Experiments were carried out on the *Basic Travel Expression Corpus* (BTEC) task [16]. This is a multilingual speech corpus which contains tourism-related sentences similar to those that are usually found in phrase books for tourists going abroad. In particular, the participants of the International Workshop on Spoken Language Translation (IWSLT 2004) were asked to test their systems on the Chinese-to-English and the Japanese-to-English task. For both translation directions different tracks were specified depending on the amount of training data that was allowed to use. We took part in the following tracks:

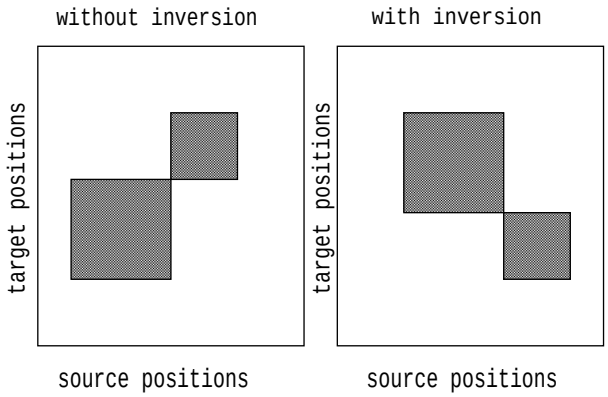

Figure 3: Illustration of monotone and inverted concatenation of two consecutive blocks.

Table 1: Statistics of the BTEC corpus for the Chinese-to-English Small Data Track

		Chinese	English
train	sentences	20 000	
	words	182 904	160 523
	singletons	3 525	2948
	vocabulary	7 643	6 982
dev	sentences	506	
	words	3 515	3 595
test	sentences	500	
	words	3 794	–

- Small Data Track:
  The training data of the MT systems was limited to the supplied corpora only. Here, we evaluated our system for both language pairs.
- Unrestricted Data Track:
  There were no limitations on the linguistic resources used to train the MT systems. We only worked on the Japanese-to-English translation direction.

The corpus statistics for these tracks are shown in Table 1 to 3. For both language pairs, 20 000 sentences randomly selected from the full BTEC corpus were supplied for training purposes, plus the CSTAR 2003 test set consisting of 506 sentence pairs as development corpus and the official 500 sentence test set for IWSLT 2004. As additional training resources for the unrestricted data track we included the full BTEC Japanese-to-English corpus and the *Spoken Language DataBase* (SLDB) [17], which consists of transcriptions of spoken dialogs in the domain of hotel reservation[1].

3.1. Evaluation specifications

So far, no generally accepted, automatic criterion exists in machine translation for the evaluation of the experimental results. Therefore, the evaluation of the translation quality was twofold:

1. Subjective Evaluation as specified by the IWSLT 2004 consortium:

[1]All corpora (BTEC, SLDB, and the CSTAR test sets) were kindly provided by ATR Spoken Language Translation Research Laboratories Kyoto, Japan.

Table 2: Statistics of the BTEC corpus for the Japanese-to-English Small Data Track

		Japanese	English
train	sentences	20 000	
	words	209 012	160 427
	singletons	4 108	2 956
	vocabulary	9 277	6 932
dev	sentences	506	
	words	4 374	3 595
test	sentences	500	
	words	4 370	–

Table 3: Statistics of the BTEC corpus for the Japanese-to-English Unrestricted Data Track

		Japanese	English
train	sentences	240 672	
	words	1 974 407	1 770 190
	singletons	8 975	3 658
	vocabulary	26 037	14 301
dev	sentences	506	
	words	3 515	3 595
test	sentences	500	
	words	3 794	–

   - Human assessments of translation quality with respect to the ”fluency” and ”adequacy” of the translation results.
   - ”Fluency” indicates how the evaluation segment sounds to a native speaker of English. The evaluator graded the level of English used in the translation from 1 (”Incomprehensible”) to 5 (”Flawless English”).
   - The ”adequacy” assessment is carried out after the fluency judgement was done. The evaluator was presented with the ”gold standard” translation and had to judge how much of the information from the original translation was expressed in the translation by selecting one of the grades from 1 (”None of it”) to 5 (”All of the information”).
2. Automatic Evaluation:
   In all experiments, the following error criteria were used:
   - WER (word error rate):
     The WER is computed as the minimum number of substitution, insertion and deletion operations that have to be performed to convert the generated sentence into the reference sentence.
   - PER (position-independent word error rate):
     A shortcoming of the WER is that it requires a perfect word order. The word order of an acceptable sentence can be different from that of the target sentence, so that the WER measure alone could be misleading. The PER compares the words in the two sentences ignoring the word order.
   - BLEU score:
     This score measures the precision of unigrams, bi-

Table 4: Translation performance of the official run submissions for the BTEC task (500 sentences).

Language Pair	Data Track	Automatic Evaluation					Subjective Evaluation	
		mWER [%]	mPER [%]	BLEU [%]	NIST	GTM [%]	Fluency	Adequacy
Chinese-to-English	Small	45.6	39.0	40.9	8.55	72.1	3.36	3.34
Japanese-to-English	Small	41.9	33.8	45.3	9.49	76.4	3.48	3.41
	Unrestricted	30.6	24.9	61.9	10.72	79.7	4.04	4.07

grams, trigrams and fourgrams with respect to a reference translation with a penalty for too short sentences [18]. BLEU measures accuracy, i.e. large BLEU scores are better.

- NIST score:
  This score is similar to BLEU. It is a weighted n-gram precision in combination with a penalty for too short sentences [4]. NIST measures accuracy, i.e. large NIST scores are better.
- GTM score:
  The General Text Matcher (GTM) [19] is a tool which measures the similarity between texts in terms of precision and recall. GTM measures accuracy, i.e. large GTM scores are better.

For the BTEC tasks, we had multiple references available. Therefore, we computed all the preceding criteria with respect to multiple references. To indicate this, we will precede the acronyms with an m (multiple) if multiple references are used.

3.2. Evaluation results

We start with the official IWSLT 2004 evaluation results for our system. Multiple system submissions for each data track were permitted, but each participant had to mark a primary system and that was going to be evaluated by humans. The results are summarized in Table 4. We see, that the subjective scores are very similar for both language pairs although the performance according to the automatic error criteria is better for the Japanese-to-English task. If we train our system on the full BTEC corpus extended by the SLDB corpus, we observe that the overall quality increases significantly and is rather high on this task. In practice, we found that the subjective accuracy measures seem to be mostly correlated with the NIST score. Hence, we optimized the model scaling factors according to the translation quality with respect to the NIST score. We also did some experiments in which we optimized the model scaling factors with respect to other error criteria, but we found out that the best overall performance is achieved by optimizing our system with respect to the NIST score. E.g., if we optimize our system on the Japanese-to-English small data track for the BLEU score, we are able to increase this score on the development set from 45.3 % to 46.7 %. Further, the mWER decreases from 41.9 % to 40.9 %, but the other error criteria deteriorate (mPER from 33.8 % to 34.4 %, NIST from 9.49 to 9.06, and GTM from 76.4 % to 74.7 %).

To investigate the effect of n-best list rescoring, we compared the performance of our system based on single-best translations with the performance on n-best lists which were successively enhanced by the models described in section 2.3. The results for the two small data tracks on the corresponding development sets are shown in Table 5 and 6. Again, all the systems have been optimized with respect to the NIST score, which serves as primary score. We see that the performance of the single-best system and that of the initial n-best list can differ due to different parameter settings for the beam search algorithm. Furthermore, we achieve a gain in performance with every model we add to the n-best list, not only in the NIST score but also in the other error criteria.

Table 5: Translation performance on the Chinese-to-English CSTAR 2003 test set (506 sentences).

System	Error Criteria			
	mWER [%]	mPER [%]	BLEU [%]	NIST
single-best	55.2	45.6	34.8	7.76
n-best list	53.4	45.3	33.6	7.63
+ IBM-1 lexicon	50.9	42.1	36.4	8.06
+ deletion model	50.6	42.2	37.1	8.07
+ 9-gram LM	50.6	42.2	38.0	8.14

Table 6: Translation performance on the Japanese-to-English CSTAR 2003 test set (506 sentences).

System	Error Criteria			
	mWER [%]	mPER [%]	BLEU [%]	NIST
single-best	48.7	38.6	44.3	9.10
+ ITG constraints	45.1	36.0	47.3	9.32
n-best list	49.5	37.3	45.0	9.32
+ IBM-1 lexicon	44.6	35.7	48.9	9.71
+ deletion model	43.2	34.7	50.1	9.80
+ 5-gram LM	42.6	34.2	51.5	9.92

- The IBM-1 lexicon is probably helpful because it captures lexical co-occurrences due to its bag-of-words characteristic [10].
- The deletion model protects the system from producing too short sentences.
- The additional language model enriches the system with knowledge about larger phrases.

Finally, to demonstrate the benefit of the ITG reordering constraints for the Japanese-to-English task we distinguish the performance of the unconstrained single-best system from the ITG constrained one in Table 6. Obviously, the unconstrained reorderings are significantly inferior to the ITG reorderings. This is not true for the Chinese-to-English task. Here, no performance gain has been achieved by constraining the reorderings.

4. Conclusions

We have presented the RWTH statistical machine translation system which is based on log-linear model combination. The main advantage comes from the large number of knowledge sources which can easily be integrated into our system in terms of feature functions. Using the *alignment templates* as main model, we incorporate shallow phrase structures: a phrase level alignment between phrases and a word level alignment between single words within the phrases. In this way, our system is able to learn word contexts and local reorderings.

Due to the fact that the alignment templates do not provide constrained reorderings and that unconstrained reordering may adversely affect the translation quality, we extended our system to cover reordering constraints. For the Japanese-to-English task the ITG constraints showed the best performance.

We included the IBM-1 lexicon, a deletion model and higher order language models as additional feature functions and applied n-best list rescoring because a straightforward integration into the dynamic programming search algorithm is not always possible.

The optimization of the model scaling factors was performed with respect to the translation quality measured by the NIST score, as this score was found out to correspond best to subjective evaluation criteria.

Participating in the International Workshop on Spoken Language Translation (IWSLT 2004), we evaluated our system on the *Basic Travel Expression Corpus* (BTEC) Chinese-to-English and Japanese-to-English tasks. On both tasks, our system produces translations of good quality. This is true especially for the unrestricted data track, for which we extended the training resources by additional corpora and obtained a rather high overall performance.

5. Acknowledgments

This work has been partially funded by the European Commission under the projects PF-Star, IST-2001-37599, and LC-Star, IST-2001-32216.

6. References

[1] P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin, "A statistical approach to machine translation," *Computational Linguistics*, vol. 16, no. 2, pp. 79–85, June 1990.

[2] F. J. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, PA, July 2002, pp. 295–302.

[3] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proc. of the 41th Annual Meeting of the Association for Computational Linguistics (ACL)*, Sapporo, Japan, July 2003, pp. 160–167.

[4] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *Proc. ARPA Workshop on Human Language Technology*, 2002.

[5] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, PA, July 2002, pp. 311–318.

[6] C. Tillmann, S. Vogel, H. Ney, A. Zubiaga, and H. Sawaf, "Accelerated DP based search for statistical translation." in *European Conf. on Speech Communication and Technology*, Rhodes, Greece, September 1997, pp. 2667–2670.

[7] F. J. Och, C. Tillmann, and H. Ney, "Improved alignment models for statistical machine translation," in *Proc. of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, University of Maryland, College Park, MD, June 1999, pp. 20–28.

[8] F. J. Och, "An efficient method for determining bilingual word classes," in *EACL '99: Ninth Conf. of the Europ. Chapter of the Association for Computational Linguistics*, Bergen, Norway, June 1999, pp. 71–76.

[9] N. Ueffing, F. J. Och, and H. Ney, "Generation of word graphs in statistical machine translation," in *Proc. Conf. on Empirical Methods for Natural Language Processing*, Philadelphia, PA, July 2002, pp. 156–163.

[10] F. J. Och, D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, and D. Radev, "A smorgasbord of features for statistical machine translation," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, Boston, MA, May 2004, pp. 161–168.

[11] A. Stolcke, "SRILM - an extensible language modeling toolkit," in *Proc. Intl. Conf. Spoken Language Processing*, Denver, CO, September 2002, pp. 901–904.

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C++*. Cambridge, UK: Cambridge University Press, 2002.

[13] R. Zens, H. Ney, T. Watanabe, and E. Sumita, "Reordering constraints for phrase-based statistical machine translation," in *COLING '04: The 20th Int. Conf. on Computational Linguistics*, Geneva, Switzerland, August 2004, pp. 205–211.

[14] D. Wu, "Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora," in *Proc. of the 14th International Joint Conf. on Artificial Intelligence (IJCAI)*, Montreal, August 1995, pp. 1328–1334.

[15] D. Wu, "Stochastic inversion transduction grammars and bilingual parsing of parallel corpora," *Computational Linguistics*, vol. 23, no. 3, pp. 377–403, September 1997.

[16] T. Takezawa, E. Sumita, F. Sugaya, H. Yamamoto, and S. Yamamoto, "Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world," in *Proc. of the Third Int. Conf. on Language Resources and Evaluation (LREC)*, Las Palmas, Spain, May 2002, pp. 147–152.

[17] T. Morimoto, N. Uratani, T. Takezawa, O. Furuse, Y. Sobashima, H. Iida, A. Nakamura, Y. Sagisaka, N. Higuchi, and Y. Yamazaki, "A speech and language database for speech translation research," in *Proc. of the 3rd Int. Conf. on Spoken Language Processing (ICSLP'94)*, Yokohama, Japan, September 1994, pp. 1791–1794.

[18] K. A. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," IBM Research Division, Thomas J. Watson Research Center, Tech. Rep. RC22176 (W0109-022), September 2001.

[19] J. P. Turian, L. Shen, and I. D. Melamed, "Evaluation of machine translation and its evaluation," Computer Science Department, New York University, Tech. Rep. Proteus technical report 03-005, 2003.