Improved Statistical Alignment Models

Franz Josef Och and **Hermann Ney**
Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen - University of Technology
D-52056 Aachen, Germany
{och,ney}@informatik.rwth-aachen.de

Abstract

In this paper, we present and compare various single-word based alignment models for statistical machine translation. We discuss the five IBM alignment models, the Hidden-Markov alignment model, smoothing techniques and various modifications. We present different methods to combine alignments. As evaluation criterion we use the quality of the resulting Viterbi alignment compared to a manually produced reference alignment. We show that models with a first-order dependence and a fertility model lead to significantly better results than the simple models IBM-1 or IBM-2, which are not able to go beyond zero-order dependencies.

1 Introduction

In statistical machine translation we set up a statistical translation model $Pr(f_1^J|e_1^I)$ which describes the relationship between a source language (SL) string f_1^J and a target language (TL) string e_1^I. In (statistical) alignment models $Pr(f_1^J, a_1^J|e_1^I)$, a 'hidden' alignment a_1^J is introduced which describes a mapping from source word f_j to a target word e_{a_j}.

We discuss here the IBM translation models IBM-1 to IBM-5 (Brown et al., 1993b) and the Hidden-Markov alignment model (Vogel et al., 1996; Och and Ney, 2000). The different alignment models we present provide different decompositions of $Pr(f_1^J, a_1^J|e_1^I)$. An alignment $\hat{a}_1^J$ for which holds

$$\hat{a}_1^J = \arg\max_{a_1^J} \; Pr(f_1^J, a_1^J|e_1^I)$$

for a specific model is called Viterbi alignment of this model.

So far, no well established evaluation criterion exists in the literature for these alignment models. For various reasons (non-unique reference translation, over-fitting and statistically deficient models) it seems hard to use training/test perplexity as in language modeling. Using translation quality is problematic, as translation quality is not well defined and as there are additional influences such as language model or decoder properties. We propose in this paper to measure the quality of an alignment model using the quality of the Viterbi alignment compared to a manually produced alignment. This allows an automatic evaluation, once a reference alignment has been produced. In addition, it results in a very precise and reliable evaluation criterion that is well suited to assess various design decisions in modeling and training of statistical alignment models.

2 Models

In this paper we use the models IBM-1 to IBM-5 from (Brown et al., 1993b) and the Hidden-Markov alignment model (HMM) from (Vogel et al., 1996; Och and Ney, 2000). All these models provide different decompositions of the probability $Pr(f_1^J, a_1^J|e_1^I)$. The alignment a_1^J may contain alignments $a_j = 0$ with the 'empty' word e_0 to account for French words that are not aligned to any English word. All models include lexicon parameters $p(f|e)$ and additional parameters describing the probability of an alignment.

We now sketch the structure of the six models:

- In IBM-1 all alignments have the same probability.
- IBM-2 uses a zero-order alignment model $p(a_j|j, I, J)$ where different alignment positions are independent from each other.
- The HMM uses a first-order model $p(a_j|a_{j-1})$ where the alignment position a_j depends on the previous alignment position a_{j-1}.
- In IBM-3 we have an (inverted) zero-order alignment model $p(j|a_j, I, J)$ with an additional fertility model $p(\phi|e)$ which describes the number of words ϕ aligned to an English word e.
- In IBM-4 we have an (inverted) first-order alignment model $p(j|j')$ and a fertility model $p(\phi|e)$.
- The models IBM-3 and IBM-4 are deficient as they waste probability mass on non-strings. IBM-5 is a reformulation of IBM-4 with a suitably refined alignment model in order to avoid deficiency.

So the main differences of these models lie in the alignment model (which may be zero-order or first-order), in the existence of an explicit fertility model and whether the model is deficient or not.

For HMM, IBM-4 and IBM-5 it is straightforward to extend the alignment parameters to include a dependence on the word classes of the words around the alignment position. In the HMM alignment model we allow for a dependence from the class $E = C(e_{a_{j-1}})$. Correspondingly, we can include similar dependencies on French and English word classes in IBM-4 and IBM-5 (Brown et al., 1993b). The classification of the words into a given number of classes (here: 50) is performed automatically by another statistical learning procedure (Kneser and Ney, 1991).

3 Training[1]

The training of all alignment models is done by the EM-algorithm using a parallel training corpus $(\mathbf{f}^{(s)}, \mathbf{e}^{(s)})$, $s = 1, \ldots, S$. In the E-step the counts for one sentence pair $(\mathbf{f}, \mathbf{e})$ are calculated. For the lexicon parameters the counts are:

$$c(f|e; \mathbf{f}, \mathbf{e}) \quad = \quad \sum_{\mathbf{a}} Pr(\mathbf{a}|\mathbf{f}, \mathbf{e}) \sum_{i,j} \delta(f, f_j)\delta(e, e_{a_j})$$

In the M-step the lexicon parameters are:

$$p(f|e) \quad \propto \quad \sum_{s} c(f|e; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})$$

Correspondingly, the alignment and fertility probabilities can be estimated.

The models IBM-1, IBM-2 and HMM have a particularly simple mathematical form so that the EM algorithm can be performed exactly, i.e. in the E-step it is possible to efficiently consider all alignments. For the HMM we do this using the Baum-Welch algorithm (Baum, 1972).

Since there is no efficient way in the fertility models IBM-3 to 5 to avoid the explicit summation over all alignments in the EM-algorithm, the counts are collected only over a subset of promising alignments. For IBM-3, IBM-4 and IBM-5 we perform the count collection only over a small number of good alignments. In order to keep the training fast we can take into account only a small fraction of all alignments. We will compare three different possibilities of using subsets of different size:

- The simplest possibility is to perform Viterbi training using only the best alignment that can be found. As the calculation of the Viterbi alignment itself is very time-consuming it is computed only approximately using the method described in (Brown et al., 1993b).
- In (Al-Onaizan et al., 1999) it was suggested to use also the neighboring alignments (i.e. alignments differing by one

[1] Our implementation of the IBM translation models is based on GIZA which is part of the publicly available toolkit for statistical machine translation EGYPT (Al-Onaizan et al., 1999).

move/swap) from the best alignment reachable.

- In (Brown et al., 1993b) an even larger set of alignments was used including also the 'pegged' alignments.

The different models are trained in succession on the same data, where the final parameter values of a simpler model serve as starting point for a more complex model. In section 8 we will show that by using the HMM instead of IBM-2 while bootstrapping to IBM-4/IBM-5 the alignment quality can be significantly improved.

4 Smoothing

To overcome the problem of over-fitting on the training data and to cope better with rare words we apply smoothing on alignment and fertility probabilities. For the alignment probabilities of the HMM (and correspondingly for IBM-4 and IBM-5) we perform an interpolation with a constant distribution:

$$p'(a_j|a_{j-1},I) = \alpha \cdot \frac{1}{I} + (1-\alpha) \cdot p(a_j|a_{j-1},I)$$

For the fertility probabilities we assume that there is a dependence on the number of letters $g(e)$ of e and estimate also a distribution $p(\phi|g)$ using the EM-algorithm. Figure 1 shows the relation between the number of letters g of a (German) word and the average fertility ($\bar{\phi}(g) = \sum_\phi \phi \cdot p(\phi|g)$). We can see that longer words have a higher fertility.

The fertility distribution used in training is then computed as follows:

$$p'(\phi|e) = \frac{n(e)}{\beta + n(e)} p(\phi|e) + \frac{\beta}{\beta + n(e)} p(\phi|g(e))$$

Here $n(e)$ denotes the frequency of e in the training corpus. This ensures that for frequent words, i.e. $n(e) \gg \beta$, the specific distribution $p(\phi|e)$ dominates and for rare words, i.e. $n(e) \ll \beta$, the general distribution $p(\phi|g(e))$ dominates.

The interpolation parameters α and β are optimized with respect to alignment quality on a validation corpus.

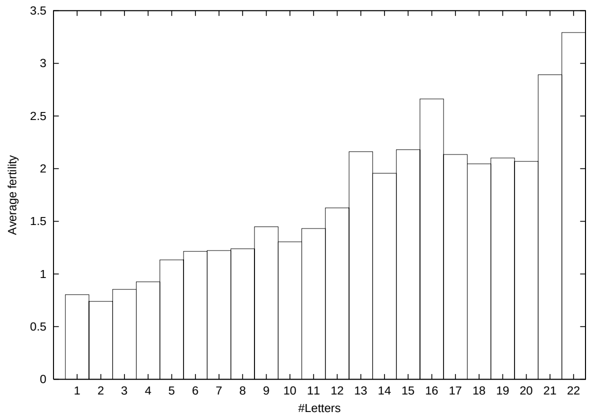

Figure 1: Average fertility as a function of the length (in letters) of a German word (on VERBMOBIL task, see later).

5 Is deficiency a problem?

When using the EM-algorithm on IBM-3 and IBM-4, we observed that during the EM-iterations more and more words are aligned to the empty word. This results in a bad alignment quality as too many words are aligned to the empty word. This does not occur when using the other models. We believe that the reason of this lies in the fact that IBM-3 and IBM-4 are deficient.

The use of the EM-algorithm guarantees that the likelihood an alignment model assigns to the training corpus is steadily increasing. This is true for deficient and for non-deficient models likewise. However, for deficient models the likelihood can be increased simply by reducing the amount of deficiency. In IBM-3 and IBM-4 as defined in (Brown et al., 1993b) the distortion model for real words is deficient, but the distortion model for the empty word is non-deficient, so the EM-algorithm can increase likelihood by simply aligning more and more words to the empty word. [2]

Therefore, we changed IBM-3 and IBM-4 slightly to obtain also a deficient distortion model for the empty word. The distortion probability is set to $p(j) = 1/J$ for every French word aligned to the empty word.

[2]This effect did not occur in (Brown et al., 1993b) as IBM-3 and IBM-4 were not trained directly.

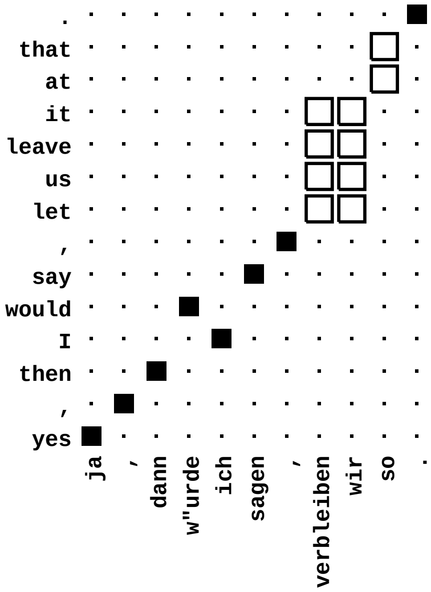

Figure 2: Example of a manual alignment with *S(ure)* (filled dots) and *P(ossible)* connections.

6 Evaluation methodology

In the following, we present an annotation scheme for single-word based alignments and a corresponding evaluation criterion. For a different approach to assess alignment quality see (Ahrenberg et al., 2000).

It is well known that manually performing a word alignment is a complicated and ambiguous task (Melamed, 1998). Therefore, we developed an annotation scheme that makes it possible to annotate explicitly the ambiguous alignments. We allowed human experts who performed the annotation to specify two different kinds of alignments: an S (sure) alignment which is used for alignments that are unambiguous and a P (possible) alignment which is used for alignments that might or might not exist. The P relation is used especially to align words within idiomatic expressions, free translations, and missing function words ($S \subseteq P$).

The thus obtained reference alignment may contain many-to-one and one-to-many relationships. Figure 2 shows an example of a manually aligned sentence with S and P relations.

The quality of an alignment $A = \{(j, a_j) | a_j > 0\}$ is then computed by appropriately redefined precision and recall measures:

$$recall = \frac{|A \cap S|}{|S|}, \quad precision = \frac{|A \cap P|}{|A|}$$

and the following error rate:

$$AER(S, P; A) = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|} \quad .$$

Thereby, a recall error can only occur if a S(ure) alignment is not found and a precision error can only occur if a found alignment is not even P(ossible).

The set of sentence pairs for which the manual alignment is produced is randomly selected from the training corpus. As the alignment is learned unsupervised, these sentence pairs may also be used in training.

Normally, the annotation is performed by two annotators, producing sets S_1, P_1, S_2, P_2. To increase the quality of the reference alignment the annotators are presented the mutual errors and are asked to improve their alignment if possible. From these alignments we finally generate a reference alignment which contains only those S(ure) connections where both annotators agree and it contains all the P(ossible) connections from both annotators. This can be done by forming the intersection of the sure alignments ($S = S_1 \cap S_2$) and the union of the possible alignments ($P = P_1 \cup P_2$). Thereby, we enforce that, if we compare the sure alignments of every single annotator with the combined reference alignment we obtain an AER of zero percent.

7 Generalized alignments

The baseline alignment model does not permit a source word to be aligned with two or more target words. Therefore, lexical correspondences like *'Zahnarzttermin'* for *dentist's appointment* cause problems because a single source word must be mapped on two or more target words.

To solve this problem, we perform a training in both translation directions (source to target, target to source). Thus we obtain two alignment vectors a_1^J and b_1^I for each sentence pair. In the following, $A_1 = \{(a_j, j) | a_j > 0\}$ and $A_2 = \{(i, b_i) | b_i > 0\}$ denote the sets of links in the two Viterbi alignments. We increase the quality of the alignments with respect to precision, recall or AER by combining A_1 and A_2 into one alignment matrix A using the following combination methods:

- Intersection: $A = A_1 \cap A_2$
- Union: $A = A_1 \cup A_2$
- Refined: In a first step the intersection $A = A_1 \cap A_2$ is determined. The elements within A are justified by both Viterbi alignments and are therefore very reliable. We now extend the alignment A iteratively by adding links (i, j) occurring only in A_1 or in A_2 if neither f_j nor e_i have an alignment in A or if the following conditions hold:
  - the link (i, j) has a horizontal neighbor $(i-1, j)$, $(i+1, j)$ or a vertical neighbor $(i, j-1)$, $(i, j+1)$ that is already in A, and
  - the set $A \cup \{(i, j)\}$ does not contain alignments with both horizontal and vertical neighbors.

Obviously, the intersection leads to an alignment that has only one-to-one alignments with higher precision and a lower recall. The union leads to a higher recall and a lower precision of the combined alignment. We typically observe that the refined combination is able to produce an alignment with better recall and precision.

8 Experiments

We present results on the Verbmobil and the Hansards task (Table 1). For both tasks we manually aligned a randomly chosen subset of the training corpus (Table 2). From this corpus the first 100 sentences were used as validation corpus to optimize the smoothing parameters and the remaining sentences were used as test corpus.

In the following graphs, we display the AER for every iteration of the EM-algorithm.

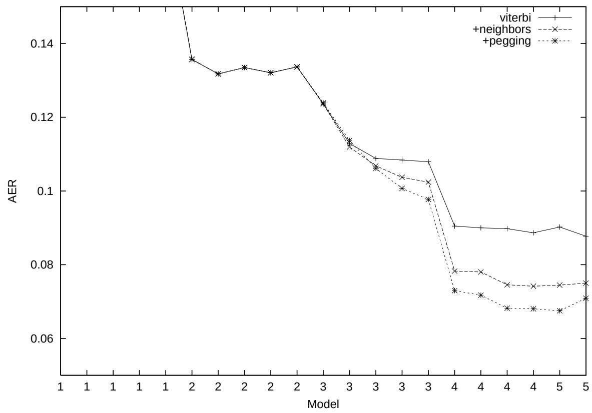

Figure 3: Effect of using more alignments in training of IBM-3/4/5 (Verbmobil task.)

Unless noted otherwise, we used for training of IBM-3/4 our modified version described in section 5.

The number of alignments in training

Figure 3 compares the results obtained by using different numbers of alignments in the training of the sophisticated alignment models on the Verbmobil task. In order to reduce training time we restricted the number of pegged alignments by using only those alignments where $Pr(\mathbf{f}, \mathbf{a}|\mathbf{e})$ is not too much smaller than the probability of the Viterbi alignment. If we use only the Viterbi alignment, the results are significantly worse than additionally using the neighborhood of the Viterbi alignment. By doing 'pegging', we obtain an additional small improvement.

Table 3 shows the computing time for performing one iteration of the EM-algorithm. Using a larger set of alignments significantly increases the training time for the models IBM-4 and IBM-5. As 'pegging' yields only a moderate improvement, all following results are obtained using the neighborhood of the Viterbi alignment.

IBM-2 or HMM

Figure 4 compares the results of using IBM-2 or HMM in bootstrapping the fertility on the Verbmobil task. The HMM alignment model yields significantly better results than IBM-2. The best results are obtained if IBM-3 is omitted in the training and the HMM

Table 1: Training corpora sizes.

	Languages		Words		Vocabulary	
Corpus	SL/TL	Sentences	SL	TL	SL	TL
Verbmobil	English/German	34k	343 076	329 625	3 505	5 936
Hansards(50k)	French/English	50k	825 713	751 849	19 900	25 000
Hansards(200k)	French/English	200k	3 273 640	2 980 160	44 475	34 865
Hansards(500k)	French/English	500k	8 173 413	7 440 097	64 293	50 323
Hansards(1500k)	French/English	1500k	24 338 195	22 163 092	100 270	78 333

Table 2: Manually annotated test corpora.

	Words		
Corpus	SL	TL	Sentences
Verbmobil	3233	3109	354
Hansards	8749	7946	500

Table 3: Computing time on the Verbmobil task (on 600 Mhz Pentium II machine).

	seconds per iteration		
alignment set	IBM-3	IBM-4	IBM-5
Viterbi	17	220	870
+neighbors	25	400	1600
+pegging	190	3500	33000

model parameters are used to directly estimate the IBM-4 model parameters. In the later iterations, IBM-4 is able to reduce the advantage of using HMM. But in the end we still obtain better results when using in bootstrapping HMM (AER: 5.8 %) instead of IBM-2 (AER: 7.4 %).

In the Hansards(50k) task (see Figure 5), the error rates are higher especially because of the high vocabulary size. The use of HMM in training yields an even stronger reduction in AER. Interestingly, already the AER of the final iteration of HMM (18.0%) yields better results than the best EM-iteration when using IBM-2 in bootstrapping (20.0%). We conclude that it is important to start the training of the sophisticated alignment models with good initial parameters.

The use of IBM-3 after HMM makes results worse, but finally IBM-4 produces best results (15.0%). Astonishingly, IBM-5 produces worse results than IBM-4. This is maybe because IBM-5 has a lot more training parameters and the distortion model uses only a dependence on the French word class. For the following experiments we use training scheme $1 \rightarrow HMM \rightarrow 4$.

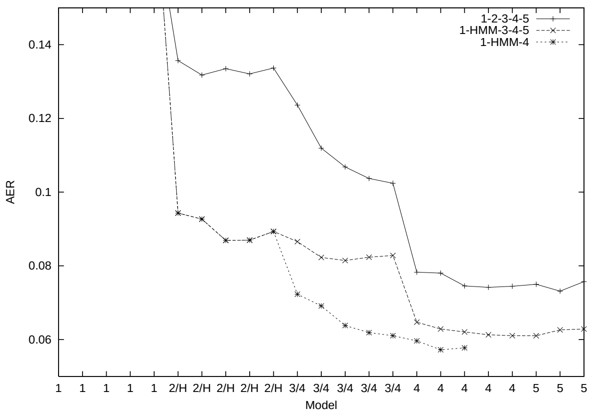

Figure 4: Comparison of using IBM-2 or HMM in bootstrapping IBM-3/4/5 (Verbmobil task).

Effect of Smoothing

Figure 6 shows the effect of using our modified IBM-4 (section 5) and smoothing the alignment/fertility probabilities. We see that using the standard version of IBM-4 yields a higher AER which is mainly due to a worse recall. Without smoothing, we also observe early over-fitting: AER increases after the second iteration of HMM. Analyzing the alignments shows that the smoothing of fertility probabilities also significantly reduces the problem that rare words often form 'garbage collectors' in that they tend to align to a lot of words (see (Brown et al., 1993a)).

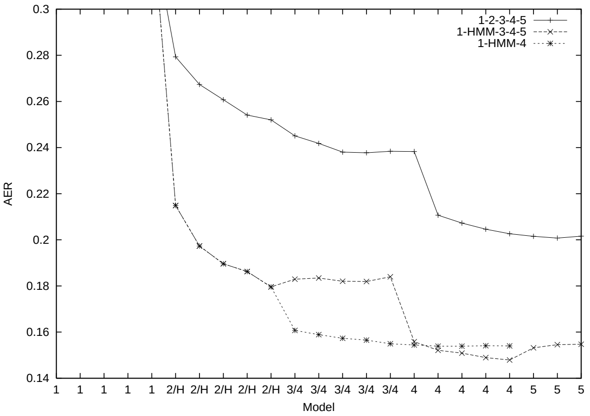

Figure 5: Comparison of using IBM-2 or HMM in bootstrapping IBM-3/4/5 (HANSARDS(50k) task).

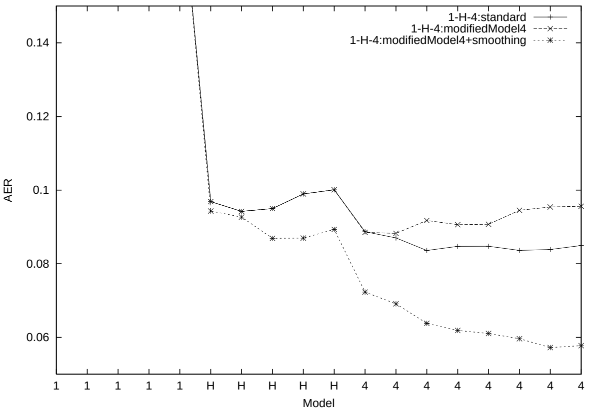

Figure 6: Effect of smoothing (VERBMOBIL task).

Using a larger training corpus

Table 4 shows the effect of using different amounts of training data. As expected, more training data helps to improve alignment quality for all models. However, for IBM-1 the relative improvement is very small compared to the relative improvement using HMM and IBM-4.

Generalized Alignments

Table 5 shows precision, recall and AER of the last iteration of IBM-4 for both translation directions. Especially for the language pair German-English (VERBMOBIL task) we observe that by using German as source language the AER is much higher than by using English as source language. This is because the baseline alignment representation as a vector a_1^J does in that case forbid that the often occurring German word compounds align to more than only one English word.

Table 4: Effect of using different amount of training data (HANSARDS task, training scheme $1 \rightarrow \text{HMM} \rightarrow 4$).

	AER [%]		
Corpus	IBM-1	HMM	IBM-4
HANSARDS(50k)	34.3	18.0	15.6
HANSARDS(200k)	31.3	14.3	12.5
HANSARDS(500k)	30.3	12.8	10.7
HANSARDS(1500k)	29.4	11.0	9.4

The effect of merging alignments by forming the intersection, the union or the refined combination of the Viterbi alignments (see section 7) of both translation directions is shown in Table 6. By using the refined combination we can increase precision and recall on all tasks. The lowest AER on the VERBMOBIL task is obtained using the refined combination method. The lowest AER on the HANSARDS task is obtained using intersection.

By forming a union or intersection of the alignments we can obtain recall or precision values (but not both) over 96 %.

9 Conclusion

We have discussed various extensions to statistical alignment models. An evaluation criterion, i.e. the alignment error rate, was suggested and results on different tasks were presented. We have shown that sophisticated alignment models with a first-order dependence and a fertility model lead to significantly better results than the simple models IBM-1 or IBM-2. We have described various heuristics that improve precision, recall or both by combining Viterbi alignments of both translation directions.

Further improvements in producing better alignments are expected from making use of cognates, and from statistical alignment models that are based on word groups rather than single words.

Table 5: Alignment quality in last iteration of IBM-4 of both translation directions.

	SL → TL			TL → SL		
Corpus	prec	rec	AER	prec	rec	AER
VERBMOBIL	93.2	95.5	5.8	90.0	87.9	10.9
HANSARDS(50k)	80.5	91.2	15.6	80.0	90.8	16.0
HANSARDS(200k)	84.3	93.1	12.5	84.2	93.4	12.4
HANSARDS(500k)	86.5	94.2	10.7	86.9	94.4	10.3
HANSARDS(1500k)	88.1	94.9	9.4	88.5	95.0	9.0

Table 6: Effect of combination of IBM-4 Viterbi alignments from both translation directions.

	Intersection			Union			Refined		
Corpus	prec	rec	AER	prec	rec	AER	prec	rec	AER
VERBMOBIL	97.9	85.4	8.0	87.3	98.0	8.6	93.3	96.4	5.4
HANSARDS(50k)	95.7	85.6	9.0	72.6	96.6	20.2	85.9	92.3	11.7
HANSARDS(200k)	96.7	89.0	6.8	77.5	97.5	16.3	88.3	94.5	9.4
HANSARDS(500k)	96.9	90.9	5.8	80.7	97.8	13.8	90.1	95.1	8.0
HANSARDS(1500k)	96.8	91.9	5.3	83.0	98.0	12.1	90.4	95.6	7.6

Acknowledgment

This work has been partially supported as part of the Verbmobil project (contract number 01 IV 701 T4) by the German Federal Ministry of Education, Science, Research and Technology and as part of the EuTrans project by the by the European Community (ESPRIT project number 30268).

References

L. Ahrenberg, M. Merkel, H. A. Sagvall, and J. Tiedemann. 2000. Evaluation of word alignment systems. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC)*, pages 1255–1261, Athens, Greece, May/June.

Y. Al-Onaizan, J. Curin, M. Jahr, K. Knight, J. Lafferty, I. D. Melamed, F. J. Och, D. Purdy, N. A. Smith, and D. Yarowsky. 1999. Statistical machine translation, final report, JHU workshop. `http://www.clsp.jhu.edu/ws99/projects/mt/final_report/mt-final-report.ps`.

L.E. Baum. 1972. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. *Inequalities*, 3:1–8.

P. Brown, S. A. Della Pietra, V. J. Della Pietra, M. J. Goldsmith, J. Hajic, R. L. Mercer, and S. Mohanty. 1993a. But dictionaries are data too. In *Human Language Technology*, pages 202–205.

P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. 1993b. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

R. Kneser and H. Ney. 1991. Forming Word Classes by Statistical Clustering for Statistical Language Modelling. In *1. Quantitative Linguistics Conference*, September.

I. D. Melamed. 1998. Manual annotation of translational equivalence: The Blinker project. Technical Report 98-07, IRCS.

F. J. Och and H. Ney. 2000. A comparison of alignment models for statistical machine translation. In *Proc. of the 18th Int. Conf. on Computational Linguistics*, Saarbrücken, Germany, August.

S. Vogel, H. Ney, and C. Tillmann. 1996. HMM-based word alignment in statistical translation. In *COLING '96: The 16th Int. Conf. on Computational Linguistics*, pages 836–841, Copenhagen, August.