Improvements in Phrase-Based Statistical Machine Translation

Richard Zens and Hermann Ney
Chair of Computer Science VI
RWTH Aachen University
{zens,ney}@cs.rwth-aachen.de

Abstract

In statistical machine translation, the currently best performing systems are based in some way on phrases or word groups. We describe the baseline phrase-based translation system and various refinements. We describe a highly efficient monotone search algorithm with a complexity linear in the input sentence length. We present translation results for three tasks: Verbmobil, Xerox and the Canadian Hansards. For the Xerox task, it takes less than 7 seconds to translate the whole test set consisting of more than 10K words. The translation results for the Xerox and Canadian Hansards task are very promising. The system even outperforms the alignment template system.

1 Introduction

In statistical machine translation, we are given a source language ('French') sentence $f_1^J = f_1 \ldots f_j \ldots f_J$, which is to be translated into a target language ('English') sentence $e_1^I = e_1 \ldots e_i \ldots e_I$. Among all possible target language sentences, we will choose the sentence with the highest probability:

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

$$= \operatorname*{argmax}_{e_1^I} \left\{ Pr(e_1^I) \cdot Pr(f_1^J | e_1^I) \right\} \quad (2)$$

The decomposition into two knowledge sources in Equation 2 is known as the source-channel approach to statistical machine translation (Brown et al., 1990). It allows an independent modeling of target language model $Pr(e_1^I)$ and translation model $Pr(f_1^J|e_1^I)$[1]. 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. 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.

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 (Och and Ney, 2002), we obtain:

$$Pr(e_1^I|f_1^J) = \exp\left(\sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J) \right) \cdot Z(f_1^J)$$

Here, $Z(f_1^J)$ denotes the appropriate normalization constant. As a decision rule, we obtain:

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

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 model scaling factors λ_1^M are trained according to the maximum entropy principle, e.g. using the GIS algorithm. Alternatively, one can train them with respect to the final translation quality measured by some error criterion (Och, 2003).

The remaining part of this work is structured as follows: in the next section, we will describe the baseline phrase-based translation model and the extraction of bilingual phrases. Then, we will describe refinements of the baseline model. In Section 4, we will describe a monotone search algorithm. Its complexity is linear in the sentence length. The next section contains the statistics of the corpora that were used. Then, we will investigate the degree of monotonicity and present the translation results for three tasks: Verbmobil, Xerox and Canadian Hansards.

[1]The notational convention will be as follows: we use the symbol $Pr(\cdot)$ to denote general probability distributions with (nearly) no specific assumptions. In contrast, for model-based probability distributions, we use the generic symbol $p(\cdot)$.

2 Phrase-Based Translation

2.1 Motivation

One major disadvantage of single-word based approaches is that contextual information is not taken into account. The lexicon probabilities are based only on single words. For many words, the translation depends heavily on the surrounding words. In the single-word based translation approach, this disambiguation is addressed by the language model only, which is often not capable of doing this.

One way to incorporate the context into the translation model is to learn translations for whole phrases instead of single words. Here, a phrase is simply a sequence of words. So, the basic idea of phrase-based translation is to segment the given source sentence into phrases, then translate each phrase and finally compose the target sentence from these phrase translations.

2.2 Phrase Extraction

The system somehow has to learn which phrases are translations of each other. Therefore, we use the following approach: first, we train statistical alignment models using GIZA++ and compute the Viterbi word alignment of the training corpus. This is done for both translation directions. We take the union of both alignments to obtain a symmetrized word alignment matrix. This alignment matrix is the starting point for the phrase extraction. The following criterion defines the set of bilingual phrases $\mathcal{BP}$ of the sentence pair $(f_1^J; e_1^I)$ and the alignment matrix $A \subseteq J \times I$ that is used in the translation system.

$$\begin{aligned} \mathcal{BP}(f_1^J, e_1^I, A) \;=\; \Big\{ \Big(f_{j_1}^{j_2}, e_{i_1}^{i_2}\Big) :& \\ \forall (j,i) \in A : j_1 \le j \le j_2 \leftrightarrow i_1 \le i \le i_2 & \\ \wedge \exists (j,i) \in A : j_1 \le j \le j_2 \wedge i_1 \le i \le i_2 \Big\} & \end{aligned}$$

This criterion is identical to the alignment template criterion described in (Och et al., 1999). It means that two phrases are considered to be translations of each other, if the words are aligned only within the phrase pair and not to words outside. The phrases have to be contiguous.

2.3 Translation Model

To use phrases in the translation model, we introduce the hidden variable S. This is a segmentation of the sentence pair $(f_1^J; e_1^I)$ into K phrases $(\tilde{f}_1^K; \tilde{e}_1^K)$. We use a one-to-one phrase alignment, i.e. one source phrase is translated by exactly one target phrase. Thus, we obtain:

$$\begin{aligned} Pr(f_1^J|e_1^I) &= \sum_S Pr(f_1^J, S|e_1^I) && (3) \\ &= \sum_S Pr(S|e_1^I) \cdot Pr(f_1^J|S, e_1^I) && (4) \\ &\approx \max_S \Big\{ Pr(S|e_1^I) \cdot Pr(\tilde{f}_1^K|\tilde{e}_1^K) \Big\} && (5) \end{aligned}$$

In the preceding step, we used the maximum approximation for the sum over all segmentations. Next, we allow only translations that are monotone at the phrase level. So, the phrase $\tilde{f}_1$ is produced by $\tilde{e}_1$, the phrase $\tilde{f}_2$ is produced by $\tilde{e}_2$, and so on. Within the phrases, the reordering is learned during training. Therefore, there is no constraint on the reordering within the phrases.

$$\begin{aligned} Pr(\tilde{f}_1^K|\tilde{e}_1^K) &= \prod_{k=1}^{K} Pr(\tilde{f}_k|\tilde{f}_1^{k-1}, \tilde{e}_1^K) && (6) \\ &= \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) && (7) \end{aligned}$$

Here, we have assumed a zero-order model at the phrase level. Finally, we have to estimate the phrase translation probabilities $p(\tilde{f}|\tilde{e})$. This is done via relative frequencies:

$$p(\tilde{f}|\tilde{e}) = \frac{N(\tilde{f}, \tilde{e})}{\sum_{\tilde{f}'} N(\tilde{f}', \tilde{e})} \qquad (8)$$

Here, $N(\tilde{f}, \tilde{e})$ denotes the count of the event that $\tilde{f}$ has been seen as a translation of $\tilde{e}$. If one occurrence of $\tilde{e}$ has $N > 1$ possible translations, each of them contributes to $N(\tilde{f}, \tilde{e})$ with $1/N$. These counts are calculated from the training corpus.

Using a bigram language model and assuming Bayes decision rule, Equation (2), we obtain the following search criterion:

$$\begin{aligned} \hat{e}_1^I &= \operatorname*{argmax}_{e_1^I} \Big\{ Pr(e_1^I) \cdot Pr(f_1^J|e_1^I) \Big\} && (9) \\ &= \operatorname*{argmax}_{e_1^I} \Big\{ \prod_{i=1}^{I} p(e_i|e_{i-1}) && (10) \\ &\qquad \cdot \max_S p(S|e_1^I) \cdot \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \Big\} \\ &\approx \operatorname*{argmax}_{e_1^I, S} \Big\{ \prod_{i=1}^{I} p(e_i|e_{i-1}) \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \Big\} && (11) \end{aligned}$$

For the preceding equation, we assumed the segmentation probability $p(S|e_1^I)$ to be constant. The result is a simple translation model. If we interpret this model as a feature function in the direct approach, we obtain:

$$h_{\text{phr}}(f_1^J, e_1^I, S, K) = \log \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k)$$

We use the maximum approximation for the hidden variable S. Therefore, the feature functions are dependent on S. Although the number of phrases K is implicitly given by the segmentation S, we used both S and K to make this dependency more obvious.

3 Refinements

In this section, we will describe refinements of the phrase-based translation model. First, we will describe two heuristics: word penalty and phrase penalty. Second, we will describe a single-word based lexicon model. This will be used to smooth the phrase translation probabilities.

3.1 Simple Heuristics

In addition to the baseline model, we use two simple heuristics, namely word penalty and phrase penalty:

$$h_{\text{wp}}(f_1^J, e_1^I, S, K) = I \quad (12)$$

$$h_{\text{pp}}(f_1^J, e_1^I, S, K) = K \quad (13)$$

The word penalty feature is simply the target sentence length. In combination with the scaling factor this results in a constant cost per produced target language word. With this feature, we are able to adjust the sentence length. If we set a negative scaling factor, longer sentences are more penalized than shorter ones, and the system will favor shorter translations. Alternatively, by using a positive scaling factors, the system will favor longer translations.

Similar to the word penalty, the phrase penalty feature results in a constant cost per produced phrase. The phrase penalty is used to adjust the average length of the phrases. A negative weight, meaning real costs per phrase, results in a preference for longer phrases. A positive weight, meaning a bonus per phrase, results in a preference for shorter phrases.

3.2 Word-based Lexicon

We are using relative frequencies to estimate the phrase translation probabilities. Most of the longer phrases are seen only once in the training corpus. Therefore, pure relative frequencies overestimate the probability of those phrases. To overcome this problem, we use a word-based lexicon model to smooth the phrase translation probabilities. For a source word f and a target phrase $\tilde{e} = e_{i_1}^{i_2}$, we use the following approximation:

$$p(f|e_{i_1}^{i_2}) \approx 1 - \prod_{i=i_1}^{i_2} (1 - p(f|e_i))$$

This models a disjunctive interaction, also called noisy-OR gate (Pearl, 1988). The idea is that there are multiple independent causes $e_{i_1}^{i_2}$ that can generate an event f. It can be easily integrated into the search algorithm. The corresponding feature function is:

$$h_{\text{lex}}(f_1^J, e_1^I, S, K) = \log \prod_{k=1}^{K} \prod_{j=j_{k-1}+1}^{j_k} p(f_j|\tilde{e}_k)$$

Here, j_k and i_k denote the final position of phrase number k in the source and the target sentence, respectively, and we define $j_0 := 0$ and $i_0 := 0$.

To estimate the single-word based translation probabilities $p(f|e)$, we use smoothed relative frequencies. The smoothing method we apply is absolute discounting with interpolation:

$$p(f|e) = \frac{\max\{N(f,e) - d, 0\}}{N(e)} + \alpha(e) \cdot \beta(f)$$

This method is well known from language modeling (Ney et al., 1997). Here, d is the nonnegative discounting parameter, $\alpha(e)$ is a normalization constant and β is the normalized backing-off distribution. To compute the counts, we use the same word alignment matrix as for the extraction of the bilingual phrases. The symbol $N(e)$ denotes the unigram count of a word e and $N(f,e)$ denotes the count of the event that the target language word e is aligned to the source language word f. If one occurrence of e has $N > 1$ aligned source words, each of them contributes with a count of $1/N$. The formula for $\alpha(e)$ is:

$$\alpha(e) = \frac{1}{N(e)} \left(\sum_{f:N(f,e)>d} d + \sum_{f:N(f,e)\leq d} N(f,e) \right)$$

$$= \frac{1}{N(e)} \sum_{f} \min\{d, N(f,e)\}$$

This formula is a generalization of the one typically used in publications on language modeling. This generalization is necessary, because the lexicon counts may be fractional whereas in language modeling typically integer counts are used. Additionally, we want to allow discounting values d greater than one. One effect of the discounting parameter d is that all lexicon entries with a count less than d are discarded and the freed probability mass is redistributed among the other entries.

As backing-off distribution $\beta(f)$, we consider two alternatives. The first one is a uniform distribution and the second one is the unigram distribution:

$$\beta_1(f) = \frac{1}{V_f} \quad (14)$$

$$\beta_2(f) = \frac{N(f)}{\sum_{f'} N(f')} \quad (15)$$

Here, V_f denotes the vocabulary size of the source language and $N(f)$ denotes the unigram count of a source word f.

4 Monotone Search

The monotone search can be efficiently computed with dynamic programming. The resulting complexity is linear in the sentence length. We present the formulae for a bigram language model. This is only for notational convenience. The generalization to a higher order language model is straightforward. For the maximization problem in (11), we define the quantity $Q(j,e)$ as the maximum probability of a phrase sequence that ends with the language word e and covers positions 1 to j of the source sentence. $Q(J+1,\$)$ is the probability of the optimum translation. The \$ symbol is the sentence boundary marker. We obtain the following dynamic programming recursion.

$$
\begin{aligned}
Q(0,\$) &= 1 \\
Q(j,e) &= \max_{\substack{e',\tilde{e},\\ j-M\leq j'<j}} \left\{ p(f_{j'+1}^{j}|\tilde{e}) \cdot p(\tilde{e}|e') \cdot Q(j',e') \right\} \\
Q(J+1,\$) &= \max_{e'} \left\{ Q(J,e') \cdot p(\$|e') \right\}
\end{aligned}
$$

Here, M denotes the maximum phrase length in the source language. During the search, we store back-pointers to the maximizing arguments. After performing the search, we can generate the optimum translation. The resulting algorithm has a worst-case complexity of $O(J \cdot M \cdot V_e \cdot E)$. Here, V_e denotes the vocabulary size of the target language and E denotes the maximum number of phrase translation candidates for a source language phrase. Using efficient data structures and taking into account that not all possible target language phrases can occur in translating a specific source language sentence, we can perform a very efficient search.

This monotone algorithm is especially useful for language pairs that have a similar word order, e.g. Spanish-English or French-English.

5 Corpus Statistics

In the following sections, we will present results on three tasks: Verbmobil, Xerox and Canadian Hansards. Therefore, we will show the corpus statistics for each of these tasks in this section. The training corpus (Train) of each task is used to train a word alignment and then extract the bilingual phrases and the word-based lexicon. The remaining free parameters, e.g. the model scaling factors, are optimized on the development corpus (Dev). The resulting system is then evaluated on the test corpus (Test).

Verbmobil Task. The first task we will present results on is the German–English Verbmobil task (Wahlster, 2000). The domain of this corpus is appointment scheduling, travel planning, and hotel reservation. It consists of transcriptions of spontaneous speech. Table 1 shows the corpus statistics of this task.

Xerox task. Additionally, we carried out experiments on the Spanish–English Xerox task. The corpus consists of technical manuals. This is a rather limited domain task. Table 2 shows the training, development and test corpus statistics.

Canadian Hansards task. Further experiments were carried out on the French–English Canadian Hansards task. This task contains the proceedings of the Canadian parliament. About 3 million parallel sentences of this bilingual data have been made available by the Linguistic Data Consortium (LDC). Here, we use a subset of the data containing only sentences with a maximum length of 30 words. This task covers a large variety of topics, so this is an open-domain corpus. This is also reflected by the large vocabulary size. Table 3 shows the training and test corpus statistics.

Table 1: Statistics of training and test corpus for the Verbmobil task (PP=perplexity).

		German	English
Train	Sentences	58 073	
	Words	519 523	549 921
	Vocabulary	7 939	4 672
Dev	Sentences	276	
	Words	3 159	3 438
	Trigram PP	-	28.1
Test	Sentences	251	
	Words	2 628	2 871
	Trigram PP	-	30.5

Table 2: Statistics of training and test corpus for the Xerox task (PP=perplexity).

		Spanish	English
Train	Sentences	55 761	
	Words	752 606	665 399
	Vocabulary	11 050	7 956
Dev	Sentences	1012	
	Words	15 957	14 278
	Trigram PP	–	28.1
Test	Sentences	1125	
	Words	10 106	8 370
	Trigram PP	–	48.3

6 Degree of Monotonicity

In this section, we will investigate the effect of the monotonicity constraint. Therefore, we compute how many of the training corpus sentence pairs can be produced with the monotone phrase-based search. We compare this to the number of sentence pairs that can be produced with a nonmonotone phrase-based search. To make these numbers more realistic, we use leaving-one-out. Thus phrases that are extracted from a specific sentence pair are not used to check its monotonicity. With leaving-one-out it is possible that even the nonmonotone search cannot generate a sentence pair. This happens if a sentence pair contains a word that occurs only once in the training corpus. All phrases that might produce this singleton are excluded because of the leaving-one-out principle. Note

Table 3: Statistics of training and test corpus for the Canadian Hansards task (PP=perplexity).

		French	English
Train	Sentences	1.5M	
	Words	24M	22M
	Vocabulary	100 269	78 332
Dev	Sentences	500	
	Words	9 043	8 195
	Trigram PP	–	57.7
Test	Sentences	5432	
	Words	97 646	88 773
	Trigram PP	–	56.7

that all these monotonicity consideration are done at the phrase level. Within the phrases arbitrary reorderings are allowed. The only restriction is that they occur in the training corpus.

Table 4 shows the percentage of the training corpus that can be generated with monotone and nonmonotone phrase-based search. The number of sentence pairs that can be produced with the nonmonotone search gives an estimate of the upper bound for the sentence error rate of the phrase-based system that is trained on the given data. The same considerations hold for the monotone search. The maximum source phrase length for the Verbmobil task and the Xerox task is 12, whereas for the Canadian Hansards task we use a maximum of 4, because of the large corpus size. This explains the rather low coverage on the Canadian Hansards task for both the nonmonotone and the monotone search.

For the Xerox task, the nonmonotone search can produce 75.1% of the sentence pairs whereas the monotone can produce 65.3%. The ratio of the two numbers measures how much the system deteriorates by using the monotone search and will be called the *degree of monotonicity*. For the Xerox task, the degree of monotonicity is 87.0%. This means the monotone search can produce 87.0% of the sentence pairs that can be produced with the nonmonotone search. We see that for the Spanish-English Xerox task and for the French-English Canadian Hansards task, the degree of monotonicity is rather high. For the German-English Verbmobil task it is significantly lower. This may be caused by the rather free word order in German and the long range reorderings that are necessary to translate the verb group.

It should be pointed out that in practice the monotone search will perform better than what the preceding estimates indicate. The reason is that we assumed a perfect nonmonotone search, which is difficult to achieve in practice. This is not only a hard search problem, but also a complicated modeling problem. We will see in the next section that the monotone search will perform very well on both the Xerox task and the Canadian Hansards task.

Table 4: Degree of monotonicity in the training corpora for all three tasks (numbers in percent).

	Verbmobil	Xerox	Hansards
nonmonotone	76.3	75.1	59.7
monotone	55.4	65.3	51.5
deg. of mon.	72.6	87.0	86.3

7 Translation Results

7.1 Evaluation Criteria

So far, in machine translation research a single generally accepted criterion for the evaluation of the experimental results does not exist. Therefore, we use a variety of different criteria.

- 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, bigrams, trigrams and fourgrams with respect to a reference translation with a penalty for too short sentences (Papineni et al., 2001). 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 (Doddington, 2002). NIST measures accuracy, i.e. large NIST scores are better.

For the Verbmobil task, we have multiple references available. Therefore on this task, we compute 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. For the other two tasks, only single references are used.

7.2 Translation Systems

In this section, we will describe the systems that were used. On the one hand, we have three different variants of the single-word based model IBM4. On the other hand, we have two phrase-based systems, namely the alignment templates and the one described in this work.

Single-Word Based Systems (SWB). First, there is a monotone search variant (Mon) that translates each word of the source sentence from left to right. The second variant allows reordering according to the so-called IBM constraints (Berger et al., 1996). Thus up to three words may be skipped and translated later. This system will be denoted by IBM. The third variant implements special German-English reordering constraints. These constraints are represented by a finite state automaton and optimized to handle the reorderings of the German verb group. The abbreviation for this variant is GE. It is only used for the German-English Verbmobil task. This is just an extremely brief description of these systems. For details, see (Tillmann and Ney, 2003).

Phrase-Based System (PB). For the phrase-based system, we use the following feature functions: a trigram language model, the phrase translation model and the word-based lexicon model. The latter two feature functions are used for both directions: $p(f|e)$ and $p(e|f)$. Additionally, we use the word and phrase penalty feature functions. The model scaling factors are optimized on the development corpus with respect to mWER similar to (Och, 2003). We use the Downhill Simplex algorithm from (Press et al., 2002). We do not perform the optimization on N-best lists but we retranslate the whole development corpus for each iteration of the optimization algorithm. This is feasible because this system is extremely fast. It takes only a few seconds to translate the whole development corpus for the Verbmobil task and the Xerox task; for details see Section 8. 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 (Och, 2003). It is also possible to optimize any other parameter, e.g. the discounting parameter for the lexicon smoothing.

Alignment Template System (AT). The alignment template system (Och et al., 1999) is similar to the system described in this work. One difference is that the alignment templates are not defined at the word level but at a word class level. In addition to the word-based trigram model, the alignment template system uses a class-based fivegram language model. The search algorithm of the alignment templates allows arbitrary reorderings of the templates. It penalizes reorderings with costs that are linear in the jump width. To make the results as comparable as possible, the alignment template system and the phrase-based system start from the same word alignment. The alignment template system uses discriminative training of the model scaling factors as described in (Och and Ney, 2002).

7.3 Verbmobil Task

We start with the Verbmobil results. We studied smoothing the lexicon probabilities as described in Section 3.2. The results are summarized in Table 5. We see that the uniform smoothing method improves translation quality. There is only a minor improvement, but it is consistent among all evaluation criteria. It is statistically significant at the 94% level. The unigram method hurts performance. There is a degradation of the mWER of 0.9%. In the following, all phrase-based systems use the uniform smoothing method.

Table 5: Effect of lexicon smoothing on the translation performance [%] for the German-English Verbmobil task.

system	mWER	mPER	BLEU	NIST
unsmoothed	37.3	21.1	46.6	7.96
uniform	37.0	20.7	47.0	7.99
unigram	38.2	22.3	45.5	7.79

The translation results of the different systems are shown in Table 6. Obviously, the monotone phrase-based system outperforms the monotone single-word based system. The result of the phrase-based system is comparable to the nonmonotone single-word based search with the IBM constraints. With respect to the mPER, the PB system clearly outperforms all single-word based systems.

If we compare the monotone phrase-based system with the nonmonotone alignment template system, we see that the mPERs are similar. Thus the lexical choice of words is of the same quality. Regarding the other evaluation criteria, which take the word order into account, the nonmonotone search of the alignment templates has a clear advantage. This was already indicated by the low degree of monotonicity on this task. The rather free word order in German and the long range dependencies of the verb group make reorderings necessary.

Table 6: Translation performance [%] for the German-English Verbmobil task (251 sentences).

system	variant	mWER	mPER	BLEU	NIST
SWB	Mon	42.8	29.3	38.0	7.07
	IBM	37.1	25.0	47.8	7.84
	GE	35.4	25.3	48.5	7.83
PB		37.0	20.7	47.0	7.99
AT		30.3	20.6	56.8	8.57

7.4 Xerox task

The translation results for the Xerox task are shown in Table 7. Here, we see that both phrase-based systems clearly outperform the single-word based systems. The PB system performs best on this task. Compared to the AT system, the BLEU score improves by 4.1% absolute. The improvement of the PB system with respect to the AT system is statistically significant at the 99% level.

Table 7: Translation performance [%] for the Spanish-English Xerox task (1125 sentences).

System	WER	PER	BLEU	NIST
SWB IBM	38.8	27.6	55.3	8.00
PB	26.5	18.1	67.9	9.07
AT	28.9	20.1	63.8	8.76

7.5 Canadian Hansards task

The translation results for the Canadian Hansards task are shown in Table 8. As on the Xerox task, the phrase-based systems perform better than the single-word based systems. The monotone phrase-based system yields even better results than the alignment template system. This improvement is consistent among all evaluation criteria and it is statistically significant at the 99% level.

Table 8: Translation performance [%] for the French-English Canadian Hansards task (5432 sentences).

System	Variant	WER	PER	BLEU	NIST
SWB	Mon	65.2	53.0	19.8	5.96
	IBM	64.5	51.3	20.7	6.21
PB		57.8	46.6	27.8	7.15
AT		61.1	49.1	26.0	6.71

8 Efficiency

In this section, we analyze the translation speed of the phrase-based translation system. All experiments were carried out on an AMD Athlon with 2.2GHz. Note that the systems were not optimized for speed. We used the best performing systems to measure the translation times.

The translation speed of the monotone phrase-based system for all three tasks is shown in Table 9. For the Xerox task, the translation process takes less than 7 seconds for the whole 10K words test set. For the Verbmobil task, the system is even slightly faster. It takes about 1.6 seconds to translate the whole test set. For the Canadian Hansards task, the translation process is much slower, but the average time per sentence is still less than 1 second. We think that this slowdown can be attributed to the large training corpus. The system loads only phrase pairs into memory if the source phrase occurs in the test corpus. Therefore, the large test corpus size for this task also affects the translation speed.

In Fig. 1, we see the average translation time per sentence as a function of the sentence length. The translation times were measured for the translation of the 5432 test sentences of the Canadian Hansards task. We see a clear linear dependency. Even for sentences of thirty words, the translation takes only about 1.5 seconds.

Table 9: Translation Speed for all tasks on a AMD Athlon 2.2GHz.

	Verbmobil	Xerox	Hansards
avg. sentence length	10.5	13.5	18.0
seconds / sentence	0.006	0.007	0.794
words / second	1642	1448	22.8

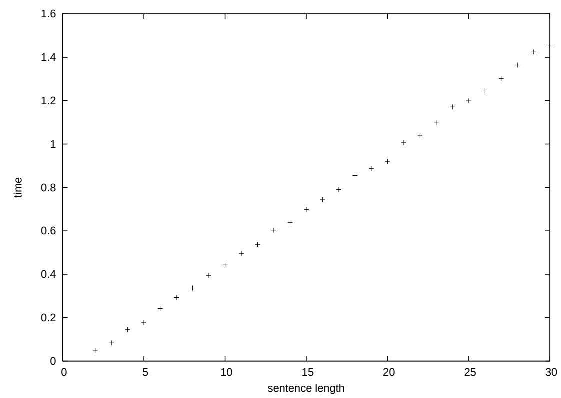

Figure 1: Average translation time per sentence as a function of the sentence length for the Canadian Hansards task (5432 test sentences).

9 Related Work

Recently, phrase-based translation approaches became more and more popular. Some examples are the alignment template system in (Och et al., 1999; Och and Ney, 2002) that we used for comparison. In (Zens et al., 2002), a simple phrase-based approach is described that served as starting point for the system in this work. (Marcu and Wong, 2002) presents a joint probability model for phrase-based translation. It does not use the word alignment for extracting the phrases, but directly generates a phrase alignment. In (Koehn et al., 2003), various aspects of phrase-based systems are compared, e.g. the phrase extraction method, the underlying word alignment model, or the maximum phrase length. (Tomas and Casacuberta, 2003) describes a linear interpolation of a phrase-based and an alignment template-based approach.

10 Conclusions

We described a phrase-based translation approach. The basic idea of this approach is to remember all bilingual phrases that have been seen in the word-aligned training corpus. As refinements of the baseline model, we described two simple heuristics: the word penalty feature and the phrase penalty feature. Additionally, we presented a single-word based lexicon with two smoothing methods. The model scaling factors were optimized with respect to the mWER on the development corpus.

We described a highly efficient monotone search algorithm. The worst-case complexity of this algorithm is linear in the sentence length. This leads to an impressive translation speed of more than 1000 words per second for the Verbmobil task and for the Xerox task. Even for the Canadian Hansards task the translation of sentences of length 30 takes only about 1.5 seconds.

The described search is monotone at the phrase level. Within the phrases, there are no constraints on the reorderings. Therefore, this method is best suited for language pairs that have a similar order at the level of the phrases learned by the system. Thus, the translation process should require only local reorderings. As the experiments have shown, Spanish-English and French-English are examples of such language pairs. For these pairs, the monotone search was found to be sufficient. The phrase-based approach clearly outperformed the single-word based systems. It showed even better performance than the alignment template system.

The experiments on the German-English Verbmobil task outlined the limitations of the monotone search. As the low degree of monotonicity indicated, reordering plays an important role on this task. The rather free word order in German as well as the verb group seems to be difficult to translate. Nevertheless, when ignoring the word order and looking at the mPER only, the monotone search is competitive with the best performing system.

For further improvements, we will investigate the usefulness of additional models, e.g. modeling the segmentation probability. Also, slightly relaxing the monotonicity constraint in a way that still allows an efficient search is of high interest. In spirit of the IBM reordering constraints of the single-word based models, we could allow a phrase to be skipped and to be translated later.

Acknowledgment

This work has been partially funded by the EU project TransType 2, IST-2001-32091.

References

A. L. Berger, P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, J. R. Gillett, A. S. Kehler, and R. L. Mercer. 1996. Language translation apparatus and method of using context-based translation models, United States patent, patent number 5510981, April.

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. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, June.

G. Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proc. ARPA Workshop on Human Language Technology*.

P. Koehn, F. J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *Proc. of the Human Language Technology Conf. (HLT-NAACL)*, pages 127–133, Edmonton, Canada, May/June.

D. Marcu and W. Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. Conf. on Empirical Methods for Natural Language Processing*, pages 133–139, Philadelphia, PA, July.

H. Ney, S. Martin, and F. Wessel. 1997. Statistical language modeling using leaving-one-out. In S. Young and G. Bloothooft, editors, *Corpus-Based Methods in Language and Speech Processing*, pages 174–207. Kluwer.

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

F. J. Och, C. Tillmann, and H. Ney. 1999. 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*, pages 20–28, University of Maryland, College Park, MD, June.

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

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

J. Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., San Mateo, CA. Revised second printing.

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

C. Tillmann and H. Ney. 2003. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational Linguistics*, 29(1):97–133, March.

J. Tomas and F. Casacuberta. 2003. Combining phrase-based and template-based aligned models in statistical translation. In *Proc. of the First Iberian Conf. on Pattern Recognition and Image Analysis*, pages 1020–1031, Mallorca, Spain, June.

W. Wahlster, editor. 2000. *Verbmobil: Foundations of speech-to-speech translations*. Springer Verlag, Berlin, Germany, July.

R. Zens, F. J. Och, and H. Ney. 2002. Phrase-based statistical machine translation. In *25th German Conference on Artificial Intelligence (KI2002)*, pages 18–32, Aachen, Germany, September. Springer Verlag.