Phrase-Based Statistical Machine Translation

Richard Zens, Franz Josef Och, and Hermann Ney

Human Language Technology and Pattern Recognition
Lehrstuhl für Informatik VI
Computer Science Department
RWTH Aachen – University of Technology
Germany

Abstract. This paper is based on the work carried out in the framework of the VERBMOBIL project, which is a limited-domain speech translation task (German-English). In the final evaluation, the statistical approach was found to perform best among five competing approaches.
In this paper, we will further investigate the used statistical translation models. A shortcoming of the single-word based model is that it does not take contextual information into account for the translation decisions. We will present a translation model that is based on bilingual phrases to explicitly model the local context. We will show that this model performs better than the single-word based model. We will compare monotone and non-monotone search for this model and we will investigate the benefit of using the sum criterion instead of the maximum approximation.

1 Introduction

In this paper, we will study some aspects of the phrase-based translation (PBT) approach in statistical machine translation. The baseline system we are using has been developed in the VERBMOBIL project [17].

In the final project evaluation [13], several approaches were evaluated on the same test data. In addition to a classical rule-based approach [4] and a dialogue-act based approach [12] there were three data-driven approaches, namely an example-based [1], a substring-based [2] and a statistical approach developed in the authors' group. The data-driven approaches were found to perform significantly better than the other two approaches. Out of the data-driven approaches the statistical approach performed best, e.g. the error rate for the statistical approach was 29% instead of 62% for the classical rule-based approach.

During the progress of the VERBMOBIL project, different variants of statistical translation systems have been implemented and experimental tests have been performed for text and speech input [7,10]. The major variants were:

- the single-word based approach (SWB), see Sect. 2.2
- the alignment template approach (AT), see Sect. 2.3

The evaluation showed that the AT approach performs much better than the SWB variant. It is still an open question, which components of the AT system are responsible for the improvement of the translation quality.

M. Jarke et al. (Eds.): KI 2002, LNAI 2479, pp. 18–32, 2002.

In this paper, we will review the baseline system and we will describe in detail a method to learn phrasal translations. We will compare SWB to phrase-based translation, monotone to non-monotone search, and the sum criterion to maximum approximation.

2 Review of the Baseline System

2.1 Bayes Decision Rule

The goal of machine translation is to automatically transfer the meaning of a source language sentence $f_1^J = f_1, ..., f_j, ..., f_J$ into a target language sentence $e_1^I = e_1, ..., e_i, ..., e_I$. In statistical machine translation, the conditional probability $Pr(e_1^I|f_1^J)$[1] is used to describe the correspondence between the two sentences. This model can be used directly for translation by solving the following maximization problem:

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

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

In the second equation, we have applied Bayes theorem. The decomposition into two knowledge sources makes the modeling easier. Now, we have two models:

1. the language model $Pr(e_1^I)$ and
2. the translation model $Pr(f_1^J|e_1^I)$.

The language model describes the correctness of the target language sentence. It helps to avoid syntactically incorrect sentences. A detailed description of language models can be found in [6]. This paper will focus on the translation model.

The resulting architecture for the statistical translation approach is shown in Fig. 1 with the translation model further decomposed into lexicon and alignment model.

2.2 Single Word-Based Translation Models

Concept. A key issue in modeling the string translation probability $Pr(f_1^J|e_1^I)$ is the question of how we define the correspondence between the words of the target sentence and the words of the source sentence. In typical cases, we can assume a sort of pairwise dependence by considering all word pairs (f_j, e_i) for a given sentence pair $(f_1^J; e_1^I)$. Models describing these types of dependencies are referred to as alignment models [3,16].

When aligning the words in parallel texts, we typically observe a strong localization effect. Figure 2 illustrates this effect for the language pair German-English. In many cases, although not always, there is an additional property: over large portions of the source string, the alignment is monotone.

[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 $p(\cdot)$.

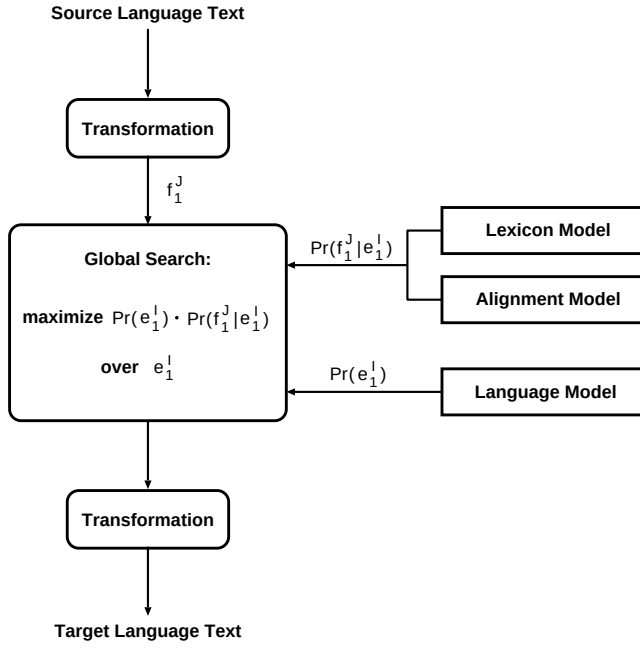

Fig. 1. Architecture of the translation approach based on Bayes decision rule

Basic Alignment Models. To arrive at a quantative specification, we define the alignment mapping: $j \rightarrow i = a_j$, which assigns a word f_j in position j to a word e_i in position $i = a_j$. We rewrite the probability for the translation model by introducing the 'hidden' alignments $a_1^J := a_1...a_j...a_J$ for each sentence pair $(f_1^J; e_1^I)$. to structure this probability distribution, we factorize it over the positions in the source sentence and limit the alignment dependencies to a first-order dependence:

$$Pr(f_1^J|e_1^I) = \sum_{a_1^J} Pr(f_1^J, a_1^J|e_1^I) \tag{3}$$

$$= \sum_{a_1^J} Pr(a_1^J|e_1^I) \cdot Pr(f_1^J|a_1^J, e_1^I) \tag{4}$$

$$= p(J|e_1^I) \cdot \sum_{a_1^J} \prod_{j=1}^{J} [p(a_j|a_{j-1}, I, J) \cdot p(f_j|e_{a_j})] \tag{5}$$

Here, we have the following probability distributions:

- the sentence length probability: $p(J|e_1^I)$, which is included here for completeness, but can be omitted without loss of performance;
- the lexicon probability : $p(f|e)$;
- the alignment probability: $p(a_j|a_{j-1}, I, J)$.

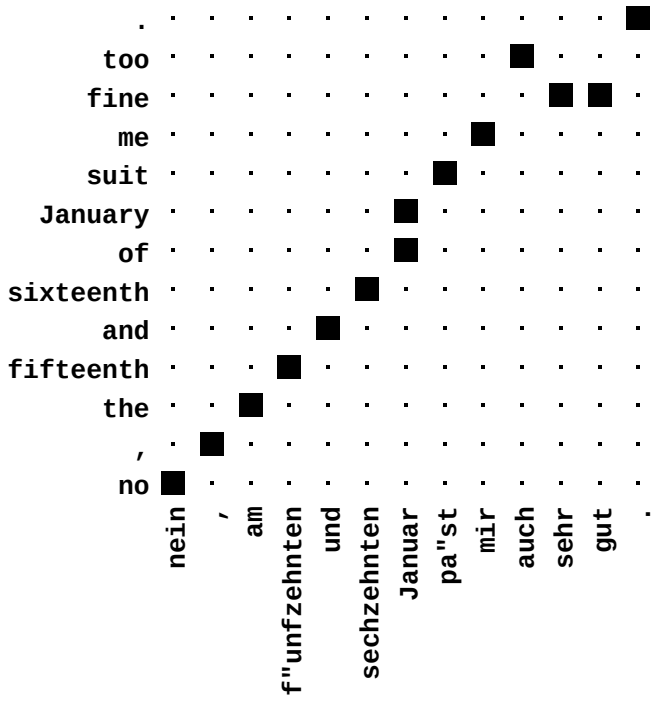

Fig. 2. Word aligned sentence pair

There are various ways to model the alignment probability. For further details, see [3,16]. Some of these models take one-to-many alignments explicitly into account, but the lexicon probabilities $p(f|e)$ are still based on single words in each of the two languages.

We will show the results for two search variants: a monotone search (later referred to as MON) that produces only monotone translations and a quasi-monotone search procedure (later referred to as GE). This proceeds from left to right along the positions of the source sentence but allows for a small number of source positions that are not processed monotonically. The word reorderings of the source sentence positions are restricted to the words of the German verb-group. For further details, see [15].

2.3 The Alignment Template Approach

The key element of the AT approach [10] are the *alignment templates*. These are pairs of source and target language phrases together with an alignment between the words within the phrases. The AT model decomposes the translation probability $Pr(f_1^J|e_1^I)$ by introducing two hidden variables: the sequence of alignment

templates z_1^K and the alignments within the templates $\tilde{a}_1^K$.

$$Pr(f_1^J|e_1^I) = \sum_{z_1^K, \tilde{a}_1^K} Pr(\tilde{a}_1^K|e_1^I) \cdot Pr(z_1^K|\tilde{a}_1^K, e_1^I) \cdot Pr(f_1^J|z_1^K, \tilde{a}_1^K, e_1^I) \quad (6)$$

$$= \sum_{z_1^K, \tilde{a}_1^K} \prod_{k=1}^{K} p(\tilde{a}_k|\tilde{a}_{k-1}) \cdot p(z_k|\tilde{e}_k) \cdot p(\tilde{f}_k|z_k, \tilde{e}_k) \quad (7)$$

There are three probability distributions:

- the phrase alignment probability $p(\tilde{a}_k|\tilde{a}_{k-1})$
- the probability of applying an alignment template $p(z_k|\tilde{e}_k)$
- the phrase translation probability $p(\tilde{f}_k|z_k, \tilde{e}_k)$

The AT approach uses a non-monotone search algorithm. The model scaling factors are trained with maximum entropy [9]. This is an extremely brief description of the AT model. For further details, see [10].

3 Phrase-Based Translation

3.1 Motivation

One major disadvantage of the single-word based (SWB) approach is that contextual information is not taken into account. As already said, the lexicon probabilities are based only on single words. For many words, the translation depends heavily on the surrounding words. In the SWB translation, this disambiguation is done completely by the language model. Often the language model is not capable of doing this. An example is shown in Fig. 3.

SOURCE	was halten Sie vom Hotel Gewandhaus ?
TARGET	what do you think about the hotel Gewandhaus ?
SWB	what do you from the hotel Gewandhaus ?
PBT	what do you think of hotel Gewandhaus ?

Fig. 3. Translation example

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 (PBT) is to segment the given source sentence into phrases, then to translate each phrase and finally compose the target sentence from these phrase translations as seen in Fig. 4. As seen in the last phrase pair of the example, punctuation marks are treated as normal words.

SOURCE: abends würde ich gerne entspannen und vielleicht in die Sauna gehen .	
source segmentation	translation
abends	in the evening
würde ich gerne entspannen	I would like to relax
und	and
vielleicht in die Sauna gehen	maybe go to the sauna
.	.
TARGET: in the evening I would like to relax and maybe go to the sauna .	

Fig. 4. Example for phrase based translation

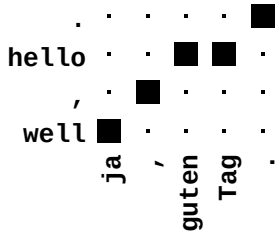

Fig. 5. Word aligned sentence pair

source phrase	target phrase
ja	well
ja,	well,
ja, guten Tag	well, hello
ja, guten Tag.	well, hello.
,	,
, guten Tag	, hello
, guten Tag.	, hello.
guten Tag	hello
guten Tag.	hello.
.	.

Fig. 6. Extracted bilingual phrases

3.2 Bilingual Phrases

Basically, a bilingual phrase is a pair of m source words and n target words. For extraction from a bilingual word aligned training corpus, we pose two additional constraints:

1. the words are consecutive and
2. they are consistent with the word alignment matrix.

This consistency means that the m source words are aligned only to the n target words and vice versa. The following criterion defines the set of bilingual phrases $\mathcal{BP}$ of the sentence pair $(f_1^J; e_1^I)$ that is consistent with the word alignment matrix A:

$$\mathcal{BP}(f_1^J, e_1^I, A) = \left\{ \left(f_j^{j+m}, e_i^{i+n} \right) : \forall (i', j') \in A : j \leq j' \leq j+m \leftrightarrow i \leq i' \leq i+n \right\}$$

This criterion is identical to the alignment template criterion described in [10]. Figure 5 is an example of a word aligned sentence pair. Figure 6 shows the bilingual phrases extracted from this sentence pair according to the defined criterion.

The extraction of the bilingual phrases from a bilingual word aligned training corpus can be done in a straightforward way. The algorithm in Fig. 7 computes the set $\mathcal{BP}$ with the assumption that the alignment is a function $A : \{1, ..., J\} \rightarrow \{1, ..., I\}$. It can be easily extended to deal with general alignments.

INPUT: f_1^J, e_1^I, A

FOR $i_2 = 1$ TO I DO

  FOR $i_1 = 1$ TO i_2 DO

    $SP = \{j | \exists i : i_1 \leq i \leq i_2 \wedge A(j) = i\}$

    IF consec(SP) THEN

      $j_1 = \min\{SP\}$

      $j_2 = \max\{SP\}$

      $\mathcal{BP} = \mathcal{BP} \cup \{(f_{j_1}^{j_2}, e_{i_1}^{i_2})\}$

OUTPUT: $\mathcal{BP}$

Fig. 7. Algorithm `extract-BP` for extracting bilingual phrases

3.3 Phrase-Based Translation Model

To use the bilingual phrases in the translation model we introduce the hidden variable B. 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. So, we obtain:

$$Pr(f_1^J|e_1^I) = \sum_B Pr(f_1^J, B|e_1^I) \tag{8}$$

$$= \sum_B Pr(B|e_1^I) \cdot Pr(f_1^J|B, e_1^I) \tag{9}$$

$$= \alpha(e_1^I) \cdot \sum_B Pr(\tilde{f}_1^K|\tilde{e}_1^K) \tag{10}$$

Here, we assume that all segmentations have the same probability $\alpha(e_1^I)$. Next, we allow only monotone translations. This will result in a very efficient search. 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.

$$Pr(\tilde{f}_1^K|\tilde{e}_1^K) = \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \tag{11}$$

Finally, we have to estimate the phrase translation probabilities $p(\tilde{f}|\tilde{e})$. This is done via relative frequencies:

$$p(\tilde{f}|\tilde{e}) = \frac{N(\tilde{f}, \tilde{e})}{N(\tilde{e})} \tag{12}$$

Here $N(\tilde{e})$ is the count of the phrase $\tilde{e}$. $N(\tilde{f}, \tilde{e})$ 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. If during the test an unknown word occurs, which was not seen in the training, this word is translated by itself.

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

$$\hat{e}_1^I = \underset{e_1^I}{\operatorname{argmax}} \left\{ Pr(e_1^I) \cdot Pr(f_1^J | e_1^I) \right\} \tag{13}$$

$$= \underset{e_1^I}{\operatorname{argmax}} \left\{ \prod_{i=1}^{I} p(e_i|e_{i-1}) \cdot \alpha(e_1^I) \cdot \sum_B \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \right\} \tag{14}$$

$$\approx \underset{e_1^I}{\operatorname{argmax}} \left\{ \prod_{i=1}^{I} p(e_i|e_{i-1}) \cdot \sum_B \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k)^\lambda \right\} \tag{15}$$

In the last step, we omitted the segmentation probability $\alpha(e_1^I)$. We also introduced the translation model scaling factor λ [14]. Using the maximum approximation for the sum over all segmentations, we obtain the following search criterion:

$$\hat{e}_1^I \approx \underset{e_1^I, B}{\operatorname{argmax}} \left\{ \prod_{i=1}^{I} p(e_i|e_{i-1}) \cdot \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k)^\lambda \right\} \tag{16}$$

3.4 Monotone Search

The monotone search can be efficiently computed with dynamic programming. For the maximization problem in (16), we define the quantity $Q(j, e)$ as the maximum probability of a phrase sequence that ends with the word e and covers positions 1 to j of the source sentence. $Q(J+1, \$)$ is the probability of the optimal translation. The \$ symbol is the sentence boundary marker. When finishing a hypothesis, we have to apply the conditional probability $p(\$|e')$, which denotes the probability of the sentence end after the word e'. We obtain the following dynamic programming recursion:

$$Q(0, \$) = 1 \tag{17}$$

$$Q(j, e) = \max_{\substack{0 \le j' < j, \\ e', \tilde{e}}} Q(j', e') \cdot p(f_{j'+1}^{j}|\tilde{e})^\lambda \cdot p(\tilde{e}|e') \tag{18}$$

$$Q(J+1, \$) = \max_{e'} Q(J, e') \cdot p(\$|e') \tag{19}$$

During the search, we store back-pointers to the maximizing arguments. So after performing the search, we can generate the optimal translation. This method will be later referred to as **MonMax**. The resulting algorithm has a worst-case complexity of $O(J^2 \cdot V_e \cdot |\{\tilde{e}\}|)$. Here, V_e denotes the vocabulary size of the target language and $|\{\tilde{e}\}|$ denotes the number of target language phrases. 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.

For the search criterion in (15), we define the quantity $Q(j, e_1^i)$ as the maximum probability of a phrase sequence that results in the word sequence e_1^i and covers the positions 1 to j of the source sentence. $Q(J+1, \hat{e}_1^I)$ is the probability of the optimal translation $\hat{e}_1^I$.

$$Q(0, \$) = 1 \tag{20}$$

$$Q(j, e_1^i) = \sum_{\substack{0 \le j' < j \\ 0 \le i' < i}} Q(j', e_1^{i'}) \cdot p(f_{j'+1}^j | e_{i'+1}^i)^\lambda \cdot p(e_{i'+1}^i | e_{i'}) \tag{21}$$

$$Q(J+1, \hat{e}_1^I) = \max_{e_1^I} Q(J, e_1^I) \cdot p(\$ | e_I) \tag{22}$$

This method will be later referred to as **MonSum**. The resulting algorithm has a worst-case complexity of $O(J^2 \cdot {V_e}^I \cdot |\{\tilde{e}\}|)$. Because of the sum criterion it is not allowed to apply language model recombination. This results in the factor ${V_e}^I$. In most statistical translation systems the maximum approximation is used, e.g. [3,5,10,18], but we will show in Sect. 4 that the sum criterion yields better results. These monotone algorithms are especially usefull for language pairs that have a similar word order, e.g. Spanish-Catalan or Italian-English.

3.5 Non-monotone Search

An analysis of the monotone translation results for the language pair German-English shows that many translation errors are due to the monotony constraint. Therefore in this section, we describe a way to extend the search described above to allow non-monotone translations. The idea is to use a reordering graph (RG) to restrict the number of possible word orders.

Reordering Graph. A RG is a directed acyclic graph with one start node and one goal node. The nodes are numbered from 0 (start) to N (goal). The numbering must be consistent with a topological order of the graph. Each node is marked with a coverage vector. This is a bit vector b of size J (the source sentence length) with the property $b[j] = 1$ iff the source position j is already covered, i.e. translated. The RG has the additional property that for each node its coverage vector differs from the coverage vector of each predecessor by exactly one bit. The start and the goal node are marked with 0^J and 1^J. We define $Pred(n)$ as the set of all predecessors (direct and indirect) of the node n. We define $S(n_1, n_2)$ as the source words covered by n_2 but not by n_1 in the same order as in the source sentence.

We gain a RG by removing non-needed information from a word graph generated by the SWB search and combining equivalent nodes, i.e. nodes with the same coverage vector.

Search. The search on the RG can be done by dynamic programming. The idea is similar to the monotone search, but instead of going over all source

positions j, we go over all nodes n of the RG from 0 to N. When visiting a node the topological sorting guarantees that all its predecessors have already been processed. Using maximum approximation, the quantity $Q(n, e)$ is defined as the maximum probability of a phrase sequence ending with the word e and translating $S(0, n)$.
We obtain the following dynamic programming recursion:

$$Q(0, \$) = 1 \tag{23}$$

$$Q(n, e) = \max_{\substack{n' \in Pred(n), \\ e', \tilde{e}}} Q(n', e') \cdot p(S(n', n)|\tilde{e})^{\lambda} \cdot p(\tilde{e}|e') \tag{24}$$

$$Q(N+1, \$) = \max_{e'} Q(N, e') \cdot p(\$|e') \tag{25}$$

This method will be later referred to as **ExtMax**. The equations for the sum criterion are analog. We define the quantity $Q(n, e_1^i)$ as the maximum probability of a phrase sequence that results in the word sequence e_1^i and translating $S(0, n)$. The method using the sum criterion will be later referred to as **ExtSum**.
We obtain the following dynamic programming recursion:

$$Q(0, \$) = 1 \tag{26}$$

$$Q(n, e_1^i) = \sum_{\substack{n' \in Pred(n), \\ 0 \leq i' < i}} Q(n', e_1^{i'}) \cdot p(S(n', n)|e_{i'+1}^i)^{\lambda} \cdot p(e_{i'+1}^i|e_{i'}) \tag{27}$$

$$Q(N+1, \hat{e}_1^I) = \max_{e_1^I} Q(N, e_1^I) \cdot p(\$|e_I) \tag{28}$$

3.6 Pruning

To further speed up the search and reduce the memory requirements, we apply threshold and histogram pruning. Note that with applying these pruning techniques the sum criterion is only approximately fulfilled. This is because if a hypothesis is pruned away, further contributions of extensions of this hypothesis are lost.

Threshold Pruning. The idea of threshold pruning is to remove all hypotheses that have a low probability relative to the best hypothesis. We need a threshold pruning parameter q, with $0 \leq q \leq 1$. We define $Q_0(j)$ as the maximum probability of all hypotheses for a source sentence position j. We prune a hypothesis iff:

$$Q(j, e_1^i) < q \cdot Q_0(j)$$

Histogram Pruning. The idea of histogram pruning is to restrict the maximum number of hypotheses for each source sentence position. So, only a fixed number of the best hypotheses is kept.

4 Results

4.1 Corpora

We present results on the VERBMOBIL task, which is a speech translation task in the domain of appointment scheduling, travel planning, and hotel reservation [17]. Table 1 shows the corpus statistics of this task. We use a training corpus, which is used to train the translation model and the language model, a development corpus, which is used to estimate the model scaling factors, and a test corpus.

Table 1. Characteristics of training corpus (Train, PM=punctuation marks), manual lexicon (Lex), development corpus (Dev), test corpus (Test)

		No Preprocessing		With Preprocessing	
		German	English	German	English
Train	Sentences	58 073			
	Words incl. PM	519 523	549 921	522 933	548 874
	Words excl. PM	418 979	453 632	421 689	456 629
	Singletons	3 453	1 698	3 570	1 763
	Vocabulary	7 940	4 673	8 102	4 780
Lex	Entries	12 779			
	Extended Vocabulary	11 501	6 867	11 904	7 089
Dev	Sentences	276			
	Words	3 159	3 438	3 172	3 445
	Trigram PP	-	28.1	-	26.3
Test	Sentences	251			
	Words	2 628	2 871	2 640	2 862
	Trigram PP	-	30.5	-	29.9

4.2 Criteria

So far, in machine translation research does not exist one generally accepted criterion for the evaluation of the experimental results. Therefore, we use a large variety of different criteria. In all experiments, we use the following error 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 target sentence. This performance criterion is widely used in speech recognition.
- PER (position-independent word error rate):
  A shortcoming of the WER is the fact 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. To overcome this problem, we introduce as additional measure the PER. This measure compares the words in the two sentences ignoring the word order.
- mWER (multi-reference word error rate):
  For each test sentence, there is not only used a single reference translation, as for the WER, but a whole set of reference translations. For each translation hypothesis, the edit distance (number of substitutions, deletions and insertions) to the most similar sentence is calculated [8].
- BLEU score:
  This score measures the precision of unigrams, bigrams, trigrams and fourgrams with respect to a whole set of reference translations with a penalty for too short sentences [11]. Unlike all other evaluation criteria used here, BLEU measures accuracy, i.e. the opposite of error rate. Hence, large BLEU scores are better.
- SSER (subjective sentence error rate):
  For a more detailed analysis, subjective judgments by test persons are necessary. Each translated sentence was judged by a human examiner according to an error scale from 0.0 to 1.0. A score of 0.0 means that the translation is semantically and syntactically correct, a score of 0.5 means that a sentence is semantically correct but syntactically wrong and a score of 1.0 means that the sentence is semantically wrong [8].
- IER (information item error rate):
  The test sentences are segmented into information items; for each of them, the translation candidates are assigned either "OK" or an error class. If the intended information is conveyed and there are no syntactic errors, the sentence is counted as correct [8].
- ISER (information item semantic error rate):
  This criterion is like the IER, but does not take into account slight syntactic errors.

4.3 PBT Variants

Table 2 shows the results for the PBT variants presented in this paper. As one may expect, the non-monotone variant yields better results than the monotone one. For the sum criterion, there is an improvement of the SSER of 7.8% absolute, which is 19.6% relative. We conclude that a for German-English translation non-monotone search is important to obtain good translation results. Typically in statistical translation systems the maximum approximation is used. Because of the simplicity of the presented PBT model, the sum over all segmentations can be carried out. Using the sum criterion instead of the maximum approximation improves translation quality. For the monotone search, there is an improvement of the SSER of 1.1% and for the non-monotone search of 0.6%.

4.4 Comparison with Other Systems

We compare the PBT results to the two other statistical translation systems, namely the SWB approach (see Sect. 2.2 and [15]) and the AT approach (see

Table 2. Comparison of different PBT variants

System	Variant	WER	PER	mWER	BLEU	SSER	IER	ISER
PBT	MonMax	46.9	33.3	42.0	42.1	40.8	40.2	19.0
	MonSum	45.9	32.2	40.9	43.4	39.7	40.0	19.2
PBT	ExtMax	42.5	30.4	36.7	49.5	32.5	33.0	17.7
	ExtSum	42.3	30.1	36.3	50.0	31.9	31.9	16.8

Table 3. Comparison of different translation systems

System	Variant	WER	PER	mWER	BLEU	SSER	IER	ISER
SWB	MON	49.0	35.2	43.4	37.0	47.0	51.7	33.2
	GE	41.9	31.4	35.9	47.5	35.1	39.0	21.6
PBT	MonSum	45.9	32.2	40.9	43.4	39.7	40.0	19.2
	ExtSum	42.3	30.1	36.3	50.0	31.9	31.9	16.8
AT		39.2	29.3	33.1	51.1	30.5	33.9	17.4

Sect. 2.3 and [10]). Some translation examples are shown in Table 4. In [2,13] an example-based approach is mentioned that is to some extend similar to PBT. The results are not included because they are evaluated on a different test set and therefore not comparable.

As Table 3 shows, the monotone PBT outperforms by far the monotone SWB translation. There is an improvement of the SSER of 7.3% absolute, which is 15.5% relative. The non-monotone PBT yields better results than the non-monotone SWB variant. There is an improvement of the SSER of 3.2%. So, this rather simple and straightforward phrase-based model performs better than the more complicated SWB model. We conclude that incorporating the local context into the translation model is important to achieve good translation results. One way to do this is the use of bilingual phrases.

On the other hand, PBT does not reach the performance of the AT approach, which is still 1.4% better. A possible reason is the generalization capability of the AT approach.

5 Conclusion

In this paper, we have presented a statistical translation model, which is based on bilingual phrases. Compared to the two other statistical approaches, this is a rather simple method, which results in a very efficient dynamic programming search algorithm. In the result section, we have compared this model to the SWB and AT approaches.

The major conclusions are:

1. Using bilingual phrases instead of single words in the translation model significantly improves translation quality.

Table 4. Translation examples

SOURCE	wollen wir am Abend Essen gehen ?
TARGET	would you like to go out for a meal in the evening ?
PBT MON	we will want evening go out to eat ?
PBT EXT	do we want to go out to eat in the evening ?

SOURCE	ich würde am dreißigsten gern mit dem Zug fahren .
TARGET	I would like to take the train on the thirtieth .
PBT MON	I would thirtieth like to go by train .
PBT EXT	on the thirtieth I would like to go by train .

SOURCE	dann müssen wir noch die Rückreise klären .
TARGET	then we still have to arrange the return journey .
PBT MON	then we still have to the return trip clarify .
PBT EXT	then we still have to clarify the return trip .

SOURCE	am Mittwoch fahren wir mit dem Zug wieder zurück nach Hamburg .
TARGET	on Wednesday we will go back by train to Hamburg .
SWB GE	on Wednesday we go by train from Hamburg again .
PBT MON	on Wednesday we go by train again back to Hamburg .

SOURCE	das Flugzeug ist dann um zwölf Uhr fünfundzwanzig in Hannover .
TARGET	the plane will arrive at Hanover at twenty-five past twelve .
SWB GE	the flight is at eleven twenty five in Hanover .
PBT MON	the plane is at twelve twenty-five in Hanover .

SOURCE	ich buche in dem Königshotel zwei Einzelzimmer mit Dusche .
TARGET	I will book two single rooms with a shower at the Königshotel .
SWB GE	I will book the Königshotel two singles with shower .
PBT MON	I will book in the Königshotel two single rooms with shower .
PBT EXT	I will book two single rooms in the Königshotel with shower .

2. For translating from German to English a non-monotone search is essential to produce good translations.
3. The sum criterion performs better than the maximum approximation.

Further investigations will concern the segmentation probability $Pr(B|e_1^I)$, which has so far been omitted. The use of hierarchical phrases, e.g. the pattern pairs in [2], might be interesting. Smoothing the phrase probabilities could result in additional improvements.

References

1. Auerswald, M.: Example-based machine translation with templates. [17] 418–427
2. Block, H.U.: Example-based incremental synchronous interpretation. [17] 411–417

3. Brown, P.F., Della Pietra, S.A., Della Pietra, V.J., Mercer, R.L.: The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics **19** (1993) 263–311
4. Emele, M.C., Dorna, M., Lüdeling, A., Zinsmeister, H., Rohrer, C.: Semantic-based transfer. [17] 359–376
5. Germann, U., Jahr, M., Knight, K., Marcu, D., Yamada, K.: Fast decoding and optimal decoding for machine translation. In: Proc. of the 39th Annual Meeting of the Association for Computational Linguistics (ACL), Toulouse, France (2001) 228–235
6. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing. MIT Press, Cambridge, MA (1999)
7. Ney, H., Nießen, S., Och, F.J., Sawaf, H., Tillmann, C., Vogel, S.: Algorithms for statistical translation of spoken language. IEEE Trans. on Speech and Audio Processing **8** (2000) 24–36
8. Nießen, S., Och, F.J., Leusch, G., Ney, H.: An evaluation tool for machine translation: Fast evaluation for MT research. In: Proc. of the Second Int. Conf. on Language Resources and Evaluation (LREC), Athens, Greece (2000) 39–45
9. Och, F.J., Ney, H.: Discriminative training and maximum entropy models for statistical machine translation. In: Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL). (2002) 8 pages To appear.
10. Och, F.J., Tillmann, C., Ney, H.: 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 (1999) 20–28
11. Papineni, K.A., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center (2001)
12. Reithinger, N., Engel, R.: Robust content extraction for translation and dialog processing. [17] 428–437
13. Tessiore, L., v. Hahn, W.: Functional validation of a machine interpretation system: Verbmobil. [17] 611–631
14. Tillmann, C.: Word re-ordering and dynamic programming based search algorithms for statistical machine translation. PhD thesis, Computer Science Department, RWTH Aachen, Germany (2001)
15. Tillmann, C., Ney, H.: Word re-ordering and DP-based search in statistical machine translation. In: COLING '00: The 18th Int. Conf. on Computational Linguistics, Saarbrücken, Germany (2000) 850–856
16. Vogel, S., Ney, H., Tillmann, C.: HMM-based word alignment in statistical translation. In: COLING '96: The 16th Int. Conf. on Computational Linguistics, Copenhagen, Denmark (1996) 836–841
17. Wahlster, W., ed.: Verbmobil: Foundations of speech-to-speech translations. Springer Verlag, Berlin, Germany (2000)
18. Wang, Y.Y., Waibel, A.: Modeling with structures in statistical machine translation. In: COLING-ACL '98: 36th Annual Meeting of the Association for Computational Linguistics and 17th Int. Conf. on Computational Linguistics. Volume 2, Montreal, Canada (1998) 1357–1363