The RWTH Phrase-based Statistical Machine Translation System

Richard Zens, Oliver Bender, Saša Hasan, Shahram Khadivi,
Evgeny Matusov, Jia Xu, Yuqi Zhang, Hermann Ney

Human Language Technology and Pattern Recognition
Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen University, D-52056 Aachen, Germany
{zens,bender,hasan,khadivi,matusov,xujia,yzhang,ney}@cs.rwth-aachen.de

Abstract

We give an overview of the RWTH phrase-based statistical machine translation system that was used in the evaluation campaign of the International Workshop on Spoken Language Translation 2005.

We use a two pass approach. In the first pass, we generate a list of the N best translation candidates. The second pass consists of rescoring and reranking this N-best list. We will give a description of the search algorithm as well as the models that are used in each pass.

We participated in the supplied data tracks for manual transcriptions for the following translation directions: Arabic-English, Chinese-English, English-Chinese and Japanese-English. For Japanese-English, we also participated in the C-Star track. In addition, we performed translations of automatic speech recognition output for Chinese-English and Japanese-English. For both language pairs, we translated the single-best ASR hypotheses. Additionally, we translated Chinese ASR lattices.

1. Introduction

We give an overview of the RWTH phrase-based statistical machine translation system that was used in the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2005.

We use a two pass approach. First, we generate a word graph and extract a list of the N best translation candidates. Then, we apply additional models in a rescoring/reranking approach.

This work is structured as follows: first, we will review the statistical approach to machine translation and introduce the notation that we will use in the later sections. Then, we will describe the models and algorithms that are used for generating the N-best lists, i.e., the first pass. In Section 4, we will describe the models that are used to rescore and rerank this N-best list, i.e., the second pass. Afterward, we will give an overview of the tasks and discuss the experimental results.

1.1. Source-channel approach to SMT

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

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

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

This decomposition into two knowledge sources is known as the source-channel approach to statistical machine translation [1]. It allows an independent modeling of the target language model $Pr(e_1^I)$ and the 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. The argmax operation denotes the search problem, i.e., the generation of the output sentence in the target language.

1.2. Log-linear model

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

$$Pr(e_1^I|f_1^J) = \frac{\exp\left(\sum_{m=1}^M \lambda_m h_m(e_1^I, f_1^J)\right)}{\sum_{e'^{I'}_1} \exp\left(\sum_{m=1}^M \lambda_m h_m(e'^{I'}_1, f_1^J)\right)} \quad (3)$$

The denominator represents a normalization factor that depends only on the source sentence f_1^J. Therefore, we can omit it during the search process. As a decision rule, we obtain:

$$\hat{e}_1^{\hat{I}} = \operatorname*{argmax}_{I,e_1^I} \left\{ \sum_{m=1}^M \lambda_m h_m(e_1^I, f_1^J) \right\} \quad (4)$$

[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 $p(\cdot)$.

This approach is a generalization of the source-channel approach. It has the advantage that additional models $h(\cdot)$ 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 an error criterion [3]. For the IWSLT evaluation campaign, we optimized the scaling factors with respect to a linear interpolation of WER, PER, BLEU and NIST using the Downhill Simplex algorithm from [4].

1.3. Phrase-based approach

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. This idea is illustrated in Figure 1. Formally, we define a segmentation of a given sentence pair (f_1^J, e_1^I) into K blocks:

$$k \quad \rightarrow \quad s_k := (i_k; b_k, j_k), \text{ for } k = 1 \ldots K. \tag{5}$$

Here, i_k denotes the last position of the k^{th} target phrase; we set $i_0 := 0$. The pair (b_k, j_k) denotes the start and end positions of the source phrase that is aligned to the k^{th} target phrase; we set $j_0 := 0$. Phrases are defined as nonempty contiguous sequences of words. We constrain the segmentations so that all words in the source and the target sentence are covered by exactly one phrase. Thus, there are no gaps and there is no overlap. For a given sentence pair (f_1^J, e_1^I) and a given segmentation s_1^K, we define the bilingual phrases as:

$$\tilde{e}_k := e_{i_{k-1}+1} \ldots e_{i_k} \tag{6}$$

$$\tilde{f}_k := f_{b_k} \ldots f_{j_k} \tag{7}$$

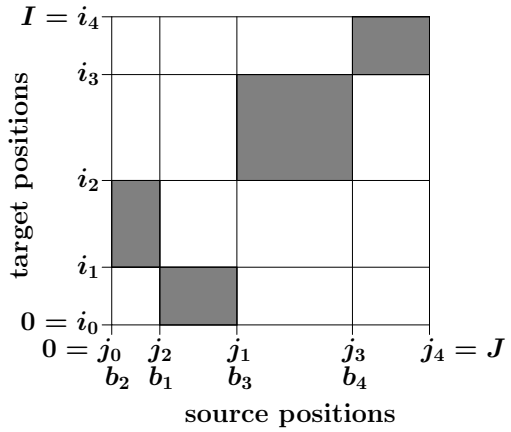

Figure 1: Illustration of the phrase segmentation.

Note that the segmentation s_1^K contains the information on the phrase-level reordering. The segmentation s_1^K is introduced as a hidden variable in the translation model. Therefore, it would be theoretically correct to sum over all possible segmentations. In practice, we use the maximum approximation for this sum. As a result, the models $h(\cdot)$ depend not only on the sentence pair (f_1^J, e_1^I), but also on the segmentation s_1^K, i.e., we have models $h(f_1^J, e_1^I, s_1^K)$.

2. Search algorithms

The RWTH phrase-based system supports two alternative search strategies that will be described in this section.

Translating a source language word graph. The first search strategy that our system supports takes a source language word graph as input and translates this graph in a monotone way [5]. The input graph can represent different reorderings of the input sentence so that the overall search can generate nonmonotone translations. Using this approach, it is very simple to experiment with various reordering constraints, e.g., the constraints proposed in [6].

Alternatively, we can use ASR lattices as input and translate them without changing the search algorithm, cf. [7]. A disadvantage when translating lattices with this method is that the search is monotone. To overcome this problem, we extended the monotone search algorithm from [5, 7] so that it is possible to reorder the target phrases. We implemented the following idea: while traversing the input graph, a phrase can be skipped and processed later.

Source cardinality synchronous search. For single-word based models, this search strategy is described in [8]. The idea is that the search proceeds synchronously with the cardinality of the already translated source positions. Here, we use a phrase-based version of this idea. To make the search problem feasible, the reorderings are constrained as in [9].

Word graphs and N-best lists. The two described search algorithms generate a word graph containing the most likely translation hypotheses. Out of this word graph we extract N-best lists. For more details on word graphs and N-best list extraction, see [10, 11].

3. Models used during search

We use a log-linear combination of several models (also called feature functions). In this section, we will describe the models that are used in the first pass, i.e., during search. This is an improved version of the system described in [12]. More specifically the models are: a phrase translation model, a word-based translation model, a deletion model, word and phrase penalty, a target language model and a reordering model.

3.1. Phrase-based model

The phrase-based translation model is the main component of our translation system. The hypotheses are generated by concatenating target language phrases. The pairs of source and corresponding target phrases are extracted from the word-aligned bilingual training corpus. The phrase extraction algorithm is described in detail in [5]. The main idea is to extract phrase pairs that are consistent with the word alignment. Thus, the words of the source phrase are aligned only

to words in the target phrase and vice versa. This criterion is identical to the alignment template criterion described in [13].

We use relative frequencies to estimate the phrase translation probabilities:

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

Here, the number of co-occurrences of a phrase pair $(\tilde{f},\tilde{e})$ that are consistent with the word alignment is denoted as $N(\tilde{f},\tilde{e})$. If one occurrence of a target phrase $\tilde{e}$ has $N > 1$ possible translations, each of them contributes to $N(\tilde{f},\tilde{e})$ with $1/N$. The marginal count $N(\tilde{e})$ is the number of occurrences of the target phrase $\tilde{e}$ in the training corpus. The resulting feature function is:

$$h_{\text{Phr}}(f_1^J, e_1^I, s_1^K) = \log \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \quad (9)$$

To obtain a more symmetric model, we use the phrase-based model in both directions $p(\tilde{f}|\tilde{e})$ and $p(\tilde{e}|\tilde{f})$.

3.2. Word-based lexicon model

We are using relative frequencies to estimate the phrase translation probabilities. Most of the longer phrases occur 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.

The score of a phrase pair is computed similar to the IBM model 1, but here, we are summing only within a phrase pair and not over the whole target language sentence:

$$h_{\text{Lex}}(f_1^J, e_1^I, s_1^K) = \log \prod_{k=1}^{K} \prod_{j=b_k}^{j_k} \sum_{i=i_{k-1}+1}^{i_k} p(f_j|e_i) \quad (10)$$

The word translation probabilities $p(f|e)$ are estimated as relative frequencies from the word-aligned training corpus. The word-based lexicon model is also used in both directions $p(f|e)$ and $p(e|f)$.

3.3. Deletion model

The deletion model [14] is designed to penalize hypotheses that miss the translation of a word. For each source word, we check if a target word with a probability higher than a given threshold τ exists. If not, this word is considered a deletion. The feature simply counts the number of deletions. Last year [15], we used this model during rescoring only, whereas this year, we integrated a within-phrase variant of the deletion model into the search:

$$h_{\text{Del}}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} \sum_{j=b_k}^{j_k} \prod_{i=i_{k-1}+1}^{i_k} [\, p(f_j|e_i) < \tau \,] \quad (11)$$

The word translation probabilities $p(f|e)$ are the same as for the word-based lexicon model. We use $[\cdot]$ to denote a true or false statement [16], i.e., the result is 1 if the statement is true, and 0 otherwise. In general, we use the following convention:

$$[\,\mathcal{C}\,] = \begin{cases} 1, & \text{if condition } \mathcal{C} \text{ is true} \\ 0, & \text{if condition } \mathcal{C} \text{ is false} \end{cases} \quad (12)$$

3.4. Word and phrase penalty model

In addition, we use two simple heuristics, namely word penalty and phrase penalty:

$$h_{\text{WP}}(f_1^J, e_1^I, s_1^K) = I \quad (13)$$

$$h_{\text{PP}}(f_1^J, e_1^I, s_1^K) = K \quad (14)$$

These two models affect the average sentence and phrase lengths. The model scaling factors can be adjusted to prefer longer sentences and longer phrases.

3.5. Target language model

We use the SRI language modeling toolkit [17] to train a standard n-gram language model. The smoothing technique we apply is the modified Kneser-Ney discounting with interpolation. The order of the language model depends on the translation direction. For most tasks, we use a trigram model, except for Chinese-English, where we use a fivegram language model. The resulting feature function is:

$$h_{\text{LM}}(f_1^J, e_1^I, s_1^K) = \log \prod_{i=1}^{I} p(e_i|e_{i-n+1}^{i-1}) \quad (15)$$

3.6. Reordering model

We use a very simple reordering model that is also used in, for instance, [13, 15]. It assigns costs based on the jump width:

$$h_{\text{RM}}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} |b_k - j_{k-1} - 1| + J - j_k \quad (16)$$

4. Rescoring models

The usage of N-best lists in machine translation has several advantages. It alleviates the effects of the huge search space which is represented in word graphs by using a compact excerpt of the N best hypotheses generated by the system. Especially for small tasks, such as the IWSLT supplied data track, rather small N-best lists are already sufficient to obtain good oracle error rates, i.e., the error rate of the best hypothesis with respect to an error measure (such as WER or BLEU). N-best lists are suitable for easily applying several rescoring techniques because the hypotheses are already fully generated. In comparison, word graph rescoring techniques need specialized tools which traverse the graph appropriately. Additionally, because a node within a word graph

allows for many histories, one can only apply local rescoring techniques, whereas for N-best lists, techniques can be used that consider properties of the whole target sentence.

In the next sections, we will present several rescoring techniques.

4.1. Clustered language models

One of the first ideas in rescoring is to use additional language models that were not used in the generation procedure. In our system, we use clustered language models based on regular expressions [18]. Each hypothesis is classified by matching it to regular expressions that identify the type of the sentence. Then, a cluster-specific (or sentence-type-specific) language model is interpolated into a global language model to compute the score of the sentence:

$$h_{\text{CLM}}(f_1^J, e_1^I) = \log \sum_c \left[\mathcal{R}_c(e_1^I)\right] \left(\alpha_c p_c(e_1^I) + (1-\alpha_c) p_g(e_1^I)\right), \quad (17)$$

where $p_g(e_1^I)$ is the global language model, $p_c(e_1^I)$ the cluster-specific language model, and $\left[\mathcal{R}_c(e_1^I)\right]$ denotes the true-or-false statement (cf. Equation 12) which is 1 if the c^{th} regular expression $\mathcal{R}_c(\cdot)$ matches the target sentence e_1^I and 0 otherwise.[2]

4.2. IBM model 1

IBM model 1 rescoring rates the quality of a sentence by using the probabilities of one of the easiest single-word based translation models:

$$h_{\text{IBM1}}(f_1^J, e_1^I) = \log \left(\frac{1}{(I+1)^J} \prod_{j=1}^{J} \sum_{i=0}^{I} p(f_j|e_i) \right) \quad (18)$$

Despite its simplicity, this model achieves good improvements [14].

4.3. IBM1 deletion model

During the IBM model 1 rescoring step, we make use of another rescoring technique that benefits from the IBM model 1 lexical probabilities:

$$h_{\text{Del}}(f_1^J, e_1^I) = \sum_{j=1}^{J} \prod_{i=0}^{I} \left[\, p(f_j|e_i) < \tau \,\right] \quad (19)$$

We call this the IBM1 deletion model. It counts all source words whose lexical probability given each target word is below a threshold τ. In the experiments, τ was chosen between 10^{-1} and 10^{-4}.

4.4. Hidden Markov alignment model

The next step after IBM model 1 rescoring is HMM rescoring. We use the HMM to compute the log-likelihood of a sentence pair (f_1^J, e_1^I):

$$h_{\text{HMM}}(f_1^J, e_1^I) = \log \sum_{a_1^J} \prod_{j=1}^{J} \left(p(a_j|a_{j-1}, I) \cdot p(f_j|e_{a_j})\right) \quad (20)$$

In our experiments, we use a refined alignment probability $p(a_j - a_{j-1}|G(e_{a_j}), I)$ that conditions the jump widths of the alignment positions $a_j - a_{j-1}$ on the word class $G(e_{a_j})$. This is the so-called homogeneous HMM [19].

4.5. Word penalties

Several word penalties are used in the rescoring step:

$$h_{\text{WP}}(f_1^J, e_1^I) = \begin{cases} I & (a) \\ I/J & (b) \\ 2|I-J|/(I+J) & (c) \end{cases} \quad (21)$$

The word penalties are heuristics that affect the generated hypothesis length. In general, sentences that are too short should be avoided.

5. Integrating ASR and MT

In the experiments on coupling speech recognition and machine translation, we used the phrase-based MT system described in Section 2 to translate ASR lattices. In addition to the models described in Section 3, we use the acoustic model and the source language model of the ASR system in the log-linear model. These models are integrated into the search and the scaling factors are also optimized.

A significant obstacle for integrating speech recognition and translation is the mismatch between the vocabularies of the ASR and MT system. For the Chinese-English task, the number of out-of-vocabulary (OOV) words was rather high. Ideally, the vocabulary of the recognition system should be a subset of the translation system source vocabulary. In the IWSLT evaluation, we had no control over the recognition experiments. For this reason, the reported improvements might have been larger with a proper handling of the vocabularies.

6. Tasks and corpora

The experiments were carried out on the *Basic Travel Expression Corpus* (BTEC) task [20]. This is a multilingual speech corpus which contains tourism-related sentences similar to those that are found in phrase books. The corpus statistics are shown in Table 1. For the supplied data track, 20 000 sentences training corpus and two test sets (C-Star'03 and IWSLT'04) were made available for each language pair. As additional training resources for the C-Star track, we used the full BTEC for Japanese-English and the *Spoken Language DataBase* (SLDB) [21], which consists of transcriptions of spoken dialogs in the domain of hotel reservations[3].

[2] The clusters are disjunct, thus only one regular expression matches.

[3] The Japanese-English training corpora (BTEC, SLDB) that we used in the C-Star track were kindly provided by ATR Spoken Language Translation

Table 2: Statistics for the Chinese ASR lattices of the three test sets.

Test Set	WER [%]	GER [%]	Density
C-Star'03	41.4	16.9	13
IWSLT'04	44.5	20.2	13
IWSLT'05	42.0	18.2	14

For the Japanese-English supplied data track, the number of OOVs in the IWSLT'05 test set is rather high, both in comparison with the C-Star'03 and IWSLT'04 test sets and in comparison with the number of OOVs for the other language pairs. As for any data-driven approach, the performance of our system deteriorates due to the high number of OOVs. Using the additional corpora in the C-Star track, we are able to reduce the number of OOVs to a noncritical number.

As the BTEC is a rather clean corpus, the preprocessing consisted mainly of tokenization, i.e., separating punctuation marks from words. Additionally, we replaced contractions such as *it's* or *I'm* in the English corpus and we removed the case information. For Arabic, we removed the diacritics and we split common prefixes: Al, w, f, b, l. There was no special preprocessing for the Chinese and the Japanese training corpora.

We used the C-Star'03 corpus as development set to optimize the system, for instance, the model scaling factors and the GIZA++ [19] parameter settings. The IWSLT'04 test set was used as a blind test corpus. After the optimization, we added the C-Star'03 and the IWSLT'04 test sets to the training corpus and retrained the whole system.

We performed speech translation experiments on the Chinese-English and Japanese-English supplied data tracks. For Japanese-English we translated the single-best ASR hypotheses only, whereas for Chinese-English we also translated ASR lattices. The preprocessing and postprocessing steps are the same as for text translation.

Table 2 contains the Chinese ASR word lattice statistics for the three test sets. The ASR WER and the graph error rate (GER) were measured at the word level (and not at the character level). The GER is the minimum WER among all paths through the lattice.

7. Experimental results

The automatic evaluation criteria are computed using the IWSLT 2005 evaluation server. For all the experiments, we report the two accuracy measures BLEU [22] and NIST [23] as well as the two error rates WER and PER. For the primary submissions, we also report the two accuracy measures Meteor [24] and GTM [25]. All those criteria are computed with respect to multiple references (with the exception of English-Chinese where only one reference is available).

Research Laboratories, Kyoto, Japan.

Table 4: Progress over time: comparison of the RWTH systems of the years 2004 and 2005 for the supplied data track on the IWSLT'04 test set.

Translation Direction	System	BLEU [%]	NIST	WER [%]	PER [%]
Chin.-Engl.	2004	40.4	8.59	52.4	42.2
	2005	46.3	8.73	47.4	39.7
Jap.-Engl.	2004	44.8	9.41	50.0	37.7
	2005	49.8	9.52	46.5	36.8

7.1. Primary submissions

The translation results of the RWTH primary submissions are summarized in Table 3. Note that for English-Chinese, only one reference was used. Therefore the scores are in a different range.

7.2. Results for text input

In Table 4, we compare the translation performance of the RWTH 2004 system [15] and our current system. The evaluation is done on the IWSLT'04 test set for the supplied data track using the IWSLT 2005 evaluation server. Note that the reported numbers for the 2004 system differ slightly from the numbers in [15] due to a somewhat different computation. We observe significant improvements for all evaluation criteria and for both language pairs. For the Chinese-English system, for instance, the BLEU score increases by 4.9% and the WER decreases by 5%. Similar improvements are obtained for the Japanese-English system.

In Table 5, we present some translation examples for Japanese-English. As already mentioned in the previous section, our data-driven approach suffers from the high number of OOVs for the supplied data track. This becomes apparent when looking at the translation hypotheses. Furthermore, the incorporation of additional training data improves the translation quality significantly, not only in terms of the official results (cf. Table 3) but also when considering the examples in Table 5. In all three examples, the C-Star data track system is able to produce one of the reference translations. On the other hand, the output of the supplied data track system is of much lower quality. In the first example, we see the effect of a single unknown word. In the second example, the word choice is more or less correct, but the fluency of the output is very poor. The translation in the final example is entirely incomprehensible for the supplied data track system.

The effects of the N-best list rescoring for the IWSLT'04 test set are summarized in Table 6. On the development set (C-Star'03), which was used to optimize the model scaling factors, all models gradually help to enhance the overall performance of the system, e.g., BLEU is improved from 45.5% to 47.4%. For the IWSLT'04 blind test set, the results are not as smooth, but still the overall system (using all models that were described in Section 4) achieves improvements in

Table 1: Corpus statistics after preprocessing.

		Supplied Data Track				C-Star Track	
		Arabic	Chinese	Japanese	English	Japanese	English
Train	Sentences	20 000				240 672	
	Running Words	180 075	176 199	198 453	189 927	1 951 311	1 775 213
	Vocabulary	15 371	8 687	9 277	6 870	26 036	14 120
	Singletons	8 319	4 006	4 431	2 888	8 975	3 538
C-Star'03	Sentences	506					
	Running Words	3 552	3 630	4 130	3 823	4 130	3 823
	OOVs (Running Words)	133	114	61	65	34	–
IWSLT'04	Sentences	500					
	Running Words	3 597	3 681	4 131	3 837	4 131	3 837
	OOVs (Running Words)	142	83	71	58	36	–
IWSLT'05	Sentences	506					
	Running Words	3 562	3 918	4 226	3 909	4 226	3 909
	OOVs (Running Words)	146	90	293	69	10	–

Table 3: Official results for the RWTH primary submissions on the IWSLT'05 test set.

Data Track	Input	Translation Direction	Accuracy Measures				Error Rates	
			BLEU [%]	NIST	Meteor [%]	GTM [%]	WER [%]	PER [%]
Supplied	Manual	Arabic-English	54.7	9.78	70.8	65.6	37.1	31.9
		Chinese-English	51.1	9.57	66.5	60.1	42.8	35.8
		English-Chinese	20.0	5.09	12.6	55.2	61.2	52.7
		Japanese-English	40.8	7.86	58.6	48.6	53.6	44.4
	ASR	Chinese-English	38.3	7.39	54.0	48.8	56.5	47.2
		Japanese-English	42.7	8.53	62.0	49.6	51.2	41.2
C-Star	Manual	Japanese-English	77.6	12.91	85.4	78.7	24.3	18.6

Table 5: Translation examples for the Japanese-English supplied and C-Star data tracks.

Data Track	Translation
Supplied	*What would you like*
C-Star	*What would you like for the main course*
Reference	*What would you like for the main course*
Supplied	*Is that flight two seats available*
C-Star	*Are there two seats available on that flight*
Reference	*Are there two seats available on that flight*
Supplied	*Have a good I anything new*
C-Star	*I prefer something different*
Reference	*I prefer something different*

Table 6: Rescoring: effect of successively adding models for the Chinese-English IWSLT'04 test set.

System	BLEU [%]	NIST	WER [%]	PER [%]
Baseline	45.1	8.56	48.9	40.1
+CLM	45.9	8.24	48.6	40.7
+IBM1	45.9	8.48	47.8	39.7
+WP	45.4	8.91	47.8	39.4
+Del	46.0	8.71	47.8	39.6
+HMM	46.3	8.73	47.4	39.7

all evaluation criteria. In Table 7, we show some examples where the impact of the rescoring models can be seen.

7.3. Results for ASR input

The translation results for the IWSLT'05 test set for ASR input in the Chinese-English supplied data track are summarized in Table 8.

We report the results for the two search strategies described in Section 2. Using the first strategy (Graph), we are able to translate ASR lattices. We observe significant improvements in translation quality over the translations of the single-best (1-Best) recognition results. This is true for the monotone search (Mon) as well as for the version which allows for reordering of target phrases (Skip). The improvements are consistent among all evaluation criteria.

Table 7: Translation examples for the Chinese-English supplied data track: effect of rescoring.

System	Translation
Baseline	*Your coffee or tea*
+Rescoring	*Would you like coffee or tea*
Reference	*Would you like coffee or tea*
Baseline	*A room with a bath*
+Rescoring	*I would like a twin room with a bath*
Reference	*A twin room with bath*
Baseline	*How much is that will be that room*
+Rescoring	*How much is that room including tax*
Reference	*How much is the room including tax*
Baseline	*Onions*
+Rescoring	*I would like onion*
Reference	*I would like onions please*

Table 8: Translation results for ASR input in the Chinese-English supplied data track on the IWSLT'05 test set (*: late submissions).

System		Input	BLEU [%]	NIST	WER [%]	PER [%]
Graph	Mon*	1-Best	31.1	6.18	62.1	52.7
		Lattice	34.1	7.20	58.3	48.1
	Skip	1-Best	33.1	6.51	61.3	51.7
		Lattice	35.1	7.53	57.7	47.2
SCSS (primary)		1-Best	38.3	7.39	56.5	47.2
+Rescoring*			40.2	7.33	55.1	46.5

Using the second search strategy (SCSS), we are limited to the single-best ASR hypotheses as input. This is the same system that is used to translate the manual transcriptions. Despite the limitation to the single-best hypotheses, this system performs best in terms of the automatic evaluation measures (except for the NIST score).

The RWTH Chinese-English primary systems for ASR did not include rescoring. After the evaluation, we applied the rescoring techniques (described in Section 4) to the primary system. The improvements from rescoring are similar to the text system, e.g., 1.9% for the BLEU score.

Even if our primary system did not use lattices, a subjective comparison of the two systems showed positive effects when translating lattices for a large number of sentences. Recognition errors that occur in the single-best ASR hypotheses are often corrected when lattices are used. Some translation examples for improvements with lattices are shown in Table 9.

Table 9: Translation examples for ASR input in the Chinese-English supplied data track.

Input	Translation
1-Best	*Is there a pair of room with a bath*
Lattice	*I would like a twin room with a bath*
Reference	*A double room including a bath*
1-Best	*Please take a picture of our*
Lattice	*May I take a picture here*
Reference	*Am I permitted to take photos here*
1-Best	*I'm in a does the interesting*
Lattice	*I'm in an interesting movie*
Reference	*A good movie is on*

8. Conclusions

We have described the RWTH phrase-based statistical machine translation system that was used in the evaluation campaign of the IWSLT 2005. We use a two pass approach. In the first pass, we use a dynamic programming beam search algorithm to generate an N-best list. The second pass consists of rescoring and reranking of this N-best list.

One important advantage of our data-driven machine translation systems is that virtually the same system can be used for the different translation directions. Only a marginal portion of the overall performance can be attributed to language-specific methods.

We have shown significant improvements compared to the RWTH system of 2004 [15].

We have shown that the translation of ASR lattices can yield significant improvements over the translation of the ASR single-best hypotheses.

9. Acknowledgments

This work was partly funded by the DFG (Deutsche Forschungsgemeinschaft) under the grant NE572/5-1, project "Statistische Textübersetzung" and by the European Union under the integrated project TC-Star (Technology and Corpora for Speech to Speech Translation, IST-2002-FP6-506738, http://www.tc-star.org).

10. References

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

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

[3] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proc. of the 41th Annual Meeting of the Asso-*

ciation for Computational Linguistics (ACL), Sapporo, Japan, July 2003, pp. 160–167.

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

[5] R. Zens, F. J. Och, and H. Ney, "Phrase-based statistical machine translation," in *25th German Conf. on Artificial Intelligence (KI2002)*, ser. Lecture Notes in Artificial Intelligence (LNAI), M. Jarke, J. Koehler, and G. Lakemeyer, Eds., vol. 2479. Aachen, Germany: Springer Verlag, September 2002, pp. 18–32.

[6] S. Kanthak, D. Vilar, E. Matusov, R. Zens, and H. Ney, "Novel reordering approaches in phrase-based statistical machine translation," in *43rd Annual Meeting of the Assoc. for Computational Linguistics: Proc. Workshop on Building and Using Parallel Texts: Data-Driven Machine Translation and Beyond*, Ann Arbor, MI, June 2005, pp. 167–174.

[7] E. Matusov and H. Ney, "Phrase-based translation of speech recognizer word lattices using loglinear model combination," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, Cancun, Mexiko, Nov/Dec 2005, to appear.

[8] C. Tillmann and H. Ney, "Word reordering and a dynamic programming beam search algorithm for statistical machine translation," *Computational Linguistics*, vol. 29, no. 1, pp. 97–133, March 2003.

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

[10] R. Zens and H. Ney, "Word graphs for statistical machine translation," in *43rd Annual Meeting of the Assoc. for Computational Linguistics: Proc. Workshop on Building and Using Parallel Texts: Data-Driven Machine Translation and Beyond*, Ann Arbor, MI, June 2005, pp. 191–198.

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

[12] R. Zens and H. Ney, "Improvements in phrase-based statistical machine translation," in *Proc. of the Human Language Technology Conf. (HLT-NAACL)*, Boston, MA, May 2004, pp. 257–264.

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

[14] F. J. Och, D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, and D. Radev, "Syntax for statistical machine translation," Johns Hopkins University 2003 Summer Workshop on Language Engineering, Center for Language and Speech Processing, Baltimore, MD, Tech. Rep., August 2003.

[15] O. Bender, R. Zens, E. Matusov, and H. Ney, "Alignment Templates: the RWTH SMT System," in *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, Kyoto, Japan, September 2004, pp. 79–84.

[16] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*, 2nd ed. Reading, Mass.: Addison-Wesley Publishing Company, 1994.

[17] A. Stolcke, "SRILM – an extensible language modeling toolkit," in *Proc. Int. Conf. on Spoken Language Processing*, vol. 2, Denver, CO, 2002, pp. 901–904.

[18] S. Hasan and H. Ney, "Clustered language models based on regular expressions for SMT," in *Proc. of the 10th Annual Conf. of the European Association for Machine Translation (EAMT)*, Budapest, Hungary, May 2005.

[19] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, March 2003.

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

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

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

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

[24] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *43rd Annual Meeting of the Assoc. for Computational Linguistics: Proc. Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*, Ann Arbor, MI, June 2005.

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