The RWTH Machine Translation System for IWSLT 2007

Arne Mauser, David Vilar, Gregor Leusch,
Yuqi Zhang and Hermann Ney

Human Language Technology and Pattern Recognition
Lehrstuhl für Informatik 6, Computer Science Department
RWTH Aachen University, D-52056 Aachen, Germany
{mauser,vilar,leusch,yzhang,ney}@cs.rwth-aachen.de

Abstract

The RWTH system for the IWSLT 2007 evaluation is a combination of several statistical machine translation systems. The combination includes Phrase-Based models, a n-gram translation model and a hierarchical phrase model. We describe the individual systems and the method that was used for combining the system outputs. Compared to our 2006 system, we newly introduce a hierarchical phrase-based translation model and show improvements in system combination for Machine Translation. RWTH participated in the Italian-to-English and Chinese-to-English translation directions.

1. Introduction

The RWTH system for the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2007 is a combination of different statistical machine translation systems: several phrase-based, an n-gram based and a hierarchical MT system. In this paper, we describe the individual systems and the algorithm used to combine them.

In the evaluation, the system participated in the Italian-to-English and Chinese-to-English translation directions. It was ranked first for Italian-to-English correct transcription and second for automatic recognition output. For Chinese-to-English, the primary system was ranked fourth, the best secondary system was ranked second.

The following section describes the statistical approach to machine translation. Section 3 forms the main part of this paper and gives a detailed description of the models and submodels used in the combination. The method for system combination is described in Section 4. The task, system setup and results are discussed in Sections 5, 6 and 7 separately for the two translation directions. Finally, Section 9 presents the conclusions drawn from the 2007 IWSLT evaluation.

2. Statistical Machine Translation

In statistical machine translation , we are given a source language sentence $f_1^J = f_1 \ldots f_j \ldots f_J$, which is to be translated into a target language 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 translation probability:

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

We model this probability directly using a log-linear model:

$$Pr(e_1^I | f_1^J) = \frac{\exp\left(\sum_{m=1}^M \lambda_m h_m(e_1^I, f_1^J)\right)}{\sum\limits_{e'^{I'}_1} \exp\left(\sum_{m=1}^M \lambda_m h_m(e'^{I'}_1, f_1^J)\right)} \tag{2}$$

This equation can be considered as a generalization of the source-channel approach presented in [1]. The $h_m(\cdot)$ represent feature functions which can be bilingual, and thus represent the correspondence between source and target language, or monolingual, which represent additional features like grammaticality of the output. Typically, the features are statistical models or simple heuristics.

This approach 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 [2]. For the IWSLT evaluation campaign, we optimized the scaling factors with respect to the BLEU measure, using the Downhill Simplex algorithm from [3].

The denominator in Equation 2 represents a normalization factor that depends only on the source sentence f_1^J. Therefore, we can omit it in the search process. As a decision rule, we obtain:

$$\hat{e}_1^{\hat{I}} \quad = \quad \operatorname*{argmax}_{I, e_1^I} \left\{ \sum_{m=1}^M \lambda_m h_m(e_1^I, f_1^J) \right\} \tag{3}$$

Current state-of-the-art machine translation systems have a clearly dominating bilingual model guiding the translation process (i.e. a phrase-based model) and additional submodels. The systems developed by RWTH differ mainly in this primary model and share most of the additional submodels.

3. The SMT Models

In this section we describe the different models that contributed to the system combination. Each of the models has one or more unique feature functions that form the core of the model and direct the search. Additionally, there are feature functions, that are used by all models. These common features are the target language model, word penalty and word lexica. We will first describe the core features of all models and then briefly explain the common features.

3.1. Phrase-based Model

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:

$$k \quad \rightarrow \quad s_k := (i_k; b_k, j_k), \text{ for } k = 1 \ldots K. \tag{4}$$

Here, i_k denotes the last position of the k^{th} target phrase and 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:

$$\tilde{e}_k \quad := \quad e_{i_{k-1}+1} \ldots e_{i_k} \tag{5}$$

$$\tilde{f}_k \quad := \quad f_{b_k} \ldots f_{j_k} \tag{6}$$

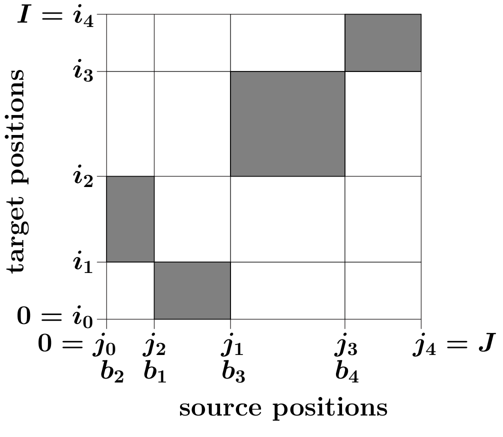

Figure 1: Illustration of the phrase segmentation.

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 $h(f_1^J, e_1^I, s_1^K)$.

The pairs of source and corresponding target phrases are extracted from the word-aligned bilingual training corpus by the phrase extraction algorithm described in [4]. The main idea is to extract phrase pairs that are consistent with the word alignment, meaning that the words of the source phrase are aligned only to words in the target phrase and vice versa.

We use relative frequencies to estimate the phrase translation probabilities:

$$p(\tilde{f}|\tilde{e}) = \frac{N(\tilde{f}, \tilde{e})}{N(\tilde{e})} \tag{7}$$

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:

$$h_{\text{Phr}}(f_1^J, e_1^I, s_1^K) = \log \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \tag{8}$$

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

Depending on the language pair, we used one of three different types of reordering models.

Jump Reordering. We use a very simple reordering model that is also used in, for instance, [5, 6]. It assigns costs based on the jump width:

$$h_{\text{RM}}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} |b_k - j_{k-1} - 1| + J - j_K \tag{9}$$

Local Reordering. For closely related languages like Italian and English reordering within a local context forms the majority of all non-monotonicity. Common example are the change of the position of a preposition or the position of the adjective with respect to the noun it refers to. For local reordering, we allow words of the source sentence to be arbitrarily reordered within a restricted window of n positions as described in [7]. At each position, we give a fixed probability to the monotone word order and distribute the remaining probability mass among the other reordering possibilities.

Syntactic Reordering for Chinese. For Chinese-to-English translation, reordering is a difficult task. Often, word order depends on the syntactic context. This is not handled well with the standard reordering approaches as presented above. Therefore we apply a rule-based reordering model at the level of syntactic chunks.

The reordering is generated by a set of rules learned from word-aligned training data. These rules are obtained by parsing the Chinese source language sentences of a bilingual training corpus and then reordering the obtained chunks to match target word order. For a test sentence to be translated, we generate every reordering that complies with the extracted rules.

Reordering alternatives are weighted using the relative frequency of the rule in the training data. Additionally, we use a source language model that was trained on the reordered Chinese training sentences for weighting the transformed source word sequence. A more detailed description of the model can be found in [8].

3.2. Hierarchical Phrase-Based Model

The hierarchical phrase-based approach can be considered as an extension of the standard phrase-based model. In this model we allow the phrases to have "gaps", i.e. we allow non-contiguous parts of the source sentence to be translated into possibly non-contiguous parts of the target sentence. The model can be formalized as a synchronous context-free grammar [9]. The bilingual rules are of the form

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle \,, \tag{10}$$

where X is a non-terminal, γ and α are strings of terminals and non-terminals, and $\sim$ is a one-to-one correspondence between the non-terminals of α and γ. Two examples of this kind of rules for the Chinese-to-English translation direction are (borrowed from [9])

$$X \rightarrow \langle \text{yu } X_{\mathbf{1}} \text{ you } X_{\mathbf{2}}, \text{have } X_{\mathbf{2}} \text{ with } X_{\mathbf{1}} \rangle \tag{11}$$

$$X \rightarrow \langle X_{\mathbf{1}} \text{ de } X_{\mathbf{2}}, \text{the } X_{\mathbf{2}} \text{ that } X_{\mathbf{1}} \rangle \tag{12}$$

where the bold subindices in the non-terminals represent the correspondence between source and target "gaps". This model has as additional advantage that reordering is integrated as part of the model itself.

The first step in the hierarchical phrase extraction is the same as for the phrased-based model presented in Section 3.1. Having a set of initial phrases, we search for phrases which contain other smaller sub-phrases and produce a new phrase with gaps. In our system, we restricted the number of non-terminals for each hierarchical phrase to a maximum of two, which were also not allowed to be adjacent.

In the original work [9], the search is organized as a parsing process, forming an extension of the CYK algorithm. This method is further augmented to include language model scores directly in the search, rather than as a preprocessing steps. Our implementation differs from this approach. We generate the target sentences in a strictly left-to-right fashion, in the spirit of [10]. In latter paper, rules are restricted to have a non-terminal symol only at the end of the rule. In our implementation we are able to handle all rules without restriction. We achieve this by transforming the target side of the grammar rules similar into a structure similar to a Greibach normal form. This allows a better integration in our existing decoder architecture (see Section 3.5) and a straightforward inclusion of language model scores into the translation process.

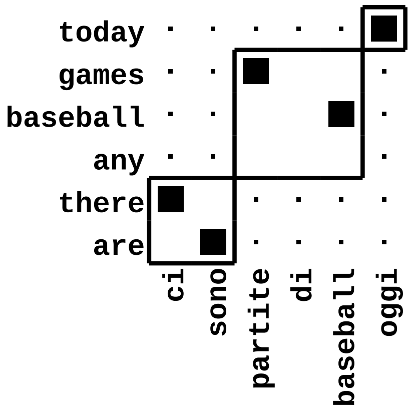

Figure 2: Example for the tuple based system. The bilingual sentence extracted is ci_sono|are_there partite_di_baseball|any_baseball_games oggi|today

3.3. Bilingual N-Gram Model

In this model, the main feature function in the log-linear model combination corresponds to the the joint probability of source and target sentence $Pr(f_1^J, e_1^I)$. Given a training sentence pair together with the corresponding alignment, we segment the source and target sentences with the same restrictions given for the phrase-based model. However in this case we try to find the smallest units such that the resulting phrase segmentation is monotonic, i.e. we only use multi-word source phrases if the alignment points cross, in a manner similar to [11]. Figure 2 shows an example of this segmentation.

Each token in the bilanguage represents the event of the source words $\tilde{f}_k$and the target words $\tilde{e}_k$being aligned in the training data. For these events, we want to model the joint probability $Pr(f_j, e_i)$. The transformation of the whole training corpus in such a way results in a *bilanguage* representation of the training corpus. On this new corpus, we apply standard language modeling techniques to train smoothed m-gram models [12].

In experimental trials a 4-gram model resulted in the best performance for most translation tasks. For better generalization we applied absolute discounting with leaving-one-out parameter estimation. Although reordering techniques can be applied for this kind of model [7], the performance of the model is normally significantly worse than the phrase-based models for language pairs with different word order. Therefore this system was only used for the Italian-to-English translation direction.

3.4. Common Models

3.4.1. *Word-based lexicon model*

The phrase translation model estimates its probabilities by relative frequencies. Most of the longer phrases or translation units however 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:

$$h_{\text{Lex}}(f_1^J, e_1^I, s_1^K) = \log \prod_{k=1}^{K} \prod_{j=b_k}^{j_k} \sum_{i=i_{k-1}+1}^{i_k} p(f_j|e_i) \qquad (13)$$

As in the phrase lexicon, 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 $p(e|f)$.

3.4.2. *Target language model*

We use the SRI language modeling toolkit [13] to train a standard n-gram language model. The resulting feature function is:

$$h_{\text{LM}}(f_1^J, e_1^I, s_1^K) = \log \prod_{i=1}^{I} p(e_i|e_{i-n+1}^{i-1}) \qquad (14)$$

The smoothing technique we apply is the modified Kneser-Ney discounting with interpolation. We used a 6-gram language model for the Chinese-to-English tasks, a 4-gram language model for the Italian-to-English task.

3.4.3. *Phrase Count Features*

The reliability of the phrase probability estimation is largely dependent on the amount and quality of the training data. Generally, the probability of rare phrases tends to be overestimated, but as they do not occur often, it might be as well errors originating from mistranslations in the training data or erroneous word alignments. Therefore, we also included features based on the actual count of the bilingual phrase pair.

$$h_{\text{C},\tau}(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} [N(\tilde{f}_k, \tilde{e}_k) \leq \tau]$$

We use $[\cdot]$ to denote a true or false statement [14], i.e., the result is 1 if the statement is true, and 0 otherwise.

The value τ determines the threshold for the phrase count feature. In the evaluation system, we used three phrase count features with τ manually chosen and ranging from 0.9 to 3.0. As actual phrase count values are fractional, fractional thresholds can also be used.

3.4.4. *Phrase penalty model*

In phrase-based MT, we usually have a large number of phrase segmentations for every source sentence. To control the number of phrases (and hence the length of the phrases, that are favored for the translation, we add a simple heuristic, the phrase penalty:

$$h_{\text{PP}}(f_1^J, e_1^I, s_1^K) \quad = \quad K \qquad (15)$$

The model scaling factors can be adjusted to prefer longer phrases. Additionally, for the hierarchical phrased-based model, having separate phrase penalties for hierarchical and normal phrases allows us to better control the contribution of each type of phrases.

3.4.5. *Word penalty*

We also use another simple heuristic, the word penalty, to control the length of the produced translation:

$$h_{\text{WP}}(f_1^J, e_1^I, s_1^K) \quad = \quad I \qquad (16)$$

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

3.5. Implementation

All models are implemented in a common software framework, called Xastur[1]. They use the same decoder and common features modules. The architecture is similar to the one presented in [15].

4. System combination

To make use of the strenghts of the different models, we generated a consensus translation out of five different MT setups using an enhanced version of the system combination approach description in [16].

For each input test sentence, the first-best hypothesis of one contributing MT system is selected as primary hypothesis, and all other n-best ("secondary"; here: $n = 10$) translations of all systems are aligned to it, allowing for word reordering. The iterative alignment procedure is based on a GIZA++ training. During the alignment step, the whole test corpus of translations is taken into account.

When the mutual word alignment of all the hypotheses for one sentence is obtained, the secondary hypotheses are then reordered to match the word order of the primary hypothesis based on the alignment. Using the monotonic alignments of secondary hypotheses with the primary one, a confusion network is constructed. Since it is not known in advance which hypothesis has the best word order, we let each hypothesis play the role of the primary translation once for each sentence, and thus construct M confusion networks

[1] **X**astur is **A** **S**tatistical **T**ranslator **U**nder **R**esearch.

(where M is the number of systems used; here $M = 5$) and unite them in a single lattice.

All arcs in the path through the confusion network representing a hypothesis of a particular MT system are weighted with a system-specific factor; the different n-best hypotheses of each systems are weighted similarly to the approach of [17]. The lattice is then rescored using a Trigram LM which is trained on the MT hypotheses. This is to give a bonus to phrases that have been hypothesized by the systems, instead of single words only. Form the resulting lattice, the best hypothesis is extracted as the result of the system combination. The factors for the individual systems, as well as a LM factor and a Word Penalty are optimized using Condor [18]. We used the the IWSLT 2005 set for the Chinese-to-English tuning and the IWSLT 2007 dev5a set for Italian-to-English.

For the Italian-to-English translation, the system combination process worked on true case input, but gave bonus to pairs of words upper case/lower case words aligned to each other. For the Chinese-to-English system combination, all input hypotheses were in lower case, and a separate true casing step was performed on the consensus translation.

5. Tasks and corpora

The experiments were carried out on the *Basic Travel Expression Corpus* (BTEC) task [19]. This is a multilingual speech corpus which contains tourism-related sentences similar to those that are found in phrase books. For the Chinese-to-English track, a 40 000 sentence pair training corpus and five test sets were made available. For the Italian-to-English track, only 20 000 sentence pairs, but 6 development sets were provided. Other resources, despite proprietary data were permitted, but were not used in this system.

6. Italian-to-English Results

For the Italian-to-English translation direction all the models described in this paper were used in the model combination.

The preprocessing of the Italian side consisted mainly in the splitting of contractions like “dell’albergo” or “un’altra” into “dell’ albergo” and “un’ altra” respectively. No corresponding transformation was used in the English side. For the phrase-based model and the hierarchical model punctuations were removed in the source side of the corpus, but not on the target side. This has shown in past evaluations to obtain the best results [20]. However the tuple model does not seem to be able to generate the correct punctuations. In this case the model was trained without punctuations in the target side, and punctuation was restored used the tools of the SRI LM toolkit [13].

The input text was lowercased, but the target text was kept in “true case”. For each word at the beginning of a sentence, the most frequent case was determined and substituted. The case for words at the beginning of sentences was then restored as a postprocessing step.

For the Italian-to-English condition 6 different development corpora were made available (numbered dev1 to dev5a and dev5b). The first 3 corpora (source plus the two longest references) were added to the training data. The dev4 and dev5b corpora were used to tune the log-linear parameters of the models, and the dev5a corpus was used to tune the parameters of the system combination. In order to translate the test data, all development data was added to the training data.

Table 1: Results for the different systems for the Italian-to-English text condition.

System	BLEU[%]	TER[%]
PBT (opt dev4, no reorder)	41.6	44.5
PBT (opt dev4, local reorder)	41.7	44.5
PBT (opt dev5b, no reorder)	42.9	43.0
PBT (opt dev5b, local reorder)	42.8	43.0
Hierarchical (opt dev5b)	42.5	43.7
Tuple (opt dev4)	33.5	50.5
System Combination	45.3	41.4

The alignment process was carried out using the GIZA++ toolkit, and the main process was common to all the models. It consisted of 4 iterations of IBM Model 1, followed of 5 iterations of the HMM Alignment model and 4 iteration of IBM Model 4. The alignment training was done in source-to-target and target-to-source directions and the resulting alignments were combined using different heuristics [21], the optimal one for each model determined using the development corpora.

Table 1 shows the systems used for this condition, 4 phrase based systems (optimized on different development corpora, with or without local reordering), a hierarchical phrase based system and a tuple model. Although this last system showed a comparable performance in the development phase, it did not generalize on the evaluation data, with nearly 9% BLEU difference absolute when compared to the best system. The system combination yields nearly 2.5% BLEU improvement over the best system.

No additional processing was made for the ASR condition. The single best output was translated with the same system as the text condition. The results, together with additional measures for the clean condition, can be found in Table 4. Actually a mistake was done in the preprocessing of the data for this condition, as some apostrophes were deleted in the Italian side of the data. Therefore the score for the ASR condition could be further improved.

7. Chinese-to-English Results

As the BTEC is a rather clean corpus, the preprocessing for Chinese-to-English consisted mainly of tokenization, i.e., separating punctuation marks from words on the English side. Additionally, we expanded contractions such as *it’s* or *I’m* in the English corpus and we removed the case information. For the Chinese source language side, we used the

Table 2: Results for the different systems for the Chinese-to-English text condition (best system).

System	BLEU[%]	TER[%]
CE-Phrase1	37.2	48.0
CE-Phrase2	36.7	48.4
CE-Phrase3	34.7	52.8
Hierarchical	33.3	51.4
CE-Phrase4	33.6	54.2
System Combination	38.5	47.2

ICTClass [22] word segmentation as sole preprocessing step.

For the word alignment we used GIZA++ and experimented with several different variants of word classes, alignment model sequences and combination heuristics. One system also used the algorithm described in [23]. All systems were optimized on the IWSLT 2004 evaluation data (dev2). We also varied the reference length method of the BLEU evaluation measure between "minimum nearest" which is the standard method used by the NIST mteval tool, and the average reference length. The IWSLT 2005 evaluation dataset (dev3) was used as blind test set.

The following systems were included in the Chinese-to-English system combination:

1. Three phrase-based systems (CE-Phrase1,2,3) using jump reordering, varying in the alignment parameters, alignment combination heuristics and optimization criterion

2. A phrase-based system using chunk-based reordering (CE-Phrase4)

3. A hierarchical phrase-based system.

The individual system performances are listed in Table 2. The hierarchical and chunk-based reordering system (CE-Phrase4) perform worse than the traditional phrase-based systems. Due to the very limited ammount of time for the Chinese-to-English track, the tuning of the different aspects of the model could not be performed to the desired extend. For both systems, no different alignment parameters or phrase segmentations were tested. On the BTEC Task, translations systems are quite sensitive to small changes in the overall training procedure. This especially holds for the syntactic models.

Truecasing was done after system combination using the SRI disambig tool with a language model trained on the supplied training data.

7.1. Progress over Time

In Table 3, we show the progress of the RWTH machine translation over the past years on the Chinese-to-English. The evaluation is done on the IWSLT 2005 test set for the supplied data track. For the 2006 system, we provide two variants. First, a system, that is only trained on 20k sentence pairs, as the systems from 2004 and 2005. Second, a system, that uses the full 40k sentence pairs that were used in the 2006 evaluation system. This makes the 2006 system comparable to the previous systems and also shows the effect of the additional data. For the 2007 system, we only report the results for the full 40k training data set. For comparison, we show two results: the best single system, the best system combination.

Table 3: Progress over time: comparison of the RWTH systems of the years 2004 to 2007 for the supplied data track on the IWSLT 2005 test set for the Chinese-to-English language pair.

System	BLEU [%]	NIST	WER [%]	PER [%]
2004	40.4	8.59	52.4	42.2
2005	46.3	8.73	47.4	39.7
2006	48.8	8.56	47.3	39.2
2006 (40k)	51.4	9.00	40.0	33.2
2007	62.4	9.64	30.7	26.0
2007 comb.	63.4	10.14	30.8	25.3

Even without the additional data, the systems improve in all scores except the NIST measure. Interestingly, using the double amount of training data only slightly improves translation quality. This can be attributed to the fact, that the coverage of the IWSLT 2004 test data is already high for the 20k sentences and the 16 references allow for a large tolerance in the MT output.

The large improvement in this year can be attributed to the extensive evaluation of different aspects of the system like like word segmentations, alignment parameters and alignment combinations. The large improvements on the development and blind test set used in the preparation seem to be due to an increasing amount of overfitting on the small and specific BTEC dataset.

8. Evaluation Results

For all the experiments, we report the two accuracy measures BLEU [24] and NIST [25] as well as the two error rates WER and PER. All those criteria are computed with respect to multiple references.

8.1. Primary submissions

The translation results of the RWTH primary submissions are summarized in Table 4. For Chinese-to-English, we also report the results of the best contrastive submissions, as it performed better than the primary submission and only differs slightly in the optimization criterion. For the primary submission we used the average sentence length as reference length for the BLEU measure, the best submission used the "minimum nearest" method, taking the length of the reference with the closest match as reference length.

9. Conclusions

We have described the RWTH machine translation system that was used in the evaluation campaign of the IWSLT 2007. It consists of a combination of different statistical machine translation systems. It was shown that the combination improved the overall system performance.

We have shown significant improvements compared to the RWTH system of 2006 [26] and have introduced new chunk-based reordering model for Chinese and a new hierarchical phrase-based system. System combination has been improved with respect to robustness against performance differences in the systems involved. Also, not only the single-best output of each system is used, but a n-best list of possible translations.

10. Acknowledgements

This paper is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023.

The authors also wish to thank Richard Zens, Daniel Stein and Jan-Thorsten Peter for their contributions and help.

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

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

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] Y. Zhang, R. Zens, and H. Ney, "Chunk-level reordering of source language sentences with automatically learned rules for statistical machine translation," in *Proceedings of SSST, NAACL-HLT 2007 / AMTA Workshop on Syntax and Structure in Statistical Translation*. Rochester, New York: Association for Computational Linguistics, April 2007, pp. 1–8.

[9] D. Chiang, "Hierarchical phrase-based translation," *Computational Linguistics*, no. 33, pp. 201–228, 2007.

[10] H. T. Taro Watanabe and H. Isozaki, "Left-to-right target generation for hierarchical phrase-based transla tion," in *COLING-ACL 2006*, Sidney, Australia, June 2006, pp. 777–784.

[11] J. Mariño, R. Banchs, J. Crego, A. de Gispert, P. Lambert, J. Fonollosa, and M. Costa-jussà, "N-gram-based Machine Translation," *Computational Linguistics*, vol. 32, no. 4, pp. 527–549, Dec 2006.

[12] F. Casacuberta and E. Vidal, "Machine translation with inferred stochastic finite-state transducers," *Computational Linguistics*, vol. 30, no. 2, pp. 205–225, 2004.

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

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

[15] A. Patry, F. Gotti, and P. Langlais, "Mood: A modular object-oriented decoder for statistical machine translation," in *5th LREC*, Genoa, Italy, May 2006, pp. 709–714.

[16] E. Matusov, N. Ueffing, and H. Ney, "Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment," in *Proceedings of EACL 2006 (11th Conference of the European Chapter of the Association for Computational Linguistics)*, Trento, Italy, April 2006, pp. 33–40.

[17] A.-V. Rosti, N. F. Ayan, B. Xiang, S. Matsoukas, R. Schwartz, and B. Dorr, "Combining outputs from multiple machine translation systems," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*. Rochester, New York: Association for Computational Linguistics, April 2007, pp. 228–235. [Online]. Available: http://www.aclweb.org/anthology/N/N07/N07-1029

[18] F. V. Berghen and H. Bersini, "CONDOR, a new parallel, constrained extension of powell's UOBYQA algorithm: Experimental results and comparison with the DFO algorithm," *Journal of Computational and Applied Mathematics*, vol. 181, pp. 157–175, 2005.

[19] 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.

Table 4: Official results for the RWTH primary submissions on the IWSLT 2007 evaluation data.

Translation Direction	Input	Accuracy Measures		Error Rates		
		BLEU [%]	NIST	TER [%]	WER [%]	PER [%]
Italian-to-English	Clean	45.3	8.21	41.4	43.1	33.9
	ASR	41.3	7.74	44.9	46.5	36.9
Chinese-to-English	Correct	37.1	6.75	50.4	51.4	45.0
(best RWTH)	Correct	38.5	6.80	47.2	47.9	43.2

[20] E. Matusov, A. Mauser, and H. Ney, "Automatic Sentence Segmentation and Punctuation Prediction for Spoken Language Translation," in *Proc. of the International Workshop on Spoken Language Translation*, Kyoto, Japan, 2006, pp. 158–165.

[21] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," in *Computational Linguistics*, 2002, to appear.

[22] H. P. Zhang, Q. Liu, X. Q. Cheng, H. Zhang, and H. K. Yu, "Chinese lexical analysis using hierarchical hidden markov model," in *Proceedings of the second SIGHAN workshop on Chinese language processing*, Morristown, NJ, July 2003, pp. 63–70.

[23] R. Zens, E. Matusov, and H. Ney, "Improved word alignment using a symmetric lexicon model," in *COLING '04: The 20th Int. Conf. on Computational Linguistics*, Geneva, Switzerland, August 2004, pp. 36–42.

[24] 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.

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

[26] A. Mauser, R. Zens, E. Matusov, S. Hasan, and H. Ney, "The RWTH Statistical Machine Translation System for the IWSLT 2006 Evaluation," in *Proc. of the International Workshop on Spoken Language Translation*, Kyoto, Japan, 2006, pp. 103–110.