# The RWTH Machine Translation System for IWSLT 2008

*David Vilar, Daniel Stein, Yuqi Zhang, Evgeny Matusov,*
*Arne Mauser, Oliver Bender, Saab Mansour and Hermann Ney*

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

## Abstract

RWTH's system for the 2008 IWSLT evaluation consists of a combination of different phrase-based and hierarchical statistical machine translation systems. We participated in the translation tasks for the Chinese-to-English and Arabic-to-English language pairs. We investigated different preprocessing techniques, reordering methods for the phrase-based system, including reordering of speech lattices, and syntax-based enhancements for the hierarchical systems. We also tried the combination of the Arabic-to-English and Chinese-to-English outputs as an additional submission.

## 1. Introduction

This year, RWTH submitted systems for the Arabic-to-English translation direction and the Chinese-to-English translation direction, for both the BTEC and challenge tasks. The submission system consists of a combination of several variations of machine translation systems, which are based on two currently widely used approaches to statistical machine translation: phrase-based and hierarchical based. We explored different reordering methods, including an efficient method for reordering speech lattices, which, however, did not get the expected improvements. Different preprocessing methods for the source languages were also investigated. An extension of the hierarchical translation model including syntax information proved to be also useful for the translation process.

RWTH's system ranked 6th in the Chinese-to-English tasks (all conditions) and third in the Arabic-to-English translation direction (all conditions). We further submitted an additional system with the combination of the best Chinese-to-English and Arabic-to-English systems, which improved the global system performance.

This paper is organized as follows: Section 2 gives a brief overview of statistical machine translation, while Section 3 discusses the baseline models we used. Section 4 presents the extensions to these models. Section 5 gives the official results obtained by RWTH in the evaluation and Section 6 concludes the paper.

## 2. Statistical Machine Translation

Following the standard formulation for statistical machine translation, we will denote the given source sentence with $f_1^J = f_1 \dots f_j \dots f_J$ and its translation with $e_1^I = e_1 \dots e_i \dots e_I$. The system chooses the translation with the highest posterior probability which gets modelled using a log-linear model:

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

The $h_m(\cdot)$ represent feature functions and the $\lambda_m$ the corresponding scaling factors. These factors are optimized using some numerical algorithm in order to maximize translation performance on a development corpus. In our case we optimize the scaling factors with respect to the BLEU measure, using the Downhill Simplex algorithm from [1].

## 3. Translation Models

In this section we will describe the baseline models we used in this year's evaluation: a phrase-based translation model and a hierarchical phrase-based model.

### 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. 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.

The pairs of source and corresponding target phrases are extracted from the word-aligned bilingual training corpus by the phrase extraction algorithm described in [2]. 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:

$$p(\tilde{f}|\tilde{e}) = \frac{N(\tilde{f}, \tilde{e})}{N(\tilde{e})} \tag{2}$$

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 in the log-linear model is:

$$h_{\text{Phr}}(f_1^J, e_1^I, s_1^K) = \log \prod_{k=1}^{K} p(\tilde{f}_k|\tilde{e}_k) \tag{3}$$

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 a different type of reordering model:

- **IBM Reordering** For the Arabic-to-English language pair a word-based reordering constrained by the IBM restrictions [3] is often enough and obtains the best results.

- **Jump Reordering** For the Chinese-to-English translation direction we use a very simple reordering model at phrase level that is also used in, for instance, [4, 5]. It assigns costs based only on the jump width.

### 3.2. Hierarchical 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 [6]. The bilingual rules are of the form

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle\,, \tag{4}$$

where $X$ is a non-terminal, $\gamma$ and $\alpha$ are strings of terminals and non-terminals, and $\sim$ is a one-to-one correspondence between the non-terminals of $\alpha$ and $\gamma$. Two examples of this kind of rules for the Chinese-to-English translation direction are

$X \rightarrow \langle$ 中 $X^{\sim 0}$ 那个 $X^{\sim 1}$, It's the $X^{\sim 1}$ in the $X^{\sim 0} \rangle$
$X \rightarrow \langle$ 也 要 $X^{\sim 0}$ 一些 $X^{\sim 1}$, like to $X^{\sim 0}$ some $X^{\sim 1}$ too$\rangle$

where the indices 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, as in the first of the examples, where the translation of the last part of the Chinese sentence (the gap $X^{\sim 1}$) gets moved to the beginning of the English sentence.

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. The scores of the phrases are again computed as relative frequencies.

### 3.3. Common Models

#### 3.3.1. Word-based Lexicon Model

The phrase translation models estimate their 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. In the case of hierarchical phrases, the gaps are simply ignored.

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 $p(e|f)$.

#### 3.3.2. Target Language Model

We use the SRI language modeling toolkit [7] to train a standard $n$-gram language model. The smoothing technique we apply is the modified Kneser-Ney discounting with interpolation. In our case we used a 6-gram language model.

#### 3.3.3. 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. See [8] for more details.

#### 3.3.4. 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), we add a simple heuristic, the phrase penalty. Additionally, for the hierarchical phrased-based model, having separate phrase penalties for paste rules and normal rules allows us to better control the contribution of each type of phrases.

#### 3.3.5. Word Penalty

We also use another simple heuristic, the word penalty, to control the length of the produced translation. These last two models affect the average sentence length. The model scaling factors can be adjusted to prefer longer sentences and longer phrases.

## 4. Extensions

### 4.1. Syntactical Features

For the hierarchical phrase-based system, we included additional features which try to capture how much a translation rule corresponds to a syntactic structure. Given the source or target part of a hierarchical rule and the parse tree of the sentence from which it was extracted, we consider the rule to be "syntactically consistent" if the original standard phrase it was extracted from and all the phrases corresponding to the gaps in the hierarchical rule correspond to the yield of nodes in the syntax tree. This can be done for the source and target parts independently. An example is shown in Figure 1. Both the original phrase "Where is the public toilet" and the sub-phrase "the public toilet" that produced a gap correspond to the yield of nodes in the syntax tree (S and NP, respectively), therefore the corresponding hierarchical rule gets a count of 1 for the syntax feature in the target part. Similarly for the source part.

These counts are added up for all occurrences of a hierarchical rule (which may be extracted from different sentences and perhaps with different syntactic properties) and normalized with the total count of the phrase. We tried different ways of smoothing the counts, for the case where the phrases do not correspond to the yield of a node completely, but a binary count seemed to work best for the IWSLT data. More details can be found in [9].

### 4.2. Chunk-based Reordering for Chinese

For the standard phrase-based model we also tried and improved reordering model based on an extended version of the method described in [10]. The Chinese input sentence is reordered by a set of syntactic chunk-level rules, which are automatically learned from the training data. The method is described in [11]. In contrast to previous work, the reordered sentences are represented as an $n$-best list instead of a lattice. The size of the $n$-best list is kept small. This method has two advantages. On the one hand, not all reorderings are translated, which improves system performance. The concept is similar to performing an aggressive pruning on the reordering lattice, where only the most promising reorderings alternatives are kept. On the other hand, there is not need for a translation system that can handle lattice based input, and thus this reordering method can be easily adapted to any translation system. An example of chunk extraction can be seen in Figure 2.

Each source reordering has a score [10], which is calculated as the product of the probability of each of the rules:

$$h_{\text{reorder}}(\pi_1^N, c_1^N) = \log(p(\pi_1^N|c_1^N)) = \sum_{k=1}^{K} \log \frac{N(\tilde{\pi}_k, \tilde{c}_k)}{N(\tilde{c}_k)} \quad (5)$$

where $c_1^N$ is the chunk sequence of the source sentence and $\pi_1^N$ represents a permutation of these chunks. The probabilities are computed as relative frequencies on the training corpus. This score is used not only to evaluate the source reorderings, but also applied to pick the best hypothesis from the multiple translations. The top $n$ reorderings are used as input to the translation system, which in term produces a list of translation alternatives. Similar to the process of $n$-best rescoring, the reordering score and the translation feature functions are used to select the best translation.

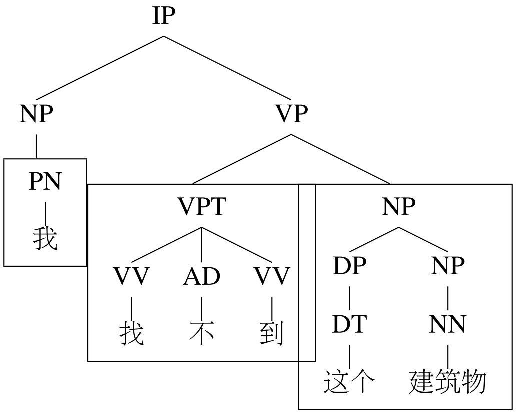

Figure 2: Example of chunk extraction.

When the word order is changed to be more similar to the target language order, the input is no longer standard Chinese. To make the training conditions more similar to the translation process, the training data is also reordered by the same set of rules. For each training sentence, the single best reordering is generated and added to the training data. The final phrase table used in the translation process is the composed from the standard phrase table expanded with the new phrases extracted from the reordered training sentences.

The training data was parsed by the tree parser from Purdue University [12], extracting the basic chunks from the tree structure. Each chunk has 1.7 words on average. The size of the source reordered $n$-best list is 5. We additionally use the jump reordering model.

### 4.3. Source Preprocessing

#### 4.3.1. Chinese

Chinese word segmentation is one of the crucial steps in the Chinese text preprocessing. We compared various segmentation methods in  [13] and found out that the unigram segmenter performs better translation results in many cases than the ictclas tool [14], which we use as baseline. Our unigram segmentation is an LDC-like segmentation without text

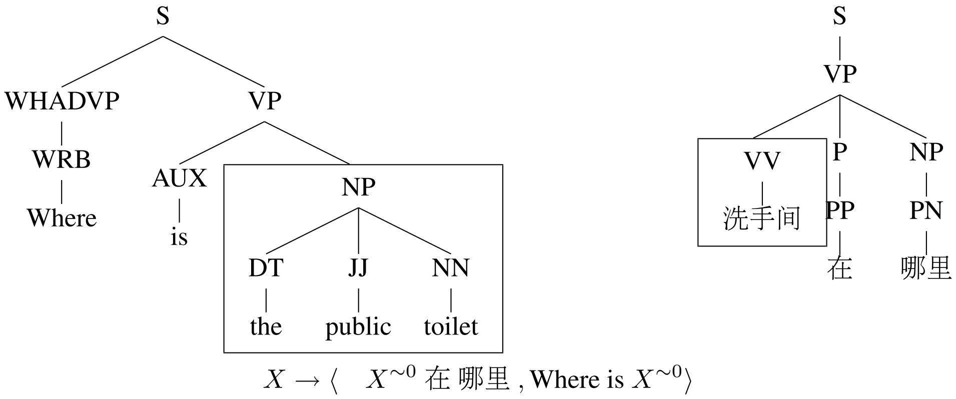
S
WHADVP
VP
WRB
Where
AUX
is
NP
DT
JJ
NN
the
public
toilet
S
VP
VV
洗手间
P
PP
在
NP
PN
哪里
$X \rightarrow \langle\ X^{\sim 0}$ 在 哪里 , Where is $X^{\sim 0}\rangle$

Figure 1: Example of a syntax-enhanced hierarchical rule.

normalization. Given a manually compiled lexicon, i.e. an LDC lexicon that contains words and their relative frequencies $P_s(f'_j)$, the best segmentation is the one that maximizes the joint probability of all words in the sentence, with the assumption that words are independent of each other:

$$f_1^J = \underset{f'^{J'}_1}{\operatorname{argmax}} Pr(f'^{J'}_1 | c_1^K) \quad (6)$$

$$\approx \underset{f'^{J'}_1}{\operatorname{argmax}} \prod_{j=1}^{J'} P_s(f'_j), \quad (7)$$

where a $c_1^K = c_1 \ldots c_K$ is a Chinese sentence in characters to be segmented into words $f_1^J = f_1 \ldots f_J$, and the maximization is taken over Chinese word sequences whose character sequence is $c_1^K$.

In order to accelerate the training process and to enhance the quality of the word alignments, we split long sentence pairs into short sub-sentence pairs using the binary segmentation described in [15] and [16]. The binary sentence segmentation method uses the lexical information to locate the optimal split point. To avoid sentence boundaries without linguistic meanings, we constrained that a split point has to be either "?" or "." in both languages. After the segmentation of long sentences we achieved an improvement of the translation performance of 0.5% in BLEU score and reduced the training time of word alignments.

### 4.3.2. Arabic

Arabic preprocessing is important for statistical machine translation systems, especially for those trained on a limited amount of data [17]. In this evaluation we tried different preprocessing techniques for Arabic.

We experimented with two different preprocessing tools, which use varying tagging levels to infer segmentation: the MADA [18] tool, a full morphological disambiguation tool, and MorphTagger [19] a POS-tagging tool. Both systems use the Buckwalter Arabic Morphological Analyzer [20] to limit the possible set of analyzes and are both trained on the Penn Arabic Treebank part 1 v3.0 [21]. Three segmentation schemes were also tested: splitting only the prefixes w+, f+, l+,k+, b+, s+ (PRE), splitting additionally the determiner Al+ (PRE+DET) and spliting the pronominal suffixes (PRE+SUF). Additionally we tested normalizing Yaa ({ي, ى} → ى) and Alef ({ا, أ, إ, آ, ٱ} → ا).

Our internal experiments show that normalization and splitting the Arabic determiner reduces the BLEU score when using MorphTagger, while for the MADA tool there was no apparent change in the translation performance. Experimental results also show that the difference between the segmentation techniques seems not to be crucial, and the extra normalization done by MADA (collapsing some verb forms and pronouns) seems the main cause for the lack of improvement when trying different schemes. Normalization usually hurts performance as it actually changes the Arabic letters and can eventually change the meaning of the sentence. For example, the Arabic word "على" means "over" but "علي" could mean the proper name "Ali". Separating the Arabic determiner also has a negative effect on translation performance. For example, in Arabic, the phrase "the handsome boy" is written as "ال+ولد ال+جميل" which corresponds to "the boy the handsome" (note the repetition of the determiner "ال" in Arabic). This causes the translation system to overgenerate articles in the English side.

The submitted system for MorphTagger uses the PRE+SUF scheme and no normalization, while for MADA we submitted the default system which uses the PRE scheme and normalization.

## 4.4. Translation of Speech Lattices

For the Chinese-to-English translation direction, in the ASR conditions we tested the extension of the phrase-based statistical machine translation system that allows for direct ASR word lattice translation presented in [22]. We used the word lattices with acoustic and source language model scores provided by the organizers. In the translation system these scores are included in the log-linear phrase-based MT model framework.

The search on text input in the phrase-based SMT system is cardinality-synchronous, i.e. at each step, translation hypotheses with the same number of covered words are expanded. To use this type of search for word lattices, we define the cardinality in terms of slots, i.e. word hypotheses which would be merged to one time slot in a confusion network (CN). However, we do not explicitly construct a CN, but label the arcs of the original word lattice with slot information. Thus, in contrast to a CN, only the true search space of the original lattice is explored in the search. By using this type of search, we can efficiently perform reordering for lattices and yet can avoid the theoretical drawbacks of the confusion networks.

In [22], the translation system was trained using character-based segmentation of Chinese, in order to avoid any mismatch between the ASR and MT Chinese vocabularies. This approach, however, is inferior to an explicit Chinese word segmentation (ictclas) in terms of MT quality. Therefore, we introduce a solution for mapping the ASR vocabulary to MT vocabulary. First, we transform the ASR word lattices to character lattices. The slots for translation are defined in terms of characters (the duration of each character from a word with $m$ characters is assumed to be $1/m$ of the word's duration). Then, each character lattice is composed with a character-to-word mapping transducer. This transducer maps Chinese character sequences to words from the MT vocabulary and includes alternative mappings (e.g. the identity mapping of each character). The lattice size increases after this mapping, but this increase is moderate and still allows for efficient translation. The cardinality of the covered slots in the search is counted based on characters, with the extension that one word can cover several slots at once. The idea to use a mapping transducer for a single sentence to represent alternative Chinese word segmentations was introduced in [23]. Here we successfully applied a similar transducer to ASR word lattices.

Despite previous favorable results on other tasks, we were not able to obtain significant improvements over the single-best ASR translation on the development data. We attribute this to the fact that the translation model trained only on 20K sentence pairs was to weak to differentiate between good and bad ASR hypotheses. The expansion of the lattices using alternative word segmentations introduces additional ambiguity: so far we have not used probabilities for the segmentation alternatives. Nevertheless, examples show that in many cases the ASR errors can be avoided when word lattices are translated (see Table 1).

Table 1: Examples of improved speech translation quality when ASR word lattices are used as input for translation.

| | |
|---|---|
| single-best | Hurry up. Can you. Some? |
| lattice | It is too expensive. Can you make it cheaper? |
| reference | Too expensive. Can you make it cheaper? |
| single-best | Where is the bus stop? |
| lattice | The bus stop here, please. |
| reference | Here is the bus stop. |
| single-best | How long time? |
| lattice | How long will it take to get there? |
| reference | How much time will it take? |

### 4.5. System Combination

For system combination we used our approach from last year's evaluation campaign [8], which is based on an enhanced version of the system combination approach described in [24]. The method is based on the generation of a consensus translation out of the output of different translation systems.

The core of the method consists in building a confusion network for each sentence by aligning and combining the (single-best) translation hypothesis from one MT system with the translations produced by the other MT systems (and the other translations from the same system, if $n$-best lists are used in combination). For each sentence, each MT system is selected once as "primary" system, and the other hypotheses are aligned to this hypothesis. The resulting confusing networks are combined into one word graph, which is then weighted with system-specific factors, similar to the approach of [25], and a trigram LM trained on the MT hypotheses. The translation with the best total score within this word graph is selected as consensus translation. The scaling factors of these models are optimized using the Condor toolkit [26] to achieve optimal BLEU score on the dev set.

## 5. Experimental Results

In this year's evaluation RWTH participated in the Arabic-to-English and Chinese-to-English translation directions. For this last language pair, we participated in both evaluation tasks, BTEC and Challenge. As training data we used the provided training data and additionally a part of the HIT-corpus[1] for the Chinese-to-English translation direction. For this last data we selected those sentences from which 60% of the words were also present in the IWSLT data. In this way we hope to select those sentences of the corpus that are more related to the IWSLT task.

Preprocessing of the English part was common for both language pairs and consisted basically in tokenization (separation of punctuation marks) and expansion of contractions like "it's" or "I'm" present in the training data, which were then redone after the translation process.

For the source side we applied the ictclas word segmenter for Chinese [14] and the Mada toolkit [18] for the Arabic part as baseline. Additionally, we tested the methods described in

[1] http://mitlab.hit.edu.cn/

Section 4.3.

For the word alignment we used GIZA++ and experimented with several different variants of word classes, alignment model sequences and combination heuristics. All systems were optimized on the IWSLT 2004 evaluation data. The IWSLT 2005 evaluation dataset was used as blind test set and as development set to optimize the system combination weights. All systems were optimized for the BLEU score.

### 5.1. Chinese-to-English

For the Chinese-to-English translation direction, following systems participated in the system combination:

1. A "baseline" phrase-based system as described in Section 3.1, without any enhancement of Section 4.
2. A phrase-based model with chunk level reordering, as presented in Section 4.2.
3. A phrase-based system with the enhanced preprocessing of Section 4.3.1 (only for the text condition due to time constraints).
4. A hierarchical system as presented in Section 3.2.
5. A hierarchical system with syntax enhancements, as explained in Section 4.1.

#### *5.1.1. BTEC Task*

The results for the BTEC task can be seen in Table 2. For the "correct recognition result" (CRR) condition, the best individual system in terms of BLEU score is the phrase based system (PBT) using the word segmentation discussed in Section 4.3.1, in terms of TER, however the baseline phrase-based system performs better. The chunk reordering approach does not have a big impact in the system performance. The hierarchical system is somewhat worse that the phrase based ones and the syntax enhancement help slightly in terms ob BLEU score, but have a greater impact on TER. The System combination improves in terms of BLEU over the best system, but the baseline PBT system still performs better in terms of TER.

For the ASR condition the performance of all systems deteriorate, as expected. For this condition the chunk reordering obtains an improvement on BLEU score over the baseline PBT system, but again, not on the TER score. Concerning the hierarchical system, the syntax information seems to be much more important for this condition. The system combination again achieves an improvement in BLEU score over the best system, but not for TER. The results for the lattice based approach are also present in the table, they were however not included in the system combination. It can be seen that they do not manage to improve the performance of the baseline PBT system on which they are based (see Section 4.4).

Table 2: Results for the Chinese-to-English BTEC Task

| CRR | | | | |
|---|---|---|---|---|
| System | BLEU | TER | WER | PER |
| System Combination | 46.1 | 37.7 | 43.9 | 39.4 |
| Phrase Based (PBT) | 42.5 | 36.6 | 45.3 | 40.6 |
| PBT + Chunk Reordering | 42.6 | 39.9 | 47.8 | 42.4 |
| PBT + New Segmentation | 44.3 | 40.3 | 47.3 | 42.0 |
| Hierarchical | 41.2 | 41.5 | 48.1 | 42.7 |
| Hierarchical + Syntax | 41.4 | 40.6 | 47.3 | 42.8 |

| ASR | | | | |
|---|---|---|---|---|
| System | BLEU | TER | WER | PER |
| System Combination | 39.7 | 42.5 | 49.6 | 44.5 |
| Phrase Based (PBT) | 37.3 | 41.2 | 50.0 | 45.1 |
| PBT + Chunk Reordering | 38.5 | 42.8 | 51.2 | 46.4 |
| Hierarchical | 31.6 | 49.6 | 56.5 | 49.5 |
| Hierarchical + Syntax | 36.6 | 44.1 | 51.4 | 47.0 |
| Lattices | 32.2 | 48.6 | 57.1 | 51.5 |

#### *5.1.2. Challenge Task*

For the challenge task we used basically the same systems as for the BTEC task. The results can be found in Table 3. For the text condition the new word segmentation obtains a big improvement over the baseline system in terms of BLEU score, however it lags somewhat behind in terms of TER score. The chunk reordering approach helps a bit in terms of BLEU score but not in TER. The syntax approach decreases the BLEU score of the baseline hierarchical system but gets an improvement in TER. The system combination achieves the best scores both in terms of BLEU score and TER score for this evaluation condition.

For the ASR condition the hierarchical system enhanced with syntax information obtains the best improvements for a single system, both in terms of BLEU score and TER. The chunk reordering also obtains improvements for both measures when compared to the baseline PBT system. In this case, the system combination again obtains the best results, as in the text condition.

### 5.2. Arabic-to-English

For the Arabic-to-English task four systems participated in the system combination:

1. Phrase-based system with MADA preprocessing
2. Phrase-based system with MorphTagger preprocessing
3. Hierarchical system with MADA preprocessing
4. Hierarchical system with MorphTagger preprocessing

No syntax trees were available for the Arabic side. Mainly due to lack of time, we did not test the hierarchical system with syntax information in the target side only.

Table 3: Results for the Chinese-to-English Challenge Task

| CRR | | | | |
|---|---|---|---|---|
| System | BLEU | TER | WER | PER |
| System Combination | 39.1 | 40.7 | 48.3 | 44.1 |
| Phrase Based (PBT) | 32.1 | 42.7 | 51.9 | 47.8 |
| PBT + Chunk Reordering | 32.6 | 43.6 | 52.5 | 48.5 |
| PBT + New Segmentation | 37.2 | 41.8 | 49.3 | 44.5 |
| Hierarchical | 30.7 | 47.1 | 54.6 | 48.9 |
| Hierarchical + Syntax | 30.2 | 45.5 | 53.6 | 48.5 |
| ASR | | | | |
| System | BLEU | TER | WER | PER |
| System Combination | 34.3 | 43.6 | 51.1 | 46.1 |
| Phrase Based (PBT) | 27.8 | 46.0 | 55.4 | 51.1 |
| PBT + Chunk Reordering | 29.4 | 45.7 | 55.0 | 50.5 |
| Hierarchical | 26.4 | 51.0 | 59.2 | 51.9 |
| Hierarchical + Syntax | 30.2 | 45.6 | 53.7 | 48.6 |
| Lattices | 25.0 | 56.6 | 62.8 | 56.7 |

It can be seen that the MorphTagger tool obtains consistently better results than the MADA preprocessing in terms of BLEU for the PBT system. The effect on TER is however very little. The opposite is true for the Hierarchical system, however, where the MorphTagger reduces TER, but gets no improvement for the BLEU score. The PBT system outperforms the hierarchical system in this condition. The same conclusions apply both for the text condition and for the ASR condition.

### 5.3. Arabic&Chinese-to-English

We submitted an additional contrastive submission, where we combined the best performing systems for both language pairs. This was possible for the BTEC task, as the texts to translate were, at the same time, translations of each other. The results can be found in Table 5. It can be seen that the translation performance increases in all measures. This suggests that the translation task for each language pair encounters different difficulties, and the combination of both can improve system performance.

## 6. Conclusions

We have presented the RWTH system for the 2008 IWSLT evaluation. The system is a combination of different statistical machine translation approaches. A phrase-based system and a hierarchical system are taken as baseline models and different extensions including improved reordering models and syntax extensions are investigated. The results on the Chinese-to-English and Arabic-to-English task have been reported. Furthermore, the combination of the best systems for each language pair increases the translation performance.

Table 4: Results for the Arabic-to-English BTEC Task

| CRR | | | | |
|---|---|---|---|---|
| System | BLEU | TER | WER | PER |
| System Combination | 53.5 | 33.0 | 37.6 | 33.9 |
| PBT + MADA | 50.0 | 33.7 | 39.7 | 36.0 |
| PBT + MorphTagger | 51.8 | 33.8 | 38.1 | 33.9 |
| Hierarchical + MADA | 49.2 | 36.6 | 41.3 | 36.7 |
| Hierarchical + MorphTagger | 49.3 | 35.9 | 41.3 | 38.0 |
| ASR | | | | |
| System | BLEU | TER | WER | PER |
| System Combination | 44.5 | 37.6 | 43.4 | 39.9 |
| PBT + MADA | 42.6 | 38.2 | 45.3 | 41.7 |
| PBT + MorphTagger | 44.0 | 38.0 | 43.4 | 39.4 |
| Hierarchical + MADA | 41.3 | 42.1 | 47.7 | 42.7 |
| Hierarchical + MorphTagger | 41.3 | 40.7 | 47.2 | 43.9 |

Table 5: Results for the Arabic&Chinese-to-English BTEC Task

| CRR | | | | |
|---|---|---|---|---|
| System | BLEU | TER | WER | PER |
| System Combination | 56.2 | 31.7 | 36.0 | 32.6 |

## 7. Acknowledgements

The authors wish to thank Niklas Hoppe and Juri Ganitkevitch for their help with the systems.

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA).

## 8. References

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

[2] 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.

[3] A. L. Berger, P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, J. R. Gillett, A. S. Kehler, and R. L. Mercer, “Language Translation apparatus and method of using context-based translation models, United States Patent 5510981,” April 1996.

[4] 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.

[5] 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.

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

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

[8] A. Mauser, D. Vilar, G. Leusch, Y. Zhang, and H. Ney, "The RWTH Machine Translation System for IWSLT 2007," in *International Workshop on Spoken Language Translation*, Trento, Italy, Oct. 2007, pp. 161–168.

[9] D. Vilar, D. Stein, and H. Ney, "Analysing Soft Syntax Features and Heuristics for Hierarchical Phrase Based Machine Translation," in *Proceedings of the IWSLT 2008*, Honolulu, Hawaii, Oct 2008.

[10] Y. Zhang, R. Zens, and H. Ney, "Improved Chunk-level Reordering for Statistical Machine Translation," in *International Workshop on Spoken Language Translation*, Trento, Italy, 2007.

[11] ——, "Chunk-level reordering of source language sentences with automatically learned rules for statistical machine translation," in *Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics Annu al Meeting*, Rochester, NY, 2007.

[12] Z. Huang, D. Filimonov, and M. Harper, "Accuracy Enhancements for Mandarin Parsing," University of Maryland, Tech. Rep., 2008.

[13] J. Xu, J. Gao, K. Toutanova, and H. Ney, "Bayesian Semi-Supervised Chinese Word Segmentation for Statistical Machine Translation," in *Proceedings of the 22nd International Conference on Computational Linguistics*, Manchester, UK, August 2008.

[14] H.-P. Zhang, H.-K. Yu, D.-Y. Xiong, and Q. Liu, "HHMM-based Chinese lexical analyzer ICTCLAS," in *Proceedings of the second SIGHAN workshop on Chinese language processing*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 184–187.

[15] J. Xu, R. Zens, and H. Ney, "Sentence segmentation using IBM word alignment model 1," in *Proceedings of EAMT 2005 (10th Annual Conference of the European Association for Machine Translation)*, Budapest, Hungary, May 2005, pp. 280–287.

[16] ——, "Partitioning Parallel Documents Using Binary Segmentation," in *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL): Proceedings of the Workshop on Statistical Machine Translation*, New York City, NY, June 2006, pp. 78–85.

[17] F. Sadat and N. Habash, "Combination of Arabic preprocessing schemes for statistical machine translation," in *ACL '06: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*. Morristown, NJ, USA: Association for Computational Linguistics, 2006, pp. 1–8.

[18] N. Habash and O. Rambow, "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop," in *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2005, pp. 573–580.

[19] S. Mansour, K. Sima'an, and Y. Winter, "Smoothing a Lexicon-based POS Tagger for Arabic and Hebrew," in *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 97–103.

[20] T. Buckwalter, "Arabic Morphological Analyzer Version 1.0," University of Pennsylvania, 2002.

[21] M. Maamouri, A. Bies, T. Buckwalter, and W. Mekki, "The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus," 2004.

[22] E. Matusov, B. Hoffmeister, and H. Ney, "ASR Word Lattice Translation with Exhaustive Reordering is Possible," in *To appear in Interspeech 2008*, Sydney, Australia, Sept. 2008.

[23] J. Xu, E. Matusov, R. Zens, and H. Ney, "Integrated Chinese Word Segmentation in Statistical Machine Translation," in *International Workshop on Spoken Language Translation*, Pittsburgh, PA, USA, Oct. 2005, pp. 141–147.

[24] 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.

[25] 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.

[26] 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.