The CMU Statistical Machine Translation System for IWSLT 2005

Sanjika Hewavitharana, Bing Zhao, Almut Silja Hildebrand, Matthias Eck,
Chiori Hori, Stephan Vogel and Alex Waibel

Interactive Systems Laboratories, Language Technologies Institute
Carnegie Mellon University, Pittsburgh, USA
[sanjika, bzhao, silja, matteck, chiori, vogel+, ahw]@cs.cmu.edu

Abstract

In this paper we describe the CMU statistical machine translation system used in the IWSLT 2005 evaluation campaign. This system is based on phrase-to-phrase translations extracted from a bilingual corpus. We experimented with two different phrase extraction methods; PESA on-the-fly phrase extraction and alignment free extraction method. The translation model, language model and other features were combined in a log-linear model during decoding. We present our experiments on model adaptation for new data in a different domain, as well as combining different translation hypotheses to obtain better translations.

We participated in the supplied data track for manual transcriptions in the translation directions: Arabic-English, Chinese-English, Japanese-English and Korean-English. For Chinese-English direction we also worked on ASR output of the supplied data, and with additional data in unrestricted and C-STAR tracks.

1. Introduction

Large vocabulary text translation has been the primary focus in machine translation research during the past. Much improvements have been achieved with projects such as TIDES, which focused on large vocabulary text translation. With the availability of reliable speech recognition systems and spoken language corpora, now the focus is shifting towards speech translation; and further towards speech-to-speech translation.

With the IBM system [1] in early 90's, statistical machine translation (SMT) has been the most promising approach for machine translation. Many approaches for SMT have been proposed since then [2], [3], [4]. Whereas the original IBM system was based on purely word translation models, current SMT systems incorporate more sophisticated models.

The CMU statistical machine translation system uses phrase-to-phrase translations as the primary building blocks to capture local context information, leading to better lexical choice and more reliable local reordering. In section 2, we describe the phrase alignment approaches used by our system.

The main obstacle in using additional data for a translation task is that the new data may belong to a different domain. We explored methods of adapting both the translation model and the language model to overcome this problem, which are described in section 3.

Section 4 outlines the architecture of the decoder that combines the translation model, language model, and other models to generate the complete translation.

When translating speech recognition output, we integrate multiple translation hypotheses into a single structure and then derive the best hypothesis. This approach is described in section 5.

Finally, in section 6 we give an overview of the data and tasks and present the results of the experiments we carried out for different data conditions.

2. Phrase Alignment

In this evaluation, we applied a variation of the alignment-free approach, which is an extension to the previous work in [5] and [6] to extract bilingual phrase pairs for the supplied data tracks. In this extension, we used eleven feature functions including phrase level fertilities and phrase level IBM Model-1 probabilities aiming to locate the phrase pairs from the parallel sentences. The feature functions are then combined in a log-linear model as follows:

$$P(X|\mathbf{e},\mathbf{f}) = \frac{\exp(\sum_{m=1}^{M}\lambda_m\phi_m(X,\mathbf{e},\mathbf{f}))}{\sum_{\{X'\}}\exp(\sum_{m=1}^{M}\lambda_m\phi_m(X',\mathbf{e},\mathbf{f}))}$$

where $X \rightarrow (f_j^{j+l}, e_i^{i+k})$ corresponds to a phrase-pair candidate extracted from a given sentence-pair $(\mathbf{e},\mathbf{f})$; ϕ_m is a feature function designed to be informative for phrase extraction. Feature function weights $\{\lambda_m\}$, are the same as in our previous experiments [7]. This log-linear model serves as a performance measure function in a local search. The search starts from fetching a test-set specific source phrase (e.g. Chinese ngram); it localizes the candidate ngram's center in the English sentence; and then around the projected center, it finds out all the candidate phrase pairs ranked with the log-linear model scores. In the local search, down-hill moves are allowed

so that functional words can be attached to the left or right boundaries of the candidate phrase-pairs.

The *eleven* (M=11) feature functions that compute different aspects of phrase pair (f_j^{j+l}, e_i^{i+k}) are as follows:

- Four of them compute the phrase-level length relevance: $P(l{+}1|e_i^{i+k})$ and $P(J{-}l{-}1|e_{i' \notin [i,i+k]})$, where $e_{i' \notin [i,i+k]}$ is denoted as the remaining English words in $\mathbf{e}$: $e_{i' \notin [i,i+k]}{=}\{e_{i'}|i' \notin [i, i{+}k]\}$, and J is the length of $\mathbf{f}$. The probability is computed via dynamic programming using English word-fertility table $P(\phi|e_i)$. $P(k{+}1|f_j^{j+l})$ and $P(I{-}k{-}1|f_{j' \notin [j,j+l]})$ are computed in a similar way.

- Another four compute the IBM Model-1 scores for the phrase-pairs $P(f_j^{j+l}|e_i^{i+k})$ and $P(e_i^{i+k}|f_j^{j+l})$; the remaining parts of $(\mathbf{e}, \mathbf{f})$ excluding the phrase-pair is modeled by $P(f_{j' \notin [j,j+l]}|e_{i' \notin [i,i+k]})$ and $P(e_{i' \notin [i,i+k]}|f_{j' \notin [j,j+l]})$ using the translation lexicons of $P(f|e)$ and $P(e|f)$.

- Another two of the scores aim to bracket the sentence pair with the phrase-pair as detailed in [7].

- The last function computes the average word alignment links per source word in the candidate phrase-pair.

We assume each phrase-pair should contain at least one word alignment link. We train the IBM Model-4 with GIZA++ [8] in both directions and grow the intersection with word pairs in the union to collect the word alignment. Because of the last feature-function, our approach is no longer truly "alignment-free". More details of the log-linear model and experimental analysis of the feature-functions are given in [7].

To use the extracted phrase-pairs in the decoder, a set of eight scores for each phrase-pair are computed: relative frequency of both directions, phrase-level fertility scores for both directions computed via dynamic programming, the standard IBM Model-1 scores for both directions (i.e. $P(f_j^{j+l}|e_i^{i+k}) = \prod_{j' \in [j,j+l]} \sum_{i' \in [i,i+k]} P(f_{j'}|e_{i'})/(k{+}1)$), and the un-normalized IBM Model-1 scores for both direction (i.e. $P(f_j^{j+l}|e_i^{i+k}) = \prod_{j' \in [j,j+l]} \sum_{i' \in [i,i+k]} P(f_{j'}|e_{i'})$). The standard IBM Model-1 scores prefer short translations; the un-normalized scores prefer longer translations. The scores are combined via the optimization component of the decoder (e.g. Max-BLUE optimization) as described in section 4 in the hope of balancing the sentence length penalty.

3. Model Adaptation

The Unrestricted Data track allows the use of additional publicly available data for both translation and language models. This mainly includes data from LDC and data that is available on the Web.

The main problem with additional data is that it usually is from a different domain compared to the original data. Using this data as is, along with the supplied data hurts the performance on the development sets. Therefore, we used a translation model adaptation approach to handle this problem.

3.1. Translation Model Adaptation

We adapt the translation model to the test set by selecting a part of the additional out-of-domain data using information retrieval techniques as explained in [9].

For every source language sentence from the test set or the development set, the most similar sentences from the out-of-domain bilingual data are selected using cosine distance with TF-IDF term weights as the similarity measure. The retrieval is done on the source language side with each test sentence as a query, then the information is used to extract respective sentence pair from the bilingual corpus.

The selected sentences from the out-of-domain data together with the supplied in-domain data are used to train the translation model for the whole test set.

3.2. Language Model Perplexity for Measuring Selection Quality

An important question when selecting additional sentences is how much out-of-domain data should be added to the training corpus. Here, we used a perplexity based re-ranking method [9].

The top 1000 retrieved sentences in the source language (which is much more than the optimal number) are split into small batches of 3-10 sentences which are sequentially added to the selection. To determine how well the selection of training data fits the test sentence, we measure the perplexity of a language model trained from each selection against the respective test sentence. Each batch is classified according to whether it decreases (good batch) or increases (bad batch) the perplexity.

The batches are re-ranked using this information by putting bad batches at the end of the sorted order of sentences. After re-ranking, those sentences that are in the range of twice the lowest perplexity value are included in the final training corpus.

Still, the main selection criterion is TF-IDF information retrieval, as we look only at the e.g. top 1000 sentences returned by the retrieval and the original TF-IDF ranking is kept among the good as well as the bad batches.

This method allows to determine the size of the selection without using a development set and shows improvements over the standard method of just choosing the same number for each test sentence.

3.3. Data Weights

To balance the different sizes of the in-domain and out-of-domain training corpora we assigned a stronger weight to the in-domain data. We experimented with different weight combinations. A rule of thumb for the weight *w* for the in-domain data is as in (1) :

$$w = \left[\frac{\#lines \quad out-of-domain}{\#lines \quad in-domain} \right] \qquad (1)$$

3.4. Language Model Adaptation

We also applied a basic form of language model (domain) adaptation using additional data crawled from the Web. Based on the English in-domain supplied training data the 5000 most common 3-grams and 4-grams were used as queries for the Google Web search engine. After filtering and basic cleaning of the retrieved web pages this data can be added to the Language Model training data.

4. Decoder

The decoder combines all knowledge sources, i.e. translation model, language model, etc. to find the best translation. In the CMU SMT decoder the decoding process is organized into two states:

- Find all available word and phrase translations. These are inserted into a lattice structure, called translation lattice.
- Find the best combinations of these partial translations, such that every word in the source sentence is covered exactly once. This amounts to doing a best path search through the translation lattice, which is extended to allow for word reordering.

In addition, the system needs to be optimized. For each model used in the decoder a scaling factor can be used to modify the contribution of this model to the overall score. Varying this scaling factors can change the performance of the system considerable. Minimum error training is used to find a good set of scaling factors.

In the following sub-sections, these different steps will be described in some more detail.

4.1. Building Translation Lattice

The CMU SMT decoder can use phrase tables, generated at training time, but can also do just-in-time phrase alignment. This means that the entire bilingual corpus is loaded and the source side indexed using a suffix array [10]. For all ngrams in the test sentence, occurrences in the corpus are located using the suffix array. For a number of occurrences, where the number can be given as a parameter to the decoder, phrase alignment as described in section 2 is performed and the found target phrase added to the translation lattice.

If phrase translations have already been collected during training time, then this phrase table is loaded into the decoder and a prefix tree constructed over the source phrases. This allows for an efficient search to find all source phrases in the phrase table which match a sequence of words in the test sentence. If a source phrase is found in the phrase translation table then a new edge is added to the translation lattice for each translation associated with the source phrase.

Each edge carries not only the target phrase, but also a number of model scores. There can be several phrase translation model scores, calculated from relative frequency, word lexicon and word fertility. In addition, the sentence stretch model score and the phrase length model score are applied at this stage.

4.2. Searching for Best Path

The second stage in the decoding is finding a best path through the translation lattice, now also applying the language model. To allow for word reordering, the search algorithm is extended.

Hypotheses describe partial translations, i.e. a sequence of target language words, which are translations of some of the source words, and a score. As we use a trigram language model, we need to store only the last two words. A hypothesis can be expanded to cover additional source words. To restrict the search space only limited word reordering is done. Essentially, decoding runs from left to right over the source sentence, but words can be skipped within a restricted reordering window and translated later. In other words, the difference between the highest index of already translated words and the index of still untranslated words is smaller than a specified constant, which typically is 4.

When a hypothesis is expanded, the language model is applied to all target words attached to the edge over which the hypothesis is expanded. In addition, the distortion model is applied, adding a cost depending on the distance of the jump made in the source sentence.

Hypotheses are recombined whenever the models can not change the ranking of alternative hypotheses in the future. For example, when using a trigram language model, two hypotheses having the same two words at the end of the word sequences generated so far, will get the same increment in language model scores when expanded with an additional word. Therefore, only the better hypothesis needs to be expanded. The translation model and distortion model require that only the hypotheses which cover the same source words are compared.

As typically too many hypotheses are generated, pruning is necessary. This means that coarser equivalence classes are used to compare hypotheses, but also to keep not only one hypothesis in one equivalence class, as done in recombination, but to keep all hypotheses, which are close to the best one. Pruning can be done with more

equivalence classes and smaller beam, or coarser equivalence classes and wider beams. For example, comparing all hypotheses, which have translated the same number of source words, no matter what the final two words are, would be working with a small number of equivalence classes in pruning. The CMU SMT decoder allows two different recombination and pruning settings.

4.3. Optimizing the System

Each model contributes to the total score of the translation hypotheses. As these models are only approximations to the real phenomena they are supposed to describe, and as they are trained on varying, but always limited data, their reliability is restricted. However, the reliability of one model might be higher than the reliability of another model. So, we should put more weight on this model in the overall decision. This can be done by doing a log-linear combination of the models. In other words, each model score is weighted and we have to find an optimal set of these weights or scaling factors. When dealing with two or three models, grid search is still feasible. When adding more and more features (models) this no longer is the case and automatic optimization needs to be done. We use the Minimum Error Training as described in [11], which uses rescoring of the n-best list to find the scaling factors with maximize BLEU or NIST score.

Starting with some reasonably chosen model weights a first decoding for some development test set is done. An n-best list is generated, typically a 1000-best list. Then a multi-linear search is performed, for each model weight in turn. The weight, for which the change gives the best improvement in the MT evaluation metric, is then fixed to the new value, and the search repeated, till no further improvement is possible.

The optimization is therefore based on an n-best list, which resulted from sub-optimal model weights, and contained only a limited number of alternative translations. To eliminate any restricting effect, a new full translation is done with the new model weights. The resulting new n-best list is then merged to the old n-best list, and the entire optimization process repeated. Typically, after three iterations of doing translation plus optimization, translation quality, as measured by the MT evaluation metric, converges.

5. ROVER on SMT n-best Hypotheses

To improve the translation accuracy of the ASR output, we integrate multiple translation hypotheses and select the best translation. Multiple translations can be obtained either by translating each of the n-best hypotheses produced by a speech recognizer, or selecting the n-best translations by a machine translation system.

5.1. ROVER

The ROVER approach is useful for integrating multiple word sequences [12]. The word sequences can be integrated based on the edit distance between the sequences, and then represented as a word transition network (WTN) which has the same structure as a confusion network (CN). A WTN differs from CN in that the score of each arc is determined based on the occurrences of words aligned to the same position in the WTN unlike posterior probabilities in CN obtained by speech recognition. Figure 1 shows an example of a word transition network. The integrated multiple word sequence begins with $<s>$ and ends with $</s>$. In each column, words aligned to the same position are included. The symbol "@" is a special word indicating the possibility of deletion.

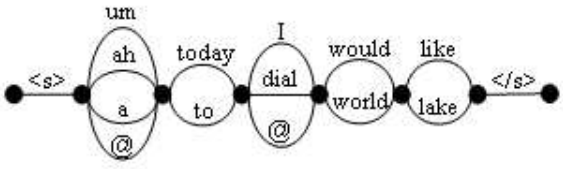

Figure 1: Word Transition Network.

To select the best translation from a WTN, we consider two methods. Given a WTN, one method is to simply choose the best scored word sequence $\hat{W}$ such that:

$$\hat{W} = \underset{W \in WTN}{\arg\max} \sum_{n=1}^{|W|} P_{WTN}(w_n) \tag{2}$$

where $P_{WTN}(w_n)$ is a score of w_n in the WTN that can be calculated as the proportion of the number of occurrences of w_n to the sum of occurrences of words in the same column; $|W|$ is the length of the word sequence *W*.

5.2. ROVER combined with Language model

When ROVER system is combined with a language model, it helps to increase the recognition performances considerably for multiple ASR system outputs [13]. We search for the best sequence using both the score of each arc and probabilities given by a language model of the target language such that:

$$\hat{W} = \underset{W \in WTN}{\arg\max} \prod_{n=1}^{|W|} P_{WTN}(w_n) P_{LM}(w_n | w_{n-2} w_{n-1})^{\lambda} \tag{3}$$

where $P_{LM}(w_n|w_{n-2}w_{n-1})$ is the language model score given by a trigram language model; λ is the scaling factor for the language model. By using a language model, the selected word sequence is expected to be fluent and grammatically correct. The best word sequence can easily be found by using a dynamic programming technique.

Table 1: *Corpus statistics for the supplied data.*

		Supplied Data Track					
		Arabic	Chinese		Japanese	Korean	English
			Manual	ASR			
Training	Sentences	20,000					
	Words	131,711	176,199		198,453	208,763	183,452
	Vocabulary	26,116	8,687		9,277	9,132	6,956
C-STAR'03	Sentences	506					
	Words	2,579	3,511	2,835	4,130	4,084	-
	Vocabulary	1,322	913	1,024	920	976	-
	Unknown Words	441	117	245	70	95	-
IWSLT'04	Sentences	500					
	Words	2,712	3,590	2,896	4,131	-	-
	Vocabulary	1,399	975	1,068	945	-	-
	Unknown Words	484	116	223	61	-	-
IWSLT'05	Sentences	506					
	Words	2,607	3,743	3,003	4,226	4,563	-
	Vocabulary	1,387	963	1,091	975	969	-
	Unknown Words	468	155	249	169	84	-

5.3. Consolidation on ROVER

In consolidation, removing recognition errors, retaining as much information of the original sentence as possible and reconstructing a fluent sentence are important factors. We define the consolidation score as:

$$S(V) = \sum_{m=1}^{M} \{\lambda_L L(v_m|v_1...v_{m-1}) + \lambda_C C(v_m) + sp.d(v_{m-1}, v_m) + ip\} \quad (4)$$

where sp is a skip penalty ($sp < 0$); $d(v_{m-1}, v_m)$ is the number of skipped words between v_{m-1} and v_m; ip is an insertion penalty [14]. The skip penalty is incorporated to avoid high compression of the original sentence because high compression of a sentence often alters the meaning of the sentence. The insertion penalty is used to control the overall compression ratio.

6. Evaluation

The evaluations were primarily based on the Basic Travel Expression Corpus (BTEC) which contains conversations in tourism-related activities. The corpus was originally created in Japanese and English by ATR [15] and was later extended to other languages.

We participated in the supplied data track for the translation directions Arabic-English, Chinese-English, Japanese-English and Korean-English. For Chinese-English direction we also worked on ASR output. In both unrestricted and C-STAR tracks, we participated for Chinese-English direction.

For each translation direction, except Korean-English, two development sets (C-STAR'03 and IWSLT'04) were made available. For Korean-English only C-STAR'03 test set was available. Table 1 shows corpus statistics for the training and test sets.

As a preprocessing step, we separated punctuations from words in the English (target) side and converted the text into lowercase. No preprocessing was done on any of the source side data.

We report translation results using the well known evaluation metrics BLEU [16] and NIST [17]. For our primary system and the best system, we report results also in WER, PER, METEOR [18] and GTM [19].

6.1. Supplied Data Track

During the evaluation our primary focus was on the Chinese-English direction. We applied both PESA and Alignment-Free phrase extraction methods to the supplied data track. In building phrase tables using the Alignment Free method, we extracted phrase-pairs with source side up to 8-gram in length. PESA online phrase extraction method can extract phrases up to full length of the sentence.

Table 2 summarizes the official translation results for our primary submissions. We also give contrastive results for the Arabic-English and Chinese-English directions in Table 3.

The primary submission for Chinese-English direction was based on PESA alignment optimized towards BLEU metric. Submissions for other language pairs were based on the Alignment-Free method optimized towards NIST. This resulted in the discrepancy between the BLEU and NIST scores for the Chinese-English direction.

Table 2: *Official results for the CMU primary submission on IWSLT'05 test set.*

Data Track	Input	Translation Direction	BLEU	NIST	WER [%]	PER [%]	METEOR	GTM
Supplied	Manual	AR-EN	40.9	8.74	50.8	43.0	0.64	0.58
		CH-EN	44.4	6.19	58.1	49.9	0.52	0.48
		JP-EN	39.3	8.00	51.3	45.9	0.56	0.52
		KR-EN	35.8	8.17	47.0	38.0	0.65	0.59
	ASR	CH-EN	36.3	6.53	46.9	36.5	0.67	0.61
Unrestricted	Manual	CH-EN	47.1	9.35	54.7	45.5	0.58	0.47
C-STAR	Manual	CH-EN	52.7	10.02	42.0	32.6	0.71	0.64

Table 3: *Contrastive results for Chinese-English and Arabic-English supplied data tracks.*

Data Track	Input	Translation Direction	BLEU	NIST	WER [%]	PER [%]	METEOR	GTM
Supplied	Manual	AR-EN	46.4	9.05	46.0	38.7	0.66	0.61
		CH-EN	46.4	9.28	47.0	39.2	0.64	0.57

Table 5: *Human judgement for Chinese-English supplied data track. All metrics range between [0-4].*

System	Fluency	Adequacy	Meaning Mtns.
CMU Primary	2.88	1.35	1.34
CMU Contrast.	2.82	2.54	2.50

The contrastive results in Table 3 are based on the Alignment-Free phrase extraction approach. Compared to PESA which uses only lexical probabilities, Alignment-Free method uses more features as explained in section 2. This resulted in better scores compared to the primary submission. Also it seems optimizing towards NIST score gives a better balance between different evaluation metrics.

Table 4 gives translation results for all three test sets. We optimized the system for C-STAR'03 test set and used IWSLT'04 as the unseen test data. In most translation directions we see comparable results between IWSLT'04 and IWSLT'05 test sets.

We also conducted a subjective evaluation for submissions on Chinese-English supplied data track: primary submission (CMU Primary) and contrastive submission (CMU Contrast.). Table 5 gives the results. Evaluator followed the same guidelines as IWSLT'05 subjective evaluation specifications [20]. These results further indicates that Alignment Free approach produced better translations.

6.2. Using Additional Data

For the Unrestricted and C-STAR data tracks it is possible to use additional bilingual and monolingual training data. We used the TIDES data (Chinese newswire) as an additional source for parallel bilingual data. This data provides approximately 9 million lines of parallel texts in about 140 million words. All available data (the supplied data and the final test set) was re-segmented based on the segmentation of the TIDES data. We also replaced contractions like *I'm* or *We'll* with their respective written forms. We selected 86,826 sentences from the TIDES corpus using the translation model adaptation technique described in section 3.

For the C-STAR track we also used the full BTEC corpus as additional in-domain data.

Table 6 gives an overview of all available bilingual data. As explained in Section 3.4 we also used a language model adaptation technique in the Unrestricted Data track which added 1.8 million sentences (with 18 million words) to the language model training data. This additional data decreased the language model perplexity by over 50 (on Development set 1, C-STAR'03) compared to using only the supplied data. We used this data as additional language modeling data on the actual test set for the Unrestricted Data track.

Table 6: *Additional bilingual training data*

	# Lines	# Words (English)	# Words (Chinese)
Supplied Data	20,000	183,452	175,690
TIDES Data	9,106,599	144,030,404	135,486,265
Selected by TMA	86,826	1,649,132	1,662,906
Full BTEC Data	193,326	1,215,594	1,140,031

Translation results for the Unrestricted data track on C-STAR'03 set and the Test set are shown in Table 7. Using only the resegmented data did not give any improvement over the original segmentation. The translation model adaptation (TMA) alone improved the results

Table 4: *Translation result for all test sets.*

Phrase Alignment	C-STAR'03		IWSLT'04		Test	
	BLEU	NIST	BLEU	NIST	BLEU	NIST
AR-EN	44.8	8.14	40.3	8.10	40.9	8.74
CH-EN (PESA)	41.2	5.04	41.1	5.43	44.4	6.19
CH-EN (Al. Free)	40.3	8.10	42.8	8.82	46.4	9.28
JP-EN	50.4	7.50	49.1	7.68	39.3	8.00
KR-EN	37.9	7.66	-	-	35.8	8.17

with further improvements when also using the adapted language model (LMA).

Table 7: *Translation results for Chinese-English unrestricted data track.*

	C-STAR'03		Test set	
	BLEU	NIST	BLEU	NIST
Baseline				
New Segment.	40.6	8.23	43.5	9.02
+TMA	43.2	7.43	46.5	9.23
+TMA +LMA	43.1	7.75	47.1	9.35

Table 8 illustrates the scores for the C-STAR data track. Using the full BTEC corpus alone gives a slight improvement in BLEU scores but leads to a considerably low NIST score especially on the C-STAR03 development set. Adding the selected data from the TIDES corpus further improves all scores.

Table 8: *Translation results for Chinese-English C-STAR track.*

	C-STAR'03		Test set	
	BLEU	NIST	BLEU	NIST
Baseline				
New Segment.	40.6	8.23	43.5	9.02
+Full BTEC	42.8	6.44	49.4	8.15
+TMA	45.8	8.39	52.7	10.02

6.3. Results on ASR Output

The Chinese ASR 1-best was translated into English. The 100 best translation hypotheses were merged into a ROVER network and the best path was selected based on the ROVER score (ROVER), the language model score (LM) and the consolidation score (CON). The best scaling factors were experimentally determined using the Dev1 (C-STAR'03) set. Table 9 shows the evaluation result. The performance drastically dropped when using only the rover score. Combining ROVER with the language model helped to increase NIST scores significantly in Dev 2(IWSLT'04) set and the final test set. In addition, the consolidation enhanced the BLEU scores. Although the WER is comparable among all experiments, both BLEU and NIST scores have increased.

Table 9: *Evaluation results for ASR 1-best translations. The numbers in parenthesis show the average number of words in a sentence.*

		Score	Dev1	Dev2	Test
ASR 1-best		BLEU	35.5	33.0	36.3
		NIST	6.25	4.72	6.53
		WER	60.8	61.5	59.9
			(3.55)	(4.75)	(5.44)
MT 1000 best for ASR 1-best	ROVER	BLEU	34.8	33.9	34.5
		NIST	4.57	5.59	4.28
		WER	71.1	66.7	60.9
			(3.63)	(4.68)	(4.59)
	ROVER + LM	BLEU	36.3	34.3	34.2
		NIST	4.87	7.49	7.20
		WER	60.4	63.8	65.2
			(4.82)	(6.17)	(6.33)
	ROVER + LM + SUM	BLEU	37.3	35.4	37.2
		NIST	3.42	7.40	6.58
		WER	60.7	60.5	61.1
			(4.29)	(5.38)	(5.57)

7. Conclusions

In this paper we described the CMU statistical machine translation system that was used for the IWSLT 2005 evaluation campaign. We experimented with two phrase extraction methods; one which uses only lexical probabilities, and another method which uses additional features such as fertility and alignment. For the Chinese-English direction we also experimented with using additional data, both in-domain and out-of-domain, for model adaptation. Results indicate that this adaptation helps to increase the accuracy.

We did further experiments in integrating multiple translation hypotheses using the ROVER approach and choosing the best translation. This showed some interesting results. However further investigations are required to fully explore the potential of this approach.

Optimizing model parameters towards one metric seems to have a negative effect on other metrics. This was

especially evident when optimized towards high BLEU scores. A better approach would be optimizing the translation system using a linear combination of the different metrics.

8. References

[1] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[2] Y. Wang and A. Waibel, "Fast decoding for statistical machine translation," in *Proc. of the ICSLP 98*, Sidney, Australia, December 1998, pp. 2775–2778.

[3] F. J. Och and H. Ney, "Improved statistical alignment models," in *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, Hongkong, China, October 2000, pp. 440–447.

[4] K. Yamada and K. Knight, "A syntax-based statistical translation model," in *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, July 2001, pp. 523–530.

[5] S. Vogel, "PESA: Phrase pair extraction as sentence splitting," in *Proc. of the Machine Translation Summit X*, Phuket, Thailand, September 2005.

[6] B. Zhao and S. Vogel, "A generalized alignment-free phrase extraction," in *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, Ann Arbor, Michigan, June 2005, pp. 141–144.

[7] B. Zhao and A. Waibel, "Learning a log-linear model with bilingual phrase-pair features for statistical machine translation," in *Proceedings of the SigHan Workshop*, Jeju, Korea, October 2005.

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

[9] A. S. Hildebrand, M. Eck, S. Vogel, and A. Waibel, "Adaptation of the translation model for statistical machine translation based on information retrieval," in *Proc. of the EAMT 2005*, Budapest, Hungary, May 2003, pp. 133–142.

[10] Y. Zhang and S. Vogel, "Competitive grouping in integrated phrase segmentation and alignment model," in *Proc. of the ACL Workshop on Building and Using Parallel Texts*, Ann Arbor, Michigan, June 2005, pp. 159–162.

[11] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, 2003, pp. 160–167.

[12] J. G. Fiscus, "A postprocessing system to yield reduced error word rates: Recognizer output voting error reduction (ROVER)," in *Proc. of the IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997, pp. 347–354.

[13] H. Schwenk and J.-L. Gauvain, "Improved rover using language model information," in *Proc. of the ISCA ITRW Workshop Automatic Speech Recognition: Challenges for the new Millenium*, 2000, pp. 47–52.

[14] C. Hori and A. Waible, "Spontaneous speech consolidation for spoken language applications," in *Proc. of Interspeech 2005*, Lisbon, Portugal, September 2005.

[15] T. Takezawa, E. Sumita, F. Sugaya, H. Yamamoto, and S. Yamamoto, "Towards 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, Canary Islands, Spain, May 2002, pp. 147–152.

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

[17] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *In Proceedings of the Human Language Technology Conference (HLT)*, San Diego, CA, March 2002.

[18] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, Michigan, June 2005, pp. 65–72.

[19] J. P. Turian, L. Shen, and I. D. Melamed, "Evaluation of machine translation and its evaluation," in *Proc. of the Machine Translation Summit IX*, New Orleans, LA, September 2003.

[20] M. Eck and C. Hori, "Overview of the iwslt2005 evaluation campaign," in *the same proceedings*.