The NICT/ATR Speech Translation System for IWSLT 2008

Masao Utiyama[†], *Andrew Finch*[†,‡], *Hideo Okuma*[†,‡], *Michael Paul*[†,‡],
Hailong Cao[†], *Hirofumi Yamamoto*[†,‡], *Keiji Yasuda*[†,‡], *Eiichiro Sumita*[†,‡]

[†]National Institute of Information and Communications Technology
‡Advanced Telecommunications Research Laboratories
Hikaridai 2-2-2, Keihanna Science City, 619-0288 Kyoto, Japan
mutiyama@nict.go.jp

Abstract

This paper describes the National Institute of Information and Communications Technology/Advanced Telecommunications Research Institute International (NICT/ATR) statistical machine translation (SMT) system used for the IWSLT 2008 evaluation campaign. We participated in the Chinese–English (Challenge Task), English–Chinese (Challenge Task), Chinese–English (BTEC Task), Chinese–Spanish (BTEC Task), and Chinese–English–Spanish (PIVOT Task) translation tasks. In the English–Chinese translation Challenge Task, we focused on exploring various factors for the English–Chinese translation because the research on the translation of English–Chinese is scarce compared to the opposite direction. In the Chinese–English translation Challenge Task, we employed a novel clustering method, where training sentences similar to the development data in terms of the word error rate formed a cluster. In the pivot translation task, we integrated two strategies for pivot translation by linear interpolation.

1. Introduction

This paper describes the NICT/ATR SMT system used in the International Workshop on Spoken Language Translation (IWSLT) 2008 evaluation campaign. We participated in the following translation tasks: Chinese–English (Challenge Task), English–Chinese (Challenge Task), Chinese–English (BTEC Task), Chinese–Spanish (BTEC Task), and Chinese–English–Spanish (PIVOT Task). Although our theme for each task was different, our systems were based on a fairly common phrase-based machine translation system [1], which was built within the framework of a feature-based exponential model. The model has the following features:

- Phrase translation probability form source to target
- Inverse phrase translation probability
- Lexical weighting probability from source to target
- Inverse lexical weighting probability
- Phrase penalty
- Language model probability
- Lexical reordering probability
- Simple distance-based distortion model
- Word penalty

The decoder used for the training and decoding was the in-house multi-stack phrase-based decoder **CleopATRa**. The decoder can operate on the same principles as the MOSES decoder [2]. For the training of SMT models, we used a training toolkit adapted from the MOSES decoder. We used GIZA++ [3] for word alignment and SRILM [4] for language modeling. We used 5-gram language models trained with modified Knesser–Ney smoothing. The language models were trained with SMT training corpora on the target side. Minimum error rate training (MERT) was used to tune the decoder's parameters on the basis of the bilingual evaluation understudy (BLEU) score, and training was performed using the standard technique developed by Och [5].

2. English–Chinese (Challenge Task)

English–Chinese translation has been researched to a lesser extent than Chinese-English translation. Thus, we examined various factors affecting English–Chinese translation.

Table 1 summarizes the BLEU scores for correct recognition results (CRR). The BLEU scores [6] for "devset" are obtained with the small Challenge Task devset corpus (comprising 251 sentences). The devset corpus was also used for MERT.[1] Thus, the results in Table 1 for devset were obtained from closed experiments. The results for "devset3" (506 sentences) were obtained by using the parameters tuned on devset (open experiments). The BLEU scores were calculated based on Chinese character n-grams. When calculating BLEU scores, we removed out-of-vocabulary (OOV) words from the machine translated text and ignored punctuation.

[1]We used 3-gram language models for performing MERT and used 5-gram language models for translating test text. This is because using 3-gram language models for MERT gave better results than using 5-gram language models, as observed in our previous experiments.

system	devset	devset3
org	0.4379	0.3606
dict	0.4720	0.4105
cldc	0.4311	0.3416
all	0.4960	0.4445
all+dict+cldc	0.5191	0.4493
all+questions+declarations	0.5020	0.4410
all+dict+cldc+q.+d.	0.5126	0.4512

Table 1: Comparison of BLEU scores for correct recognition results

2.1. Chinese word segmentation

Zhang et al. [7] have shown that Chinese word segmentation (CWS) has a significant impact on Chinese–English SMT. We examined the effect of CWS on the abovementioned English–Chinese task. We compared the original CWS in the supplied BTEC training corpus (approximately 20,000 sentences) with a re-segmentation of the same corpus.

Our CWS was performed with a tool developed by Zhang et al. [8]. The tool was a dictionary-based hybrid CWS system. Its lexicon and language model were obtained as follows. First, we created a hybrid training corpus by combining all the training corpora [9]–AS, CITYU, MSR, and PKU–from the Second International Chinese Word Segmentation Bakeoff. The hybrid corpus was used to train a conditional random field (CRF)-based CWS system. The CRF-based segmenter was then used to segment a Chinese corpus created from the training data used for the 2005 NIST MT evaluation campaign. The language model of the dictionary-based hybrid CWS was trained using the segmented training data. Subsequently, we extracted a lexicon of 100,000 words from the most frequently occurring words in the segmented training data. In our experiment, this 100,000-word lexicon was appended to approximately 11,000 words that were extracted from the supplied BTEC training corpus. Note that a lexicon and a language model are the only resources needed for building the dictionary-based CWS system described by Zhang et al. [8].

In Table 1, "org" and "dict" show the BLEU scores obtained when using the supplied BTEC training corpus with the original segmentation and the re-segmentation by our CWS system, respectively. For devset3 (used in open experiments), the BLEU score obtained using the original segmentation was 0.3606, while that obtained using the re-segmentation was 0.4105. A substantial improvement of 4.99% BLEU was observed. Thus, we conclude that CWS is vital for English–Chinese SMT as well as for Chinese–English SMT.

2.2. Additional corpus

We investigated the performance of SMT systems trained with additional corpora obtained from external resources because the performance of SMT systems is fundamentally bounded by their training corpora. We used the Chinese Olympic corpus (comprising about 52,000 sentences); this corpus is distributed by the Chinese Linguistic Data Consortium (Code: 2004-863-0009), which we refer to as CLDC hereafter

We segmented the CLDC using the tool developed by Zhang et al. The BLEU score obtained by using only CLDC was 0.3416 for devset3 as shown in the "cldc" row of Table 1. It is lower than that for "dict" by 6.89% BLEU. This implies that the supplied BTEC corpus is more suitable than CLDC for devset3. This is not surprising because devset3 and the supplied training corpus were extracted from the same corpus. However, "dict" also outperformed "cldc" for devset. This suggests that the supplied BTEC corpus is more suitable for the IWSLT 2008 Challenge Task than CLDC, even though the sentences in devset were extracted from a *non*-BTEC corpus.

We combined the training data for the "dict" and the "cldc" systems to form a training corpus comprising approximately 72,000 sentences. The BLEU score for the system using this corpus is shown in the "all" row of Table 1. The BLEU score for the "all" system was 0.4445 for devset3, which was higher than that obtained for the "dict" system by 3.4% BLEU. Thus, CLDC was very effective in improving BLEU scores.

These experiments confirmed the well-known fact that the addition of relevant data improves the SMT performance.

2.3. Dynamic Model Interpolation

We investigated the effects of training data clustering on the performance of SMT. Yamamoto and Sumita [10] divided the training data into topics and built topic-dependent models for SMT. We explored *natural* clustering of sentences for our investigation.

2.3.1. Clustering by corpora

For the first experiment, we used two clusters. The first cluster was the supplied corpus and the second was CLDC. We also used the combined corpus described above. Consequently, we built three SMT systems from these three corpora: "dict", "cldc" and "all", which has been described above. We *dynamically* combined all of the component models (phrase-table, reordering-table, language model, etc) of the SMT systems under a single framework [1].

Our decoder, **CleopATRa**, can linearly interpolate all the models from all the sub-systems (in this case three systems) according to a vector of interpolation weights that are supplied for each sentence to be decoded. In order to perform the interpolation, prior to the search, the decoder must first merge the phrase-tables from each sub-system. Every phrase from all of the phrase-tables is used during the decoding. Phrases that occur in one sub-systems' table, but do not occur in another sub-system's table will be used, but will receive no support (zero probability) from those sub-systems that did

not acquire this phrase during training. The search process proceeds as in a typical multi-stack phrase-based decoder.

In this experiment, the weight for "all" was set by tuning the parameter using "devset" in order to optimize performance of the system with respect to the BLEU score. The abovementioned weight determined the amount of probability mass to be assigned to "all", and it was constant during the decoding of all sentences. The remainder of the probability mass was dynamically divided among sub-classes ("dict" and "cldc"), sentence-by-sentence at run-time. The fraction that is assigned to each class is simply the probability of the source sentence belonging to that particular class (class membership probability): this probability is assigned by a classifier.[2]

We used a maximum entropy (ME) classifier[3] to determine which class to which the input source sentence belongs using a set of lexical features. We used the set of words in the input source sentence as features. Examples of the training data for the ME classifier are shown in Table 2. "C0" and "C1" indicate that the corresponding sentences are in the supplied BTEC corpus and the Chinese Olympic corpus, respectively. The accuracy of the classifier was 0.788 when we applied cross-validation on the training corpus.

Class	Features
C0	please input your pin number
C0	we want to have a table near the window
C1	yes please
C1	thank you sir

Table 2: Examples of training data for our ME classifier when clustering by corpora

In Table 1, "all+dict+cldc" shows the BLEU scores for this setting. The BLEU score for devset3 was 0.4493; this score is higher than that for "all" by 0.48% BLEU. Thus, it is effective to use clusters based on the corpora. Note that we also attempted bilingual clustering [10]. However, the results were not as good as those of "all+dict+cldc".

2.3.2. Clustering by sentence type

For the second experiment, we clustered training sentences on the basis of their sentence type [1]. We partitioned the "all" training data into *questions* and *declarations (sentences that were not questions)* based on the punctuation marks in the target-side (Chinese) sentences. Subsequently, we trained an ME classifier by using the set of 1-grams, 2-grams, and 3-grams in the input sentence. We added "<s>" and "</s>" to the beginning and end of the input sentence when we obtained features.

[2]Actually, we interpolated the class membership probability with a uniform prior probability. The weight for the class membership probability was the classification accuracy of the classifier.

[3]http://www2.nict.go.jp/x/x161/members/mutiyama/software.html

Examples of the training data for the ME classifier are shown in Table 3. "Q" and "D" indicate that the corresponding sentences are questions and declarations, respectively. The accuracy of the classifier was 0.962 when we applied cross-validation on the training corpus.

We developed two SMT systems, one from questions and one from declarations. The SMT system for questions was tuned using the question sentences in devset and that for declaration was tuned using the declaration sentences in devset. Consequently, we used approximately 100 sentences for MERT. These two SMT systems were combined with "all" as described in the previous section.

In Table 1, "all+questions+declarations" shows the BLEU scores for this setting. The BLEU score for devset3 was 0.4410, which was lower than that obtained for "all" by 0.35% BLEU. Thus, it is not beneficial to use clusters based on sentence type. This observation is consistent with that of Finch and Sumita [1], who reported that clustering by sentence type does not result in any improvement in the performance of Chinese–English SMT.

2.3.3. Combining all SMT systems

For the third experiment, we interpolated all the SMT systems used in the first and second experiments. In other words, we interpolated "all", "dict", "cldc", "question" and "declaration" systems. We assigned a weight of w_1 to "all" system, w_2 to "dict" and "cldc" systems, and w_3 to "question" and "declaration" systems, where $w_1 + w_2 + w_3 = 1$. These weights, which were determined by a grid search on devset, were fixed during the decoding of all sentences. The weight w_2 (w_3) was dynamically divided among "dict" and "cldc" ("question" and "declaration") systems sentence-by-sentence at run time using the ME classifiers as described above.

In Table 1, "all+dict+cldc+q.+d." shows the BLEU scores for this setting. The BLEU score for devset3 was 0.4512, which was higher than those for "all", "all+dict+cldc", and "all+questions+declarations" systems. Thus, we concluded that it is beneficial to combine all the SMT systems.

Based on these experiments, we decided to use the "all+dict+cldc+q.+d." system for this task.

2.4. Reranking of N-best sentences

Each English sentence in a list of N-best English sentences, which was made from automatic speech recognition (ASR) results, had three scores. We added an SMT score obtained by translating each English sentence into a Chinese sentence. As a result, we had the following: (1) an English ASR output, (2) a Chinese translation, and (3) four scores for each of the N-best sentences. We assigned weights to these scores and summed these weighted scores to obtain the score for each sentence. Then, we reranked the N-best sentences by using these new scores.

Class	Features
Q	<s>_where <s>_where_do where where_do where_do_i do ...
Q	<s>_how <s>_how_long how how_long how_long_is long ...
D	<s>_the <s>_the_light the the_light the_light_was ...
D	<s>_i <s>_i_have i i_have i_have_a have have have_a ...

Table 3: Examples of training data for our ME classifier when clustering by sentence type

We compared two types of N-best reranking strategies.

The first strategy, SMT-reranking, is based on Chinese translations. We applied MERT on the N-best of devset using N-best Chinese translations and reference Chinese translations with respect to BLEU score. That is, we attempted to obtain Chinese translations that were similar to the reference Chinese translations.

The second strategy, ASR-reranking, is based on English ASR outputs. We applied MERT on the N-best of devset using N-best English ASR outputs and correct English recognition results as references. That is, we attempted to obtain English ASR outputs that were similar to CRR.

The BLEU scores for CRR, 1-BEST, and 20-BEST[4] inputs are shown in Table 4. This table shows that the ASR-reranking strategy is more effective than the SMT-reranking strategy for the data in open experiments (devset3). Consequently, we used the ASR-reranking strategy in this task.

input	devset	devset3
CRR	0.5126	0.4512
1-BEST	0.4473	0.4034
20-BEST (SMT-reranking)	0.4701	0.4007
20-BEST (ASR-reranking)	0.4614	0.4050

Table 4: Comparison of BLEU scores for the "all+dict+cldc+q.+d." system

2.5. Results for testset

We used the "all+dict+cldc+q.+d." system for this task. The BLEU scores for the testset are shown in Table 5. Punctuation and case were restored by using the SRILM toolkit.

input	case+punc	no-case+no-punc
CRR	0.4176	0.4122
1-BEST	0.3653	0.3594
20-BEST (SMT-reranking)	0.3704	0.3675

Table 5: Comparison of BLEU scores for the testset

2.6. Results without MERT

After the final submission, we repeated the same experiments as above, except that we did not perform MERT in these experiments. The results are shown in Tables 6 and 7. Comparing these tables with Tables 1 and 4, we concluded that MERT using a small amount of development data had a negative effect on the BLEU scores. Note that in Tables 6 and 7, the BLEU scores for devset are the results of the open experiments while, in Tables 1 and 4, the BLEU scores for devset are the results of the closed experiments.

system	devset	devset3
org	0.4282	0.4301
dict	0.4462	0.4363
cldc	0.4399	0.3834
all	0.4963	0.4710
all+dict+cldc	0.4966	0.4691
all+questions+declarations	0.5055	0.4743
all+dict+cldc+q.+d.	0.5070	0.4745

Table 6: Comparison of BLEU scores for correct recognition results (without MERT, cf. Table 1)

input	devset	devset3
CRR	0.5070	0.4745
1-BEST	0.4298	0.4276
20-BEST (SMT-reranking)	0.4534	0.4222
20-BEST (ASR-reranking)	0.4425	0.4232

Table 7: Comparison of BLEU scores for the "all+dict+cldc+q.+d." system (without MERT, cf. Table 4)

3. Chinese–English (Challenge Task)

3.1. Corpora

For the experiments used in this section, we took parallel sentence pairs from all of the supplied BTEC training, development, and the Chinese Olympic corpora. We used the small challenge corpus for evaluation of our models during the development process, and for partitioning the training and development data (described later). Table 8 shows the sizes of the corpora used in this task.

3.2. Clustering

The whole parallel data was partitioned according to distance from the challenge development corpus. We used the whole challenge corpus as a reference set, and calculated the word

[4] 20-BEST ASR outputs were distributed by the IWSLT organizers as the default N-best lists. N-best outputs for N greater than 20 may lead to better performance.

Corpus	Sentences	Tokens
(1) train en (MERT)	72196	603591
(2) train zh (MERT)	72196	555372
(3) train en (Submission)	106245	881005
(4) train zh (Submission)	106245	791298
(5) train (in domain) en (MERT)	32803	210018
(6) train (in domain) zh (MERT)	32803	194732
(7) train (in domain) en (Subm.)	52745	344711
(8) train (in domain) zh (Subm.)	52745	309435
(9) dev zh	1577	9274
(10) dev en	1577	9878-11184

Table 8: Corpus size

error rate (WER) of each sentence in the training corpus to the closest sentence (with respect to edit distance) in the reference set. We chose a WER threshold to partition the corpus such that the corpus was partitioned into two approximately equal-sized parts. This threshold was not determined empirically, but set heuristically to maintain a balance between specificity of the class and data size. In future, we would like to run experiments to determine the optimal value for this parameter.

3.3. System Combination

For the final submission, the development data (Table 8, (9)(10)) was added to the training corpus for the system. By constructing new training data by making bi-lingual pairs from pairing each source sentence with all of its reference translations the number of training sentences increased significantly (by around 50%) (Table 8, (3)(4)). The decoder parameters learned by a previous experiment that tuned on the development data (Table 8, (1)(2)(5)(6)) were used for decoding.

Two systems were built and then combined in the decoder by interpolation of the scores from all models in the MT system [1].

One system was trained on all of the training data (Table 8, (3)(4)). The second system was trained only on the segment of the training data that contained only sentences close to the challenge development set in terms of WER (Table 8, (7)(8)). During decoding a single interpolation weight needed to be found. This was obtained by using a simple grid search to maximize the BLEU score. The optimal weights were 0.9 for the model trained on all of the data and a weight of 0.1 for the domain-specific model.

The development data (Table 8, (9)(10)) were also partitioned according to distance from the small amount of supplied challenge-task. The intuition being that since the test data was likely to be similar in character to the challenge development data, selecting similar development data to tune on would yield parameters more suited to decoding the test data. The challenge development data itself was believed to be a little too small to give reliable estimates of the MERT-tuned parameters.

3.4. Pre-processing

The English training data were pre-processed using the tokenization tool supplied for the NIST MT06 evaluation campaign. This tokenizer was slightly modified to handle cases in the CLDC data where sentence final punctuation was not separated from the last and first words of the surrounding sentences by white space. The English data was lowercase, and stripped of punctuation before training. The punctuation being restored in a later post-processing step. The Chinese tokenizer used was an in-house tokenizer as described in Section 2.1.

3.5. Post-processing

Out of vocabulary words were removed from the MT output as it is common for these to have a detrimental affect on automatic evaluation schemes, and in particular the BLEU score. Punctuation and case were restored using the hidden n-gram technique.

3.6. Decoding Conditions

The decoding was performed by the in-house CleopaATRa decoder. This decoder works according to the same principles as MOSES, but for purposes of these experiments was configured to interpolate the scores from two MT systems during the decoding. The interpolation weights were set once for the entire decoding run, rather then dynamically for each sentence. Other training conditions were the same as those described in Section 1. The phrase-table for the model trained on all of the data consisted of 1.16M phrase pairs, whereas that for the topic specific model only contained 304K pairs.

4. PIVOT Task

We integrate two strategies for pivot translation by linear interpolation. Here, we named these strategies *PseudoCorpus* and *PhraseTableComposition* and the integrated strategy *LinearInterpolate*. First, we integrated two types of *PseudoCorpus* systems, and then, we integrated the resulting system with the *PhraseTableComposition* system. As a reference, we have also presented another strategy named *Cascade*.

4.1. Pivot Translation Strategies

In this paper, we refer to bilingual corpora between languages X and E and languages Y and E as the "X-E corpus" and "Y-E corpus," respectively.

4.2. Cascade

Assuming that we have a bilingual corpus between languages X and language E, and one between languages Y and E, the simplest and easiest method to translate between languages

X and Y is to translate through the pivot language E. We cascade the language X to language E (X2E) SMT system and the language E to the language Y (E2Y) SMT system to form a cascaded system. In this cascaded system, we translate an input sentence x in language X into e in language E using the X2E system; e is then translated into y in language Y by the E2Y system.

4.3. Pseudo Corpus

This strategy was introduced by Gispert [11]. To implement this strategy, we first develop a language E to language X SMT system (we will call this the E2X system) using the X-E corpus. Then, we form a pseudo corpus X' by translating corpus E of the "Y-E corpus" to language X using the N-BEST outputs of the E2X system. Subsequently, we train models of the language X to language Y SMT system by using corpus Y of the "Y-E corpus" and the newly created pseudo corpus X'. After developing models as described above, we remove all of phrase table entries that have OOV words on the source side of the phrase table. We call the system developed above the X'2Y system and the strategy *PseudoCorpusX*. In this strategy, the source side of the phrase table is not completely reliable.

In a manner similar to that described above, we will develop another X2Y SMT system. First, we prepare a language E to language Y SMT system using the "Y-E corpus," named the E2Y system. Then, we form a pseudo corpus Y' by translating corpus E of the"X-E corpus" to language Y, using the N-BEST outputs of the E2Y system. Subsequently, we train models of the language X to language Y SMT system by using corpus X of the "X-E corpus" and the newly created corpus Y'. After developing the models, as described above, we remove all the phrase table entries that have OOV words on the target side of the phrase table. We will call the system developed above the X2Y' system and the strategy *PseudoCorpusY*. In this system, the target side of the phrase table is not completely reliable.

For training these systems, we develop and use a language model using corpus Y of the "Y-E corpus."

4.4. Phrase Table Composition

This strategy was introduced by Utiyama [12]. In order to implement this strategy, we first develop the X2E system using the "X-E corpus" and the E2Y system using the "Y-E corpus." Then, we compose a new phrase table from the phrase tables of the X2E and E2Y systems.

For the purpose of integrating two models, we extend this strategy to include the lexicalized reordering model.

4.5. Linear Interpolation

This strategy is used to develop new models from those described above, by linear interpolation: the phrase translation model and the lexicalized reordering model. First, we interpolate two *PseudoCorpus* models. These models are developed from different bilingual corpora. Second, we interpolate the resulting model and the *PhraseTableComposition* model. Each of the three models used in this strategy has different characteristics.

	Chinese	English	English	Spanish
no. of sentences	20000		19972	
no. of tokens	135518	160138	159959	147560
ave. of tokens	6.78	8.01	8.01	7.34
vocabulary	9172	7117	7168	9811

Table 9: Training Corpus Size

Development set size					
	Sents	Tkns	Ave. Tkns	Voc.	Refs
Zh→En	506	3209	6.34	930	16
En→Zh	506	3260	6.44	831	1
En→Es	506	3260	6.44	831	16
Zh→Es	506	3209	6.34	930	16

Table 10: Development Set Size

4.6. Experiments

The language pairs in the IWSLT 2008 PIVOT Task are Chinese–English and English–Spanish. The Languages X, Y, and E in section 4.1 correspond to Chinese, Spanish, and English (the pivot language), respectively.

4.6.1. Training

Table 9 shows the sizes of both of the bilingual corpora. Both bilingual corpora are not merged to form a single bilingual corpus, but have 149 English sentences and 4684 English words in common. For the *PhraseTableComposition* strategy, we compare the effect of both with and without the composed lexicalized reordering model. For the *PseudoCorpus* strategy, we make a pseudo corpus using 100-best outputs. For the *LinearInterpolation* strategy, we first form a model by the best linear interpolation of the *PseudoCorpusEs* and *PseudoCorpusZh* models; then, we form a model from the linear interpolation of the abovementioned model and the *PhraseTableComposition* model. In this paper, Es, Zh, and En stand for Spanish, Chinese, and English, respectively.

Table 10 shows the size of each development corpus set.

The development set is a trilingual corpus of Chinese, English, and Spanish. Zh→En is used for the *Cascade* and *PhraseTableComposition* strategies. En→Zh is used for the *PseudoCorpus* strategy. En→Es is used for the *Cascade*, *PseudoCorpus*, and *PhraseTableComposition* strategies. Zh→Es is used for the *PseudoCorpus*, *PhraseTableComposition*, and *LinearInterpolation* strategies.

The interpolation ratio between the *PseudoCorpusEs* model and the *PseudoCorpusZh* model was 7:3, as derived using the development set. The best interpolation ratio between the model integrated two *PseudoCorpus* strategies and the *PhraseTableComposition* model was 9:1.

Training and decoding were performed in the manner de-

Task	Direction	Strategy	BLEU	NIST	WER	METEOR	(BLEU+METEOR)/2
Pivot	Zh→Es	*Cascade*	25.29	5.331	58.99	45.33	35.31
		PC Es	27.40	5.605	57.86	46.87	37.14
		PC Zh	28.60	5.594	55.46	47.64	38.12
		PTC	27.03	5.189	58.90	45.26	36.15
		LI PC Es/PC Zh	29.91	**5.983**	**54.52**	**49.47**	39.69
		LI PC Es/PC Zh/PTC	**30.50**	5.708	54.84	49.20	**39.85**
	Zh→En		40.95	7.625	47.74	60.76	50.86
	En→Zh		14.67	4.243	75.83	39.52	27.10
	En→Es		55.21	9.400	30.33	73.08	64.15
BTEC	Zh→En		50.01	8.382	37.78	68.08	59.05
	Zh→Es		32.73	6.759	50.87	53.61	43.17

Table 11: Auto evaluation scores of each system

scribed in Section 1.

4.6.2. Results

Table 11 shows the auto-evaluation results for each of the strategies for devset by the BLEU, NIST [13], WER, and METEOR [14] scores. As a reference, we include the Zh→En, En→Zh, and En→Es scores of the SMT system. Additionally, we include the results of the IWSLT 2008 BTEC Task in this table as the upper bound of the PIVOT Task. *PC*, *PTC*, and *LI* stand for *PseudoCorpus*, *PhraseTableComposition*, and *LinearInterpolation*, respectively. *LRM* stands for Lexicalized Reordering Model.

The relative (BLEU+METEOR)/2 score for the *LinearInterpolation* strategy *LI PC Es/PC Zh/PTC* is 0.92 points below its Chinese–Spanish BTEC task score.

5. Conclusions

We participated in Chinese–English (Challenge Task), English–Chinese (Challenge Task), Chinese–English (BTEC Task), Chinese–Spanish (BTEC Task), and Chinese–English–Spanish (PIVOT Task) translation tasks. In the English–Chinese translation Challenge Task, we observed that Chinese word segmentation and external resources had a significant impact on the translation results. In the Chinese–English translation Challenge Task, we used a novel clustering method. Finally, in the PIVOT Task, we integrated two strategies for pivot translations by linear interpolation.

6. References

[1] A. Finch and E. Sumita, "Dynamic model interpolation for statistical machine translation," in *Proceedings of the Third Workshop on Statistical Machine Translation*, 2008, pp. 208–215.

[2] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, 2007, pp. 177–180.

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

[4] A. Stolcke, "SRILM - an extensible language modeling toolkit," in *ICSLP*, 2002.

[5] F. J. Och, "Minimum error rate training in statistical machine translation," in *ACL*, 2003.

[6] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *ACL*, 2002.

[7] R. Zhang, K. Yasuda, and E. Sumita, "Improved statistical machine translation by multiple Chinese word segmentation," in *Proceedings of the Third Workshop on Statistical Machine Translation*, 2008, pp. 216–223.

[8] R. Zhang, G. Kikui, and E. Sumita, "Subword-based tagging by conditional random fields for Chinese word segmentation," in *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, 2006, pp. 193–196.

[9] T. Emerson, "The second international Chinese word segmentation bakeoff," in *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, 2005.

[10] H. Yamamoto and E. Sumita, "Bilingual cluster based models for statistical machine translation," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 514–523.

[11] A. de Gispert and J. B. Mari, "Catalan-English statistical machine translation without parallel corpus: Bridging through spanish," in *Processdings of LREC 5th Workshop on Strategies for developing Machine Translation for Minority Languages*, 2006.

[12] M. Utiyama and H. Isahara, "A comparison of pivot methods for phrase-based statistical machine translation," in *Proceedings of NAACL HTL 2007*, 2007, pp. 484–491.

[13] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *Proceedings of the HLT Conference*, San Diego, California, 2002.

[14] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005, pp. 65–72.