The TÜBİTAK-UEKAE Statistical Machine Translation System for IWSLT 2008

Coşkun Mermer, Hamza Kaya, Ömer Farukhan Güneş, Mehmet Uğur Doğan

National Research Institute of Electronics and Cryptology (UEKAE)
The Scientific and Technological Research Council of Turkey (TÜBİTAK)
Gebze, Kocaeli 41470, Turkey
{coskun,hamzaky,farukhan,mugur}@uekae.tubitak.gov.tr

Abstract

We present the TÜBİTAK-UEKAE statistical machine translation system that participated in the IWSLT 2008 evaluation campaign. Our system is based on the open-source phrase-based statistical machine translation software *Moses*. Additionally, phrase-table augmentation is applied to maximize source language coverage; lexical approximation is applied to replace out-of-vocabulary words with known words prior to decoding; and automatic punctuation insertion is improved. We describe the preprocessing and postprocessing steps and our training and decoding procedures. Results are presented on our participation in the classical Arabic-English and Chinese-English tasks as well as the new Chinese-Spanish direct and Chinese-English-Spanish pivot translation tasks.

1. Introduction

In this paper, we report on our second participation in the IWSLT evaluation campaign. Among the six translation tasks in IWSLT 2008, we participated in the following:

- Arabic-to-English (BTEC Task)
- Chinese-to-English (BTEC Task)
- Chinese-to-Spanish (BTEC Task)
- Chinese-to-English-to-Spanish (Pivot Task)

We built our baseline system based on the open-source phrase-based statistical machine translation software *Moses*. Among shared corpora and tools, we used only the supplied training data and the Buckwalter Arabic Morphological Analyzer. In order to cope with previously unseen words during decoding, we used the run-time lexical approximation method, which replaces an out-of-vocabulary word with the closest known word having the same feature. This system obtained very good translation results in last year's evaluation campaign, especially in the clean transcript condition [1].

We trained separate translation, target language, source punctuation and target recasing models for each translation task. Both correct recognition results (CRR) and 1-best automatic speech recognition (ASR) outputs are translated. We used BLEU scores to test and tune our systems. The results of our run submissions are reported in terms of the official BLEU and METEOR metrics, as well as six other automatic evaluation metrics.

2. Training

2.1. Corpora

We used only the supplied BTEC corpora [2] in developing our systems. For BTEC_AE and BTEC_CE tasks, six development sets were supplied, so we also included *devsets1-3* in the training corpus. *Devsets1-3* all have 16 English reference segments per source segment. In order to obtain better phrase alignments and to increase the system's target phrase coverage, all reference segments in these data sets were included in the training set with their corresponding source segments. Tables 1 and 2 show the corpora used in training and development, respectively.

Table 1: Corpora used in training

Task	Corpora	Sentence pairs
BTEC_AE	*train, devsets1-3*	44,164
BTEC_CE	*train, devsets1-3*	44,164
BTEC_CS	*train*	19,972
PIVOT_CE	*train*	20,000
PIVOT_ES	*train*	19,972

Table 2: Corpora used in development

Task	Corpora	Source sentences	English references per sentence
BTEC_AE	*devsets4-6*	1478	6/7
BTEC_CE	*devset4-6**	1478	6/7
BTEC_CS	*devset3*	506	16
PIVOT_CE	*devset3*	506	16
PIVOT_ES	*devset3*	506	16

* ASR outputs were not available for *devset6*, so only *devsets4-5* were used for developing the BTEC_CE ASR system.

The English sides of the final training corpora were used to generate 3-gram target language models for each translation task. For this purpose, the SRI language modeling toolkit [3] was used with modified Kneser-Ney discounting and interpolation.

2.2. Sentence splitting

Before translation model training, multi-sentence segments are split so as to prevent erroneous word alignments across sentence boundaries. The splitting is done automatically on segments with equal number of sentence boundary punctuations in both the source and the target. The resulting number of segments in each corpus are shown in Table 3.

Table 3: Number of segments in the training corpora before and after automatic splitting

Corpus	Number of segments before splitting	Number of segments after splitting
BTEC_AE	44,164	**49,325**
BTEC_CE	44,164	**49,277**
BTEC_CS	20,040	**23,308**
PIVOT_CE	20,000	**22,563**
PIVOT_ES	20,040	**23,856**

In the BTEC_CS and BTEC_ES training corpora, the Spanish sides sometimes contained two almost identical Spanish sentences in the same segment. We automatically split them and duplicated the corresponding English segment if there was an even number of sentences in the Spanish segment and the edit distance between the two halves was less than three substitutions. This processing was done prior to the automatic splitting mentioned above. As evident from comparing Tables 1 and 3, 68 such segments were found in each of BTEC_CS and BTEC_ES training corpora.

2.3. Orthographical normalization

One of our goals from last year was to investigate the striking discrepancy between the performance of our system in correct recognition result (CRR) and ASR output conditions in the Arabic-to-English task. Last year, we optimized our systems for the CRR condition and used the same systems to translate the ASR outputs. This approach yielded the results in Table 4 [1].

Table 4: Official BLEU scores of the submitted Arabic-English system in IWSLT 2007

Input condition	BLEU	Rank
Correct recognition result	49.23	1/11
ASR output	36.79	8/10

So this year we developed our ASR systems using only the ASR output parts of the provided development sets. We found that in the supplied Arabic corpora, eight Arabic characters (“ ٔ ”, “ ٕ ”, “ ٓ ”, “ ٖ ”, “ ٗ ”, “ ٘ ”, “ ٙ ”, “ ٚ ”) that were present in the training corpus were never used in the developments sets for the ASR output condition. In addition, the *alef* variants “ أ ” and “ إ ”never occurred at the beginning of a word.

Hence we “orthographically normalized” the training corpus to match the ASR output orthography by removing all occurrences of the mentioned eight characters, also replacing all occurrences of “ آ ” and word-initial occurrences of “ أ ” and “إ ” with “ ا ” (*alef*). Table 5 shows the effect of this normalization on the performance of ASR output translation measured by BLEU.

Table 5: Effect of orthographical normalization on ASR output translation BLEU scores in the BTEC_AE task

	devset4	*devset5*	*devset6*
Original orthography	23.14	19.96	37.67
Normalized orthography	**23.95**	**20.29**	**41.32**

Note the significant improvement especially in *devset6*, which was the test set in 2007. We also tried this normalization for CRR translation, as have been done in [4]. In this setting, input Arabic sentences are applied the same normalization before decoding. Table 6 shows the results for the CRR condition. Note that differently from last year, the punctuation marks in the *devset6* CRR were removed in order to conform with this year’s evaluation specifications.

Table 6: Effect of orthographical normalization on CRR translation performance in the BTEC_AE task

	devset4	*devset5*	*devset6*
Original orthography	26.33	21.11	48.08
Normalized orthography	**27.08**	**22.17**	**48.85**

BLEU scores were improved in all development sets. Therefore, we used the orthographically normalized translation models in our submitted Arabic-to-English systems for both ASR and CRR conditions.

2.4. Phrase table augmentation

In our system, the word alignments are generated by *GIZA++* [5] using IBM Model-4 [6] and the phrase-based translation model generation is performed by the scripts provided in the *Moses* toolkit [7]. Phrase pairs are extracted using the *grow-diag-final-and* heuristic [8] and all the phrase pairs are stored along with their translation model parameters in a “phrase table”. However, there may be some source-language words in the training corpus without a one-word entry in the phrase table. To avoid out-of-vocabulary treatment of these words in previously unseen contexts, we append them to the list of phrases extracted by the *Moses* `phrase-extract` module. The target phrases in these phrase-pairs are selected from *GIZA++* word alignments, specifically those with lexical translation probabilities above a relative threshold. Table 7 shows the effect of this process on the phrase table size.

Table 7: Phrase table augmentation. |vcb|: Source vocabulary size. |pt|: Default phrase table size. $|vcb_{missing}|$: Portion of source vocabulary without a one-word entry in the default phrase table. |Δpt|: New phrase-pairs added to the phrase table. $|pt_{augmented}|$: Augmented phrase table size.

Corpus	BTEC			PIVOT	
	AE	CE	CS	CE	ES
|vcb|	17,720	8,757	8,412	9,186	7,074
|pt|	410,346	395,211	217,728	216,563	302,583
$\|vcb_{missing}\|$	7,626	4,158	4,539	5,321	1,688
|Δpt|	20,610	13,190	16,619	21,122	3,754
$\|pt_{augmented}\|$	**430,956**	**408,401**	**234,347**	**237,685**	**306,337**

2.5. Training the punctuation model

Source language punctuation is modeled by training a 3-gram language model on a punctuated corpus. Punctuation insertion is performed before translation, using the SRILM tool `hidden-ngram`.

Last year, our punctuator had a bug in the punctuation selection which contributed to the strikingly low scores in the ASR task (see Table 4). Basically, during postprocessing the punctuation decisions, we were selecting the predictions

made at the sentence ends instead of beginnings. However, especially in Arabic, sentence-beginning words are more predictive in determining whether the sentence is a question or a declaration. After correcting this bug, we obtained the improved results shown in Table 8.

Table 8: BLEU scores of last year's system after correcting the punctuation bug (BTEC_AE ASR output condition)

	devset4	*devset5*	*devset6*
Buggy punctuation	22.43	19.58	36.81
Corrected punctuation	**23.41**	**20.29**	**38.87**

Last year, we trained the punctuator with artificially merged sentences [1]. The motivation was that when the training corpus was used directly to train a punctuation model, the punctuator failed to recognize the internal sentence boundaries in most of the multi-sentence segments in *devsets4-5*. We suspected this was because *devsets4-5* contained relatively more multi-sentence segments than the training set. Therefore, in order to train the punctuator with more occurrences of segment-internal sentence boundaries, we had artificially merged 10 segments in the training set and thus trained the punctuator.

Despite the improved BLEU scores on *devsets4-5*, this technique did not work as expected in the 2007 evaluation set, i.e., *devset6*, as shown in the first two rows of Table 9.

Table 9: BLEU scores with automatic punctuator trained on different merging strategies (BTEC_AE ASR output condition)

N	*devset4*	*devset5*	*devset6*
1	22.88	19.34	43.92
10	24.83	**21.25**	43.71
2	**24.88**	20.86	44.02
2 (modified)	24.79	20.84	**44.26**

Faced with this conflicting behavior, this year we tuned our systems according to *devset6* while still trying to achieve improvement for *devset4-5* from the baseline setting of N = 1. We artificially merged 2 sentences at a time (third row in Table 9) to improve the BLEU score on *devset6*. Also, we noticed that our punctuator had a tendency to incorrectly insert question marks in the middle of sentences. In real utterances, a question mark rarely appears in the middle of a segment because a question usually marks the end of a dialogue turn. So we constrained our artificially-merged corpus to not have any non-final question marks (the last row in Table 9), which resulted in some more improvement and was selected as the model used in the submitted systems.

2.6. Other pre-/postprocessing

We tokenized and lowercased all training data sets. Also, we performed Buckwalter transliteration on all Arabic corpora.

The Spanish corpora have additional punctuation marks ("¿" and "¡") at the beginning of question and exclamation sentences. Those punctuation marks were removed from the training sets. Accordingly, when generating a Spanish output, "¿" and "¡" were added to the beginning of sentences in a postprocessing step if the sentence-final punctuations were "?" and "!", respectively.

3. Decoding

For decoding, we used *Moses* [7], which is a phrase-based beam-search decoder that uses a log-linear model, with the following default scoring functions:

- source-to-target phrase translation score,
- target-to-source phrase translation score,
- source-to-target lexical translation score,
- target-to-source lexical translation score,
- language model score,
- word count penalty, and
- distortion penalty.

3.1. Run-time lexical approximation

The basic premise of lexical approximation [1] is to replace a previously unseen word in the input sentence with a known word that has the same feature. It is applied twice before decoding:

- In the first step (LA#1), the feature function returns the morphological root(s) of the word according to Buckwalter Arabic Morphological Analyzer [9].
- The still-remaining unknown words go through a second step (LA#2), in which the feature function returns an orthographical normalization of the word obtained by removing all the vowels and diacritics.

Among the candidate replacements that share the same feature with the OOV word in question, the one with the least edit-distance is selected. In case of a tie, the more-frequently occurring candidate is chosen. In last year's evaluation, lexical approximation proved to be very effective, especially in the Arabic-to-English task.

With orthographical normalization of the training corpora this year (Section 2.3), some words that were OOVs last year could be no more OOV, *e.g.*, those which are unseen orthographical variations of known words. Therefore we investigated how lexical approximation is affected by using an orthographically normalized translation model. Tables 10 and 11 compare the number of out-of-vocabulary (OOV) words in the development sets and the OOV reduction achieved with the lexical approximation method.

Table 10: Effect of lexical approximation (LA) on OOV words when default models are used

OOV words	CRR			ASR		
	devset4	*devset5*	*devset6*	*devset4*	*devset5*	*devset6*
Input	661	795	424	735	909	374
After LA#1	185	221	108	205	270	121
After LA#2	149	172	65	180	227	76

Table 11: Effect of lexical approximation (LA) on OOV words when orthographically normalized models are used

OOV words	CRR			ASR		
	devset4	*devset5*	*devset6*	*devset4*	*devset5*	*devset6*
Input	524	637	215	568	718	213
After LA#1	166	202	81	183	254	83
After LA#2	148	172	65	168	222	76

Indeed, the number of OOV words in the input is significantly reduced by using normalized orthography, especially for *devset6*. Lexical approximation is in addition able to resolve most of the remaining OOV words. The number of non-resolved words after LA (the last row in both

tables) is very close in both cases, suggesting an overlap between the OOV reduction of the two methods (orthographical normalization and LA).

3.2. Decoding setup

As a change from last year, we applied lexical approximation *before* punctuation insertion, hoping it can lead to better punctuation. As shown in Table 12, there is a small but consistent improvement, so we used this setup throughout this year's experiments.

Table 12: Effect of decoding setup on the BTEC_AE ASR output translation BLEU scores

Decoding setup	*devset4*	*devset5*	*devset6*
Punctuation, then LA	23.41	20.25	38.32
LA, then punctuation	**23.80**	**20.30**	**38.36**

3.3. Case restoration

After decoding, target language case information is automatically restored using the *Moses* recasing tool. A lowercase-to-truecase translation model is trained and applied on the translation outputs, together with a few simple rules such as uppercasing the first letter of a sentence.

4. Results and discussion

Table 13 shows the official scores of our submitted systems according to the eight provided metrics.

It is surprising to note that the Chinese-to-Spanish translation with English as the intermediate language (pivot translation) achieves better BLEU scores than the direct translation. We had observed the opposite during our development experiments using *devset3*, as shown in Table 14. We suspect this is due to the similarity of the 2008 test set to the pivot training corpora.

Table 14: Comparison of BLEU scores between direct and pivot translation from Chinese to Spanish

Task	CRR		ASR	
	devset3	*test*	*devset3*	*test*
PIVOT_CES	25.71	**32.94**	20.77	**29.40**
BTEC_CS	**32.40**	29.07	**25.67**	26.85

Also note in Table 13 that all metrics rank pivot translation higher than direct translation except NIST, which evaluates up to 5-grams and is unique in including an information measure with each *N*-gram match, and (only for ASR condition) GTM, which does not restrict the length of *N*-grams. This suggests that direct translation may be able to correctly translate rarely-seen *N*-grams and longer *N*-grams better than pivot translation.

Per evaluation guidelines, we trained separate models for each translation task, using only the corpora supplied for that task. However, in the pivot translation scenario of Chinese-to-English-to-Spanish, it is reasonable to assume that the system developer has access to both the Chinese-English and English-Spanish corpora. So, in practice, the English sides of both parallel corpora could be combined when generating the English language model. Table 15 shows that, as expected, a consistent improvement can be achieved using a pivot-language model trained from combined corpora.

Table 15: PIVOT_CES BLEU scores obtained on *devset3* by using English language models trained on (i) separate and (ii) combined corpora

LM training corpus	CRR		ASR	
	CE	CES	CE	CES
Separate	35.92	25.71	30.21	20.77
Combined	**36.81**	**25.96**	**31.81**	**22.12**

5. Conclusion

We have presented our Arabic-to-English, Chinese-to-English, Chinese-to-Spanish, and Chinese-to-English-to-Spanish statistical machine translation systems based on publicly-available software. We described our modifications to translation model generation, automatic punctuation insertion, and treatment of OOV words and presented our training and decoding procedures. Official evaluation results with correct recognition result and ASR output conditions were reported and discussed.

6. References

[1] C. Mermer, H. Kaya and M.U. Doğan, "The TÜBİTAK-UEKAE Statistical Machine Translation System for IWSLT 2007", *Proc. of the International Workshop on Spoken Language Translation*, Trento, 2007, pp. 176-179.

[2] T. Takezawa, E. Sumita, F. Sugaya, H. Yamamoto, and S. Yamamoto. "Toward a Broad-coverage Bilingual Corpus for Speech Translation of Travel Conversation in the Real World" in *Proc. of LREC 2002*, Las Palmas, Spain, 2002.

[3] Stolcke, A., "SRILM – an extensible language modeling toolkit", in *Proc. International Conference on Spoken Language Processing*, vol. 2, Denver, USA, 2002, pp. 901-904.

[4] W. Shen, B. Delaney, T. Anderson and R. Slyh, "The MIT-LL/AFRL IWSLT 2007 MT System", *Proc. of the International Workshop on Spoken Language Translation*, Trento, 2007, pp. 95-102.

[5] Och, F. Z. and Ney, H., "A Systematic Comparison of Various Statistical Alignment Models", *Computational Linguistics*, vol. 29, no. 1, 2003, pp. 19-51.

[6] Brown, P.F., Della Pietra, S.A., Della Pietra, V.J. and Mercer, R.L., "The Mathematics of Statistical Machine Translation: Parameter Estimation", *Computational Linguistics*, vol. 19, no. 2, 1993, pp. 263-311.

[7] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A. and Herbst, E., "Moses: Open Source Toolkit for Statistical Machine Translation", *The 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, demonstration session, Prague, Czech Republic, June 2007.

[8] Koehn, P., Axelrod, A., Mayne, A.B., Callison-Burch, C., Osborne, M. and Talbot, D., Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation, in *Proc. of the International Workshop on Spoken Language Translation*, Pittsburgh, PA, 2005.

[9] Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistic Data Consortium Catalog: LDC2002L49.

Table 13: Official evaluation results (with case and punctuation).

			BLEU	NIST	WER	PER	GTM	METEOR	TER	No-output
BTEC	AE	ASR	0.4111	7.7762	0.4368	0.3916	0.7151	0.6274	38.9240	0/507
		CRR	0.4803	8.8110	0.3766	0.3312	0.7704	0.6785	33.5530	0/507
	CE	ASR	0.3335	6.5053	0.5349	0.4763	0.6461	0.5564	46.1600	0/507
		CRR	0.3765	6.9227	0.4949	0.4374	0.6769	0.5843	42.4810	0/507
	CS	ASR	0.2685	5.8936	0.5999	0.5199	0.6131	0.2986	54.1250	0/507
		CRR	0.2907	6.3387	0.5685	0.4889	0.6404	0.3189	50.6500	0/507
PIVOT	CES	ASR	0.2940	5.8271	0.5527	0.4951	0.6099	0.3062	47.9250	0/507
		CRR	0.3294	6.3124	0.5094	0.4557	0.6496	0.3306	43.9250	0/507