The XMU SMT System for IWSLT 2007

Yidong Chen, Xiaodong Shi, Changle Zhou

Department of Cognitive Science, School of Information Sciences and Technologies,
Xiamen University, Xiamen, Fujian, P. R. China
`{ydchen, mandel, dozero}@xmu.edu.cn`

Abstract

In this paper, an overview of the XMU statistical machine translation (SMT) system for the 2007 IWSLT Speech Translation Evaluation is given. Our system is a phrase-based system with a reordering model based on chunking and reordering of source language. In this year's evaluation, we participated in the open data track for Clean Transcripts for the Chinese-English translation direction. The system ranked the 12th among the 15 participating systems.

1. Introduction

This paper describes the system which participated in the 2007 IWSLT Speech Translation Evaluation of Department of Cognitive Science, Xiamen University. The system is a phrase-based Statistical Machine Translation (SMT) system with a reordering model based on chunking and reordering of source language. The steps of our model could be stated briefly as follows:

- At training time:

Firstly, align the sentence pairs and get word alignment matrixes.

Secondly, chunk the source sentences extract chunk reordering information according to the word alignment matrixes.

Thirdly, reorder the source sentences using the chunk reordering information.

Finally, train the baseline phrase-based system with the reordered source sentences and the original target sentences.

- At decoding time:

Firstly, chunk the test sentences.

Secondly, reorder the test sentences with the chunk reordering information.

Finally, translate the reordered sentences with the baseline phrase-based decoder in monotonic order.

This paper is organized as follows. Section 2 describes data preparation. Section 3 gives an overview of the translation model. In section 4, experiments and the results are reported. And finally, section 5 concludes.

2. Preparing the Data

2.1. Preprocessing

Data preprocessing is not a trivial task for machine translation system. Our experiments showed that good data preprocessing model can result in better translation quality.

Three types of preprocessing were performed on the Chinese part of the training data:

- Segmentation and Tagging: To transform Chinese characters into Chinese words and POS tags of each word.
- Chunking: To chunk the POS-tagged Chinese sentences. We used a Chinese chunker based on CRF++[1] in this evaluation.
- SBC case to DBC case: To replace numbers, English characters or punctuations in SBC case in Chinese by their DBC case. For instance, "１", "Ａ" and "。" would respectively be replaced by "1", "A" and ".".

For the English part of the training data, also two types of preprocessing were performed:

- Tokenization: To separate punctuations from words in English sentences.
- Truecasing of the first word of an English sentence: To transform the uppercase version of the beginning words of English sentences into their lowercase version if their lowercase version occur more often.

2.2. Word Alignment

To achieve n-to-n word alignment, we first run GIZA++ up to IBM model 4 in both translation directions to get an initial word alignment, and then apply "grow-diag-final" method [1] to refine it. This process could be addressed in detail as followed:

- In the *initial* step, we intersect the two alignments obtained by running GIZA++, i.e., Chinese to English and English to Chinese, and get a high-precision alignment.
- Then the intersection alignment *grows* iteratively by adding potential alignments, which exist in the union of the two alignments. The neighbors of the intersection points in alignment matrix, including left, right, up, bottom and the diagonally directions are checked, if either of the words linked by the potential alignment is not aligned previously, the potential alignment is added. This operator is done until no more neighbors can be added.
- In the *final* step, potential alignments, which exist in the union of two alignments, will be added if all their neighbors do not exist in the union alignment.

[1] Downloadable from http://crfpp.sourceforge.net/

2.3. Reordering of Chinese Part of the Training Set

At the training time, we used an algorithm similar to selection sort algorithm to perform the reordering.

Here, we regard the chunk reordering problem as a problem of finding a permutation of the chunks that is the best one according to the target language order, and thus is similar to the problem of sorting, whose aim is finding a permutation of a given integer sequence so that the integers are in ascending or descending order.

The word alignment matrix is used as a clue for how a Chinese chunk sequence should be reordered.

2.4. Phrase Extraction

Bilingual phrases can be learned from word aligned parallel corpus. As is common in most phrase-based SMT systems, we consider bilingual phrase as a pair of source and target words sequences, with the following constrains:

- The words should be consecutive in both source and target sentences.
- The word level alignment of bilingual phrase should be consistent with the alignment matrix.

The consistency means that the words of the bilingual phrase can only be aligned to each other, and not to any other words outside.

Our phrase extraction method is very similar to [2]. For a word aligned sentence pair, we enumerate all the consecutive words sequences of English sentence, and for each English phrase, find the corresponding Chinese words according to the alignment matrix, if it satisfies the two constraints above, a bilingual phrase is extracted. In addition, in order to extract more phrases, such a bilingual phrase can be extended at Chinese side since "NULL" alignment is allowed, which means a word aligned to nothing. For the same English phrase, we extend the corresponding Chinese phrase to both left and right, if the added Chinese word is not aligned, and the new phrase satisfies our definition, it is extracted as a bilingual phrase. This is done iteratively until no more words can be added.

However, we limited the length of phrases from 1 word to 6 words in our experiment, since it has been showed that longer phrases don't yield better translation quality [1]. And, to avoid a too large search space in decoding, we also limited the size of the translation table. For a Chinese phrase, only 20-best corresponding bilingual phrases were kept. We used Formula 1 to evaluate and rank the bilingual phrases with the same Chinese phrase.

$$\sum_{i=1}^{N} \lambda_i \cdot h_i(\tilde{e},\tilde{c}) \tag{1}$$

Where, $h_i(\tilde{e},\tilde{c})$ (1≤ i ≤N) denotes a phrase probability of a given bilingual phrase $(\tilde{e},\tilde{c})$, and λ_i (1≤ i ≤N) is the corresponding parameter for $h_i(\tilde{e},\tilde{c})$. In our system, N is set to be 4, in that there are four phrase probabilities for a given bilingual phrase (see 2.4 for details).

Note that, the parameters here should use the same values as their corresponding ones in the translation model (see 3.2 for details).

By using the pruned phrase table, our system could translate the test set from this evaluation at the speed of about 0.2 seconds per sentence.

2.5. Phrase Probabilities

Four phrase probabilities are defined for a given bilingual phrase in our system:

- Phrase translation probability $p(\tilde{e} \mid \tilde{c})$
- Inverse phrase translation probability $p(\tilde{c} \mid \tilde{e})$
- Phrase lexical weigh $lex(\tilde{e} \mid \tilde{c})$
- Inverse phrase lexical weight $lex(\tilde{c} \mid \tilde{e})$

We define the phrase translation probability using relative frequency as in Formula 2:

$$p(\tilde{e} \mid \tilde{c}) = \frac{N(\tilde{e},\tilde{c})}{\sum_{\tilde{e}'} N(\tilde{e}',\tilde{c})} \tag{2}$$

Where, $N(\tilde{e},\tilde{c})$ is the total number of bilingual phrase $(\tilde{e},\tilde{c})$ occurred in the training corpus.

Additional to $p(\tilde{e} \mid \tilde{c})$, we introduce a lexical weight metric that denotes how well the words of phrase $\tilde{c}$ translate to the words of phrase $\tilde{e}$. Following the description in [1], given a bilingual phrase (e_1^I, c_1^J) and its alignment a , the lexical weight is defined as Formula 3:

$$lex(e_1^I \mid c_1^J, a) = \prod_{i=1}^{I} \frac{1}{|\{j \mid (i,j) \in a\}|} \sum_{\forall (i,j) \in a} p(c_i \mid e_j) \tag{3}$$

For computing phrase lexical weight, we should know the word level alignment of bilingual phrases, as well as the word translation probability. When extracting phrases from the training corpus, the alignment information is reserved, moreover, a special token "NULL" is added to each English sentence and aligned to unaligned Chinese words, and then the word translation probability can be computed using relative frequency.

Probability $p(\tilde{c} \mid \tilde{e})$ and $lex(c_1^I \mid e_1^J, a)$ can be computed in the similar way to $p(\tilde{e} \mid \tilde{c})$ and $lex(e_1^I \mid c_1^J, a)$, respectively.

3. System Overview

3.1. Translation Model

As described in [3], we use a log-linear modeling approach, in which all knowledge sources are described as feature functions that include the given source language string c_1^J and the target language string e_1^I . Hence, the translation probability and the decision rule could be given by Formula 4 and 5, respectively.

$$\Pr(e_1^I \mid c_1^J) = \frac{\exp[\sum_{m=1}^{M} \lambda_m \cdot h_m(e_1^I, c_1^J)]}{\sum_{e_1^I} \exp[\sum_{m=1}^{M} \lambda_m \cdot h_m(e_1^I, c_1^J)]} \tag{4}$$

$$\hat{e}_1^I = \arg\max_{e_1^I}\left\{\sum_{m=1}^{M}\lambda_m \cdot h_m(e_1^I, c_1^J)\right\} \qquad (5)$$

Six features were used in our translation model:

- Phrase translation probability $p(\tilde{e}\,|\,\tilde{c})$
- Inverse phrase translation probability $p(\tilde{c}\,|\,\tilde{e})$
- Phrase lexical weigh $lex(\tilde{e}\,|\,\tilde{c})$
- Inverse phrase lexical weight $lex(\tilde{c}\,|\,\tilde{e})$
- English language model $lm(e_1^I)$
- English sentence length penalty I

3.2. Parameters

The parameters used in the translation model could be trained using discriminative training method such as minimum error rate training [4].

But due to the time limitation, we didn't implement such method. So we have to adjust the parameters by hand. Moreover, we didn't readjust the parameters according to the develop sets provided in this evaluation again due to the time limitation. On the contrary, we simply used an empirical setting, with which our decoder achieved a good performance in translating the test set from the *2005 China's National 863 MT Evaluation*. The parameter settings for our system are listed in Table 1, as followed:

Table 1: The parameter settings

Parameters	Corresponding Features	Values
λ_1	$p(\tilde{e}\,|\,\tilde{c})$	0.15
λ_2	$p(\tilde{c}\,|\,\tilde{e})$	0.03
λ_3	$lex(\tilde{e}\,|\,\tilde{c})$	0.16
λ_4	$lex(\tilde{c}\,|\,\tilde{e})$	0.03
λ_5	$lm(e_1^I)$	0.13
λ_6	I	0.48

Please note that the parameter settings listed above is not optimal for the training and test set from this evaluation.

3.3. Decoder

We used the monotone search in the decoding, as described in [5]. And the monotone search was implemented with dynamic programming.

For the maximization problem in Formula 5, we define the quantity $Q(j,e)$ as the maximum probability of a phrase sequence. Thus $Q(J+1,\$)$ is the probability of the optimal translation, where the \$ symbol is the sentence boundary marker. Given the definitions, we then obtain the following dynamic programming recursion:

$$Q(0,\$) = 1 \qquad (6)$$

$$Q(j,e) = \max_{\substack{0\le j'<j \\ e',\tilde{e}}}\left\{Q(j',e') + \sum_{m=1}^{M}\lambda_m \cdot h_m(\tilde{e}, c_{j'+1}^{j})\right\} \qquad (7)$$

$$Q(J+1,\$) = \max_{e'}\left\{Q(J,e') + p(\$\,|\,e')\right\} \qquad (8)$$

During the search, we stored back-pointers to the maximizing arguments. So after performing the search, we could generate the optimal translation, easily.

3.4. Reordering of Source Sentence

As mentioned in Section 1. In our system, a chunk-level source language reordering model is used before the traditional phrase-based decoder. Here, we use a way similar to the monotone decoding of phrase-based SMT to performing the reordering. A dynamic programming recursion similar to that of 3.3 is used.

Two kinds of data are required:

- Reordering Patterns, which is a set of triple <*CST, Perm, Prob*>. Here, *CST* is a chunk tag sequence, *Perm* is a permutation, and *Prob* is the corresponding probability.
- Chunk tag 3-gram.

These two types of data could both be trained used the training bitexts, with the Chinese part reordered at the training time.

3.5. Dealing with the Unknown Words

Words that are not covered by phrases are called unknown words. Keeping unknown words un-translated will make the translations less readable, so most phrase-based systems integrated a model to deal with them. Some systems simply dropped the unknown words [6] while other systems integrated a pre-translation model to detect and translate special unknown words such as named entities and simply dropped other unknown words [7].

In our system, no special translation models for named entities are used. Named entities are translated in the same way as other unknown words. During the decoding, an unknown word will be translated in two steps, as followed:

- Firstly, we will look up a dictionary containing more than 100,000 Chinese words for the word. All the translations will be put into the phrase table with a certain probability, and the most optimal one will be selected by the translation model. In this evaluation, the probability was set to be 10^{-7}.
- If no translations are found in the first step, the word will then be translated using a rule-based Chinese-English translation system[1].

4. Experiments

In this year's evaluation, we participated in the open data track for Clean Transcripts for the Chinese-English translation direction.

This section describes the training data we used and the results we achieved. Some discussions follow the results.

[1] Downloadable from http://59.77.17.146/download/.

4.1. Training Data

In addition to the training data provided by IWSLT 2007, we also used other training data.

All the data we used were list in Table 2.

Table 2: Training data list

Purposes	Corpus	
	Names	Amounts
Bilingual Phrases and Reordering Patterns	Training set from IWSLT 2007	177,535 sentence pairs
	Three parts from CLDC-LAC-2003-004: oral.xml, n_train.txt and life_2.xml	
English Language Model	English part of the training set from the 2005 China’s National 863 MT Evaluation	7.4M words
Chinese Chunker	LDC2005T01	18,782 trees

4.2. Results

The scores of our system in IWSLT 2007 is list in Table 3, only the BLEU-4 scores are included.

Table 3: BLEU-4 scores for XMU in IWSLT 2007

	BLEU-4
Baseline + Reordering	0.2888
Baseline	0.2742

We could learn from the scores that, after incorporating the chunk-based reordering model, the phrase-based SMT system could outperform the baseline system.

5. Conclusions

This paper describes the system which participated in the 2007 IWSLT Speech Translation Evaluation of Department of Cognitive Science, Xiamen University. The result shows that after incorporating a chunk-based reordering model, the baseline system may achieve great improvements.

6. Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 60573189), National 863 High-tech Program (Grant No. 2006AA01Z139), Natural Science Foundation of Fujian Province (Grant No.2006J0043) and the Fund of Key Research Project of Fujian Province (Grant No. 2006H0038).

7. References

[1] Koehn, Philipp, Och, Fraz Josef and Marcu Daniel, "Statistical phrase-based translation", *Proceeding of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, Edmonton, Canada, 2003, pp. 127-133.

[2] Och, Franz Josef, "Statistical Machine Translation: From Single Word Models to Alignment Templates", *Ph.D. thesis*, RWTH Adchen, Germany, 2002.

[3] Och, Fraz Josef and Ney, Hermann, "Discriminative training and maximum entropy models for statistical machine translation", *Proceeding of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, PA, 2002, pp. 295-302.

[4] Och, Franz Josef, "Minimum error rate training in statistical machine translation", *Proceeding of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*, Sapporo, Japan, 2003, pp. 160-167.

[5] Zens, Richard, Och, Franz Josef and Ney, Hermann, "Phrase-Based Statistical Machine Translation", *Proceeding of the 25th German Conference 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.

[6] Koehn, Philipp, Axelrod, Amittai, Mayne, Alexandra Birch, Callison-Burch, Chris, Osborne, Miles and Talbot, David, "Edinburgh system description for the 2005 iwslt speech translation evaluation", *Proceeding of International Workshop on Spoken Language Translation*, Pittsburgh, PA, 2005

[7] He, Zhongjun, Liu, Yang, Xiong, Deyi, Hou, Hongxu and Liu, Qun, "ICT System Description for the 2006 TC-STAR Run #2 SLT Evaluation", *Proceeding of the TC-STAR Workshop on Speech-to-Speech Translation*, Barcelona, Spain, 2006, pp. 63-68.

[8] Forney, G. D., "The Viterbi algorithm", *Proceeding of IEEE*, 61(2): 268-278, 1973