The CASIA Phrase-Based Statistical Machine Translation System for IWSLT 2007

Yu Zhou, Yanqing He, and Chengqing Zong

National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, Beijing 100080, China
{yzhou,yqhe,cqzong}@nlpr.ia.ac.cn

Abstract

This paper describes our phrase-based statistical machine translation system (CASIA) used in the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2007. In this year's evaluation, we participated in the open data track of clean text for the Chinese-to-English machine translation. Here, we mainly introduce the overview of the system, the primary modules, the key techniques, and the evaluation results.

1. Introduction

In recent years, statistical machine translation (SMT) method is becoming more and more popular. It achieves good performance for its unique merits and becomes the primary approach for most machine translation systems [1][2]. Our system used in this campaign is the phrase-based SMT system which does some improvements on the system of IWSLT 2006 [3].

The primary modules in the phrase-based system are ameliorated this year to improve the translation result. We deal with the word alignments and adopt a new flexible measure to extract the phrase translation table. We also treat with the name entities especially.

Because our mainly focus is on the open data track of the clean text for the Chinese-to-English translation this year, we employ new approaches for pre-processing and post-processing on the training, development and test data.

This paper is organized as follows: Section 2 describes the data sources and related processing steps on such data. Section 3 presents the overview of CASIA system. In Section 4, the experimental results of our system are reported and the details on analyses of the results are given. Section 5 gives the conclusion.

2. Data

In this section, we mainly describe five processing steps on the data:

- Data collection
- Data preprocessing
- Word alignments
- Phrase extraction and probability calculation
- Language model parameters

After these processing steps, all the training data, the phrase translation table and the language model parameters used in the final decoding process are obtained. Here we will describe each process step in detail.

2.1. Data collection

First of all, we download all the resources including bilingual sentence pairs and bilingual dictionaries for Chinese-English which can be obtained from the website (http://iwslt07.itc.it/menu/resources.html). Here we call such resources as NewCE_train.

Then we extract the new bilingual data which are highly correlative with the Chinese-to-English training data (CE_train) released by IWSLT 2007. We extract the new train data by justifying if all the words in the bilingual data of the NewCE_train are all falling into the CE_train word vocabulary. If the answer is 'yes', we add such bilingual sentence pairs into our CE_train to construct the new training data used in this evaluation campaign.

We use the filtered training data instead of all the free data resources because we have done a series of experiments which prove that only the new added data is highly relative to the CE_train, it can get a better result. If we add all the data arbitrarily without any restriction, it will result in worse output translations because the low relative data may be looked as the noise data in the training process.

2.2. Data preprocessing

For the Chinese part of the training data, three types of preprocessing are performed:

- Segmenting the Chinese characters into Chinese words using the free software toolkit ICTCLAS3.0 (http://www.nlp.org.cn);
- Removing the noises words or characters in the Chinese training data;
- Transforming the SBC case into DBC case;

For the English part of the training data, also three types of preprocessing are performed:

- Tokenization of the English words: which separates the punctuations with the English words;
- Removing the noises words or characters in the English training data;
- Transforming the uppercase into lowercase of the beginning character for the English words according to their statistical frequencies in the English training data.

2.3. Word alignments

Our word alignments are based on the training results of the GIZA++ toolkit (http://www.fjoch.com/GIZA++.html) under the default parameters. We obtain the initial word alignments by the method of grow-diag-final [1] on the bi-directional word alignments of GIZA++. Then we use a dictionary and a

'jumping-distance' method to modify the word alignment results.

Our dictionary is obtained from two aspects: one is from the bilingual dictionary download from the open resources on the web (http://iwslt07.itc.it/ menu/resources.html). The other is obtained from the bi-directional dictionaries generated by GIZA++. For the dictionaries generated by GIZA++, we only extract such word pairs with the highest probabilities as the final bilingual dictionary lists.

Our correcting process is described as follows: for each word pair (f_i, e_j) in bilingual sentence pair, we check them using our bilingual dictionary. Only the first five characters are used to judge whether the two English words is matching. The word pair can be divided into four categories:

(1) For the word pair (f_i, e_j) which is inexistent in the bilingual dictionary but existent in the word alignments of the sentence pair, we observe if the English word e_j is aligned to other Chinese words. If other Chinese words $f_{i'}$ co-occurs with the English word e_j in the bilingual dictionary, the link of the word pair (f_i, e_j) will be deleted. Otherwise, we use "jumping-distance" to decide whether the link of the word pair (f_i, e_j) should be kept. We observe the neighbor right N and left N Chinese words of f_i. If the position of the corresponding English word e_j is falling in the fields of ($\{j_{\min} - M,\ \ j_{\max} + M\}$), the link of ($f_i$, e_j) will be kept. Here, the $j_{\min}$ and $j_{\max}$ are the minimum and maximum index of the English words which the 2*N Chinese words are aligned to. We do the same in the converse direction.

(2) If the word pair (f_i, e_j) is inexistent both in the bilingual dictionary and the word alignments, we will not deal with such case.

(3) If the word pair (f_i, e_j) is existent in the bilingual dictionary but inexistent in word alignments we will add the word pair alignment information.

(4) If the word pair (f_i, e_j) is existent both in the bilingual dictionary and in the word alignments, we will keep the word pair alignment information.

After such process, we go on to treat with the m-1 and 1-m word alignments with the 'jumping-distance' which is similarity with the method described above.

After all the above processes we can get a new word alignments by deleting some wrongly aligned word pair links and adding some correctly aligned word pair links.

2.4. Phrase extraction and probability calculation

Among all the phrase extraction methods, Och's method ([2]) of extracting phrase pairs based on word alignments is widely used in SMT systems. But Och's phrase extraction method only obtains those phrase pairs which are totally consistent with word alignments. For the two aligned phrase ($\tilde{f}$, $\tilde{e}$), all words in $\tilde{f}$ must be aligned to the words inside $\tilde{e}$ and the same in the converse direction. Och's phrase can be defined as equation 1. So in order to overcome its weakness, we propose our method to solve the problem [4][5][6]. Our phrase is shown in equation 2.

$$(\tilde{f}, \tilde{e}) \in BP <=> \begin{cases} \forall f_i \in \tilde{f} : (f_i, e_j) \in A \rightarrow e_j \in \tilde{e} \\ AND \quad \forall e_j \in \tilde{e} : (f_i, e_j) \in A \rightarrow f_i \in \tilde{f} \end{cases} \tag{1}$$

$$(\tilde{f}, \tilde{e}) \in BP <=> \forall f_i \in \tilde{f} : (f_i, e_j) \in A \rightarrow e_j \in \tilde{e} \tag{2}$$

$$AND \begin{cases} \forall e_j \in \tilde{e} : (f_i, e_j) \in A \rightarrow f_i \in \tilde{f} \\ OR \ \dfrac{\{e_j \mid (e_j, f_i) \in A\}}{\{e_j \mid e_j \in \tilde{e}\}} \geq Threshold, \\ e_j \ is \ not \ a \ functional \ word, \\ \rightarrow \arg\max_{\{f_i \mid (f_i, e_j) \in A\}} p(f_i \mid e_j) \in \tilde{f} \end{cases}$$

We will explain the equation 2 in detail as follows: for a given source phrase $\tilde{f}$, we determine the target phrase $\tilde{e} = e_{j_1} \ldots e_{j_2}$ by judging if the target phrase is consistent with word alignments. If the answer is 'yes', we will extract the phrase pair the same way as the Och's method [2]. If the answer is 'no' we will find the set of non-consistent target words in $\tilde{e}$. Its complementary set consists of the target words in $\tilde{e}$ which are aligned inside $\tilde{f}$. Then we judge the situation using our 'flexible scale'. The procedures are described as follows:

- Compute the percentage of consistent target words in $\tilde{e}$. We use a threshold to control the percentage. In Och's method the percentage is fixed 100%. In our method we can predefine the percentage as any value. If the percentage is larger than the threshold, we perform the next procedure. Otherwise we abandon the phrase pair.
- Judge if these non-consistent target words are functional words. Here we consider those English words whose POS (part of speech) are 'DT', 'CC', 'IN', 'MD', 'PDT', 'POS', 'RP', 'TO' and 'UH' as functional words. We use the tags of part of speech defined in [7]. 'DT', 'CC', 'IN', 'MD', 'PDT', 'POS', 'RP', 'TO' and 'UH' denote respectively 'determiner', 'coordinating conjunction', 'preposition or subordinating conjunction', 'modal verb', 'pre-determiner', 'possessive ending', 'particle', 'to' or' interjection'. If the answer of our judge is 'yes' we ignore the alignment information of this functional target word. If the answer is 'no', that means the target word is a non-functional word. Then we go to the next step.
- Check if the source words that the non-consistent and non-functional target word is aligned to are all outside $\tilde{f}$. If the answer is 'yes', we replace the

target word with '#' and extract the target phrase as a non-consecutive phrase pair. If the answer is 'no', there will be some source words in $\tilde{f}$ and some of them outside $\tilde{f}$. Under such condition we may find the source word which the current target word is translated into with the maximum probability $p(f_i \mid e_j)$ in the bilingual dictionary. If the source word with maximum translation probability is outside $\tilde{f}$, we extract $\tilde{e}$ with a non-consecutive form. If the source word is in $\tilde{f}$ we extract $\tilde{f}$ and $\tilde{e}$. Finally we extend the target words beside $\tilde{e}$ which are not aligned to any source word just like Och's method.

Generally speaking, the extracted candidates of phrase pairs contain much redundant information. The number of phrase pairs is too large and greatly increase the search space of decoder. So it is necessary to select the most likely sets of translations. There are four features which are widely used to compute the phrase translation score to discriminate the phrase pairs [1]: phrase translation probability distributions based on frequency (see equation (3) and (4)) and lexical weighting probabilities based on word alignments (see equation (5) and (6)). Here $\tilde{f} = f_{i_1}^{i_2}$ and $\tilde{e} = e_{j_1}^{j_2}$ are respectively the source and target phrase and i_1, i_2, j_1, j_2 are their boundary index. $N(\tilde{f},\tilde{e})$ is the concurrent frequency of the phrase pair $(\tilde{f},\tilde{e})$. a is the word alignments of the phrase pair $(\tilde{f},\tilde{e})$.

$$\phi(\tilde{f} \mid \tilde{e}) = \frac{N(\tilde{f},\tilde{e})}{\sum_{\tilde{f}'} N(\tilde{f}',\tilde{e})} \tag{3}$$

$$\phi(\tilde{e} \mid \tilde{f}) = \frac{N(\tilde{f},\tilde{e})}{\sum_{\tilde{e}'} N(\tilde{f},\tilde{e}')} \tag{4}$$

$$lex(\tilde{f} \mid \tilde{e}, a) = \prod_{i=i_1}^{i_2} \frac{1}{|\{j \mid (i,j) \in a\}|} \sum_{\forall (i,j) \in a} p(f_i \mid e_j) \tag{5}$$

$$lex(\tilde{e} \mid \tilde{f}, a) = \prod_{j=j_1}^{j_2} \frac{1}{|\{i \mid (i,j) \in a\}|} \sum_{\forall (i,j) \in a} p(e_j \mid f_i) \tag{6}$$

2.5. Language model

The data used in the training process for language model is only the English part of the final bilingual training data used in GIZA++. We do not use all the English resources in the website for the computer memory limitation. We use the ngram-count tool in the open SRILM toolkit (http://www.speech.sri.com/projects/srilm) with Kneser-Ney smoothing method [8] to get the final 4-gram language model parameters. Here we only use the 4-gram language model based on the true English words. The features of POS (part-of-speech) and word classes are not combined in the language model.

3. System Overview

This section gives an overview of our system, including the translation model, the search algorithm, the processing with the name entities and the post-processing with the output translation results.

3.1. Phrase-based translation model

In our system, the phrase-based translation model is based on a log-linear model [9]. In the log-linear model, given the sentence *f* (source language), the translating process is searching the translation *e* (target language) with the highest probability. The translation probability and the decision rule are given as formula (7).

$$e^* = \arg\max_{e} \sum_{m=1}^{M} \lambda_m h_m(e, f) \tag{7}$$

Where $h_m(e,f)$ is a feature function and λ_m is the weight of the feature.

In the phrase-based system, we use seven features in the decoding process:

- Phrase translation probability $p(\tilde{e} \mid \tilde{c})$;
- Lexical phrase translation probability $lex(\tilde{e} \mid \tilde{c})$;
- Inversed phrase translation probability $p(\tilde{c} \mid \tilde{e})$;
- Inversed lexical phrase translation probability $lex(\tilde{c} \mid \tilde{e})$;
- English language model based on 4-gram $lm(e_1^I)$;
- English sentence length penalty I;
- Chinese phrase count penalty N.

Here, the entire λ_m are obtained by the minimum error rate training [9][10][11].

3.2. Decoder

In the phrase-based statistical machine translation system, the decoder employs a beam search algorithm that is similar to the Pharaoh decoder [1] and the decoder which is used in IWSLT06 [3]. Our decoder is somewhat different with the Pharaoh decoder: First, we adding the 'expanding F-zerowords' model; second, we use a new tracing back method. Here we only use the monotone search without any distortion model and reordering model.

- **Expanding F-zerowords**

Considering the different expression habits between Chinese and English, some words must be complemented when translating Chinese sentences into English. For example, some frequent words, such as "a, an, of, the", are difficult to extract because those words have zero fertility and correspond to NULL in IBM model 4. We call them F-zerowords. When decoding, the F-zerowords can be added after each new hypothesis, which means, a NULL is added

after each phrase in the source sentences. At the same time, in Chinese sentence there are many auxiliary words and mood words which correspond to NULL in English. We expand the F-zerowords by using two stacks (odd and even stack) instead of one stack. We will use a figure to explain the expanding process in detail.

The decoder starts with an initial hypothesis. There are two kinds of initial hypothesis: one is an empty hypothesis that means no source phrase is translated and no target phrase is generated, and the other one is expanded from the empty hypothesis by adding F-zerowords.

New hypothesis are expanded from the current existing hypotheses as follows: if the last target phrase generated in the existing hypothesis is an F-zerowords, an un-translated source phrase and its translation options are selected to expand the hypothesis. If the last target phrase is not F-zerowords, the hypothesis can be expanded as described above or by selecting one of the F-zerowords. An example of hypotheses expansion is illustrated in Figure 1. The expansion with cross is unallowable because the F-zerowords can not be added after F-zerowords.

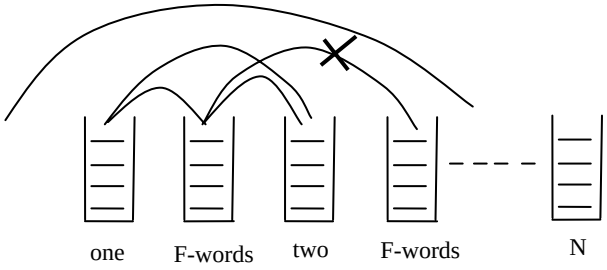

Figure 1: different hypothesis expansion approach

As is shown in Figure 1, the hypotheses are stored in different stacks and each of them has a sequence number. The hypothesis whose last target phrase is not F-zerowords and in which *p* source words have been translated accumulatively will be put into the odd stack S_{2p-1}(p=1,2……). In the same way, if the last target phrase is F-zerowords, the hypothesis will be in the even stack S_{2p}. We recombine the hypotheses and prune out the weak hypotheses that are similar to the Pharaoh decoder. Those operations will reduce the number of hypotheses and speed up the decoding.

- **New tracing back method**

In our decoder, we select the final hypothesis of the best translation in the last several stacks instead of those cover all the source words, because not all the words in source language sentence are necessary to be translated. When all the words of the source sentences have been translated, by searching not in the final stack which covers all the source words but in the final several odd stacks, we find the best translation according to the accumulative score.

3.3. Dealing with the name entities

The test data includes some name entities such as person name, location name, organization name, number and date. If we ignore such name entities, much useful information will be lost. It will result in worse translation result. Aiming at such name entities, we first identify and extract them from the test data [12] and then deal with them individually with their different characters.

- For the person name and location name, we translate them only by looking up its translations in the common phrase pair table which is obtained from the training data on word alignments;
- For the organization name, we translate them using the model based on a synchronous CFG grammar [13];
- For the number and date, we adopt the method based on the man-written rules to translate.

Finally, we add all the name entity translation pairs in the phrase pair table to combine the complete phrase translation table used in the decoding process.

3.4. Post-processing

The post-processing for the output result mainly includes:

- Transforming the lowercase of the first character of the English words into uppercase;
- Recombination the separated punctuations with its left closest English words.

4. Experiment Results

We carried a number of experiments on the Chinese-to-English translation tasks. First, we use the development data to train the parameters of our phrase-based translation model. Then we translate the Chinese test data with the parameters obtained on the development data. We will describe each step in detail and give our analysis on the experiment results.

4.1. Training, development and test data

In section 2.1, we know that more data have been filtered from the LDC resources which are combined with the CE_train as the final training data. Here we give the statistics of the training and development data which shown in table 1.

Table 1: Statistics of training data, development data and test data

data	Chinese	English
CE_train	39,950	39,950
CE_sent_filtered	188,282	188,282
CE_dict_filtered	31,132	31,132
CE_newdev1	24,192	24,192
CE_newdev2	10,423	10,423
CE_test	489	---

Here, CE_train means the Chinese-to-English training data released by IWSLT 2007; CE_sent_filtered means the bilingual sentence pairs filtered from the open resources of the bilingual sentences on the website; CE_dict_filtered means the bilingual dictionary filtered from the open resources of the bilingual dictionaries on the website (here we split the dictionary translation lists into one-to-one aligned bilingual dictionary); CE_newdev1 denotes the bilingual sentence pairs obtained by the combination of the development data IWSLT07_CE_devset1, IWSLT07_CE_devset2 and IWSLT07_CE_devset3 which are released by the IWSLT 2007; CE_newdev2 is the bilingual sentence pairs obtained by the combination of the development data IWSLT07_CE_devset4 and

IWSLT07_CE_devset5 which also are released by IWSLT 2007; CE_test means the final test set released by IWSLT 2007.

We combine the top four row data (CE_train, CE_sent_filtered, CE_dict_filtered and CE_newdev1) as our training set and look the last row data (CE_newdev2) as our development set. For the test data released by IWSLT 2007 is based on the clean text with punctuation information, so we add the punctuation information on the Chinese sentences of IWSLT07_CE_devset4_IWSLT06_C.txt and IWSLT07_ CE_devset5_IWSLT06_C.txt by hand to form the final development set. The detailed statistics are given in Table 2.

After the model parameters are obtained by the training process on our model, we add the last row data (CE_newdev2) into our former training set to form the new training set to obtain the final phrase translation table used to translate the Chinese test set under the trained parameters. The detailed statistics are given in Table 3.

Table 2: Detailed statistics of training data on development set

DEV_train	Chinese	English
Sentences	283,556	283,556
Words	1,754,932	1,900,216
Vocalbulary	11,424	10,507
Average Length	6.2	6.7

Table 3: Detailed statistics of training data on test set

TST_train	Chinese	English
Sentences	293,979	293,979
Words	1,890,984	2,051,619
Vocalbulary	11,661	11,273
Average Length	6.4	7.0

From the table 2 and 3, we may doubt why the average length is so short. This is because we add the CE_dict_filtered in the training data and the average length of the CE_dict_filtered is too short for it is just the word dictionary.

4.2. Analysis of IWSLT 2007 test results

Here we give the test results of IWSLT 2007 shown in Table 4. All the model parameters used are obtained by the minimum error training trained on the DEV_train. Then we get new phrase translation table on the TST_train set and use such model parameters as the configure parameters in the decoder to translate the test set.

As we have mentioned above, we have extracted the name entities from the test set and translated them according to their individual character. In all, we have obtained 116 bilingual name entity lists which are added in the final phrase translation table with all the four probabilities as 1.0.

Table 4: Results of IWSLT 2007 test data

System	BLEU4
Baseline	0.2730
CASIA	0.3648

Baseline means the system with the base methods on word alignments and phrase extraction. The baseline system is only looking the name entities as the common words. CASIA means the system with the new methods described in our paper.

From the translation result shown in table 4, we find that the new methods (word alignments, phrase extraction, name entity identification and translation) are effective in the SMT system. But there still much space for us to polish. First, the word alignments are ameliorated only using the features of the dictionary and the 'jumping-distance'. The two features are not strong enough to support more useful information, so more effective features should be added to improve the word alignments. Second, the new phrase extraction method can obtain more useful phrase translation pairs including the non-consecutive phrase, but the non-consecutive phrase pairs have not added into the decoder due to time limitations. Third, we only use the monotone search in the decoder without any distortion and reordering model.

5. Conclusions

In summary, this paper presents our phrase-based statistical machine translation system in IWSLT 2007 evaluation campaign. We use several new approaches in this year's campaign: word alignments, phrase extraction, name entity identification and translation. The translation result proves that the new methods are effective in the SMT system. But the system is still in the preliminary stage for we only use the basic method of phrase-based statistical machine translation method. There are much more space for us to ameliorate such as adding the semantic information into our model, putting non-consecutive phrase pair into our decoder, adding the reorder model into our decoder, re-ranking the N-best of the decoder and combining with other translation systems.

6. Acknowledgements

The research work described in this paper has been funded by the Natural Science Foundation of China under Grant No. 60575043, National Hi-Tech. Program (863) under Grant No. 2006AA01Z194, National Key Technology R&D Program under Grant No. 2006BAH03B02, and Nokia (China) Co. Ltd as well.

7. References

[1] Koehn Philipp. 2004. Pharaoh: a Beam Search Decoder for Phrase-based Statistical Machine Translation Models. In Proceedings of the 6th Conference of the Association for Machine Translation in the Americas, pages: 115-124. (http://www.isi.edu/licensed-sw/pharaoh).

[2] Franz Josef Och, Hermann Ney. 2004. The alignment template approach to statistical machine translation. Computational Linguistics, 1 June 2004.

[3] Chunguang Chai, Jinhua Du, Wei Wei, Peng Liu , Keyan Zhou, Yanqing He, Chengqing Zong. NLPR Translation System for IWSLT 2006 Evaluation Campaign. International Workshop on Spoken Language Translation (IWSLT2006),Novermber 27-28, 2006, Kyoto, Japan. Pages:91-94.

[4] Yuncun Zuo, Yu Zhou, Chengqing Zong. Multi-Engine Based Chinese-to-English Translation System.

International Workshop on Spoken Language Translation (IWSLT2004), September 30-October 1,Kyoto, Japan. Pages:73-77.

[5] Yu Zhou, Chengqing Zong, and Bo Xu. Multi-layer Filtering Based Statistical Machine Translation (in Chinese). The Journal of Chinese Information Processing,Beijing, 19(3), pages 54-59, 2005.

[6] Yanqing He, Yu Zhou, and Chengqing Zong. Flexible-Scale Based Phrase Translation Extraction (in Chinese). The 9th National workshop of JSCL-2007 in Dalian.

[7] Beatrice Santorini. 1990. Part-of-Speech Tagging Guidelines for the Penn Treebank Project, Technical report MS-CIS-90-47, Department of Computer and Information Science, University of Pennsylvania, 1990

[8] Kneser, Reinhard and Hermann Ney, 1995. Improved backing-off for m-gram language modeling. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, volume 1, pages 181-184.

[9] Och, Franz Josef, Hermann Ney. 2002. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. July 2002. Pages: 295-302.

[10] Och, Franz Josef. 2003. Minimum Error Rate Training in Statistical Machine Translation. In Proceedings of the 41st Annual Conference of the Association for Computational Linguistics (ACL). July 8-10, 2003. Sapporo, Japan. Pages: 160-167.

[11] Ashish Venugopal, Stephan Vogel. Considerations in Maximum Mutual Information and Minimum Classification Error training for Statistical Machine Translation. In the Proceedings of the Tenth Conference of the European Association for Machine Translation (EAMT-05), Budapest, Hungary May 30-31, 2005.

[12] Youzheng Wu, Jun Zhao, Bo Xu, Chinese Named Entity Recognition Model Based on Multiple Features. In Proceedings of HLT/EMNLP 2005, pages: 427~434, October 6-8, Vancouver, B.C., Canada.

[13] Chiang, David. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In Proceedings of the 43rd Annual Meeting of the ACL, pages 263-270.