Integrating Multi-source Bilingual Information for Chinese Word Segmentation in Statistical Machine Translation

Wei Chen, Wei Wei, Zhenbiao Chen, and Bo Xu

Interactive Digital Media Technology Research Center(IDMTech)
Institute of Automation, Chinese Academy of Sciences

Abstract. Chinese texts are written without spaces between the words, which is problematic for Chinese-English statistical machine translation (SMT). The most widely used approach in existing SMT systems is apply a fixed segmentations produced by the off-the-shelf Chinese word segmentation (CWS) systems to train the standard translation model. Such approach is sub-optimal and unsuitable for SMT systems. We propose a joint model to integrate the multi-source bilingual information to optimize the segmentations in SMT. We also propose an unsupervised algorithm to improve the quality of the joint model iteratively. Experiments show that our method improve both segmentation and translation performance in different data environment.

Keywords: Chinese segmentation, bilingual information, statistical machine translation.

1 Introduction

Different from most of the western languages, Chinese sentences are written without any spaces between the words. Word segmentation is therefore one of the most important steps of Chinese natural language processing tasks, such as statistical machine translation (SMT).

[1] showed that SMT system worked much better by segmenting the text into words than those treating each character as one "word". While it is difficult to define what is a "correct" Chinese word segmentation (CWS), a generally accepted point is that the definition of "correct" segmentation should vary with different tasks. For example, Chinese information retrieval systems call for a segmentation that generates shorter words, while automatic speech recognition benefits from having longer words. However, it is difficult to define and poorly understood what is a satisfactory segmentation for SMT systems. [2] and [3] showed that the F-score, which is used generally to measure the performance of a segmentation on monolingual corpus, had nothing to do with the effect of the segmentation on SMT systems as a very high F-score may produce rather poor quality translations.

In spite of this, the common approach in most SMT systems has been to use an off-the-shelf monolingual CWS method. For instance, [4] proposed the N-gram generative language modeling based approach. [5] used the hierarchical hidden Markov Model

M. Sun et al. (Eds.): CCL and NLP-NABD 2013, LNAI 8202, pp. 61–72, 2013.

(HHMM) based method. [6] applied a sliding-window maximum entropy classifier to take CWS as a task of character tagging. Then [7] used Linear-chain conditional random fields (CRFs) [8] instead to take on the role of classifier and got a better result.

By the different existing methods, the fixed segmentations are applied in translation model training process even if they are sub-optimal and raise a series of problems as follows:

– Firstly, the specifications of monolingual CWS systems are not suitable for SMT. What' s more, the "best" specification in the bilingual corpus may differ from sentence to sentence (see Figure 1(a)), and it' s difficult to find it only through monolingual CWS method.

– Secondly, monolingual CWS methods often make a large number of mistakes on out-of-vocabulary(OOV) segmentation especially named entity(NE) segmentation (see Figure 1(b)).

– Thirdly, monolingual CWS methods are not good at dealing with the ambiguities in the Chinese text and will segment randomly because each segmentation is "right" (see Figure 1(c)).

Every problem can cause a chain mistake in the SMT system. Even so, it is poorly studied how to optimize the CWS in the machine translation system. [2] proposed two approaches to combine multiple word segmentations. [3] showed that neither character-level segmentation granularity nor Chinese-Treebank-style segmentation granularity

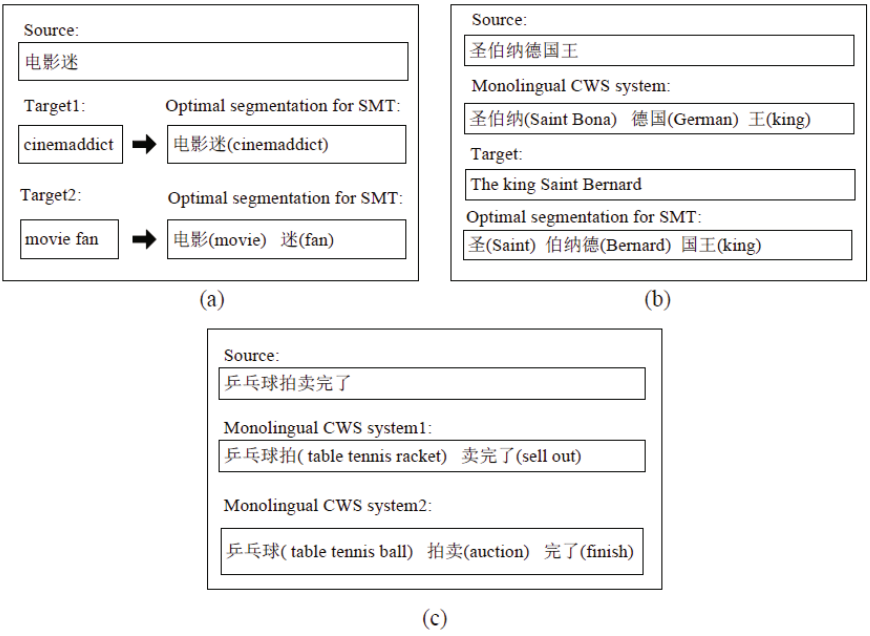


Fig. 1. Optimal segmentations in different situations using bilingual information

was suitable for SMT systems and it introduced a new feature to shorten the average word length produced by its CRF segmenter. However, the optimization in these papers is based on monolingual information and still keeps the problems above. [9] described a generative model which consisted of a unigram language model and an alignment model of both directions. Then it treated the word segmentation as a Hidden Markov Modeling problem of inserting and deleting spaces with the initial segmentations. But the approach suffers from the problems of local optimum because of the lack of linguistic specifications which introduces some mistaken alternatives. Furthermore, it couldn't address the issues of monolingual CWS systems only by the information of word alignment and called for multi-source information to be integrated.

To address the problems caused by monolingual CWS system and [9], we propose a joint translation model to integrate multi-source bilingual information into monolingual CWS methods to address the issues above. Firstly we apply a word-based translation model to rescore the alternative segmentations. We get the alternative set by the combination of CRF-based CWS system and N-gram language model based CWS system and rescore them by the way of cross-validation. Secondly, we take use of a phrase-based named entity (NE) transliteration model to integrate the information of bilingual NE into the model. Thirdly, we employ an English-Chinese dictionary and a Chinese synonym dictionary to make the model more accurate and effective. Finally, we propose an algorithm to improve the segmentation iteratively.

Our experiments show that the approach can generate a more satisfactory and correct segmentation for SMT systems and is very effective in improving the performance of machine translations.

2 Producing the Set of Alternative

2.1 Previous Work on Monolingual CWS

CRF-Based Model for CWS. CRF is an undirected graphical model trained to maximize a conditional probability [8] and is first used for CWS task by [10], which treats CWS task as a sequence tagging question. For instance, Chinese characters that begin a new word are given the START tag and Characters in the end of the words are given END tag.CRF-based model overcomes the problem of marking bias in generative models but has a shortage of prone to generate much longer word than other methods, which is harmful to SMT because it causes data sparseness.

N-gram Language Model for CWS. N-gram language model based method [4] treats CWS task as a hidden Markov modeling problem of inserting spaces into text. It defines two states between every pair of the characters of Chinese text: have a space or don' t have a space between the pair of characters. N-gram language model has much weaker ability of recognizing OOV word than CRF-based model but it generates significant shorter words than CRF-based model, which meets our demand greatly.

2.2 Combination of CWS Systems

In order to produce the set of alternative effectively and accurately, we propose an approach to combine the two models above. First, we are given a Chinese sentence

$c_1^K = (c_1, c_2, , c_K)$, where c_k indicates the character k in the sentence. Then, we get two segmentations by CWS models above: $f_{1CRF}^J = (f_1, f_2, , f_J)$ produced by CRF-based model and $f_{1N-gram}^J = (f_1, f_2, , f_J)$ produced by N-gram language model. In the sentence, we call a character as a word boundary when it is the ending (not the beginning) of a word in one of the segmentations. According to the description, we define four states of a character as follows:

(1) C_{k+}^S indicates the character k is a word boundary both in f_{1CRF}^J and $f_{1N-gram}^J$.
(2) C_{k-}^S indicates the character k is not a word boundary either in f_{1CRF}^J or $f_{1N-gram}^J$.
(3) C_{k+}^D indicates the character k is a word boundary in f_{1CRF}^J while not a word boundary in $f_{1N-gram}^J$.
(4) C_{k-}^D indicates the character k is a word boundary in $f_{1N-gram}^J$ while not a word boundary in f_{1CRF}^J.

Finally, we can describe our approach that produces the alternative segmentations as follows:

(1) every C_{k-}^D between two adjacent C_{k+}^S in f_{1CRF}^J can be converted to C_{k-}^S or keep the original state in the sentence (see Figure 2(a)).
(2) every C_{k+}^D between two adjacent C_{k+}^S in f_{1CRF}^J can be converted to C_{k+}^S or keep the original state in the sentence (see Figure 2(b)).

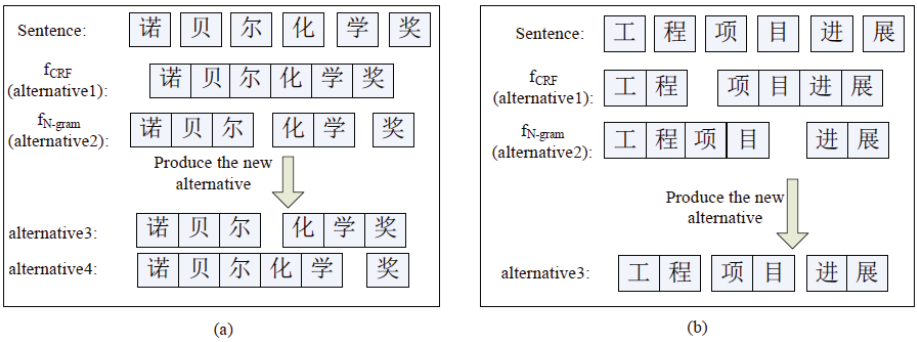


Fig. 2. Produce new alternatives in situation (1) and (2)

Then we can get the set of alternatives segmentations $set(f_1^J) = f_{1(1)}^J, f_{1(2)}^J, , f_{1(L)}^J$ by combining each character's possible states and the set of alternatives in Figure 2(a) can be described as a graph (see Figure 3).

Each path that goes through the graph from left to right indicates an alternative segmentation and each alternative segmentation will be given a fixed value as their monolingual segmentation probability for the next process.

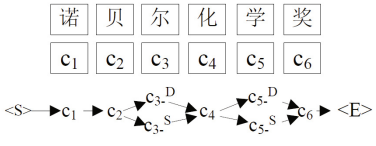


Fig. 3. The graph of the set of alternatives in Figure 2(a)

3 Joint Translation Model for Integrating Multi-source Bilingual Information

3.1 Word-Based Translation Model

For each parallel sentence (c_1^K, e_1^I) in the corpus, the observations are Chinese text c_1^K and English text e_1^I, and the hidden variable is the word segmentation f_1^J. In tranditional SMT systems, we use monolingual CWS methods to select a "best" segmentation by assuming that the probability of the segmentation is conditional independent with the English text as follows:

$$f_1^J best = \arg\max_{f_1^J} p(f_1^J|c_1^K, e_1^I) = \arg\max_{f_1^J} p(f_1^J|c_1^K)$$

however, it is proved by [9] that the assumption is harmful to the translation performance. Ignoring the assumption, we can select the "best" segmentation by the bilingual CWS probability as follows:

$$f_1^J best = \arg\max_{f_1^J} p(f_1^J|c_1^K, e_1^I) = \arg\max_{f_1^J} \frac{p(e_1^I|f_1^J, c_1^K) * p(f_1^J, c_1^K)}{p(c_1^K) * p(e_1^I)}$$

where the c_1^K in $p(e_1^I|f_1^J, c_1^K)$ can be dropped because the c_1^K is fixed given the f_1^J.

It indicates the bilingual CWS probability of each alternative segmentation is determined both by the monolingual CWS probability $p(f_1^J, c_1^K)$ and the translation probability $p(e_1^I|f_1^J)$ and the probability $p(f_1^J, c_1^K)$ of each alternative segmentation are set to a fixed value as mentioned above, it can be ignored and the bilingual CWS probability thus is

$$p(f_1^J|c_1^K, e_1^I) \propto p(e_1^I|f_1^J)$$

for each alternative segmentation f_1^J, we compute the translation probability $p(e_1^I|f_1^J)$ with our word-based translation model. Considering the computing complexity, we take use of IBM model-1 in the process. As we can' t obtain the fixed alignment of each alternative, we take every possible alignment into account. Then the translation probability is derived by

$$p(e_1^I|f_1^J) = \sum_a \mathrm{P}(e_1^I, \mathrm{a}|f_1^J) = \frac{\varepsilon}{(J+1)^I} \prod_{i=1}^{I} \sum_{j=0}^{J} t(ei|fj) \tag{1}$$

where ε is the normalization factor to make all alternative segmentations' probability sum to one and "I" indicates the number of the words of English sentence e_1^I while "J" indicates the number of the words of the alternative segmentation f_1^J. $t(e_j|f_i)$ is the translation probability from Chinese word f_j to English word e_i which is given by our word-based translation model.

To avoid the problem of over-fitting, we introduce the thought of cross validation in the process of computing translation probability. That is, we compute the translation probability of sentence pair i $(c_1^K, e_1^I)^i$ through the translation model which is trained on the corpus of the other sentence pairs without the current sentence pair $(c_1^K, e_1^I)^i$. Considering efficiency, we divide the corpus into two subsets and compute the probability of one using the translation model trained on the other subset.

3.2 English-Chinese Phrase-Based Named Entity Transliteration Model

As we mentioned in Section 1, it is really difficult for monolingual CWS methods to segment the proper names or technical terms which are defined as named entity (NE) correctly and suitably. As many different words can be the transliteration of the same English named entity since they pronounce in the same way, it causes a big problem of data sparseness, which can' t be solved by the translation model in Section 3.1.

In this section, we propose a phrase-based named entity transliteration model to fill the gap.

Firstly, we get the transliteration model using an initial NE dictionary. We convert each named entity word pair (e_i, f_j) to a "sentence pair" (l_1^Y, c_1^X) by splitting e_i by letters and f_j by characters and train a standard English-Chinese phrase-based transliteration model using the open source translation system moses.

Given the transliteration model and English word e_i, we convert the word e_i into an English "sentence" l_1^Y and derive the best transliteration of it as:

$$c_{1best}^X = \arg\max_{c_1^{X\prime}} \prod_{x=1}^{X} \phi(\bar{c_x}|\bar{l_x}) d(start_x - end_{x-1} - 1) * \prod_{x=1}^{|c|} p_{LM}(c_x|c_1 ... c_{x-1})$$

where $\phi(\bar{c_x}|\bar{l_x})$ indicates the phrase translation probability of the phrase pair $(\bar{c_x}, \bar{l_x})$. As an English named entity is generally transliterated from left to right, we don' t need to reorder the translation and the value of reordering feature $d(start_x - end_{x-1} - 1)$ is fixed to $d(0)$. What' s more, as we mentioned above, many different words pronounce in the same way. It doesn' t matter which character is chosen and each will be a "correct" transliteration of the English word ei. So the value of language model feature is set to a fixed value, too. Then the best transliteration of the "sentence" l_1^Y is derive as follows:

$$c_{1best}^X = \arg\max_{c_1^{X\prime}} \prod_{x=1}^{X} \phi(\bar{c_x}|\bar{l_x})$$

For each word pair (f_j, e_i) in the alternative segmentation and the corresponding parallel English sentence, we can integrate the feature of transliteration into the translation probability $p(e_1^I|f_1^J)$ in Formula (1). Then the probability that rescore the alternative segmentations is given by:

$$p(e_1^I|f_1^J) = \frac{\varepsilon}{(J+1)^I} \prod_{i=1}^{I} \sum_{j=0}^{J} [\lambda_1 t(ei|fj) + \lambda_2 f_{NE}(fj, c_{1best}^X)] \quad (2)$$

where λ_1 and λ_2 indicate the weights of word translation feature and named entity transliteration feature. The function of named entity transliteration feature $f_{NE}(f_j, c_{1best}^X)$ is given by:

$$f_{NE}(f_j, c_{1best}^X) = \begin{cases} 1 & \text{if the pinyin (pronunciation) of } f_j \text{ and } c_{1best}^X \text{ is the same} \\ 0 & \text{if the pinyin (pronunciation) of } f_j \text{ and } c_{1best}^X \text{ is different} \end{cases}$$

and the c_{1best}^X is given above.

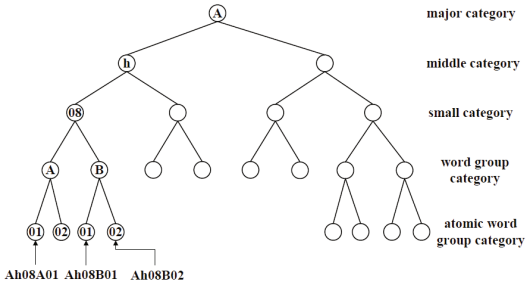
3.3 Integrating the Information of Dictionary

In order to promote the joint model to be more accurate, we put forward a dictionary-based model in this Section.

Firstly, we propose an English-Chinese translation dictionary $(e_i, T_i)^N$ where N indicates the number of items in the dictionary. Each item consists of an English word e_i and a set of the word' s translations T_i.

However, it' s impossible to collect all of the possible translations of English word e_i in the set T_i. What' s more, it' s common that replace the translation of English word with a synonym which may have a little difference in meaning with the English word.

To address the issue, we propose a dictionary of Chinese synonyms to compute the similarity of two Chinese words. The dictionary has five category' s levels and every word is given one or more codes to indicate the categories of the word. The words given the same code have the almost same meaning. Based on the tree, we define the



semantic distance of two codes $SemDist(S_1, S_2)$ as the shortest distance from the point S_1 to point S_2 in the tree. For example, $SemDist(Ah08B01, Ah08B02) = 2$, $SemDist(Ah08B01, Ah08A01) = 4$. Then we define the similarity of two codes $SemSim(S_1, S_2)$ as follows:

$$SemSim(S_1, S_2) = \begin{cases} 1/SemDist(S_1, S_2) & \text{if } S_1 \neq S_2 \\ 0 & \text{if } S_1 = S_2 \end{cases}$$

The feature function of word pair (e_j, f_i) and the similarity of two words is therefore defined by

$$f_{DICT}(f_j, e_i) = \max_{\substack{W_1=f_j \\ W_2 \in T_i}} \begin{cases} 1 & \text{if } W_1 = W_2 \\ \max\limits_{\substack{S_m \in categoryOf(W_1) \\ S_n \in categoryOf(W_2)}} SemSim(S_m, S_n) & \text{if } W_1 \neq W_2 \end{cases}$$

where the function $categoryOf(W_1)$ return the set of codes of word W_1. Finally, we extend the translation model described in Section 4.2 to

$$p(e_1^I | f_1^J) = \frac{\varepsilon}{(J+1)^I} \prod_{i=1}^{I} \sum_{j=0}^{J} [\lambda_1 t(ei|fj) + \lambda_2 f_{NE}(fj, \mathrm{c}_{1best}^X) + \lambda_3 f_{DICT}(fj, ei)] \tag{3}$$

where the λ_1, λ_2 and λ_3 indicate the weights of word-based translation model, named entity transliteration model and dictionary-based model.

4 Iterative Algorithm

In this Section, as the algorithm showed in Algorithm 1, we present an iterative process to optimize the joint model and our segmentation in an unsupervised way.

In each iteration, we optimize the segmentation by the joint model, and then update the word-based translation model $M_{trans}(1)$,$M_{trans}(2)$ and NE transliteration model M_{NE} by the optimized segmentations and the new NE dictionary D_{NE}.

The iteration will be stopped if the number of different segmentations of last iteration and current iteration is lower than a threshold h.

5 Experiment Setup

5.1 Data Set and Evaluation

We take the IWSLT machine translation task [11] for our experiment and our model is evaluated on the data track from two aspects: the segmentation performance on the training data set and the final translation performance on evaluation data set. The bilingual training corpus is a superset of corpora in the multi-domain collected from different sources including the training data of IWSLT task.

5.2 Baseline System and Translation System

We take CRF-based CWS method [7] as a baseline CWS method.

In order to highlight the translation performance, we use an out-of-the-box Moses[1] *(2010-8-13 version).* framework using GIZA++ [12] and minimum error rate training [13] to train and tune the feature weights of SMT systems. GIZA++ is used to get alignments from the bilingual training corpus with *grow-diag-final-and* option. The 4-gram LM is estimated by the SRILM toolkit [14] with interpolated modified Kneser-Ney discounting. We use the Moses decoder to produce all the system outputs, and score them with the BLEU-4 [15] score.

[1] `http://www.statmt.org/moses/index.php?n=Main.HomePage`

Algorithm 1. Iterative joint model training

Input:

Bilingual corpus $(c_1^K, e_1^I)^n$, initial NE dictionary D_{NE}, English-Chinese dictionary D_{E2C},Chinese synonyms dictionary D_{Syn}

Output:

optimized segmented bilingual corpus $(f_{1opt}^J, e_1^I)^n$

divide the corpus into two subsets $(c_1^K, e_1^I)^{1..m}$,$(c_1^K, e_1^I)^{m+1..n}$
get initial segmentations for each subset $(f_{CRF}, e_1^I)^{1..m}$ $(f_{N-gram}, e_1^I)^{1..m}$ and $(f_{CRF}, e_1^I)^{m+1..n}$ $(f_{N-gram}, e_1^I)^{m+1..n}$
train initial word-based translation model $M_{trans}(1)$ for the first subset and $M_{trans}(2)$ for the other
train initial NE transliteration model M_{NE} on D_{NE}
get the set of alternative segmentations $set_i(f_1^J)$ for each sentence pair i $(c_1^K, e_1^I)_i$
repeat
$(f_{current}, e_1^I)^n \leftarrow (f_{opt}, e_1^I)^n, (f_{1opt}^J, e_1^I)^n \leftarrow \phi$
for each sentence pair $(c_1^K, e_1^I)_i \in (c_1^K, e_1^I)^n$ **do**
    for each alternative segmentation $f_1^J \in set_i(f_1^J)$ **do**
        for each word pair (f_i, e_j) in the (f_1^J, e_1^I) **do**
            compute $t(e_j|f_i)$ by the M_{trans} that is trained on the other subset
            compute $f_{NE}(f_i, c_{1best}^I)$ by the M_{NE}
            compute $f_{DICT}(f_i, e_j)$ by the D_{E2C} and D_{Syn}
            add the word pair (f_i, e_j) to D_{NE} if $f_{NE}(f_i, c_{1best}^I) \neq 0$
        end for
        compute the score of f_1^J by the joint model
    end for
    select $f_{1best}^J \in set_i(f_1^J)$ with the highest score
    add (f_{1best}^J, e_1^I) to $(f_{1opt}^J, e_1^I)^n$
end for
retrain $M_{trans}(1)$,$M_{trans}(2)$,M_{NE} by $(f_{1opt}^J, e_1^I)^n$ and D_{NE}
until the number of different segmentations between $f_{current}$ and f_{opt} is lower than h
return f_{opt}

6 Experiment

6.1 Segmentation Performance on Training Data Set

Firstly, we compare our model on the segmentation performance with currently widely-used monolingual CWS methods.

As we mentioned above, F-score can' t measure the segmentations effectively in SMT systems and the CWS are related to SMT by a series of factors such as the specifications, OOVs, lexicons. None of these factors can be directly related to the SMT. Therefore, we compare our method with others in multiple factors on the training data as shown in Table 1.Considering computational complexity, we perform our method using only one iteration.

We can see that the number of running words generated by our method is close to the others. However, it produces a much smaller vocabulary than CRF and ICT [5] methods while keeps a high rate of unique words, which means that our method not only avoid

Table 1. Segmentation performance with different CWS methods on the training data

Method	Sents.	Tokens [M]	Voc. [K]	Unique Words[K]
ICT.	2M	18.80	214.1	41.0
CRF(base)		18.47	214.2	114.6
Our.		18.63	**133.1**	**50.2**

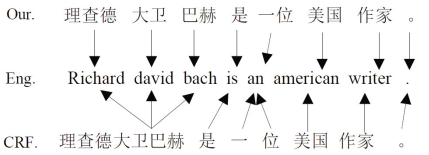


Fig. 4. Segmentations outputs with baseline and our method

data sparseness by shortening the common words, but also recognize the OOVs more accurate as the example shown in Figure 4.

6.2 Translation Performance on Task IWSLT

Then, we evaluate our method for word segmentation on the IWSLT machine translation task. The bilingual training corpus includes the training data of task and other corpus in the multi-domain collected from different sources. We take the open-source translation system moses in the evaluation and use the evaluation corpus of (IWSLT 2005) [16] to optimize the model weights of moses. Finally, we take the evaluation corpus of (IWSLT 2007) [11] to evaluate the translation performance.

For a fair comparison, we evaluate on various CWS methods including ICTCLAS [5], CRF-based method [7], N-gram language model based method [4], GS [9] and our method as shown in Table 2.

Furthermore, we replace the monolingual CWS methods, i.e., CRF-based method and N-gram language model based method, with another two monolingual methods, and then integrate parts of our joint model or our full model into them to evaluate the translation performance using the same evaluation corpus as above. The results are shown in Table 3.

Table 2. Translation performance with different CWS methods on IWSLT 2007[% BLEU]

Method	ICT.	CRF(base)	N-gram	GS	Our.
BLEU	39.62	39.25	38.22	39.99	**41.23**

It can be seen that each part of our joint model can improve the translation performance effectively. It also can be found that even if ICT system has a better translation performance than N-gram, it obvious that N-gram method are more adaptive to combine with CRF method using the joint model because N-gram is prone to segment the OOVs into characters and thus is fit for our method.

Table 3. Translation performance with integrating the joint model to another CWS methods[% BLEU]: A = word-based translation model, B = phrase-based NE transliteration model, C = dictionary-based model

Joint model	ICT.	CRF	N-gram	CRF
monolingual	39.62	39.25	38.22	39.25
+A	40.26		40.45	
+A+B	40.62		40.96	
+A+B+C	40.94		41.23	

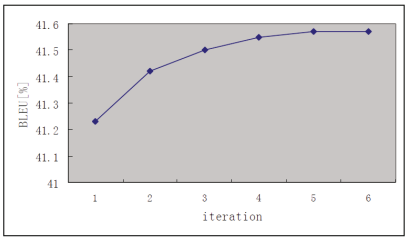


Fig. 5. Translation performance with the joint model of each iteration[% BLEU]

As our joint model can be optimized iteratively, we use 6 iterations (using N-gram and CRF methods) over the training corpus and evaluate the translation performance for each iteration as shown in Figure 5. We get the final BLEU at iteration 6 in Figure 5 is 41.58.

We compare the translation outputs using our method with the baseline method and list two examples in Table 4.

Table 4. Translation outputs with baseline and our methods

	Example1	Example2
Eval	咖啡还没有上来。	我也是啊。他们真的很棒啊。
Base	not coffee .	they also is . I really wonderful .
Our.	coffee hasn't come yet .	me too . they are really wonderful .
REF	my coffee hasn't come yet .	me , too . They play really well .

7 Conclusion and Future Work

In this paper, we showed that it is effective to improve the performance of SMT system by introducing multi-source bilingual information to CWS system.We proposed a joint model and an iterative algorithm and our experiments showed that our method outperformed the other CWS approaches in terms of not only the word segmentation performance but also the translation quality. It is also proved that each sub-model of our joint model is effective and the iterative algorithm works well. In future work, we plan to make our joint model more accurate to select the segmentation for SMT system better.

References

[1] Xu, J., Zens, R., Ney, H.: Do we need Chinese word segmentation for statistical machine translation. In: Proc. of the Third SIGHAN Workshop on Chinese Language Learning, Barcelona, Spain (2004)
[2] Zhang, R., Yasuda, K., Sumita, E.: Improved Statistical Machine Translation by Multiple Chinese Word Segmentation. In: Proceedings of the Third Workshop on Statistical Machine Translation, pp. 216–223 (2008)
[3] Chang, P.-C., Galley, M., Manning, C.D.: Optimizing Chinese Word Segmentation for Machine Translation Performance. In: Proceedings of the Third Workshop on Statistical Machine Translation, pp. 224–232 (2008)
[4] Teahan, W.J., Wen, Y., McNab, R., Witten, I.H.: A Compression-based Algorithm for Chinese Word Segmentation. Computational Linguistics 26(3), 375–393 (2000)
[5] Zhang, H.-P., Yu, H.-K., Xiong, D.-Y., Liu, Q.: HHMM-based Chinese lexical analyzer ICTCLAS. In: Proceedings of the Second SIGHAN Workshop on Chinese Language Learning, pp. 184–187 (2003)
[6] Xue, N.: Chinese Word Segmentation as Character Tagging. Computational Linguistics and Chinese Language Processing 8(1), 29–48 (2003)
[7] Tseng, H., Chang, P., Andrew, G., Jurafsky, D., Manning, C.D.: A conditional random field word segmenter for Sighan bakeoff 2005. In: Proc. of the Fourth SIGHAN Workshop on Chinese Language Processing (2005)
[8] Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc. 18th International Conf. on Machine Learning (2001)
[9] Xu, J., Gao, J., Toutanova, K., Ney, H.: Bayesian Semi-Supervised Chinese Word Segmentation for Statistical Machine Translation. In: Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pp. 1017–1024 (2008)
[10] Peng, F., Feng, F., McCallum, A.: Chinese segmentation and new word detection using conditional random fields. In: Proceedings of the 20th International Conference on Computational Linguistics, p. 562 (2004)
[11] IWSLT: International workshop on spoken language translation home page (2007), http://www.slt.atr.jp/IWSLT2007
[12] Och, F.J., Ney, H.: Improved statistical alignment models. In: Proceedings of ACL, pp. 440–447 (2000)
[13] Och, F.J.: Minimum error rate training in statistical machine translation. In: Proceedings of ACL, pp. 160–167 (2003)
[14] Stolcke, A.: SRILM - An extensible language modeling toolkit. In: Proceedings of ICSLP, pp. 901–904 (2002)
[15] Papineni, K., Roukos, S., Ward, T., Zhu, W.: BLEU: A method for automatic evaluation of machine translation. In: Proceedings of ACL, pp. 311–318 (2002)
[16] IWSLT: International workshop on spoken language translation home page (2005), http://www.slt.atr.jp/IWSLT2005