Pseudo-word for Phrase-based Machine Translation

Xiangyu Duan **Min Zhang** **Haizhou Li**
Institute for Infocomm Research, A-STAR, Singapore
{Xduan, mzhang, hli}@i2r.a-star.edu.sg

Abstract

The pipeline of most Phrase-Based Statistical Machine Translation (PB-SMT) systems starts from automatically word aligned parallel corpus. But word appears to be too fine-grained in some cases such as non-compositional phrasal equivalences, where no clear word alignments exist. Using words as inputs to PB-SMT pipeline has inborn deficiency. This paper proposes pseudo-word as a new start point for PB-SMT pipeline. Pseudo-word is a kind of basic multi-word expression that characterizes minimal sequence of consecutive words in sense of translation. By casting pseudo-word searching problem into a parsing framework, we search for pseudo-words in a monolingual way and a bilingual synchronous way. Experiments show that pseudo-word significantly outperforms word for PB-SMT model in both travel translation domain and news translation domain.

1 Introduction

The pipeline of most Phrase-Based Statistical Machine Translation (PB-SMT) systems starts from automatically word aligned parallel corpus generated from word-based models (Brown et al., 1993), proceeds with step of induction of phrase table (Koehn et al., 2003) or synchronous grammar (Chiang, 2007) and with model weights tuning step. Words are taken as inputs to PB-SMT at the very beginning of the pipeline. But there is a deficiency in such manner that word is too fine-grained in some cases such as non-compositional phrasal equivalences, where clear word alignments do not exist. For example in Chinese-to-English translation, "想" and "would like to" constitute a *1*-to-*n* phrasal equivalence, "多少钱" and "how much is it" constitute a *m*-to-*n* phrasal equivalence. No clear word alignments are there in such phrasal equivalences. Moreover, should basic translational unit be word or coarse-grained multi-word is an open problem for optimizing SMT models.

Some researchers have explored coarse-grained translational unit for machine translation. Marcu and Wong (2002) attempted to directly learn phrasal alignments instead of word alignments. But computational complexity is prohibitively high for the exponentially large number of decompositions of a sentence pair into phrase pairs. Cherry and Lin (2007) and Zhang et al. (2008) used synchronous ITG (Wu, 1997) and constraints to find non-compositional phrasal equivalences, but they suffered from intractable estimation problem. Blunsom et al. (2008; 2009) induced phrasal synchronous grammar, which aimed at finding hierarchical phrasal equivalences.

Another direction of questioning word as basic translational unit is to directly question word segmentation on languages where word boundaries are not orthographically marked. In Chinese-to-English translation task where Chinese word boundaries are not marked, Xu et al. (2004) used word aligner to build a Chinese dictionary to resegment Chinese sentence. Xu et al. (2008) used a Bayesian semi-supervised method that combines Chinese word segmentation model and Chinese-to-English translation model to derive a Chinese segmentation suitable for machine translation. There are also researches focusing on the impact of various segmentation tools on machine translation (Ma et al. 2007; Chang et al. 2008; Zhang et al. 2008). Since there are many *1*-to-*n* phrasal equivalences in Chinese-to-English translation (Ma and Way. 2009), only focusing on Chinese word as basic translational unit is not adequate to model *1*-to-*n* translations. Ma and Way (2009) tackle this problem by using word aligner to bootstrap bilingual segmentation suitable for machine translation. Lambert and Banchs (2005) detect bilingual multi-word ex-

Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 148–156,
Uppsala, Sweden, 11-16 July 2010. ©2010 Association for Computational Linguistics

pressions by monotonically segmenting a given Spanish-English sentence pair into bilingual units, where word aligner is also used.

IBM model 3, 4, 5 (Brown et al., 1993) and Deng and Byrne (2005) are another kind of related works that allow *1*-to-*n* alignments, but they rarely questioned if such alignments exist in word units level, that is, they rarely questioned word as basic translational unit. Moreover, *m*-to-*n* alignments were not modeled.

This paper focuses on determining the basic translational units on both language sides without using word aligner before feeding them into PB-SMT pipeline. We call such basic translational unit as pseudo-word to differentiate with word. Pseudo-word is a kind of multi-word expression (includes both unary word and multi-word). Pseudo-word searching problem is the same to decomposition of a given sentence into pseudo-words. We assume that such decomposition is in the Gibbs distribution. We use a measurement, which characterizes pseudo-word as minimal sequence of consecutive words in sense of translation, as potential function in Gibbs distribution. Note that the number of decomposition of one sentence into pseudo-words grows exponentially with sentence length. By fitting decomposition problem into parsing framework, we can find optimal pseudo-word sequence in polynomial time. Then we feed pseudo-words into PB-SMT pipeline, and find that pseudo-words as basic translational units improve translation performance over words as basic translational units. Further experiments of removing the power of higher order language model and longer max phrase length, which are inherent in pseudo-words, show that pseudo-words still improve translational performance significantly over unary words.

This paper is structured as follows: In section 2, we define the task of searching for pseudo-words and its solution. We present experimental results and analyses of using pseudo-words in PB-SMT model in section 3. The conclusion is presented at section 4.

2 Searching for Pseudo-words

Pseudo-word searching problem is equal to decomposition of a given sentence into pseudo-words. We assume that the distribution of such decomposition is in the form of Gibbs distribution as below:

$$P(Y \mid X) = \frac{1}{Z_X} \exp\left(\sum_k Sig_{y_k}\right) \qquad (1)$$

where *X* denotes the sentence, *Y* denotes a decomposition of *X*. *Sig* function acts as potential function on each multi-word y_k, and Z_X acts as partition function. Note that the number of y_k is not fixed given *X* because *X* can be decomposed into various number of multi-words.

Given *X*, Z_X is fixed, so searching for optimal decomposition is as below:

$$\hat{Y} = \underset{Y}{ARGMAX}\, P(Y \mid X) = \underset{Y_1^K}{ARGMAX} \sum_k Sig_{y_k} \qquad (2)$$

where Y_1^K denotes *K* multi-word units from decomposition of *X*. A multi-word sequence with maximal sum of *Sig* function values is the search target — pseudo-word sequence. From (2) we can see that *Sig* function is vital for pseudo-word searching. In this paper *Sig* function calculates sequence significance which is proposed to characterize pseudo-word as minimal sequence of consecutive words in sense of translation. The detail of sequence significance is described in the following section.

2.1 Sequence Significance

Two kinds of definitions of sequence significance are proposed. One is monolingual sequence significance. *X* and *Y* are monolingual sentence and monolingual multi-words respectively in this monolingual scenario. The other is bilingual sequence significance. *X* and *Y* are sentence pair and multi-word pairs respectively in this bilingual scenario.

2.1.1 Monolingual Sequence Significance

Given a sentence $w_1, \ldots, w_n$, where w_i denotes unary word, monolingual sequence significance is defined as:

$$Sig_{i,j} = \frac{Freq_{i,j}}{Freq_{i-1,j+1}} \qquad (3)$$

where $Freq_{i,j}$ ($i \leq j$) represents frequency of word sequence $w_i, \ldots, w_j$ in the corpus, $Sig_{i,j}$ represents monolingual sequence significance of a word sequence $w_i, \ldots, w_j$. We also denote word sequence $w_i, \ldots, w_j$ as *span*[*i, j*], whole sentence as *span*[*1, n*]. Each span is also a multi-word expression.

Monolingual sequence significance of *span*[*i, j*] is proportional to *span*[*i, j*]'s frequency, while is inversely proportion to frequency of expanded span (*span*[*i-1, j+1*]). Such definition characterizes minimal sequence of consecutive words which we are looking for. Our target is to find pseudo-word sequence which has maximal sum of spans' significances:

$$pw_1^K = \underset{span_1^K}{ARGMAX} \sum_{k=1}^{K} Sig_{span_k} \quad (4)$$

where *pw* denotes pseudo-word, *K* is equal to or less than sentence's length. $span_k$ is the *k*th span of *K* spans $span_1^K$. Equation (4) is the rewrite of equation (2) in monolingual scenario. Searching for pseudo-words pw_1^K is the same to finding optimal segmentation of a sentence into *K* segments $span_1^K$ (*K* is a variable too). Details of searching algorithm are described in section 2.2.1.

We firstly search for monolingual pseudo-words on source and target side individually. Then we apply word alignment techniques to build pseudo-word alignments. We argue that word alignment techniques will work fine if non-existent word alignments in such as non-compositional phrasal equivalences have been filtered by pseudo-words.

2.1.2 Bilingual Sequence Significance

Bilingual sequence significance is proposed to characterize pseudo-word pairs. Co-occurrence of sequences on both language sides is used to define bilingual sequence significance. Given a bilingual sequence pair: *span-pair*[i_s, j_s, i_t, j_t] (source side *span*[i_s, j_s] and target side *span*[i_t, j_t]), bilingual sequence significance is defined as below:

$$Sig_{i_s,j_s,i_t,j_t} = \frac{Freq_{i_s,j_s,i_t,j_t}}{Freq_{i_s-1,j_s+1,i_t-1,j_t+1}} \quad (5)$$

where *Freq* denotes the frequency of a span-pair. Bilingual sequence significance is an extension of monolingual sequence significance. Its value is proportional to frequency of *span-pair*[i_s, j_s, i_t, j_t], while is inversely proportional to frequency of expanded *span-pair*[i_s-1, j_s+1, i_t-1, j_t+1]. Pseudo-word pairs of one sentence pair are such pairs that maximize the sum of *span-pairs*' bilingual sequence significances:

$$pwp_1^K = \underset{span-pair_1^K}{ARGMAX} \sum_{k=1}^{K} Sig_{span-pair_k} \quad (6)$$

pwp represents pseudo-word pair. Equation (6) is the rewrite of equation (2) in bilingual scenario. Searching for pseudo-word pairs pwp_1^K is equal to bilingual segmentation of a sentence pair into optimal $span\text{-}pair_1^K$. Details of searching algorithm are presented in section 2.2.2.

2.2 Algorithms of Searching for Pseudo-words

Pseudo-word searching problem is equal to decomposition of a sentence into pseudo-words. But the number of possible decompositions of the sentence grows exponentially with the sentence length in both monolingual scenario and bilingual scenario. By casting such decomposition problem into parsing framework, we can find pseudo-word sequence in polynomial time. According to the two scenarios, searching for pseudo-words can be performed in a monolingual way and a synchronous way. Details of the two kinds of searching algorithms are described in the following two sections.

2.2.1 Algorithm of Searching for Monolingual Pseudo-words (SMP)

Searching for monolingual pseudo-words is based on the computation of monolingual sequence significance. Figure 1 presents the search algorithm. It is performed in a way similar to CKY (Cocke-Kasami-Younger) parser.

```
Initialization: $W_{i,i} = Sig_{i,i}$;
                $W_{i,j} = 0,\ (i \neq j)$;
for $d = 2 \ldots n$ do
    for all $i, j$ s.t. $j-i=d-1$ do
        for $k = i \ldots j-1$ do
            $v = W_{i,k} + W_{k+1,j}$
            if $v > W_{i,j}$ then
                $W_{i,j} = v$;
        $u = Sig_{i,j}$
        if $u > W_{i,j}$ then
            $W_{i,j} = u$;
```

Figure 1. Algorithm of searching for monolingual pseudo-words (SMP).

In this algorithm, $W_{i,j}$ records maximal sum of monolingual sequence significances of sub spans of *span*[*i, j*]. During initialization, $W_{i,i}$ is initialized as $Sig_{i,i}$ (note that this sequence is word w_i only). For all spans that have more than one word ($i \neq j$), $W_{i,j}$ is initialized as zero.

In the main algorithm, *d* represents span's length, ranging from *2* to *n*, *i* represents start position of a span, *j* represents end position of a span, *k* represents decomposition position of *span*[*i,j*]. For *span*[*i, j*], $W_{i,j}$ is updated if higher sum of monolingual sequence significances is found.

The algorithm is performed in a bottom-up way. Small span's computation is first. After maximal sum of significances is found in small spans, big span's computation, which uses small spans' maximal sum, is continued. Maximal sum of significances for whole sentence ($W_{1,n}$, *n* is sentence's length) is guaranteed in this way, and optimal decomposition is obtained correspondingly.

The method of fitting the decomposition problem into CKY parsing framework is located at steps 7-9. After steps 3-6, all possible decompositions of *span*[i, j] are explored and $W_{i,j}$ of optimal decomposition of *span*[i, j] is recorded. Then monolingual sequence significance $Sig_{i,j}$ of *span*[i, j] is computed at step 7, and it is compared to $W_{i,j}$ at step 8. Update of $W_{i,j}$ is taken at step 9 if $Sig_{i,j}$ is bigger than $W_{i,j}$, which indicates that *span*[i, j] is non-decomposable. Thus whether *span*[i, j] should be non-decomposable or not is decided through steps 7-9.

2.2.2 Algorithm of Synchronous Searching for Pseudo-words (SSP)

Synchronous searching for pseudo-words utilizes bilingual sequence significance. Figure 2 presents the search algorithm. It is similar to ITG (Wu, 1997), except that it has no production rules and non-terminal nodes of a synchronous grammar. What it cares about is the span-pairs that maximize the sum of bilingual sequence significances.

```
Initialization:  if $i_s = j_s$ or $i_t = j_t$ then
                     $W_{i_s,j_s,i_t,j_t} = Sig_{i_s,j_s,i_t,j_t}$;
                 else
                     $W_{i_s,j_s,i_t,j_t} = 0$;
1: for $d_s = 2 \dots n_s$, $d_t = 2 \dots n_t$ do
2:     for all $i_s, j_s, i_t, j_t$ s.t. $j_s - i_s = d_s - 1$ and $j_t - i_t = d_t - 1$ do
3:         for $k_s = i_s \dots j_s - 1$, $k_t = i_t \dots j_t - 1$ do
4:             $v = max\{W_{i_s,k_s,i_t,k_t} + W_{k_s+1,j_s,k_t+1,j_t},$
                   $W_{i_s,k_s,k_t+1,j_t} + W_{k_s+1,j_s,i_t,k_t}\}$
5:             if $v > W_{i_s,j_s,i_t,j_t}$ then
6:                 $W_{i_s,j_s,i_t,j_t} = v$;
7:         $u = Sig_{i_s,j_s,i_t,j_t}$
8:         if $u > W_{i_s,j_s,i_t,j_t}$ then
9:             $W_{i_s,j_s,i_t,j_t} = u$;
```

Figure 2. Algorithm of Synchronous Searching for Pseudo-words(SSP).

In the algorithm, W_{i_s,j_s,i_t,j_t} records maximal sum of bilingual sequence significances of sub span-pairs of *span-pair*[i_s, j_s, i_t, j_t]. For *1*-to-*m* span-pairs, *W*s are initialized as bilingual sequence significances of such span-pairs. For other span-pairs, *W*s are initialized as zero.

In the main algorithm, d_s/d_t denotes the length of a span on source/target side, ranging from *2* to n_s/n_t (source/target sentence's length). i_s/i_t is the start position of a span-pair on source/target side, j_s/j_t is the end position of a span-pair on source/target side, k_s/k_t is the decomposition position of a *span-pair*[i_s, j_s, i_t, j_t] on source/target side.

Update steps in Figure 2 are similar to that of Figure 1, except that the update is about span-pairs, not monolingual spans. Reversed and non-reversed alignments inside a span-pair are compared at step 4. For *span-pair*[i_s, j_s, i_t, j_t], W_{i_s,j_s,i_t,j_t} is updated at step 6 if higher sum of bilingual sequence significances is found.

Fitting the bilingually searching for pseudo-words into ITG framework is located at steps 7-9. Steps 3-6 have explored all possible decompositions of *span-pair*[i_s, j_s, i_t, j_t] and have recorded maximal W_{i_s,j_s,i_t,j_t} of these decompositions. Then bilingual sequence significance of *span-pair*[i_s, j_s, i_t, j_t] is computed at step 7. It is compared to W_{i_s,j_s,i_t,j_t} at step 8. Update is taken at step 9 if bilingual sequence significance of *span-pair*[i_s, j_s, i_t, j_t] is bigger than W_{i_s,j_s,i_t,j_t}, which indicates that *span-pair*[i_s, j_s, i_t, j_t] is non-decomposable. Whether the *span-pair*[i_s, j_s, i_t, j_t] should be non-decomposable or not is decided through steps 7-9.

In addition to the initialization step, all span-pairs' bilingual sequence significances are computed. Maximal sum of bilingual sequence significances for one sentence pair is guaranteed through this bottom-up way, and the optimal decomposition of the sentence pair is obtained correspondingly.

- **Algorithm of Excluded Synchronous Searching for Pseudo-words (ESSP)**

The algorithm of SSP in Figure 2 explores all span-pairs, but it neglects NULL alignments, where words and "empty" word are aligned. In fact, SSP requires that all parts of a sentence pair should be aligned. This requirement is too strong because NULL alignments are very common in many language pairs. In SSP, words that should be aligned to "empty" word are programmed to be aligned to real words.

Unlike most word alignment methods (Och and Ney, 2003) that add "empty" word to account for NULL alignment entries, we propose a method to naturally exclude such NULL alignments. We call this method as Excluded Synchronous Searching for Pseudo-words (ESSP).

The main difference between ESSP and SSP is in steps 3-6 in Figure 3. We illustrate Figure 3's span-pair configuration in Figure 4.

```
Initialization: if $i_s = j_s$ or $i_t = j_t$ then
                    $W_{i_s,j_s,i_t,j_t} = Sig_{i_s,j_s,i_t,j_t}$;
                else
                    $W_{i_s,j_s,i_t,j_t} = 0$;
for $d_s = 2 ... n_s$, $d_t = 2 ... n_t$ do
    for all $i_s, j_s, i_t, j_t$ s.t. $j_s-i_s=d_s-1$ and $j_t-i_t=d_t-1$ do
        for $k_{s1}=i_s+1 ... j_s$, $k_{s2}=k_{s1}-1 ... j_s-1$
            $k_{t1}=i_t+1 ... j_t$, $k_{t2}=k_{t1}-1 ... j_t-1$ do
            $v = \max\{W_{i_s,k_{s1}-1,i_t,k_{t1}-1} + W_{k_{s2}+1,j_s,k_{t2}+1,j_t}$,
                       $W_{i_s,k_{s1}-1,k_{t2}+1,j_t} + W_{k_{s2}+1,j_s,i_t,k_{t1}-1}\}$
            if $v > W_{i_s,j_s,i_t,j_t}$ then
                $W_{i_s,j_s,i_t,j_t} = v$;
        $u = Sig_{i_s,j_s,i_t,j_t}$
        if $u > W_{i_s,j_s,i_t,j_t}$ then
            $W_{i_s,j_s,i_t,j_t} = u$;
```

Figure 3. Algorithm of Excluded Synchronous Searching for Pseudo-words (ESSP).

The solid boxes in Figure 4 represent excluded parts of *span-pair*[i_s, j_s, i_t, j_t] in ESSP. Note that, in SSP, there is no excluded part, that is, $k_{s1}=k_{s2}$ and $k_{t1}=k_{t2}$.

We can see that in Figure 4, each monolingual span is configured into three parts, for example: *span*[i_s, $k_{s1}-1$], *span*[k_{s1}, k_{s2}] and *span*[$k_{s2}+1$, j_s] on source language side. k_{s1} and k_{s2} are two new variables gliding between i_s and j_s, *span*[k_{s1}, k_{s2}] is source side excluded part of *span-pair*[i_s, j_s, i_t, j_t]. Bilingual sequence significance is computed only on pairs of blank boxes, solid boxes are excluded in this computation to represent NULL alignment cases.

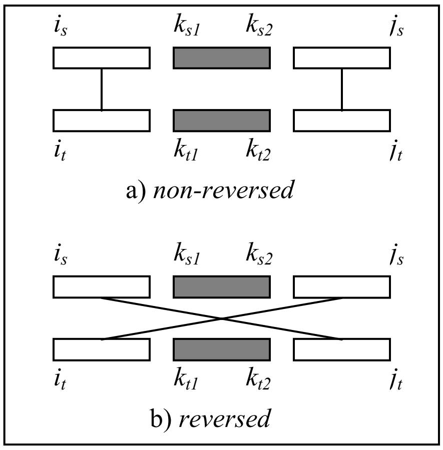

Figure 4. Illustration of excluded configuration.

Note that, in Figure 4, solid box on either language side can be void (i.e., length is zero) if there is no NULL alignment on its side. If all solid boxes are shrunk into void, algorithm of ESSP is the same to SSP.

Generally, span length of NULL alignment is not very long, so we can set a length threshold for NULL alignments, eg. $k_{s2}-k_{s1} \leqslant EL$, where EL denotes *Excluded Length* threshold. Computational complexity of the ESSP remains the same to SSP's complexity $O(n_s^3.n_t^3)$, except multiply a constant EL^2.

There is one kind of NULL alignments that ESSP can not consider. Since we limit excluded parts in the middle of a span-pair, the algorithm will end without considering boundary parts of a sentence pair as NULL alignments.

3 Experiments and Results

In our experiments, pseudo-words are fed into PB-SMT pipeline. The pipeline uses GIZA++ model 4 (Brown et al., 1993; Och and Ney, 2003) for pseudo-word alignment, uses Moses (Koehn et al., 2007) as phrase-based decoder, uses the SRI Language Modeling Toolkit to train language model with modified Kneser-Ney smoothing (Kneser and Ney 1995; Chen and Goodman 1998). Note that MERT (Och, 2003) is still on original words of target language. In our experiments, pseudo-word length is limited to no more than six unary words on both sides of the language pair.

We conduct experiments on Chinese-to-English machine translation. Two data sets are adopted, one is small corpus of IWSLT-2008 BTEC task of spoken language translation in travel domain (Paul, 2008), the other is large corpus in news domain, which consists Hong Kong News (LDC2004T08), Sinorama Magazine (LDC2005T10), FBIS (LDC2003E14), Xinhua (LDC2002E18), Chinese News Translation (LDC2005T06), Chinese Treebank (LDC2003E07), Multiple Translation Chinese (LDC2004T07). Table 1 lists statistics of the corpus used in these experiments.

	small		large	
	Ch →	En	Ch →	En
Sent.	23k		1,239k	
word	190k	213k	31.7m	35.5m
ASL	8.3	9.2	25.6	28.6

Table 1. Statistics of corpora, "Ch" denotes Chinese, "En" denotes English, "Sent." row is the number of sentence pairs, "word" row is the number of words, "ASL" denotes average sentence length.

For small corpus, we use CSTAR03 as development set, use IWSLT08 official test set for test. A *5*-gram language model is trained on English side of parallel corpus. For large corpus, we use NIST02 as development set, use NIST03 as test set. Xinhua portion of the English Gigaword3 corpus is used together with English side of large corpus to train a *4*-gram language model.

Experimental results are evaluated by case-insensitive BLEU-4 (Papineni et al., 2001). Closest reference sentence length is used for brevity penalty. Additionally, NIST score (Doddington, 2002) and METEOR (Banerjee and Lavie, 2005) are also used to check the consistency of experimental results. Statistical significance in BLEU score differences was tested by paired bootstrap re-sampling (Koehn, 2004).

3.1 Baseline Performance

Our baseline system feeds word into PB-SMT pipeline. We use GIZA++ model 4 for word alignment, use Moses for phrase-based decoding. The setting of language model order for each corpus is not changed. Baseline performances on test sets of small corpus and large corpus are reported in table 2.

	small	Large
BLEU	0.4029	0.3146
NIST	7.0419	8.8462
METEOR	0.5785	0.5335

Table 2. Baseline performances on test sets of small corpus and large corpus.

3.2 Pseudo-word Unpacking

Because pseudo-word is a kind of multi-word expression, it has inborn advantage of higher language model order and longer max phrase length over unary word. To see if such inborn advantage is the main contribution to the performance or not, we unpack pseudo-word into words after GIZA++ aligning. Aligned pseudo-words are unpacked into $m \times n$ word alignments. PB-SMT pipeline is executed thereafter. The advantage of longer max phrase length is removed during phrase extraction, and the advantage of higher order of language model is also removed during decoding since we use language model trained on unary words. Performances of pseudo-word unpacking are reported in section 3.3.1 and 3.4.1. Ma and Way (2009) used the unpacking after phrase extraction, then re-estimated phrase translation probability and lexical reordering model. The advantage of longer max phrase length is still used in their method.

3.3 Pseudo-word Performances on Small Corpus

Table 3 presents performances of SMP, SSP, ESSP on small data set. $pw_{ch}pw_{en}$ denotes that pseudo-words are on both language side of training data, and they are input strings during development and testing, and translations are also pseudo-words, which will be converted to words as final output. $w_{ch}pw_{en}/pw_{ch}w_{en}$ denotes that pseudo-words are adopted only on English/Chinese side of the data set.

We can see from table 3 that, ESSP attains the best performance, while SSP attains the worst performance. This shows that excluding NULL alignments in synchronous searching for pseudo-words is effective. SSP puts overly strong alignment constraints on parallel corpus, which impacts performance dramatically. ESSP is superior to SMP indicating that bilingually motivated searching for pseudo-words is more effective. Both SMP and ESSP outperform baseline consistently in BLEU, NIST and METEOR.

There is a common phenomenon among SMP, SSP and ESSP. $w_{ch}pw_{en}$ always performs better than the other two cases. It seems that Chinese word prefers to have English pseudo-word equivalence which has more than or equal to one word. $pw_{ch}pw_{en}$ in ESSP performs similar to the baseline, which reflects that our direct pseudo-word pairs do not work very well with GIZA++ alignments. Such disagreement is weakened by using pseudo-words on only one language side ($w_{ch}pw_{en}$ or $pw_{ch}w_{en}$), while the advantage of pseudo-words is still leveraged in the alignments.

Best ESSP ($w_{ch}pw_{en}$) is significantly better than baseline (p<0.01) in BLEU score, best SMP ($w_{ch}pw_{en}$) is significantly better than baseline (p<0.05) in BLEU score. This indicates that pseudo-words, through either monolingual searching or synchronous searching, are more effective than words as to being basic translational units.

Figure 5 illustrates examples of pseudo-words of one Chinese-to-English sentence pair. Gold standard word alignments are shown at the bottom of figure 5. We can see that “front desk” is recognized as one pseudo-word in ESSP. Because SMP performs monolingually, it can not consider “前台” and “front desk” simultaneously. SMP only detects frequent monolingual multi-words as pseudo-words. SSP has a strong constraint that all parts of a sentence pair should be aligned, so source sentence and target sentence have same length after merging words into

	SMP			SSP			ESSP			baseline
	$pw_{ch}pw_{en}$	$w_{ch}pw_{en}$	$pw_{ch}w_{en}$	$pw_{ch}pw_{en}$	$w_{ch}pw_{en}$	$pw_{ch}w_{en}$	$pw_{ch}pw_{en}$	$w_{ch}pw_{en}$	$pw_{ch}w_{en}$	
BLEU	0.3996	0.4155	0.4024	0.3184	0.3661	0.3552	0.3998	**0.4229**	0.4147	0.4029
NIST	7.4711	**7.6452**	7.6186	6.4099	6.9284	6.8012	7.1665	7.4373	7.4235	7.0419
METEOR	0.5900	**0.6008**	0.6000	0.5255	0.5569	0.5454	0.5739	0.5963	0.5891	0.5785

Table 3. Performance of using pseudo-words on small data.

pseudo-words. We can see that too many pseudo-words are detected by SSP.

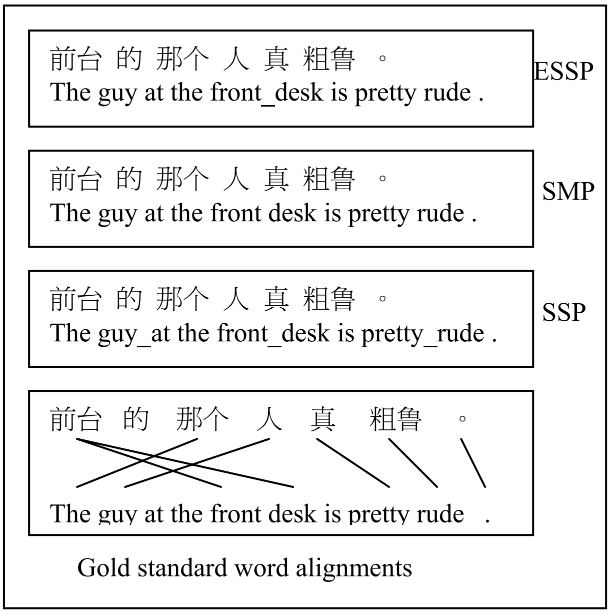
前台 的 那个 人 真 粗鲁 。
The guy at the front_desk is pretty rude .
ESSP
前台 的 那个 人 真 粗鲁 。
The guy at the front desk is pretty rude .
SMP
前台 的 那个 人 真 粗鲁 。
The guy_at the front_desk is pretty_rude .
SSP
前台 的 那个 人 真 粗鲁 。
The guy at the front desk is pretty rude .
Gold standard word alignments

Figure 5. Outputs of the three algorithms ESSP, SMP and SSP on one sentence pair and gold standard word alignments. Words in one pseudo-word are concatenated by "_".

3.3.1 Pseudo-word Unpacking Performances on Small Corpus

We test pseudo-word unpacking in ESSP. Table 4 presents its performances on small corpus.

	$\text{unpacking}_{\text{ESSP}}$			baseline
	$pw_{ch}pw_{en}$	$w_{ch}pw_{en}$	$pw_{ch}w_{en}$	
BLEU	0.4097	**0.4182**	0.4031	0.4029
NIST	**7.5547**	7.2893	7.2670	7.0419
METEOR	**0.5951**	0.5874	0.5846	0.5785

Table 4. Performances of pseudo-word unpacking on small corpus.

We can see that pseudo-word unpacking significantly outperforms baseline. $w_{ch}pw_{en}$ is significantly better than baseline ($p<0.04$) in BLEU score. Unpacked pseudo-word performs comparatively with pseudo-word without unpacking. There is no statistical difference between them. It shows that the improvement derives from pseudo-word itself as basic translational unit, does not rely very much on higher language model order or longer max phrase length setting.

3.4 Pseudo-word Performances on Large Corpus

Table 5 lists the performance of using pseudo-words on large corpus. We apply SMP on this task. ESSP is not applied because of its high computational complexity. Table 5 shows that all three configurations ($pw_{ch}pw_{en}$, $w_{ch}pw_{en}$, $pw_{ch}w_{en}$) of SMP outperform the baseline. If we go back to the definition of sequence significance, we can see that it is a data-driven definition that utilizes corpus frequencies. Corpus scale has an influence on computation of sequence significance in long sentences which appear frequently in news domain. SMP benefits from large corpus, and $w_{ch}pw_{en}$ is significantly better than baseline ($p<0.01$). Similar to performances on small corpus, $w_{ch}pw_{en}$ always performs better than the other two cases, which indicates that Chinese word prefers to have English pseudo-word equivalence which has more than or equal to one word.

	SMP			baseline
	$pw_{ch}pw_{en}$	$w_{ch}pw_{en}$	$pw_{ch}w_{en}$	
BLEU	0.3185	**0.3230**	0.3166	0.3146
NIST	8.9216	**9.0447**	8.9210	8.8462
METEOR	0.5402	**0.5489**	0.5435	0.5335

Table 5. Performance of using pseudo-words on large corpus.

3.4.1 Pseudo-word Unpacking Performances on Large Corpus

Table 6 presents pseudo-word unpacking performances on large corpus. All three configurations improve performance over baseline after pseudo-word unpacking. $pw_{ch}pw_{en}$ attains the best BLEU among the three configurations, and is significantly better than baseline ($p<0.03$). $w_{ch}pw_{en}$ is also significantly better than baseline ($p<0.04$). By comparing table 6 with table 5, we can see that unpacked pseudo-word performs comparatively with pseudo-word without unpacking. There is no statistical difference between them. It shows that the improvement derives from pseudo-word itself as basic translational unit, does not rely very much on higher language model order or longer max phrase length setting. In fact, slight improvement in $pw_{ch}pw_{en}$ and $pw_{ch}w_{en}$ is seen after pseudo-word unpacking, which indicates that higher language model order and longer max phrase length impact the performance in these two configurations.

	Unpacking$_{\text{SMP}}$			Baseline
	$pw_{ch}pw_{en}$	$w_{ch}pw_{en}$	$pw_{ch}w_{en}$	
BLEU	**0.3219**	0.3192	0.3187	0.3146
NIST	8.9458	8.9325	**8.9801**	8.8462
METEOR	**0.5429**	0.5424	0.5411	0.5335

Table 6. Performance of pseudo-word unpacking on large corpus.

3.5 Comparison to English Chunking

English chunking is experimented to compare with pseudo-word. We use FlexCRFs (Xuan-Hieu Phan et al., 2005) to get English chunks. Since there is no standard Chinese chunking data and code, only English chunking is executed. The experimental results show that English chunking performs far below baseline, usually 8 absolute BLEU points below. It shows that simple chunks are not suitable for being basic translational units.

4 Conclusion

We have presented pseudo-word as a novel machine translational unit for phrase-based machine translation. It is proposed to replace too fine-grained word as basic translational unit. Pseudo-word is a kind of basic multi-word expression that characterizes minimal sequence of consecutive words in sense of translation. By casting pseudo-word searching problem into a parsing framework, we search for pseudo-words in polynomial time. Experimental results of Chinese-to-English translation task show that, in phrase-based machine translation model, pseudo-word performs significantly better than word in both spoken language translation domain and news domain. Removing the power of higher order language model and longer max phrase length, which are inherent in pseudo-words, shows that pseudo-words still improve translational performance significantly over unary words.

References

S. Banerjee, and A. Lavie. 2005. *METEOR: An automatic metric for MT evaluation with improved correlation with human judgments.* In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization (ACL'05). 65–72.

P. Blunsom, T. Cohn, C. Dyer, M. Osborne. 2009. *A Gibbs Sampler for Phrasal Synchronous Grammar Induction.* In Proceedings of ACL-IJCNLP, Singapore.

P. Blunsom, T. Cohn, M. Osborne. 2008. *Bayesian synchronous grammar induction.* In Proceedings of NIPS 21, Vancouver, Canada.

P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer. 1993. *The mathematics of machine translation: Parameter estimation.* Computational Linguistics, 19:263–312.

P.-C. Chang, M. Galley, and C. D. Manning. 2008. *Optimizing Chinese word segmentation for machine translation performance.* In Proceedings of the 3rd Workshop on Statistical Machine Translation (SMT'08). 224–232.

Chen, Stanley F. and Joshua Goodman. 1998. *An empirical study of smoothing techniques for language modeling.* Technical Report TR-10-98, Harvard University Center for Research in Computing Technology.

C. Cherry, D. Lin. 2007. *Inversion transduction grammar for joint phrasal translation modeling.* In Proc. of the HLTNAACL Workshop on Syntax and Structure in Statistical Translation (SSST 2007), Rochester, USA.

D. Chiang. 2007. *Hierarchical phrase-based translation.*Computational Linguistics, 33(2):201–228.

Y. Deng and W. Byrne. 2005. *HMM word and phrase alignment for statistical machine translation.* In Proc. of HLT-EMNLP, pages 169–176.

G. Doddington. 2002. *Automatic evaluation of machine translation quality using n-gram cooccurrence statistics.* In Proceedings of the 2nd International Conference on Human Language Technology (HLT'02). 138–145.

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, pages 181−184, Detroit, MI.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan,W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, E. Herbst. 2007. *Moses: Open source toolkit for statistical machine translation.* In Proc. of the

45th Annual Meeting of the ACL (ACL-2007), Prague.

P. Koehn, F. J. Och, D. Marcu. 2003. *Statistical phrasebased translation.* In Proc. of the 3rd International conference on Human Language Technology Research and 4th Annual Meeting of the NAACL (HLT-NAACL 2003), 81–88, Edmonton, Canada.

P. Koehn. 2004. *Statistical Significance Tests for Machine Translation Evaluation.* In Proceedings of EMNLP.

P. Lambert and R. Banchs. 2005. *Data Inferred Multi-word Expressions for Statistical Machine Translation.* In Proceedings of MT Summit X.

Y. Ma, N. Stroppa, and A. Way. 2007. *Bootstrapping word alignment via word packing.* In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07). 304–311.

Y. Ma, and A. Way. 2009. *Bilingually Motivated Word Segmentation for Statistical Machine Translation.* In ACM Transactions on Asian Language Information Processing, 8(2).

D. Marcu,W.Wong. 2002. *A phrase-based, joint probability model for statistical machine translation.* In Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002), 133–139, Philadelphia. Association for Computational Linguistics.

F. J. Och. 2003. *Minimum error rate training in statistical machine translation.* In Proc. of ACL, pages 160–167.

F. J. Och and H. Ney. 2003. *A systematic comparison of various statistical alignment models.* Computational Linguistics, 29(1):19–51.

Xuan-Hieu Phan, Le-Minh Nguyen, and Cam-Tu Nguyen. 2005. *FlexCRFs: Flexible Conditional Random Field Toolkit,* http://flexcrfs.sourceforge.net

K. Papineni, S. Roukos, T. Ward, W. Zhu. 2001. *Bleu: a method for automatic evaluation of machine translation,* 2001.

M. Paul, 2008. *Overview of the IWSLT 2008 evaluation campaign.* In Proc. of Internationa Workshop on Spoken Language Translation, 20-21 October 2008.

A. Stolcke. (2002). *SRILM - an extensible language modeling toolkit.* In Proceedings of ICSLP, Denver, Colorado.

D. Wu. 1997. *Stochastic inversion transduction grammars and bilingual parsing of parallel corpora.* Computational Linguistics, 23(3):377–403.

J. Xu, Zens., and H. Ney. 2004. *Do we need Chinese word segmentation for statistical machine translation?* In Proceedings of the ACL Workshop on Chinese Language Processing SIGHAN'04). 122–128.

J. Xu, J. Gao, K. Toutanova, and H. Ney. 2008. *Bayesian semi-supervised chinese word segmentation for statistical machine translation.* In Proceedings of the 22nd International Conference on Computational Linguistics (COLING'08). 1017–1024.

H. Zhang, C. Quirk, R. C. Moore, D. Gildea. 2008. *Bayesian learning of non-compositional phrases with synchronous parsing.* In Proc. of the 46th Annual Conference of the Association for Computational Linguistics: Human Language Technologies (ACL-08:HLT), 97–105, Columbus, Ohio.

R. Zhang, K. Yasuda, and E. Sumita. 2008. *Improved statistical machine translation by multiple Chinese word segmentation.* In Proceedings of the 3rd Workshop on Statistical Machine Translation (SMT'08). 216–223.