

A Post-Reordering Model based Maximum Entropy in Statistic Machine Translation

Jinying Kong^{1,2}, Yating Yang^{1,3}, Xiao Li^{1,3} and Lei Wang^{1,3}

¹*Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences*

²*University of Chinese Academy of Sciences*

³*Xinjiang Laboratory of Minority Speech and Language Information Processing
konguestc@163.com*

Abstract

Reordering model is crucial in statistic machine translation (SMT). This paper proposes a novel post-reordering model which reorders translation output after decoding. As none additional syntactic tool used and the location our model worked, our model can be utilized in SMT of any language pairs and any translation model. Unlike other methods, our model design four orientations (monotony, swapping, discontinue monotony, discontinue swapping) to capture more possible reordering operations with sentence. In order to carry out long-distance reordering in SMT, we put forward a new stack shift-reduce algorithm to recombine any two phrases which are not necessary neighbor. According to the experiment, SMT combined with our model have a significant raise in BLEU compared with the state-of-art SMT. What is more, our stack shift-reduce algorithm do produce new output sentence efficiently.

Keywords: *Reordering model, Statistical Machine Translation, Maximum Entropy*

1. Introduction

Currently, Statistical Machine Translation which is more convenient than human translation has been widely used, while its performance is not so satisfied [1]. Although the result of machine translation (MT) get better and better, there are still some problems in MT. One of the most obvious issue that influence translation quality is the output word ordering. Now MT output can almost translate the right words or phrases, but the union of output words is just like a bag of words. It is because of that Word reordering is a difficult task for decoders when the languages involved have a significant difference in syntax. Unfortunately, languages we faced to translate always have different Syntactic structure. It is also very challenging due to lacking of effective methods to model the difference between target and source languages.

Generally speaking ,phrase based machine translation system (MTS) is good at local word ordering ,but fail at capturing long-distance word ordering especially for language pairs with significant different syntax [2]. Hierarchical phrase based MT get good performance for reordering but has large amount of computation [3]. Because sentence with N words has $n!$ orders, but only few of them is right. Hence, word reordering is a hot but difficult topic of research in SMT (statistical machine translation). The ordering problem search not only the sentence which is fluent, but also the sentence which interpret original input sentence best. Here is an example between translation sentence and reference sentence in Figure 1. In the Figure 1, translation sentence produced by phrase based MT has a significant error in words order. Actually, this phenomenon is very common in SMT. In spite of complex language model and translation model to help

produce the translation, different syntax and other different linguistic phenomenon also obstacle the MTS to get the right order. And that's the problem what this paper want to solve.

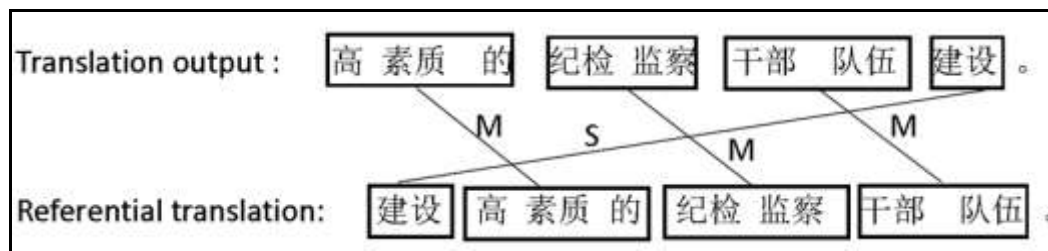


Figure 1. An Example between Output and Reference

The model to solve this problem is called reordering model in SMT. Nowadays the reordering problem have been studied in many research works among various languages pair. The recently attractive approaches for this challenge such as pre-ordering and reordering model embed in translation model have good performance. But a crucial issue on these methods is language dependent; those effective methods need a syntactic parser commonly.

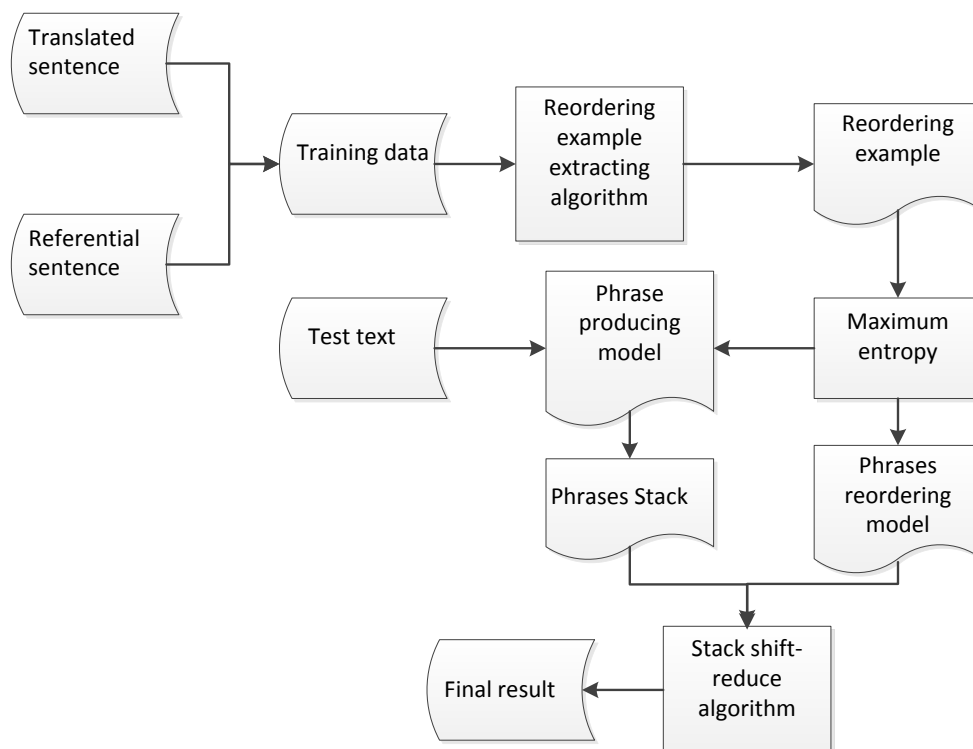


Figure 2. Maximum Entropy Post-Reordering Model Framework

In this work we apply a Maximum Entropy reordering model in post-processing to address the reordering problem in most MTS which is language independent. First, we train two Maximum Entropy models: a phrase producing model and a phrases reordering model. Phrase producing model based Maximum Entropy is to produce new phrase which is suitable for reordering, and the phrases reordering model is to reorder the new phrases that we got before. Then we use the trained phrase producing model to get new phrase which produced by adjacent words monotonously merging or swapping merging, and put

the new phrases into a stack. Thirdly, while the phrases stack popping out its elements, we use the phrases reordering model to reorder these phrases by judging whether the popped two phrase is monotonous union or swap union. Figure 2 describe our job roughly.

The remainder of this paper is organized as follows. Section 2 discusses previous work in the field and how that is related to our paper. From the perspective of problem, Section 3 is dedicated to describe post-reordering model based Maximum Entropy and explain how our model solves this issue. Section 4 introduce our experiment to proven the efficiency of our approach. At last section, we discuss our experiment result and point to our future work.

2. Related Work

Although there are many works on reordering model for English-to-Chinese or Chinese-to-Japan, or other common bilingual language. Compare with them, reordering methods for general bilingual language has received little attention. In consideration of these, we introduce a general bilingual reordering model in this paper. Although rarely method without syntactic parsing have been proposed nowadays, we also can learn some useful information from method with syntactic parsing. There are many criterion we can use to classify the approaches addressing the ordering problem. Here we first differentiate reordering model by the location where they are applied.

As research continues, researcher find that reordering is a both theoretically and practically challenge in SMT. Some scholars make the word order of source sentence similarity to the target sentence's or do some other processing on the training set prior to the translation decoding. We call these works pre-ordering as these done prior to the translation decoding. Furthermore, we can divided these work into two part: syntax dependent or syntax independent. Syntax dependent have some representative job such as: (Collins *et al.* 2005) used linguistically-motivated hand-written rules for Chinese-to-English translation [4]; (Li *et al.* 2007) use rules to reorder the source sentences on the chunk level and provide a source-reordering lattice [5]; (Karthik *et al.* 2011) consider source sentence reordering problem as a ATSP [6]. Syntax independent have some representative jobs such as: (Costa-jussa *et al.* 2006) use reordering table derived from phrase tables to reorder [7]; (Dyer *et al.* 2010) advanced such a pre-ordering based translation to a novel unified approach of long-distance pre-ordering and decoding, with discriminative context-free reordering and finite-state phrase translation [8]; (Maria *et al.* 2012) reorders source words according to the corresponding target word order suggested by an initial word alignment without any Syntactic tool [9]. The scholars combining reordering model into translation model to reward good reordering operations or penalize bad operations is the other category, there are many works has been done about this domain: (Och *et al.* 2004) proposed a distortion model which simply penalize phrase displacements proportionally to the amount of no monotonicity [10]; (Tillman *et al.* 2004) introduce a lexicalized reordering model that models orientation of phrases by monotone, swap, and discontinuous [11]; (Han *et al.* 2012) introduced a refined reordering approach by importing an existing reordering method(Head-driven phrase structure grammars) for English proposed in (Isozaki *et al.* 2010b) [12-13]. There is still other category: Post-ordering a MT output where some reordering phenomena was not correctly solved by MT. Our job belongs this category, and these related approaches have been proposed: (Sudoh *et al.* 2011) see the post-ordering as a string-to-tree problem [14]; (Goto *et al.* 2012) modeled the post-ordering by parsing and created training data for a post-ordering parser using a language-dependent rule called head-finalization [15]; (Katsuhiko *et al.* 2013) put forward a Shift-Reduce Word Reordering model to handle this issue [16].

Otherwise, we can differentiate reordering model by the translation model what they applied: Word based MT, phrase based MT, hierarchical phrase based MT, and syntax based MT. This part is similar to the above, so we no longer describe in detail.

The most related work to our job is Maximum Entropy Based Phrase Reordering Model: (Xiong *et al.* 2006) use maximum entropy model which distinguish two orientations (monotone, swap) with respect to the neighbor phrase to capture the ordering features, and combine the Maximum Entropy model to phrase based MT [17]. What we should draw attention is the decoding algorithm they use is a novel ITG model which propose by (Wu *et al.* 1997) [18]. The novel decoding algorithm help make some long-distance reordering operation available. It reminds us that an effective decoding algorithm is vital in reordering model.

3. Maximum Entropy Post-Reordering Model

Syntactic information is useful to guide reordering, but reordering methods that heavily depend on detailed syntactic information are sensitive to those parsing error [19]. To efficiently utilize reordering information, we put forward a post-reordering model based Maximum Entropy.

We can obviously find the translation we got from MT always having significantly difference with reference. Some caused by wrong word or phrase selection, others caused by wrong order. The first problem is solved by translation model, and the second problem is solved by reordering model. This paper put forward a reordering method to decrease the translation order problem. Comparing translation with reference, we use machine learning to model the right translation order. And then, a novel words combining algorithm will reorder the translation sentence with the model we got before. Like (Galley *et al.* 2008) [20], we reorder the output sentence with a shift-reducing stack. To efficiency utilize reordering information, Maximum Entropy has been imported in our approach. As the location our model applied, the method we describe in this paper called Maximum Entropy post-reordering model .We expatiate our model in the following three units: model preparation, reordering example extraction algorithm, stack shift-reduce algorithm.

3.1. Model Preparation

As the research of the reordering problem moves along, we find that there are always some stationary order errors in settled SMT. Taken our Figure 1 as an example, this error is always happen in Uyghur-to-Chinese SMT. Some other stationary errors may happen in English-to-Chinese SMT or other SMT too. If we can capture this stationary error and learn how to change this wrong order to the right order, we can easily get a more fluent translation.

To realize our idea, our method have three parts: phrase producing model and phrases reordering model to capture the stationary errors; a novel stack shift-reduce algorithm to change the wrong translation order to right. To get these two models, we first need some training data which consist of translation sentence got by MTs that you want to improve and the corresponding reference. With the training data, we can extract reordering examples for our phrase producing model and phrase reordering model. The principle we used to extract reordering examples are described as follows.

In the first time scanning the translate sentence, we extract reordering example for the phrase producing model in these three principles:

$$if (p_{\phi(i+1)} - p_{\phi(i)} = 1), o_i = m; \quad (1)$$

$$else if (p_{\phi(i+1)} - p_{\phi(i)} = -1), o_i = s; \quad (2)$$

$$else o_i = d; \quad (3)$$

In these formulas, m means monotonously merging, s means swapped merge, d means discontinue monotonous merge. ‘m’ or ‘s’ is the operation to current phrase,

'd' is the operation to produce new phrase. ' $p_{\phi(i)}$ ' means the location number in target sentence of word i from original translation sentence.

The primary job of phrase producing model is producing new phrases, therefore we put our main energy to solve the problem how to merge the adjacent word phrases. Take this into consideration: phrase based MT is good at location reordering, we just make a little modifications of the sentence order in this step. After this step has been run, we got a sentence consisted of new phrases.

The phrase model only consider three orientations types: (m) monotone order, (s) swap with previous phrase, or (d) discontinuous, phrases reordering model considers four orientation types: monotone order (m), swap with previous phrase (s), discontinuous monotone order (dm), and discontinuous swap with previous phrase (ds). Compare with the phrase producing model, phrases reordering model has a stronger reordering ability.

From second time to the last time scanning the translate sentence, we use these four principles to extract reordering examples for the phrases reordering model.

$$if(p_{\phi(i+1)} - p_{\phi(i)} = 1), o_i = m \quad (4)$$

$$else if(p_{\phi(i+1)} - p_{\phi(i)} = -1), o_i = s \quad (5)$$

$$else if(p_{\phi(i+1)} - p_{\phi(i)} > 1), o_i = dm \quad (6)$$

$$else if(p_{\phi(i+1)} - p_{\phi(i)} < -1), o_i = ds \quad (7)$$

In these principles, 'm', 's' and ' $p_{\phi(i)}$ ' are the same meaning with above. 'dm' means monotonous order but not adjacent, 'ds' means swapped order but not adjacent. 'dm' and 'ds' are the long distance reordering operations ,m and s are location reordering operations.

3.2. Reordering Example Extraction Algorithm

A reordering example consists of two adjacent phrases from translation sentence and its orientation in referential sentence. With the extracted reordering examples, we can get features for our Maximum Entropy post-reordering model. Taken example by (Xiong *et al.* 2006), our model design two kinds of features, lexical features and length features.

For a block $b^i = (t, r)$ we extracted before, we use t_f to denote the first word of original translation fragment, r_f to denote the first word of referential translation fragment r. In the same way, we use t_l to denote the last word of original translation fragment, r_l to denote the last word of referential translation fragment. Otherwise, we use n to record the words number of original translation fragment.

Unlike (Xiong *et al.*,2006), our model not only use the first word as the features, but also use last word and fragment length as the features. The four principles extracting template for our model features are illustrated as follows. Figure 3 is an illustration of reordering examples extracted from the sentence pair in Figure 1.

$$E = b^1 \{ t_f \bullet t_l \} \oplus b^2 \{ t_f \bullet t_l \} \oplus |engt h_1 \oplus |engt h_2 \quad (8)$$

$$\exists e \in E, e = \{ b^1, b^2, |engt h_1, |engt h_2 \} \quad (9)$$

$$p_{b^2\{t_{(r_f)}\}} - p_{b^1\{t_{(r_f)}\}} = 1, o_i = m \quad (10)$$

$$p_{b^2\{t_{(r_l)}\}} - p_{b^1\{t_{(r_l)}\}} = -1, o_i = s \quad (11)$$

$$p_{b^2\{t_{(r)}\}} - p_{b^1\{t_{(r)}\}} < -1, o_i = ds \quad (12)$$

$$p_{b^2\{t_{(r)}\}} - p_{b^1\{t_{(r)}\}} > 1, o_i = dm \quad (13)$$

FIRST1=纪检 FIRST2=监察 SECOND1=干部 SECOND2=队伍 LEN1=2 LEN2=2 M
 FIRST1=监察 FIRST2=干部 SECOND1=队伍 LEN1=2 LEN2=1 M
 FIRST1=干部 FIRST2=队伍 SECOND1=建设 LEN1=2 LEN2=1 D-s

Figure 3. An Illustration of Reordering Example

As different location the model applied between our method and Xiong, we use more features than Xiong. On one hand, we use both first word and last word of translation fragment to record all possible merging. On the other hand, in order to distinguish the different phrases which have the same first word and the same last word, we bring in the fragment length as the other features. Integrated reordering principle and reordering example, we proposed a reordering example extracting algorithm which is shown in Algorithm1.

Algorithm 1: Reordering Example Extraction

```

1 : Input: translation output and referential sentence pair  $(t_0, r_0) (t_1, r_1) \dots (t_n, r_n)$ , the
    phrase length l;
2 :  $T = \{w_0 w_1 w_2 \dots w_{t.length-1}\}$ 
3 :  $\mathfrak{R} = \{w'_0 w'_1 \dots w'_{r.length-1}\}$ 
4 : E={};    (extracting result)
5 : for i=0,...l do
6 :   for k=0,...n do
7 :     for j=0,...len-1 do
8 :       phrase1 =  $\{w^j w_{j+1} w_{j+2} \dots w_{j+i}\}$ 
9 :       if  $\mathfrak{R}$ .isMatch(phrase1) then    (if referential sentence has matched phrase1)
10 :        phrase2 = {}
11 :        while  $\mathfrak{R}$ .isMatch(phrase2) && (phrase2.length <= j) then do
12 :          phrase2 +=  $\{w^{(j+i)++}\}$ 
13 :           $p1_{head_r} = \mathfrak{R}.get(phrase1)$                 (get the head number of phrase1)
14 :           $p1_{tail_r} = \mathfrak{R}.get(phrase1) + phrase1.length - 1$ 
15 :           $p2_{head_r} = \mathfrak{R}.get(phrase2)$ 
16 :           $p2_{tail_r} = \mathfrak{R}.get(phrase2) + phrase2.length - 1$ 
17 :          if i==1 then
18 :            if  $(p2_{head_r} - p1_{tail_r}) == 1$  then, add (phrase1,phrase2,m) to E
19 :            else if  $(p2_{tail_r} - p1_{head_r}) == -1$  then add (phrase1,phrase2,s) to E
20 :            else then add (p1,p2,d) to E
21 :            else then
22 :              if  $(p2_{head_r} - p1_{tail_r}) = 1$  then, add (phrase1,phrase2,m) to E
23 :              else if  $(p2_{tail_r} - p1_{head_r}) = -1$  then add (phrase1,phrase2,s) to E

```

```

24 : else if ( $p^2_{tail_r} - p^1_{head_r}$ ) < -1 then add (phrase1,phrase2,ds) to E
25 : else if ( $p^2_{head_r} - p^1_{tail_r}$ ) > 1 then add (phrase1,phrase2,dm) to E
26 : return E;

```

With the reordering example and the features, we can train the phrase producing model and phrases reordering model smoothly by Maximum Entropy. After this step, we get two Maximum Entropy models: phrase producing model and phrases reordering model.

3.3. Stack Shift-Reduce Algorithm

(Xiong *et al*, 2006) use a novel ITG scheme to achieve his reordering model which can handle some long-distance reordering problem. In this paper, we propose a novel stack shift-reduce algorithm which can handle long-distance reordering problem better. First of all, we push all our sentence to stack with the unit of word. After this step has been done, we got a stack filled with words. Then while pushing the words out, we use our trained phrase producing model to produce new phrases. The phrases produced in this step may not conform to syntax, but is more suitable to reorder. Thirdly, we got a stack filled with new phrases while the new phrases pushing into the stack. Finally, we use reordering model to reassociate the new phrases which the stack popped. This process will iterate until the stack is empty. After all these have been done, we get the ultimate output. The algorithm we design to reassociate the new phrases consist of these two parts : a words reordering part which merge the neighbor words by monotony or swapping like normal method, a phrases reordering part algorithm is illustrated as algorithm2.

Algorithm2 stack shift-reduce algorithm

```

1 : Input : new phrases stack :stack1, temporary reordering stack:stack2,
2 :  $stack1 = \{phrase_0, phrase_1, phrase_2, \dots, phrase_n\}$ 
3 : phrase1= stack1.pop() (Get the first element of s1)
4 : While (!s1.isEmpty())||(!s2.isEmpty()) do
5 : phrase2=s1.pop()
6 : operation =o(phrase1,phrase2) (Judging operation between phrase1 and phrase2)
7 : if operation ==M then
8 : phrase1+=phrase2
9 : add phrase1 to s1
10 : else if operation == S then
11 : phrase2+=phrase1
12 : add phrase2 to s1
13 : else if operation == DM then
14 : add phrase1 to s1
15 : add phrase2 to s2
16 : else if operation ==DS then
17 : add phrase1 to s2
18 : add phrase2 to s1
19 : phrase1= s1.pop()
20 : if s1.isEmpty()&&(!s2.isEmpty()) then (If s2 is not empty and s1 is empty)
21 : s1=s2 (Assign the value of s2 to s1 )
22 : s2.clear() (Empty s2)
23 : return phrase1 (Phrase1 is the output)

```

Unlike normal method, our stack shift-reduce algorithm has a novel Characteristics: We can reorder the phrases under any conditions if the phrases

reordering model is good enough. For example, we can successfully reorder sequence like “A F E C D B” that right order is “A B C D E F”. This property help us to settle the long distance reordering problem and other complex reordering problems. Table 1 and Table 2 are the example of our algorithm processing the Chinese sentence ‘高素质的纪检监察干部队伍建设。’.

Table 1. An Example of Word Merging Process

Phrase1	Phrase2	Operation	Stack
“高”	“素质”	M push	“高素质”
“高素质”	“的”	D	“高素质” “的”
“的”	“纪检”	D	“高素质” “的” “纪检”
“纪检”	“监察”	M push	“高素质” “的” “纪检监察”
“纪检监察”	“干部”	D	“高素质” “的” “纪检监察” “干部”
“干部”	“队伍”	M push	“高素质” “的” “纪检监察” “干部队伍”
“干部队伍”	“建设”	D	“高素质” “的” “纪检监察” “干部队伍” “建设”
“建设”	。	D	“高素质” “的” “纪检监察” “干部队伍” “建设” 。
。	Null	Push	“高素质” “的” “纪检监察” “干部队伍” “建设” 。

When we finished the word reordering, we got a new sentence with reunion phrases as the illustration of Table 1. After the word reordering, the example get a new sentence: ‘高素质 的 纪检监察 干部队伍 建设 。

Here has a little skill, we always firstly put the ‘.’ to our output sentence. So we usually finish the words reordering in the last word. The next step is phrase reordering, we get our first two elements with stack popped when this step start.

Table 2. An Example of Phrases Reordering Process

Phrase1	Phrase2	operation	Stack1	Stack2
“干部队伍”	“建设”	DM s2 push	“高素质”“的” “纪检监察” “干部队伍”	“建设”
“纪检监察”	“干部队伍”	M s1 push	“高素质”“的” “纪检监察” “干部队伍”	“建设”
“的”	“纪检监察干部 队伍”	DM s2 push	“高素质” “的”	“纪检监察干部队伍” “建设”
“高素质”	“的”	M s1 push	“高素质的”	“纪检监察干部队伍” “建设”
“高素质的”	Null	s2 Push	Empty	“高素质的” “纪检监察干部队伍” “建设”
“高素质的”	“纪检监察干部 队伍”	M s2 Push	Empty	“高素质的纪检监察干部队伍” “建设”
“高素质的纪检监察干部 队伍”	“建设”	S s2 Push	Empty	“建设高素质的纪检监察 干部队伍”
“建设高素质的纪检监察 干部队伍”	Null	Output	Empty	Empty

Table 2 provides an example of execution of our stack shift-reduce algorithm and phrases reordering model, which produced a perfect output with our model.

We finally notice that the post-reordering algorithm presented in this section always have too much reordering operations. Therefore LM was applied to restrain the reordering operation. With the LM introduced, a swap merging operation can only happen when the LM perplexity score got by swapped sentence fragment more than the LM perplexity score got by monotone sentence fragment .

4. Experiment

As lacking of syntactic tool in Uyghur, reordering model of Uyghur-to-Chinese MT always has no progress. Our model is suitable for Uyghur-to-Chinese MT in theory, and this is our original object to tackle. Meanwhile, English-to-Chinese MT

is hot spot in reordering model searching; we intend to contrast our model with the state-of-art reordering model. Therefore, we investigated the advantage of our post-reordering model by the following Uyghur-to-Chinese and English-to-Chinese translation experiment with our model and baseline SMTs.

4.1. Setup

We compare our proposed post-ordering model with two baseline SMTs: a standard phrase based SMT (PBMT) implemented by [21]. Moses a hierarchical phrase based SMT (HPBMT) implemented by Moses. Otherwise, we respectively use these systems to train Uyghur-to-Chinese and English-to-Chinese translation models for contrast test.

Table 3. Training Corpus

	Training	Dev(sentence)	Test
Uyghur&&Chinese	4,700	700	700
English&&Chinese	90,000	1000	1000

The corpus we used to test is shown in Table 3. These corpus from our lab are in news domain. Those model are trained in the standard manner with Moses3.0¹ which published in 3rd Feb 2015, grow-diag-final-and heuristics for symmetric word alignment, and max-phrase-length we limited is 11. Otherwise, we use Giza++² for word alignment and SRILM³ for 5-gram LM scoring during decoding. Our Maximum Entropy model is realized by Apache OpenNlp⁴ library which is a machine learning based toolkit for the processing of natural language text.

PBMT we test in experiment select msd-bidirectional-fe used in the WMT baselines as its lexicalized reordering model to make a better result. HPBMT, usually regarded as the state-of-art MTS with strong reordering ability, just select included simple distance model for contrast test. After training, we use default set of Moses to MERT our translation model [22].

4.2. Results

Table 4 shows the results in BLEU in case-insensitive evaluation with the compared methods [23]. PBMT group means the category of phrase based MT; HPBMT group means the category of hierarchical phrase based MT; PR&&PBMT group means the category of phrase based MT used Maximum Entropy post-reordering model; PR&&HPBMT means the category of hierarchical phrase based MT used Maximum Entropy post-reordering model.

Table 4. The Result of Experiment

¹ <http://www.statmt.org/moses/>

² <http://www.statmt.org/moses/giza/GIZA++.html>

³ <http://www.speech.sri.com/projects/srilm/>

⁴ <http://opennlp.apache.org/index.html>

Group	Test Set(sentences)					
	English-to-Chinese			Uyghur-to-Chinese		
	100	500	1,000	80	400	700
PBMT(BLEU)	27.73	25.25	26.26	28.73	28.12	27.60
HPBMT(BLEU)	28.90	26.12	27.25	28.94	28.45	28.50
PR&&PBMT(BLEU)	28.03	25.50	26.55	29.19	28.40	27.90
PR&&HPBMT(BLEU)	29.26	26.29	27.50	29.19	28.60	28.75

In table4, the groups used Maximum Entropy post-reordering model always get the higher score of the average raise of 0.31 than phrased based MT. Meanwhile, the groups used Maximum Entropy post-reordering model always get the higher score of the average raise of 0.24 than the hierarchical phrase based MT. Otherwise, we found phrased based MT used our model sometimes have better performance than the hierarchical phrase based MT.

5. Discussion and Conclusion

From the result of Table 4. We can obviously obtain these conclusions.

1) Our model has positive correlation with the population and the quality of corpus, the better BLEU which the original translation get, the more raise which our model achieve. This is because of that the quality of two Maximum Entropy model having closely relationship with the quality of corpus.

2) Our unsupervised model can effectively reduce the reordering problem compared with these two baseline systems. We got significantly good result in our experiment which proven our model is useful.

3) Our two Maximum Entropy models enhanced orientation detection ability drastically. From the results of extracting reordering example and the results of reordering, we found more reordering operation in sentence.

4) Our stack shift-reduce algorithm can easily handle long distance reordering. Compared Uyghur-to-Chinese MT with English-to-Chinese MT, we found that result of Uyghur-to-Chinese MT used our model usually better than the result of English-to-Chinese. As Uyghur is SOV (subject-object-verb) language, Chinese and English are SVO (subject-verb-object) language, so the Uyghur-to-Chinese MT have more long-distance reordering phenomenon than English-to-Chinese. The result of Uyghur-to-Chinese better than the result of English-to-Chinese proven this view.

Although our model maybe weak in location reordering than the other reordering model, and we ignore sparse data problem, our model still get a significantly success. The reasons why our model can have a good performance are at least as follows. Firstly, we have more orientations defined in our reordering model that help us detect more possible operations. Secondly, Our stack shift-reduce algorithm make long distance reordering realization, so that we can simply reorder two phrases which is not neighbor. Thirdly, we train five features for Maximum Entropy that can capture the useful reordering information between original translation and reference. What is more, we specially use a phrase producing model to produce new phrase which is more adaptable for reordering. Finally, we can smoothly adjust our parameter of the models to make sure the fluent output.

6. Future Work

This paper proposed a useful post-reordering model which is especially fit for syntactic tool lacking language MT such as Uyghur-to-Chinese. Moreover, our model can easily merged with the other reordering method like pre-reordering model and reordering model embed in translation model. All in all, Translation sentence

ordering problem is a linguistic phenomenon. We can use statistical method or machine learning technique to decrease this problem, but far away to completely solve it.

To achieve better performance, we plan to apply some useful syntactic information to our model. Maybe we can not translate the source language perfectly in the near future, but we can make sure the translation will be fluent with useful syntactic information applying in post-reordering model. Of course, our next step job also involve mixing together our novel post-reordering model with other good reordering model to find another robust reordering model. Finally, our stack shift-reduce algorithm would be an effective decoding method with some improvement.

Acknowledgments

This research was partially supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDA06030400), the Xinjiang High-Tech Industrialization Project (Grant No. 201412101) and Young Creative Sci-Tech Talents Cultivation Project of Xinjiang Uyghur Autonomous Region (Grant No. 2013731021), the West Light Foundation of Chinese Academy of Sciences (Grant No. LHXZ201301 XBBS201216).

References

- [1] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra and R. L. Mercer, "The Mathematics of Statistical Machine Translation: Parameter estimation", *J. Computational Linguistics*, vol.19, no.2, (1993), pp.263-311.
- [2] Koehn, Philipp, F. J. Och and D. Marcu, "Statistical Phrase-Based Translation", Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-volume, Edmonton, Canada, (2003).
- [3] D. Chiang, "Hierarchical phrase-based translation", *J. Computational Linguistics*, vol.33, no.2, (2007), pp.201-228.
- [4] M. Collins, P. Koehn and I. Kučerov`a, "Clause restructuring for statistical machine translation", Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics, Michigan, USA, (2005).
- [5] C.-H. Li, D. Zhang, M. Li, M. Zhou, M. Li and Y. Guan, "A probabilistic approach to syntax-based reordering for statistical machine translation", Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Prague, Czech Republic, (2007).
- [6] K. Visweswariah, R. Rajkumar and A. Gandhe, "A Word Reordering Model for Improved Machine Translation", Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK, (2011).
- [7] M. R. Costa-juss`a and J.A. R. Fonollosa, "Statistical machine reordering", Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Sydney, Australia, (2006).
- [8] C. Dyer and P. Resnik, "Context-free reordering, finite-state translation", Proceedings of HLT-NAACL, Los Angeles, USA, (2010).
- [9] M. Holmqvist, S. Stymne, L. Ahrenberg and M. Merkel, "Alignment-based reordering for SMT", Proceedings of the Eight International Conference on Language Resources and Evaluation, Istanbul, Turkey, (2012).
- [10] F. Josef Och and H. Ney, "The alignment template approach to statistical machine translation", *J. Computational Linguistics*, vol.30, no.4, (2004), pp.417-449.
- [11] Tillmann and Christoph, "A block orientation model for statistical machine translation", Proceedings of HLT-NAACL, Boston, USA, (2004).
- [12] D. Han, K. Sudoh, X. Wu, K. Duh, H. Tsukada and M. Nagata, "Head finalization reordering for Chinese-to-Japanese machine translation", Proceedings of the Sixth Workshop on Syntax, Semantics and Structure in Statistical Translation, Jeju Island, South Korea, (2012).
- [13] H. Isozaki, K. Sudoh, H. Tsukada and K. Duh, "Head finalization: A simple reordering rule for sov languages", Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR, Uppsala, Sweden, (2010).
- [14] K. Sudoh, X. Wu, K. Duh, H. Tsukada and M. Nagata, "Syntax-Based Post-Ordering for Efficient Japanese-to-English Translation", *J. ACM Transactions on Asian Language Information Processing*, vol.12, no.3, (2013), pp.1-15.

- [15] I. Goto, M. Utiyama and E. Sumita, "Post-ordering by parsing for Japanese-English statistical machine translation", Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, Jeju Island, South Korea, (2012).
- [16] K. Sudoh, X. Wu, K. Duh, H. Tsukada and M. Nagata, "Post-ordering in statistical machine translation", Proceedings of the 13th Machine Translation Summit, Xiamen. China, (2011).
- [17] D. Xiong, Q. Liu and S. Lin, "Maximum entropy based phrase reordering model for statistical machine translation", Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia, (2006).
- [18] D. Wu, "Stochastic inversion transduction grammars and bilingual parsing of parallel corpora", J. Computational linguistics, vol.23, no.3, (1997), pp.377-403.
- [19] P. Xu, J. Kang, M. Ringgaard and F. Och, "Using a dependency parser to improve SMT for subject-object-verb languages", Proceedings of HLT-NAACL, Colorado, USA, (2009).
- [20] M. Galley and C. D. Manning, "a simple and effective hierarchical phrase reordering model", Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics Ohio, USA, (2008).
- [21] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi and E. Herbst, "Moses: Open source toolkit for statistical machine translation", Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Prague, Czech Republic, (2007).
- [22] F. J. Och, "Minimum error rate training in statistical machine translation", Proceedings of the 41th Annual Meeting of the Association for Computational Linguistics, Sapporo, Japan, (2003).
- [23] K. Papineni, S. Roukos, T. Ward and W. J. Zhu, "BLEU: a method for automatic evaluation of machine translation", Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, (2002).

Authors



Jinying Kong, he is now a Ph.D candidate in University of Chinese Academy of Sciences and his research interests include machine translation and natural language processing.



Yating Yang, she received her Ph.D from Graduate University of Chinese Academy of Sciences and her research interest is machine translation. She is currently an Associate Professor at Xinjiang Technical Institute of Physics and Chemistry, Chinese Academy of Science, China.



Xiao Li, he is now a professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His current research interests include multi-lingual information processing.



Lei Wang, he is now a professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His current research interests include multi-lingual information processing and application software.