A Novel Reordering Model Based on Multi-layer Phrase for Statistical Machine Translation

Yanqing He[1], Yu Zhou[2], Chengqing Zong[2], Huilin Wang[1]

[1]Institute of Scientific and Technical Information of China

[2]Institute of Automation, Chinese Academy of Sciences

{heyq,wanghl}@istic.ac.cn   {yzhou,cqzong}@nlpr.ia.ac.cn

Abstract

Phrase reordering is of great importance for statistical machine translation. According to the movement of phrase translation, the pattern of phrase reordering can be divided into three classes: monotone, BTG (Bracket Transduction Grammar) and hierarchy. It is a good way to use different styles of reordering models to reorder different phrases according to the characteristics of both the reordering models and phrases itself. In this paper a novel reordering model based on multi-layer phrase (PRML) is proposed, where the source sentence is segmented into different layers of phrases on which different reordering models are applied to get the final translation. This model has some advantages: different styles of phrase reordering models are easily incorporated together; when a complicated reordering model is employed, it can be limited in a smaller scope and replaced with an easier reordering model in larger scope. So this model better trade-offs the translation speed and performance simultaneously.

1 Introduction

In statistical machine translation (SMT), phrase reordering is a complicated problem. According to the type of phrases, the existing phrase reordering models are divided into two categories: contiguous phrase-based reordering models and non-contiguous phrase-based reordering models.

Contiguous phrase-based reordering models are designed to reorder contiguous phrases. In such type of reordering models, a contiguous phrase is reordered as a unit and the movements of phrase don't involve insertions inside the other phrases. Some of these models are content-independent, such as distortion models (Och and Ney, 2004; Koehn et al., 2003) which penalize translation according to jump distance of phrases, and flat reordering model (Wu, 1995; Zens et al., 2004)which assigns constant probabilities for monotone order and non-monotone order. These reordering models are simple and the contents of phrases have not been considered. So it's hard to obtain a satisfactory translation performance. Some lexicalized reordering models (Och et al., 2004; Tillmann 2004, Kumar and Byrne, 2005, Koehn et al., 2005) learn local orientations (monotone or non-monotone) with probabilities for each bilingual phrase from training data. These models are phrase-dependent, so improvements over content-independent reordering models are obtained. However, many parameters need to be estimated.

Non-contiguous phrase-based reordering models are proposed to process non-contiguous phrases and the movements of phrase involve insertion operations. This type of reordering models mainly includes all kinds of syntax-based models where more structural information is employed to obtain a more flexible phrase movement. Linguistically syntactic approaches (Yamada and Knight, 2001; Galley et al., 2004, 2006; Marcu et al., 2006; Liu et al., 2006; Shieber et al., 1990; Eisner, 2003; Quirk et al., 2005; Ding and Palmer, 2005) employ linguistically syntactic information to enhance their reordering capability and use non-contiguous phrases to

obtain some generalization. The formally syntax-based models use synchronous context-free grammar (SCFG) but induce a grammar from a parallel text without relying on any linguistic annotations or assumptions (Chiang, 2005; Xiong et al., 2006). A hierarchical phrase-based translation model (HPTM) reorganizes phrases into hierarchical ones by reducing sub-phrases to variables (Chiang 2005). Xiong et al. (2006) is an enhanced bracket transduction grammar with a maximum entropy-based reordering model (MEBTG). Compared with contiguous phrase-based reordering model, Syntax-based models need to shoulder a great deal of rules and have high computational cost of time and space. The type of reordering models has a weaker ability of processing long sentences and large-scale data, which heavily restrict their application.

The above methods have provided various phrases reordering strategies. According to the movement of phrase translation, the pattern of phrase reordering can be divided into three classes: monotone, BTG (Bracket Transduction Grammar) (Wu, 1995) and hierarchy. In fact for most sentences, there may be some phrases which have simple reordering patterns, such as monotone or BTG style. It is not necessary to reorder them with a complicated mechanism, e.g. hierarchy. It is a good idea that different reordering models are employed to reorder different phrases according to the characteristics of both the reordering models and the phrases itself. This paper thus gives a novel reordering model based on multi-layer phrase (PRML), where the source sentence is segmented into different layers of phrases on which different reordering models are applied to get the final translation. Our model has the advantages as follow: (1) PRML segments source sentence into multiple-layer phrases by using punctuation and syntactic information and the design of segmentation algorithm corresponds to each reordering model. Different reordering models are chosen for each layer of phrases. (2) In our model different reordering models can be easily integrated together to obtain a combination of multiple phrase reordering models. (3) Our model can incorporate some complicated reordering models. We limit them in relatively smaller scopes and replace them with easier reordering models in larger scopes. In such way our model better trade-offs

the translation speed and performance simultaneously. (4) Our segmentation strategy doesn't impair translation quality by controlling phrase translation tables to determine the scope of each reordering model in each source sentence. The poor phrase translations generated by the former reordering model, still have chances of being revised by the latter reordering model.

Our work is similar to the phrase-level system combination (Mellebeek et al., 2006). We share one important characteristic: we decompose input sentence into chunks and recompose the translated chunks in output. The differences are that, we segment the input sentence into multi-layer phrases and we reorder their translations with a multi-layer decoder.

The remainder of the paper is organized as follows: Section 2 gives our reordering model PRML. Section 3 presents the details of the sentence segmentation algorithm and the decoding algorithm. Section 4 shows the experimental results. Finally, the concluding remarks are given in Section 5.

2 The Model

We use an example to demonstrate our motivation. Figure 1 shows a Chinese and English sentence pair with word alignment. Each solid line denotes the corresponding relation between a Chinese word and an English word. Figure 2 shows our reordering mechanism. For the source sentence, the phrases in rectangle with round corner in row 2 obviously have a monotone translation order. For such kinds of phrase a monotone reordering model is enough to arrange their translations. Any two neighbor consecutive phrases in the ellipses in row 3 have a straight orders or inverted order. So BTG reordering model is appropriate to predict the order of this type of phrases. Inside the phrases in the ellipses in row 3 there are possibly more complicated hierarchical structures. For the phrase "通往 和平 之 路", a rule "X→⟨通往 X_1 之路, *towards the road to X_1*⟩" has the ascendancy over the monotone and BTG style of reordering model. Hierarchy style of reordering models, such as HPTM reordering model, can translate non-contiguous phrases and has the advantage of capturing the translation of such kind of phrases.

The whole frame of our model PRML is shown in Figure 3. PRML is composed of a

448

segmentation sentence module and a decoder which consists of three different styles of phrase reordering models. The source sentence is segmented into 3 layers of phrases: the original whole sentence, sub-sentences and chunks. The original whole sentence is considered as the first-layer phrase and is segmented into sub-sentences to get the second-layer phrase. By further segmenting these sub-sentences, the chunks are obtained as the third-layer phrase. The whole translation process includes three steps: **1)** In order to capture the most complicated structure of phrases inside chunks, HPTM reordering model are chosen to translate the chunks. So the translations of chunks are obtained. 2) Combine the bilingual chunks generated by step 1 with those bilingual phases generated by the MEBTG training model as the final phrase table and translate the sub-sentences with MEBTG reordering model, the translations of sub-sentences are obtained. 3) Combine the bilingual sub-sentences generated by step 2 with those bilingual phases generated by the Monotone training model as the final phrase table and translate the original whole sentences with monotone reorder-
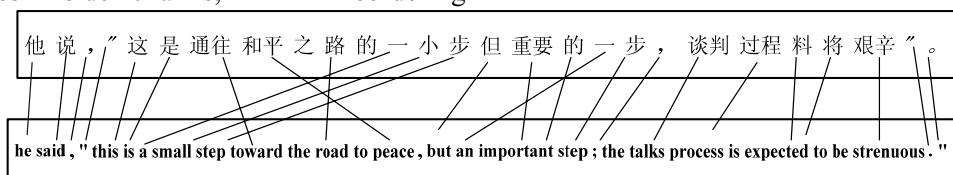

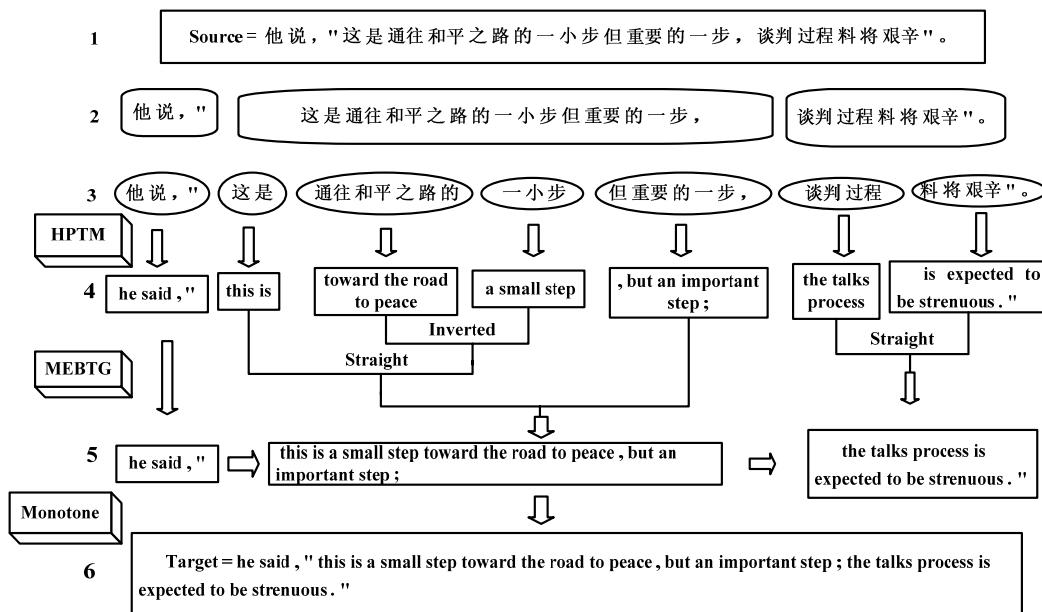Figure 1. An example of Chinese-English sentence pair with their word alignment
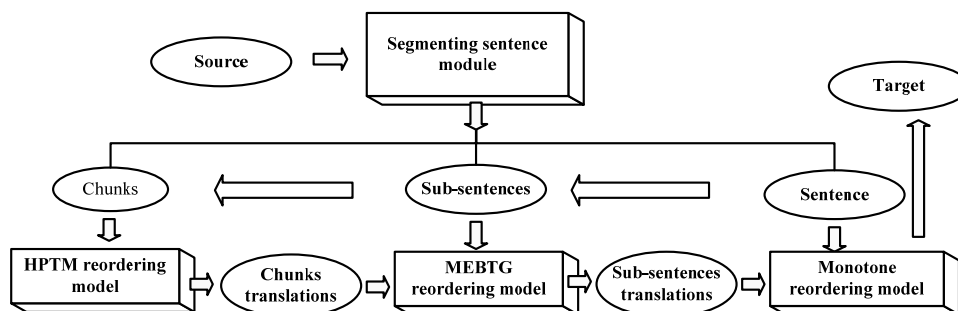

Figure 2. Diagram of Translation Using PRML.
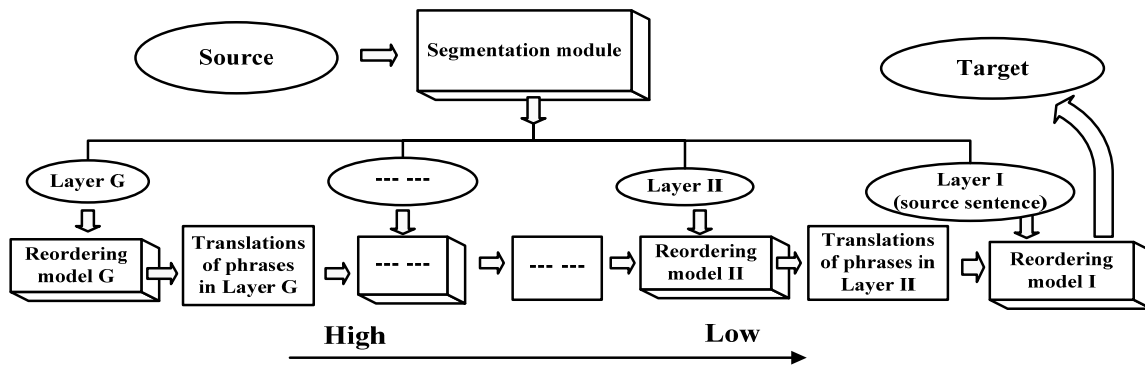

Figure 3. Frame of PRML

449

Figure 4. General frame of our model

ing model, the translations of the original whole sentences are obtained.

We also give a general frame of our model in Figure 4. In the segmentation module, an input source sentence is segmented into G layers of contiguous source strings, Layer 1, Layer 2, …, Layer G. The phrases of lower-order layer are re-segmented into the phrases of higher-order layer. The phrases of the same layer can be combined into the whole source sentence. The decoding process starts from the phrases of the highest-order layer. For each layer of phrases a reordering model is chosen to generate the translations of phrases according to their characteristics. The generated translations of phrases in the higher-order layer are fed as a new added translation source into the next lower-order reordering model. After the translations of the phrase in Layer 2 are obtained, they are fed into the Reordering model 1 as well as the source sentence (the phrase in Layer 1) to get the target translation.

Due to the complexity of the language, there may be some sentences whose structures don't conform to the pattern of the reordering models we choose. So in our segmentation module, if the sentence doesn't satisfy the segmentation conditions of current layer, it will be fed into the segmentation algorithm of the next layer. Even in the worst condition when the sentence isn't segmented into any phrase by segmentation module, it will be translated as the whole sentence to get the final translation by the highest-order reordering model.

Our model tries to grasp firstly the simple reordering modes in source sentence by the lower layer of phrase segmentations and controls more complicated reordering modes inside the higher layers of phrases. Then we choose some complicated reordering models to translate those phrases. Thus search space and computational complexity are both reduced. After obtaining the translation of higher layer's phrases, it is enough for simple reordering models to reorder them. Due to phrase segmentation some phrases may be translated poorly by the higher layer of reordering models, but they still have chances of being revised by the lower layer of reordering model because in lower layer of reordering model the input phrases have not these hard segmentation boundary and our model uses phrase translation tables to determine the scope of each reordering model.

There are two key issues in our model. The first one is how to segment the source sentence into different layers of phrases. The second one is how to choose a reordering model for different layer of phrases. In any case the design of segmenting sentence module should consider the characteristic of the reordering model of phrases.

3 Implementation

The segmentation module consists of the sub-sentence segmentation and chunk segmentation. The decoder combines three reordering models, HPTM, MEBTG, and a monotone reordering model.

3.1 Segmentation module

We define the sub-sentence as the word sequence which can be translated in monotone order. The following six punctuations: 。 ！ ？ ， ： ； in Chinese, and . ! ? , : ; in English are chosen as the segmentation anchor candidates. Except Chinese comma, all the other five punctuations can ex-

450

press one semantic end and another semantic beginning. In most of the time, it has high error risk to segment the source sentence by commas. So we get help from syntactic information of Chinese dependency tree to guarantee the monotone order of Chinese sub-sentences.

The whole process of sub-sentence segmentation includes training and segmenting.

Training: 1) The word alignment of training parallel corpus is obtained; **2)** The parallel sentence pairs in training corpus are segmented into sub-sentences candidates. For a Chinese-English sentence pair with their word alignment in training data, all bilingual punctuations are found firstly, six punctuations respectively "? ! 。 , : ; " in Chinese and "? ! . , : ;" in English. The punctuation identification number (id) sets in Chinese and English are respectively extracted. For a correct punctuation id pair (id_c, id_e), the phrase before id_e in English sentence should be the translation of the phrase before id_c in Chinese sentence, namely the number of the links[1] between the two phrases should be equal. In order to guarantees the property we calculate a bilingual alignment ratio for each Chinese-English punctuation id pair according to the following equation. For the punctuation id pair (id_c, id_e), bilingual alignment ratio consists of two value, Chinese-English alignment ratio (*CER*) and English-Chinese alignment ratio (*ECR*).

$$CER = \frac{\sum_{\substack{1 \le i \le id\_c \\ 1 \le j \le J}} \delta(A_{ij})}{\sum_{\substack{1 \le j \le id\_e \\ 1 \le i \le I}} \delta(A_{ij})} \qquad ECR = \frac{\sum_{\substack{1 \le j \le id\_e \\ 1 \le i \le I}} \delta(A_{ij})}{\sum_{\substack{1 \le i \le id\_c \\ 1 \le j \le J}} \delta(A_{ij})}$$

where $\delta(A_{ij})$ is an indicator function whose value is 1 when the word id pair (i, j) is in the word alignment and is 0 otherwise. *I* and *J* are the length of the Chinese English sentence pair. *CER* of a correct punctuation id pair will be equal to 1.0. So does *ECR*. In view of the error rate of word alignment, the punctuation id pairs will be looked as the segmentation anchor if both *CER* and *ECR* are falling into the threshold range (*minvalue*, *maxvalue*). Then all the punctuation id pairs are judged according to the same method and those punctuation id pairs

satisfying the requirement segment the sentence pair into sub-sentence pairs. **3)** The first word of Chinese sub-sentence in each bilingual sub-sentence pair is collected. We filter these words whose frequency is larger than predefined threshold to get segmentation anchor word set (*SAWS*).

Segmenting: 1) The test sentence in Chinese is segmented into segments by the six Chinese punctuation "。 ! ? , : ; " in the sentence. **2)** If the first word of a segment is in *SAWS* the punctuation at the end of the segment is chosen as the segmentation punctuation. **3)** If a segment satisfies the property of "dependency integrity" the punctuation at the end of the segment is also chosen as the segmentation punctuation. Here "dependency integrity" is defined in a dependency tree. Figure 5 gives the part output

ID	word	POS	head id	dependency type
1	美国	NR	3	NMOD
2	国会	NN	3	NMOD
3	议员	NN	4	SUB
4	表示	VV	0	ROOT
5	,	PU	4	P
6	人民币	NN	7	VMOD
7	低估	VV	9	VMOD
8	,	PU	9	P
… …	… …	… …	… …	… …

Figure 5. The part dependency parser output of a Chinese sentence.

of "lexical dependency parser"[2] for a Chinese sentence. There are five columns of data for each word which are respectively the word id, the word itself, its speech of part, the id of its head word and their dependency type. In the sentence the Chinese word sequence "*美国 国会 议员 表示* (US congressional representatives say that)" has such a property: Each word in the sequence has a dependency relation with the word which is still in the sequence except one word which has a dependency relation with the root, e.g. id 4. We define the property as "dependency integrity". Our reason is: a sub-sentence with the property of "dependency integrity" has relatively independent semantic meaning and a large possibility of monotone translation order. **4)** The union of the segmentation punctuations in step 2) and 3) are the final sub-sentence segmentation tags.

---

[1] Here a link between a Chinese word and an English word means the word alignment between them.

[2] http://www.seas.upenn.edu/ ~strctlrn/MSTParser/MSTParser.html

After sub-sentence segmentation, chunks segmentation is carried out in each sub-sentence. We define the chunks as the word sequence which can be translated in monotone order or inverted order. Here the knowledge of the "phrase structure parser"[3] and the "lexicalized dependency parser" are integrated to segment the sub-sentence into chunks. In a Chinese phrase structure parser tree the nouns phrase (NP) and preposition phrase (PP) are relatively independent in semantic expressing and relatively flexible in translation. So in the chunk segmentation, only the NP structure and PP structure in the Chinese structure parsing tree are found as phrase structure chunk. The process of chunk segmentation is described as follows: **1)** the test sub-sentence is parsed to get the phrase structure tree and dependency parsing tree; **2)** We traverse the phrase structure tree to extract sub-tree of "NP" and "PP" to obtain the phrase structure chunks. **3)** We mark off the word sequences with "dependency integrity" in the dependency tree. **4)** Both the two kinds of chunks are recombined to obtain the final result of chunk segmentation.

3.2 Decoding

Our decoder is composed of three styles of reordering models: HPTM, MEBTG and a monotone reordering model.

According to Chiang (2005), given the chunk c_{chunk}, a CKY parser finds $\hat{e}_{chunk}$, the English yield of the best derivation $\hat{D}_{hptm}$ that has Chinese yield c_{chunk}:

$$\hat{e}_{chunk} = e_{chunk}(\hat{D}_{hptm})$$
$$= e_{chunk}(\underset{C(D_{hptm})=C_{chunk}}{\arg\max}\ \Pr(D_{hptm}))$$

Here the chunks not the whole source sentence are fed into HPTM decoder to get the L-best translations and feature scores of the chunks. We combine all the chunks, their L-best translations and the feature scores into a phrase table, namely chunk phrase table. We only choose 4 translation scores (two translation probability based on frequency and two lexical weights based on word alignment) because the language model score, phrase penalty score and word penalty score will be re-calculated in the lower layer of reordering

[3] http://nlp.stanford.edu/software/lex-parser.shtml

model and need not be kept here. Meantime we change the log values of the scores into probability value. In the chunk phrase table each phrase pair has a Chinese phrase, an English phrase and four translations feature scores. In each phrase pair the Chinese phrase is one of our chunks, the English phrase is one translation of L-best of the chunk.

In MEBTG (Xiong et al., 2006), three rules are used to derive the translation of each sub-sentence: lexical rule, straight rule and inverted rule. Given a source sub-sentence $C_{sub-sent}$, it finds the final sub-sentence translation $\hat{E}_{sub-sent}$ from the best derivation $\hat{D}_{mebtg}$:

$$\hat{E}_{sub-sent} = E_{sub-sent}(\hat{D}_{mebtg})$$
$$= E(\underset{C(D_{mebtg})=C_{sub-sent}}{\arg\max}\ \Pr(D_{mebtg}))$$

Generally chunk segmentation will make some HPTM rules useless and reduce the translation performance. So in MEBTG we also use base phrase pair table which contains the contiguous phrase translation pairs consistent with word alignment. We merge the chunk phrase table and base phrase table together and feed them into MEBTG to translate each sub-sentence. Thus the K-Best translation and feature scores of each sub-sentence are obtained and then are recombined into a new phrase table, namely sub-sentence phrase table, by using the same method with chunk phrase table.

Having obtained the translation of each sub-sentence we generate the final translation of the whole source sentence by a monotone reordering model. Our monotone reordering model employs a log-linear direct translation model. Three phrase tables: chunk phrase table, sub-sentence phrase table and base phrase table are merged together and fed into the monotone decoder. Thus the decoder will automatically choose those phrases it need. In each phrase table each source phrase only has four translation probabilities for its candidate translation. So it's easy to merge them together. In such way all kinds of phrase pairs will automatically compete according to their translation probabilities. So our PRML model can automatically decide which reordering model is employed in each phrase scope of the whole source sentence. It's worth noting that the inputs of the three reordering

model have no segmentation tag. Because any segmentation for the input before decoding will influence the use of some rules or phrase pairs and may cause some rules or phrase pairs losses. It would be better to employ different phrase table to limit reordering models and let each decoder automatically decide reordering model for each segments of the input. Thus by controlling the phrase tables we apply different reordering models on different phrases. For each reordering model we perform the maximum BLEU training (Venugopal et al. 2005) on a development set. For HPTM the training is same as Chiang 2007. For MEBTG we use chunk phrase table and base table to obtain translation parameters. For monotone reordering model all the three phrase tables are merged to get translation weights.

4   Experiments

This section gives the experiments with Chinese-to-English translation task in news domain. Our evaluation metric is case-insensitive BLEU-4 (Papineni et al. 2002). We use NIST MT 2005, NIST MT 2006 and NIST MT 2008 as our test data. Our training data is filtered from the LDC corpus[4]. Table 1 gives the statistics of our data.

4.1   Evaluating translation Performance

We compare our PRML against two baselines: MEBTG system developed in house according to Xiong (2006, 2008) and HPTM system[5] in PYTHON based on HPTM reordering model (Chiang 2007). In MEBTG phrases of up to 10 words in length on the Chinese side are extracted and reordering examples are obtained without limiting the length of each example. Only the last word of each reordering example is used as lexical feature in training the reordering model by the maximum entropy based classifier[6]. We also set a swapping window size as 8 and the beam threshold as 10. It is worth noting that our MEBTG system uses cube-pruning algorithm (Chiang 2005) from bottom to up to generate the

---

Set	Language	Sentence	Vocabulary	A. S. L
Train data	Chinese	297,069	6,263	11.9
	English	297,069	8,069	13.6
NIST 05	Chinese	1,082	5669	28.2
	English	4,328	7575	32.7
NIST 06	Chinese	1,664	6686	23.5
	English	6,656	9388	28.9
NIST 08	Chinese	1,357	6,628	24.5
	English	5,428	9,594	30.8

Table 1. The statistics of training data and test data, A. S. L is average sentence length.

N-best list not the lazy algorithm of (Huang and Chiang, 2005). We also limit the length of the HPTM initial rules no more than 10 words and the number of non-terminals within two. In the decoding for the rules the beam pruning parameter is 30 and threshold pruning parameter is 1.0. For hypotheses the two pruning parameters are respectively 30 and 10. In our PRML *minvalue*=0.8, *maxvalue*=1.25, which are obtained by minimum error rate training on the development set. The predefined value for filtering *SAWS* is set as 100.

The translation performance of the three reordering model is shown in Table 2. We can find that PRML has a better performance than MEBTG with a relatively 2.09% BLEU score in NIST05, 5.60% BLEU score in NIST06 and 5.0% BLEU score in NIST08. This indicates that the chunk phrase table increases the reordering ability of MEBTG. Compared with HPTM, PRML has a comparable translation performance in NIST08. In NIST05 and NIST06 our model has a slightly better performance than HPTM. Because PRML limit hierarchical structure reordering model in chunks while HPTM use them in the whole sentence scope (or in a length scope), HPTM has a more complicated reordering mechanism than PRML. The experiment result shows even though we use easier reordering moels in larger scope, e.g. MEBTG and monoto-

Model	Nist05	Nist06	Nist08
HPTM	0.3183	0.1956	0.1525
MEBTG	0.3049	0.1890	0.1419
PRML	0.3205	0.1996	0.1495

Table 2. The translation performance

ne reordering model, we have a comparatively translation performance as HPTM.

4.2 Evaluating translation speed

Table 3 shows the average decoding time on test data for the three phrase reordering models on a double processor of a dual 2.0 Xeon machine. Time denotes mean time of per-sentence, in seconds. It is seen that PRML is the slower than MEBTG but reduce decoding time with a relatively 54.85% seconds in NIST05, 75.67% seconds in NIST06 and 65.28% seconds in NIST08. For PRML, 93.65% average decoding time in NIST05 is spent in HPTM, 4.89% time in MEBTG and 1.46% time in monotone reordering decoder.

Model	Nist05	Nist06	Nist08
HPTM	932.96	1235.21	675
MEBTG	43.46	27.16	10.24
PRML	421.20	300.52	234.33

Table 3. The average decoding time

4.3 Evaluating the performance of each layer of phrase table

In order to evaluate the performance of each reordering model, we run the monotone decoder with different phrase table in NIST05. Table 4 list the size of each phrase table. From the results in Table 5 it is seen that the performance of using three phrase tables is the best. Compared with the base phrase table, the translation performances are improved with relatively 10.86% BLEU score by adding chunk phrase table and 11% BLEU score by adding sub-sentence table. The result of row 4 has a comparable to the one in row 5. It indicates the sub-sentence phrase table has contained the information of HPTM reordering model. The case of row 4 to row 2 is the same.

Phrase table	Phrase pair
Base	732732
Chunk	86401
Sub-sentence	24710

Table 4. The size of each phrase table.

Phrase table	Reordering model	BLEU
Base	Monotone	0.2871
Base +chunk	monotone+HPTM	0.3180
Base +sub-sentence table	monotone+HPTM +MEBTG	0.3187
Base +chunk +subsentence	monotone+HPTM +MEBTG	0.3205

Table 5. The performance of phrase table

5 Conclusions

In this paper, we propose a novel reordering model based on multi-layer phrases (PRML), where the source sentence is segmented into different layers of phrases and different reordering models are applied to get the final translation. Our model easily incorporates different styles of phrase reordering models together, including monotone, BTG, and hierarchy or other more complicated reordering models. When a complicated reordering model is used, our model can limit it in a smaller scope and replace it with an easier reordering model in larger scope. In such way our model better trade-offs the translation speed and performance simultaneously.

In the next step, we will use more features to segment the sentences such as syntactical features or adding a dictionary to supervise the segmentation. And also we will try to incorporate other systems into our model to improve the translation performance.

6 Acknowledgements

References

David Chiang. 2005. *A hierarchical phrase-based model for statistical machine translation*. In Proceedings of ACL 2005, pages 263–270.

David Chiang, 2007. *Hierarchical Phrase-based Translation*. Computational Linguistics,33(2):201-228.

Yuan Ding and Martha Palmer. 2005. *Machine translation using probabilistic synchronous dependency insertion grammars*. In proceeding of 43th Meeting of the Association for Computational Linguistics, 541-548

Jason Eisner. 2003. *Learning non-isomorphic tree mappings for machine translation*. In proceedings of the 41th Meeting of the Association for Computational Linguistics (companion volume).

Michel Galley, Mark Hopkins, Kevin Knight and Daniel Marcu. 2004. *What's in a translation rule?* In proceedings of HLTNAACL- 2004.

Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, Ignacio Thayer. 2006. *Scalable Inference and Training of Context-Rich Syntactic Translation Models.* In Proceedings of the joint conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics. Sydney, Australia.

Liang Huang and David Chiang. 2005. *Better k-best parsing*. In Proceedings of the Ninth International Workshop on Parsing Technology, Vancouver, October, pages 53–64.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu, 2002. *BLEU: a method for automatic evaluation of machine translation*. In Proceedings of the 40th Annual Meeting of the ACL. page 311-318, Philadelphia, PA.

Philipp Koehn, Franz J. Och and Daniel Marcu. 2003. *Statistical phrase-based translation*. In proceedings of HLT-NAACL-03, 127-133

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne and David Talbot. 2005. *Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation*. In International Workshop on Spoken Language Translation.

Shankar Kumar and William Byrne. 2005. *Local phrase reordering models for statistical machine translation*. In Proceedings of HLT-EMNLP.

Yang Liu, Qun Liu and Shouxun Lin. 2006. *Tree-to-String Alignment Template for Statistical Machine Translation.* In proceedings of ACL-06, 609-616.

Daniel Marcu and William Wong. 2002. *A phrase-based, joint probability model for statistical machine translation.* In proceedings of EMNLP-02, 133-139.

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin Knight. 2006. *SPMT: Statistical Machine Translation with Syntactified Target Language Phrases*. In Proceedings of EMNLP-2006, 44-52, Sydney, Australia

Bart Mellebeek, Karolina Owczarzak, Josef Van Genabith, Andy Way. 2006. *Multi-Engine Machine Translation By Recursive Sentence Decomposition.* In Proceedings of AMTA 2006

Franz J. Och and Hermann Ney. 2004. *The alignment template approach to statistical machine translation.* Computational Linguistics, 30(4):417-449

Franz Josef Och, Ignacio Thayer, Daniel Marcu, Kevin Knight, Dragos Stefan Munteanu, Quamrul Tipu, Michel Galley, andMark Hopkins. 2004. *Arabic and Chinese MT at USC/ISI*. Presentation given at NIST Machine Translation Evaluation Workshop.

Chris Quirk, Arul Menezes and Colin Cherry. 2005. *Dependency treelet translation: Syntactically informed phrasal SMT*. In proceedings of the 43th Meeting of the Association for Computational Linguistics, 271-279

S. Shieber and Y. Schabes. 1990. *Synchronous tree adjoining grammars*. In proceedings of COLING-90.

Christoph Tillmann. 2004. *A block orientation model for statistical machine translation*. In HLT-NAACL, Boston, MA, USA.

Ashish Venugopal, Stephan Vogel and Alex Waibel. 2003. *Effective Phrase Translation Extraction from Alignment Models*, in Proceedings of the 41st ACL, 319-326.

Dekai Wu. 1995. *Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora.* In proceeding of IJCAL 1995, 1328-1334,Montreal, August.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. *Maximum Entropy Based phrase reordering model for statistical machine translation*. In proceedings of COLING-ACL, Sydney, Australia.

Deyi Xiong, Min Zhang, Ai Ti Aw, Haitao Mi, Qun Liu and Shouxun Lin. *Refinements in BTG-based Statistical Machine Translation.* In Proceedings of IJCNLP 2008.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In proceedings of the 39th Meeting of the ACL, 523-530.

R. Zens, H. Ney, T. Watanabe, and E. Sumita. 2004. *Reordering Constraints for Phrase-Based Statistical MachineTranslation.* In Proceedings of CoLing 2004, Geneva, Switzerland, pp. 205-211.