Phrase Reordering Model Integrating Syntactic Knowledge for SMT

Dongdong Zhang, Mu Li, Chi-Ho Li, Ming Zhou

Microsoft Research Asia
Beijing, China
{dozhang,muli,chl,mingzhou}@microsoft.com

Abstract

Reordering model is important for the statistical machine translation (SMT). Current phrase-based SMT technologies are good at capturing local reordering but not global reordering. This paper introduces syntactic knowledge to improve global reordering capability of SMT system. Syntactic knowledge such as boundary words, POS information and dependencies is used to guide phrase reordering. Not only constraints in syntax tree are proposed to avoid the reordering errors, but also the modification of syntax tree is made to strengthen the capability of capturing phrase reordering. Furthermore, the combination of parse trees can compensate for the reordering errors caused by single parse tree. Finally, experimental results show that the performance of our system is superior to that of the state-of-the-art phrase-based SMT system.

1 Introduction

In the last decade, statistical machine translation (SMT) has been widely studied and achieved good translation results. Two kinds of SMT system have been developed, one is phrase-based SMT and the other is syntax-based SMT.

In phrase-based SMT systems (Koehn et al., 2003; Koehn, 2004), foreign sentences are firstly segmented into *phrases* which consists of adjacent words. Then source phrases are translated into target phrases respectively according to knowledge usually learned from bilingual parallel corpus. Finally the most likely target sentence based on a certain statistical model is inferred by combining and reordering the target phrases with the aid of search algorithm. On the other hand, syntax-based SMT systems (Liu et al., 2006; Yamada et al., 2001) mainly depend on parse trees to complete the translation of source sentence.

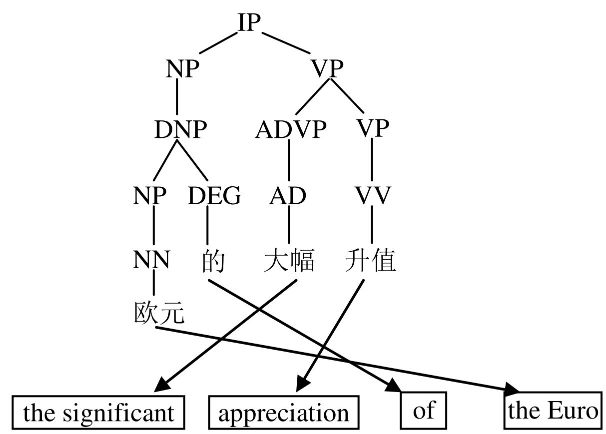

Figure 1: A reordering example

As studied in previous SMT projects, language model, translation model and reordering model are the three major components in current SMT systems. Due to the difference between the source and target languages, the order of target phrases in the target sentence may differ from the order of source phrases in the source sentence. To make the translation results be closer to the target language style, a mathematic model based on the statistic theory is constructed to reorder the target phrases. This statistic model is called as *reordering model*. As shown in Figure 1, the order of the translations of “欧元” and “的” is changed. The order of the

Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 533–540, Prague, June 2007.

translation of “欧元/的” and “大幅/升值” is altered as well. The former reordering case with the smaller distance is usually referred as *local reordering* and the latter with the longer distance reordering as *global reordering*. Phrase-based SMT system can effectively capture the local word reordering information which is common enough to be observed in training data. But it is hard to model global phrase reordering. Although syntactic knowledge used in syntax-based SMT systems can help reorder phrases, the resulting model is usually much more complicated than a phrase-based system.

There have been considerable amount of efforts to improve the reordering model in SMT systems, ranging from the fundamental distance-based distortion model (Och and Ney, 2004; Koehn et al., 2003), flat reordering model (Wu, 1996; Zens et al., 2004; Kumar et al., 2005), to lexicalized reordering model (Tillmann, 2004; Kumar et al., 2005; Koehn et al., 2005), hierarchical phrase-based model (Chiang, 2005), and maximum entropy-based phrase reordering model (Xiong et al., 2006). Due to the absence of syntactic knowledge in these systems, the ability to capture global reordering knowledge is not powerful. Although syntax-based SMT systems (Yamada et al., 2001; Quirk et al., 2005; Liu et al., 2006) are good at modeling global reordering, their performance is subject to parsing errors to a large extent.

In this paper, we propose a new method to improve reordering model by introducing syntactic information. Syntactic knowledge such as boundary of sub-trees, part-of-speech (POS) and dependency relation is incorporated into the SMT system to strengthen the ability to handle global phrase reordering. Our method is different from previous syntax-based SMT systems in which the translation process was modeled based on specific syntactic structures, either phrase structures or dependency relations. In our system, syntactic knowledge is used just to decide where we should combine adjacent phrases and what their reordering probability is. For example, according to the syntactic information in Figure 1, the phrase translation combination should take place between “大幅” and “升值” rather than between “的” and “大幅”. Moreover, the non-monotone phrase reordering should occur between “欧元/的” and “大幅/升值” rather than between “欧元/的” and “大幅”. We train a maximum entropy model, which is able to integrate rich syntactic knowledge, to estimate phrase reordering probabilities. To enhance the performance of phrase reordering model, some modification on the syntax trees are also made to relax the phrase reordering constraints. Additionally, the combination of other kinds of syntax trees is introduced to overcome the deficiency of single parse tree. The experimental results show that the performance of our system is superior to that of the state-of-art phrase-based SMT system.

The roadmap of this paper is: Section 2 gives the related work. Section 3 introduces our model. Section 4 explains the generalization of reordering knowledge. The procedures of training and decoding are described in Section 5 and Section 6 respectively. The experimental results are shown in Section 7. Section 8 concludes the paper.

2 Related Work

The Pharaoh system (Koehn et al., 2004) is well known as the typical phrase-based SMT system. Its reordering model is designed to penalize translation according to jump distance regardless of linguistic knowledge. This method just works well for language pairs that trend to have similar word-orders and it has nothing to do with global reordering.

A straightforward reordering model used in (Wu, 1996; Zens et al., 2004; Kumar et al., 2005) is to assign constant probabilities to monotone reordering and non-monotone reordering, which can be flexible depending on the different language pairs. This method is also adopted in our system for non-peer phrase reordering.

The lexicalized reordering model was studied in (Tillmann, 2004; Kumar et al., 2005; Koehn et al., 2005). Their work made a step forward in integrating linguistic knowledge to capture reordering. But their methods have the serious data sparseness problem.

Beyond standard phrase-based SMT system, a CKY style decoder was developed in (Xiong et al., 2006). Their method investigated the reordering of any two adjacent phrases. The limited linguistic knowledge on the boundary words of phrases is used to construct the phrase reordering model. The basic difference to our method is that no syntactic knowledge is introduced to guide the global phrase reordering in their system. Besides boundary words, our phrase reordering model also integrates more significant syntactic knowledge such as POS information and dependencies from the syntax tree, which can avoid some intractable phrase reordering errors.

A hierarchical phrase-based model was proposed by (Chiang, 2005). In his method, a synchronous CFG is used to reorganize the phrases into hierarchical ones and grammar rules are automatically learned from corpus. Different from his work, foreign syntactic knowledge is introduced into the synchronous grammar rules in our method to restrict the arbitrary phrase reordering.

Syntax-based SMT systems (Yamada et al., 2001; Quirk et al., 2005; Liu et al., 2006) totally depend on syntax structures to complete phrase translation. They can capture global reordering by simply swapping the children nodes of a parse tree. However, there are also reordering cases which do not agree with syntactic structures. Furthermore, their model is usually much more complex than a phrase-based system. Our method exactly attempts to integrate the advantages of phrase-based SMT system and syntax-based SMT system to improve the phrase reordering model. Phrase translation in our system is independent of syntactic structures.

3 The Model

In our work, we focus on building a better reordering model with the help of source parsing information. Although we borrow some fundamental elements from a phrase-based SMT system such as the use of bilingual phrases as basic translation unit, we are more interested in introducing syntactic knowledge to strengthen the ability to handle global reordering phenomena in translation.

3.1 Definitions

Given a foreign sentence f and its syntactic parse tree T, each leaf in T corresponds to a single word in f and each sub-tree of T exactly covers a phrase f_i in f which is called as *linguistic phrase*. Except linguistic phrases, any other phrase is regarded as *non-linguistic phrase*. The *height* of phrase f_i is defined as the distance between the root node of T and the root node of the maximum sub-tree which exactly covers f_i. For example, in Figure 1 the phrase "大幅" has the maximum sub-tree rooting at ADJP and its height is 3. The height of phrase "的" is 4 since its maximum sub-tree roots at ADBP instead of AD. If two adjacent phrases have the same height, we regard them as *peer phrases*.

In our model, we make use of *bilingual phrases* as well, which refer to source-target aligned phrase pairs extracted using the same criterion as most phrase-based systems (Och and Ney, 2004).

3.2 Model

Similar to the work in Chiang (2005), our translation model can be formulated as a weighted synchronous context free grammar derivation process. Let D be a derivation that generates a bilingual sentence pair $\langle f, e\rangle$, in which f is the given source sentence, the statistical model that is used to predict the translation probability $p(e|f)$ is defined over Ds as follows:

$$p(e|f) \propto p(D) \propto p_{lm}(e)^{\lambda_{lm}} \times \prod_{i} \prod_{X \to \langle\gamma,\alpha\rangle \in D} \phi_i(X \to \langle\gamma,\alpha\rangle)^{\lambda_i}$$

where $p_{lm}(e)$ is the language model, $\Phi_i(X \rightarrow \langle\gamma,\alpha\rangle)$ is a feature function defined over the derivation rule $X \rightarrow \langle\gamma,\alpha\rangle$, and λ_i is its weight.

Although theoretically it is ideal for translation reorder modeling by constructing a synchronous context free grammar based on bilingual linguistic parsing trees, it is generally a very difficult task in practice. In this work we propose to use a small synchronous grammar constructed on the basis of bilingual phrases to model translation reorder probability and constraints by referring to the source syntactic parse trees. In the grammar, the source / target words serve as terminals, and the bilingual phrases and combination of bilingual phrases are presented with non-terminals. There are two non-terminals in the grammar except the start symbol S: Y and Z. The general derivation rules are defined as follows:

a) Derivations from non-terminal to non-terminals are restricted to binary branching forms;
b) Any non-terminals that derives a list of terminals, or any combination of two non-terminals, if the resulting source string won't cause any cross-bracketing problems in the source parse tree (it exactly corresponds to a linguistic phrase in binary parse trees), are reduced to Y;
c) Otherwise, they are reduced to Z.

Table 1 shows a complete list of derivation rules in our synchronous context grammar. The first nine grammar rules are used to constrain phrase reordering during phrase combination. The last two rules are used to represent bilingual phrases. Rule (10) is the start grammar rule to generate the entire sentence translation.

Rule Name	Rule Content
Rule (1)	$Y \rightarrow \langle Y_1Y_2, Y_1Y_2 \rangle$
Rule (2)	$Y \rightarrow \langle Y_1Y_2, Y_2Y_1 \rangle$
Rule (3)	$Y \rightarrow \langle Z_1Z_2, Z_1Z_2 \rangle$
Rule (4)	$Y \rightarrow \langle Y_1Z_2, Y_1Z_2 \rangle$
Rule (5)	$Y \rightarrow \langle Z_1Y_2, Z_1Y_2 \rangle$
Rule (6)	$Z \rightarrow \langle Y_1Z_2, Y_1Z_2 \rangle$
Rule (7)	$Z \rightarrow \langle Z_1Y_2, Z_1Y_2 \rangle$
Rule (8)	$Z \rightarrow \langle Z_1Z_2, Z_1Z_2 \rangle$
Rule (9)	$Z \rightarrow \langle Y_1Y_2, Y_1Y_2 \rangle$
Rule (10)	$S \rightarrow \langle Y_1, Y_1 \rangle$
Rule (11)	$Z \rightarrow \langle Z_1, Z_1 \rangle$
Rule (12)	$Y \rightarrow \langle Y_1, Y_1 \rangle$

Table 1: Synchronous grammar rules

Rule (1) and Rule (2) are only applied to two adjacent peer phrases. Note that, according to the constraints of foreign syntactic structures, only Rule (2) among all rules in Table 1 can be applied to conduct non-monotone phrase reordering in our framework. This can avoid arbitrary phrase reordering. For example, as shown in Figure 1, Rule (1) is applied to the monotone combination of phrases "欧元" and "的", and Rule (2) is applied to the non-monotone combination of phrases "欧元/的" and "大幅/升值". However, the non-monotone combination of "的" and "大幅" is not allowed in our method since there is no proper rule for it.

Non-linguistic phrases are involved in Rule (3)~(9). We do not allow these grammar rules for non-monotone combination of non-peer phrases, which really harm the translation results as proved in experimental results. Although these rules violate the syntactic constraints, they not only provide the option to leverage non-linguistic translation knowledge to avoid syntactic errors but also take advantage of phrase local reordering capabilities. Rule (3) and Rule (8) are applied to the combination of two adjacent non-linguistic phrases. Rule (4)~(7) deal with the situation where one is a linguistic phrase and the other is a non-linguistic phrase. Rule (9) is applied to the combination of two adjacent linguistic phrases but their combination result is not a linguistic phrase.

Rule (11) and Rule (12) are applied to generate bilingual phrases learned from training corpus.

Table 2 demonstrates an example how these rules are applied to translate the foreign sentence "欧元/的/大幅/升值" into the English sentence "the significant appreciation of the Euro".

Step	Partial derivations	Rule
1	$S \rightarrow \langle Y_1, Y_1 \rangle$	(10)
2	$\rightarrow \langle Y_2Y_3, Y_3Y_2 \rangle$	(2)
3	$\rightarrow \langle Y_4Y_5Y_3, Y_3Y_5Y_4 \rangle$	(2)
4	→⟨欧元 Y_5Y_3, Y_3Y_5 the Euro⟩	(12)
5	→⟨欧元 的 Y_3, Y_3 of the Euro⟩	(12)
6	→⟨欧元 的 Y_6Y_7, Y_6Y_7 of the Euro⟩	(1)
7	→⟨欧元 的 大幅 Y_7, the significant Y_7 of the Euro⟩	(12)
8	→⟨欧元 的 大幅 升值, the significant appreciation of the Euro⟩	(12)

Table 2: Example of application for rules

However, there are always other kinds of bilingual phrases extracted directly from training corpus, such as ⟨欧元, the Euro⟩ and ⟨的 大幅 升值, 's significant appreciation⟩, which can produce different candidate sentence translations. Here, the phrase "的 大幅 升值" is a non-linguistic phrase. The above derivations can also be rewritten as $S \rightarrow \langle Y_1, Y_1 \rangle \rightarrow \langle Y_2Z_3, Y_2Z_3 \rangle \rightarrow$⟨欧元 Z_3, the Euro Z_3⟩→⟨欧元的 大幅 升值, the Euro 's significant appreciation⟩, where Rule (10), (4), (12) and (11) are applied respectively.

3.3 Features

Similar to the default features in Pharaoh (Koehn, Och and Marcu 2003), we used following features to estimate the weight of our grammar rules. Note that different rules may have different features in our model.

- The lexical weights $p_{lex}(\gamma|\alpha)$ and $p_{lex}(\alpha|\gamma)$ estimating how well the words in α translate the words in γ. This feature is only applicable to Rule (11) and Rule (12).
- The phrase translation weights $p_{phr}(\gamma|\alpha)$ and $p_{phr}(\alpha|\gamma)$ estimating how well the terminal words of α translate the terminal words of γ, This feature is only applicable to Rule (11) and Rule (12).
- A word penalty $\exp(|\alpha|)$, where $|\alpha|$ denotes the count of terminal words of α. This feature is only applicable to Rule (11) and Rule (12).
- A penalty exp(1) for grammar rules analogous to Pharaoh's penalty which allows the model to learn a preference for longer or shorter derivations. This feature is applicable to all rules in Table 1.
- Score for applying the current rule. This feature is applicable to all rules in Table 1. We will explain the score estimation in detail in Section 3.4.

3.4 Scoring of Rules

Based on the syntax constraints and involved non-terminal types, we separate the grammar rules into three groups to estimate their application scores which are also treated as reordering probabilities.

For Rule (1) and Rule (2), they strictly comply with the syntactic structures. Given two peer phrases, we have two choices to use one of them. Thus, we use maximum entropy (ME) model algorithm to estimate their reordering probabilities separately, where the boundary words of foreign phrases and candidate target translation phrases, POS information and dependencies are integrated as features. As listed in Table 3, there are totally twelve categories of features used to train the ME model. In fact, the probability of Rule (1) is just equal to the supplementary probability of Rule (2), and vice versa.

For Rule (3)~(9), according to the syntactic structures, their application is determined since there is only one choice to complete reordering, which is similar to the "glue rules" in Chiang (2005). Due to the appearance of non-linguistic phrases, non-monotone phrase reordering is not allowed in these rules. We just assign these rules a constant score trained using our implementation of Minimum Error Rate Training (Och, 2003b), which is 0.7 in our system.

For Rule (10)~(12), they are also determined rules since there is no other optional rules competing with them. Constant score is simply assigned to them as well, which is 1.0 in our system.

Fea.	Description
LS1	First word of first foreign phrase
LS2	First word of second foreign phrase
RS1	Last word of first foreign phrase
RS2	Last word of second foreign phrase
LT1	First word of first target phrase
LT2	First word of second target phrase
RT1	Last word of first target phrase
RT2	Last word of second target phrase
LPos	POS of the node covering first foreign phrase
RPos	POS of the node covering second foreign phrase
Cpos	POS of the node covering the combination of foreign phrases
DP	Dependency between the nodes covering two single foreign phrases respectively

Table 3: Feature categories used for ME model

4 The Generalization of Reordering Knowledge

4.1 Enriching Parse Trees

The grammar rules proposed in Section 3 are only applied to binary syntax tree nodes. For n-ary syntax trees (n>2), some modification is needed to generate more peer phrases. As shown in Figure 2(a), the syntactic tree of Chinese sentence "广东省/高新技术/产品/出口" (Guangdong/high-tech/products/export), parsed by the Stanford Parser (Klein, 2003), has a 3-ary sub-tree. Referring to its English translation result "export of high-tech products in Guangdong", we understand there should be a non-monotone combination between the phrases "广东省" and "高新技术/产品". However, "高新技术/产品" is not a linguistic phrase though its component phrases “高新技术” and “产品” are peer phrases. To avoid the conflict with the Rule (2), we just add some extra *virtual nodes* in the *n*-ary sub-trees to make sure that only binary sub-trees survive in the modified parse tree. Figure 2(b) is the modification result of the syntactic tree from Figure 2(a), where two virtual nodes with the new distinguishable POS of M are added.

In general, we add virtual nodes for each set of the continuous peer phrases and let them have the same height. Thus, for a *n*-ary sub-tree, there are $\sum_{i=1}^{n-1}(n-i) = (n-1)^2/2$ virtual nodes being added where $n>2$. The phrases exactly covered by the virtual nodes are called as *virtual peer phrases*.

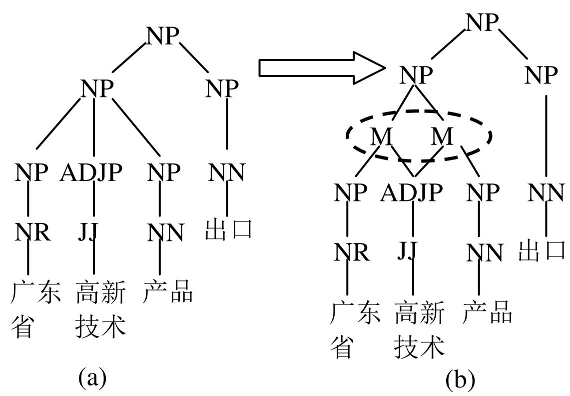

Figure 2: Example of syntax tree modification

4.2 Combination of Parse Trees

It is well known that parse errors in syntactic trees always are inescapable even if the state-of-the-art parser is used. Incorrect syntactic knowledge may harm the reordering probability estimation. To minimize the impact of parse error of a single tree, more parse trees are introduced. To support the combination of parse trees, the synchronous grammar rules are applied independently, but they will compete against each other with the effect of other models such as language model.

In our system, we combine the parse trees generated respectively by Stanford parser (Klein, 2003) and a dependency parser developed by (Zhou, 2000). Compared with the Stanford parser, the dependency parser only conducts shallow syntactic analysis. It is powerful to identify the base NPs and base VPs and their dependencies. Additionally, dependency parser runs much faster. For example, it took about three minutes for the dependency parser to parse one thousand sentences with average length of 25 words, but the Stanford parser needs about one hour to complete the same work. More importantly, as shown in the experimental results, the dependency parser can achieve the comparable quality of final translation results with Stanford parser in our system.

5 The Decoder

We developed a CKY style decoder to complete the sentence translation. A two-dimension array *CA* is constructed to store all the local candidate phrase translation and each valid cell CA_{ij} in *CA* corresponds to a foreign phrase where *i* is the phrase start position and *j* is the phrase end position. The cells in *CA* are filled in a bottom-up way. Firstly we fill in smaller cells with the translation in bilingual phrases learned from corpus. Then the candidate translation in the larger cell CA_{ij} is generated based on the content in smaller adjacent cells CA_{ik} and CA_{k+1j} with the *monotone combination* and *non-monotone combination*, where $i \le k \le j$. To reduce the cost of system resources, the well known pruning methods, such as histogram pruning, threshold pruning and recombination, are used to only keep the top *N* candidate translation in each cell.

6 Training

Similar to most state-of-the-art phrase-based SMT systems, we use the SRI toolkit (Stolcke, 2002) for language model training and Giza++ toolkit (Och and Ney, 2003) for word alignment. For reordering model training, two kinds of parse trees for each foreign sentence in the training corpus were obtained through the Stanford parser (Klein, 2003) and a dependency parser (Zhou, 2000). After that, we picked all the foreign linguistic phrases of the same sentence according to syntactic structures. Based on the word alignment results, if the aligned target words of any two adjacent foreign linguistic phrases can also be formed into two valid adjacent phrase according to constraints proposed in the phrase extraction algorithm by Och (2003a), they will be extracted as a reordering training sample. Finally, the ME modeling toolkit developed by Zhang (2004) is used to train the reordering model over the extracted samples.

7 Experimental Results and Analysis

We conducted our experiments on Chinese-to-English translation task of NIST MT-05 on a 3.0GHz system with 4G RAM memory. The bilingual training data comes from the FBIS corpus. The Xinhua news in GIGAWORD corpus is used to train a four-gram language model. The development set used in our system is the NIST MT-02 evaluation test data.

For phrase extraction, we limit the maximum length of foreign and English phrases to 3 and 5 respectively. But there is no phrase length constraint for reordering sample extraction. About 1.93M and 1.1M reordering samples are extracted from the FBIS corpus based on the Stanford parser and the dependency parser respectively. To reduce the search space in decoder, we set the histogram pruning threshold to 20 and relative pruning threshold to 0.1.

In the following experiments, we compared our system performance with that of the other state-of-the-art systems. Additionally, the effect of some strategies on system performance is investigated as well. Case-sensitive BLEU-4 score is adopted to evaluate system performance.

7.1 Comparing with Baseline SMT system

Our baseline system is Pharaoh (Koehn, 2004). Xiong's system (Xiong, et al., 2006) which used ME model to train the reordering model is also regarded as a competitor. To have a fair comparison, we used the same language model and translation model for these three systems. The experimental results are showed in Table 4.

System	Bleu Score
Pharaoh	0.2487
Xiong's System	0.2616
Our System	0.2737

Table 4: Performance against baseline system

These three systems are the same in that the final sentence translation results are generated by the combination of local phrase translation. Thus, they are capable of local reordering but not global reordering. The phrase reordering in Pharaoh depends only on distance distortion information which does not contain any linguistic knowledge. The experimental result shows that the performance of both Xiong's system and our system is better than that of Pharaoh. It proves that linguistic knowledge can help the global reordering probability estimation. Additionally, our system is superior to Xiong's system in which only use phrase boundary words to guide global reordering. It indicates that syntactic knowledge is more powerful to guide global reordering than boundary words. On the other hand, it proves the importance of syntactic knowledge constraints in avoiding the arbitrary phrase reordering.

7.2 Syntactic Error Analysis

Rule (3)~(9) in Section 3 not only play the role to compensate for syntactic errors, but also take the advantage of the capability of capturing local phrase reordering. However, the non-monotone combination for non-peer phrases is really harmful to system performance. To prove these ideas, we conducted experiments with different constrains.

Constraints	Bleu Score
All rules in Table 1 used	0.2737
Allowing the non-monotone combination of non-peer phrases	0.2647
Rule (3)~(9) are prohibited	0.2591

Table 5: About non-peer phrase combination

From the experimental results shown in Table 5, just as claimed in other previous work, the combination between non-linguistic phrases is useful and cannot be abandoned. On the other hand, if we relax the constraint of non-peer phrase combination (that is, allowing non-monotone combination for on-peer phrases), some more serious errors in non-syntactic knowledge is introduced, thereby degrading performance from 0.2737 to 0.2647.

7.3 Effect of Virtual Peer Phrases

As discussed in Section 4, for n-ary nodes (n>2) in the original syntax trees, the relationship among n-ary sub-trees is always not clearly captured. To give them the chance of free reordering, we add the virtual peer nodes to make sure that the combination of a set of peer phrases can still be a peer phrase. An experiment was done to compare with the case where the virtual peer nodes were not added to n-ary syntax trees. The Bleu score dropped to 26.20 from 27.37, which shows the virtual nodes have great effect on system performance.

7.4 Effect of Mixed Syntax Trees

In this section, we conducted three experiments to investigate the effect of constituency parse tree and dependency parse tree. Over the same platform, we tried to use only one of them to complete the translation task. The experimental results are shown in Table 6.

Surprisingly, there is no significant difference in performance. The reason may be that both parsers produce approximately equivalent parse results. However, the combination of syntax trees outperforms merely only one syntax tree. This suggests that the N-best syntax parse trees may enhance the quality of reordering model.

Situation	**Bleu Score**
Dependency parser only	0.2667
Stanford parser only	0.2670
Mixed parsing trees	0.2737

Table 6: Different parsing tree

8 Conclusion and Future Work

In this paper, syntactic knowledge is introduced to capture global reordering of SMT system. This method can not only inherit the advantage of local reordering ability of standard phrase-based SMT system, but also capture the global reordering as the syntax-based SMT system. The experimental results showed the effectiveness of our method.

In the future work, we plan to improve the reordering model by introducing N-best syntax trees and exploiting richer syntactic knowledge.

References

David Chiang. 2005. *A hierarchical phrase-based model for statistical machine translation*. In *Proceedings of ACL 2005*.

Franz Josef Och. 2003a. *Statistical Machine Translation: From Single-Word Models to Alignment Templates Thesis*.

Franz Josef Och. 2003b. *Minmum Error Rate Training in Statistical Machine Translation*. In *Proceedings for ACL 2003*.

Franz Josef Och, Hermann Ney. 2003. *A Systematic Comparison of Various Statistical Alignment Models*. Computational Linguistics, 29:19-51.

Franz Josef Och and Hermann Ney. 2004. *The alignment template approach to statistical machine translation*. Computational Linguistics, 30:417−449.

Dan Klein and Christopher D. Manning. 2003. *Accurate Unlexicalized Parsing*. In *Proceedings of ACL 2003*.

Philipp Koehn, Franz Joseph Och, and Daniel Marcu. 2003. *Statistical Phrase-Based Translation.* In *Proceedings of HLT/NAACL 2003*.

Philipp Koehn. 2004. *Pharaoh: a Beam Search Decoder for Phrased-Based Statistical Machine Translation Models.* In *Proceedings of AMTA 2004*.

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

Yang Liu, Qun Liu, Shouxun Lin. 2006. *Tree-to-String Alignment Template for Statistical Machine Translation*. In *Proceedings of COLING-ACL 2006*.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. *Dependency treelet translation: Syntactically informed phrasal SMT*. In *Proceedings of ACL 2005*.

Andreas Stolcke. 2002. *SRILM-An Extensible Language Modeling Toolkit*. In *Proceedings of ICSLP 2002*.

Christoph Tillmann. 2004. *A block orientation model for statistical machine translation*. In *Proceedings of HLT-NAACL 2004*.

Dekai Wu. 1996. *A Polynomial-Time Algorithm for Statistical Machine Translation*. In *Proceedings of ACL 1996*.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. *Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation*. In *Proceedings of COLING-ACL 2006*.

Kenji Yamada and Kevin Knight. 2001. *A syntax based statistical translation model*. In *Proceedings of ACL 2001*.

Le Zhang. 2004. Maximum Entropy Modeling Toolkit for Python and C++. Available at http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html.

R. Zens, H. Ney, T. Watanabe, and E. Sumita. 2004. *Reordering Constraints for Phrase-Based Statistical Machine Translation.* In *Proceedings of CoLing 2004*.

Ming Zhou. 2000. *A block-based robust dependency parser for unrestricted Chinese text*. The second Chinese Language Processing Workshop attached to ACL2000.