

# Dependency-based Pre-ordering for Chinese-English Machine Translation

Jingsheng Cai<sup>†\*</sup> Masao Utiyama<sup>‡</sup> Eiichiro Sumita<sup>‡</sup> Yujie Zhang<sup>†</sup>

<sup>†</sup>School of Computer and Information Technology, Beijing Jiaotong University

<sup>‡</sup>National Institute of Information and Communications Technology

joycetsai99@gmail.com

{mutiyama, eiichiro.sumita}@nict.go.jp

yjzhang@bjtu.edu.cn

## Abstract

In statistical machine translation (SMT), syntax-based pre-ordering of the source language is an effective method for dealing with language pairs where there are great differences in their respective word orders. This paper introduces a novel pre-ordering approach based on dependency parsing for Chinese-English SMT. We present a set of dependency-based pre-ordering rules which improved the BLEU score by 1.61 on the NIST 2006 evaluation data. We also investigate the accuracy of the rule set by conducting human evaluations.

## 1 Introduction

SMT systems have difficulties translating between distant language pairs such as Chinese and English. The reason for this is that there are great differences in their word orders. Reordering therefore becomes a key issue in SMT systems between distant language pairs.

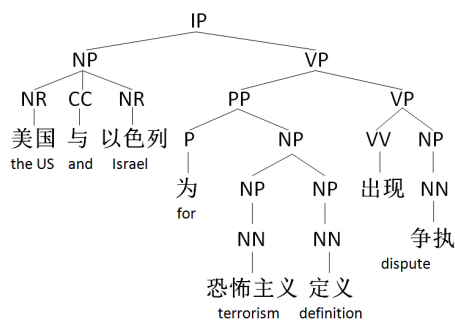
Previous work has shown that the approaches tackling the problem by introducing a pre-ordering procedure into phrase-based SMT (PBSMT) were effective. These pre-ordering approaches first parse the source language sentences to create parse trees. Then, syntactic reordering rules are applied to these parse trees with the goal of reordering the source language sentences into the word order of the target language. Syntax-based pre-ordering by employing constituent parsing have demonstrated effectiveness in many language pairs, such as English-French (Xia and McCord, 2004), German-English (Collins et al., 2005), Chinese-English (Wang et al., 2007; Zhang et al., 2008), and English-Japanese (Lee et al., 2010).

As a kind of constituent structure, HPSG (Pollard and Sag, 1994) parsing-based pre-ordering showed improvements in SVO-SOV translations, such as English-Japanese (Isozaki et al., 2010; Wu et al., 2011) and Chinese-Japanese (Han et al., 2012). Since dependency parsing is more concise than constituent parsing in describing sentences, some research has used dependency parsing in pre-ordering approaches for language pairs such as Arabic-English (Habash, 2007), and English-SOV languages (Xu et al., 2009; Katz-Brown et al., 2011). The pre-ordering rules can be made manually (Collins et al., 2005; Wang et al., 2007; Han et al., 2012) or extracted automatically from a parallel corpus (Xia and McCord, 2004; Habash, 2007; Zhang et al., 2007; Wu et al., 2011).

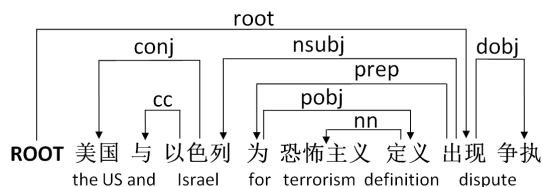
The purpose of this paper is to introduce a novel dependency-based pre-ordering approach through creating a pre-ordering rule set and applying it to the Chinese-English PBSMT system. Experiment results showed that our pre-ordering rule set improved the BLEU score on the NIST 2006 evaluation data by 1.61. Moreover, this rule set substantially decreased the total times of rule application about 60%, compared with a constituent-based approach (Wang et al., 2007). We also conducted human evaluations in order to assess its accuracy. To our knowledge, our manually created pre-ordering rule set is the first Chinese-English dependency-based pre-ordering rule set.

The most similar work to this paper is that of Wang et al. (2007). They created a set of pre-ordering rules for constituent parsers for Chinese-English PBSMT. In contrast, we propose a set of pre-ordering rules for dependency parsers. We argue that even though the rules by Wang et al. (2007) exist, it is almost impossible to automatically convert their rules into rules that are applicable to dependency parsers. In fact, we abandoned our initial attempts to automatically convert their rules into rules for dependency parsers, and

\* This work was done when the first author was on an internship in NICT.



(a) A constituent parse tree



(b) Stanford typed dependency parse tree

Figure 1: A constituent parse tree and its corresponding Stanford typed dependency parse tree for the same Chinese sentence.

spent more than two months discovering the rules introduced in this paper. By applying our rules and Wang et al.’s rules, one can use both dependency and constituency parsers for pre-ordering in Chinese-English PBSMT.

This is especially important on the point of the system combination of PBSMT systems, because the diversity of outputs from machine translation systems is important for system combination (Cer et al., 2013). By using both our rules and Wang et al.’s rules, one can obtain diverse machine translation results because the pre-ordering results of these two rule sets are generally different.

Another similar work is that of (Xu et al., 2009). They created a pre-ordering rule set for dependency parsers from English to several SOV languages. In contrast, our rule set is for Chinese-English PBSMT. That is, the direction of translation is opposite. Because there are a lot of language specific decisions that reflect specific aspects of the source language and the language pair combination, our rule set provides a valuable resource for pre-ordering in Chinese-English PBSMT.

## 2 Dependency-based Pre-ordering Rule Set

Figure 1 shows a constituent parse tree and its Stanford typed dependency parse tree for the same

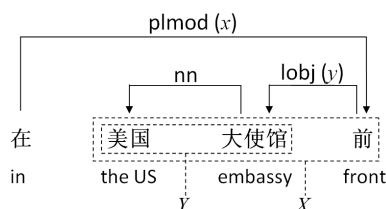


Figure 2: An example of a preposition phrase with a plmod structure. The phrase translates into “in front of the US embassy”.

Chinese sentence. As shown in the figure, the number of nodes in the dependency parse tree (i.e. 9) is much fewer than that in its corresponding constituent parse tree (i.e. 17). Because dependency parse trees are generally more concise than the constituent ones, they can conduct long-distance reorderings in a finer way. Thus, we attempted to conduct pre-ordering based on dependency parsing. There are two widely-used dependency systems – Stanford typed dependencies and CoNLL typed dependencies. For Chinese, there are 45 types of grammatical relations for Stanford typed dependencies (Chang et al., 2009) and 25 for CoNLL typed dependencies. As we thought that Stanford typed dependencies could describe language phenomena more meticulously owing to more types of grammatical relations, we preferred to use it for searching candidate pre-ordering rules.

We designed two types of formats in our dependency-based pre-ordering rules. They are:

Type-1:  $x : y$

Type-2:  $x - y$

Here, both  $x$  and  $y$  are *dependency relations* (e.g., plmod or lobj in Figure 2). We define the *dependency structure* of a dependency relation as the structure containing the dependent word (e.g., the word directly indicated by plmod, or “前” in Figure 2) and the whole subtree under the dependency relation (all of the words that directly or indirectly depend on the dependent word, or the words under “前” in Figure 2). Further, we define  $X$  and  $Y$  as the corresponding dependency structures of the dependency relations  $x$  and  $y$ , respectively. We define  $X \setminus Y$  as structure  $X$  except  $Y$ . For example, in Figure 2, let  $x$  and  $y$  denote plmod and lobj dependency relations, then  $X$  represents “前” and all words under “前”,  $Y$  represents “大使馆” and all words under “大使馆”, and  $X \setminus Y$  represents

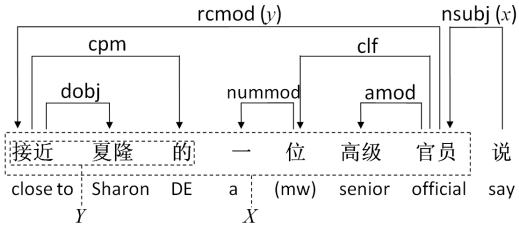


Figure 3: An example of rcmod structure within an nsubj structure. The phrase translates into “a senior official close to Sharon said”.

“前”. For Type-1,  $Y$  is a sub-structure of  $X$ . The rule repositions  $X \setminus Y$  to the position before  $Y$ . For Type-2,  $X$  and  $Y$  are ordered sibling structures under a same parent node. The rule repositions  $X$  to the position after  $Y$ .

We obtained rules as the following steps:

- 1 Search the Chinese dependency parse trees in the corpus and rank all of the structures matching the two types of rules respectively according to their frequencies. Note that while calculating the frequencies of Type-1 structures, we dismissed the structures in which  $X$  occurred before  $Y$  originally.
- 2 Filtration. 1) Filter out the structures which occurred less than 5,000 times. 2) Filter out the structures from which it was almost impossible to derive candidate pre-ordering rules because  $x$  or  $y$  was an “irrespective” dependency relation, for example, root, conj, cc and so on.
- 3 Investigate the remaining structures. For each kind of structure, we selected some of the sample dependency parse trees that contained it, tried to restructure the parse trees according to the matched rule and judged the re-ordered Chinese phrases. If the reordering produced a Chinese phrase that had a closer word order to that of the English one, this structure would be a candidate pre-ordering rule.
- 4 Conduct primary experiments which used the same training set and development set as the experiments described in Section 3. In the primary experiments, we tested the effectiveness of the candidate rules and filtered the ones that did not work based on the BLEU scores on the development set.

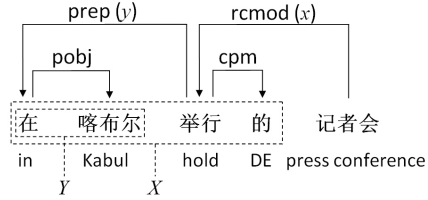


Figure 4: An example of rcmod structure with a preposition modifier. The phrase translates into “a press conference held in Kabul”.

As a result, we obtained eight pre-ordering rules in total, which can be divided into three dependency relation categories. They are: plmod (localizer modifier), rcmod (relative clause modifier) and prep (preposition modifier). Each of these categories are discussed in detail below.

**plmod** Figure 2 shows an example of a prepositional phrase with a plmod structure, which translates literally into “in the US embassy front”. In Chinese, the dependent word of a plmod relation (e.g., “前” in Figure 2) occurs in the last position of the prepositional phrase. However, in English, this kind of word (e.g., “front” in the caption of Figure 2) always occur directly after prepositions, which is to say, in the second position in a prepositional phrase. Therefore, we applied a rule **plmod : lobj** (localizer object) to reposition the dependent word of the plmod relation (e.g., “前” in Figure 2) to the position before the lobj structure (e.g., “美国大使馆” in Figure 2). In this case, it also comes directly after the preposition. Similarly, we created a rule **plmod : lcomp** (clausal complement of a localizer).

**rcmod** Figure 3 shows an example of an rcmod structure under an nsubj (nominal subject) structure. Here “mw” means “measure word”. As shown in the figure, relative clause modifiers in Chinese (e.g., “接近夏隆的” in Figure 3) occurs before the noun being modified, which is in contrast to English (e.g., “close to Sharon” in the caption of Figure 3), where they come after. Thus, we introduced a series of rules **NOUN : rcmod** to restructure rcmod structures so that the noun is moved to the head. In this example, with the application of an **nsubj : rcmod** rule, the phrase can be translated into “a senior official close to Sharon say”, which has a word order very close to English. Since a noun can be nsubj, dobj (direct object), pobj (prepositional object) and lobj in Stanford typed dependencies, we created four

Type	System	Parser	BLEU	Counts	#Sent.
-	No pre-ordering	-	29.96	-	-
Constituent	WR07	Berkeley	31.45	2,561,937	852,052
Dependency	OUR DEP 1	Berkeley Const.	31.54	978,013	556,752
	OUR DEP 2	Mate	31.57	947,441	547,084

Table 1: The comparison of four systems, including the performance (BLEU) on the test set, the total count of each rule set and the number of sentences they were applied to on the training set.

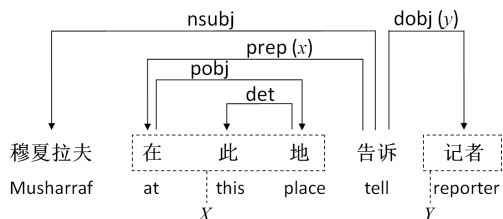


Figure 5: An example of verb phrase with a preposition modifier. The phrase translates into “Musharraf told reporters here”.

rules from the NOUN pattern. Note that for some preposition modifiers, we needed a rule **rcmod : prep** to conduct the same work. For instance, the Chinese phrase in Figure 4 can be translated into “hold in Kabul press conference” with the application of this rule.

**prep** Within verb phrases, the positions of prep structures are quite different between Chinese and English. Figure 5 shows an example of a verb phrase with a preposition modifier (prep), which literally translates into “Musharraf at this place tell reporter”. Recognizing that prep structures occur before the verb in Chinese (e.g., “在此地” in Figure 5) but after the verb in English (usually in the last position of a verb phrase, e.g., “here” in the caption of Figure 5), we applied a rule **prep - dobj** to reposition prep structures after their sibling dobj structures.

In summary, the dependency-based pre-ordering rule set has eight rules: **plmod : lobj**, **plmod : lcomp**, **nsbj : rcm**, **dobj : rcm**, **pobj : rcm**, **lobj : rcm**, **rcmod : prep**, and **prep - dobj**.

### 3 Experiments

We used the MOSES PBSMT system (Koehn et al., 2007) in our experiments. The training data, which included those data used in Wang et al. (2007), contained 1 million pairs of sentences extracted from the Linguistic Data Consortium’s par-

allel news corpora. Our development set was the official NIST MT evaluation data from 2002 to 2005, consisting of 4476 Chinese-English sentences pairs. Our test set was the NIST 2006 MT evaluation data, consisting of 1664 sentence pairs. We employed the Stanford Segmenter<sup>1</sup> to segment all of the data sets. For evaluation, we used BLEU scores (Papineni et al., 2002).

We implemented the constituent-based pre-ordering rule set in Wang et al. (2007) for comparison, which is called WR07 below. The Berkeley Parser (Petrov et al., 2006) was employed for parsing the Chinese sentences. For training the Berkeley Parser, we used Chinese Treebank (CTB) 7.0.

We conducted our dependency-based pre-ordering experiments on the Berkeley Parser and the Mate Parser (Bohnet, 2010), which were shown to be the two best parsers for Stanford typed dependencies (Che et al., 2012). First, we converted the constituent parse trees in the results of the Berkeley Parser into dependency parse trees by employing a tool in the Stanford Parser (Klein and Manning, 2003). For the Mate Parser, POS tagged inputs are required both in training and in inference. Thus, we then extracted the POS information from the results of the Berkeley Parser and used these as the pre-specified POS tags for the Mate Parser. Finally, we applied our dependency-based pre-ordering rule set to the dependency parse trees created from the converted Berkeley Parser and the Mate Parser, respectively.

Table 1 presents a comparison of the system without pre-ordering, the constituent system using WR07 and two dependency systems employing the converted Berkeley Parser and the Mate Parser, respectively. It shows the BLEU scores on the test set and the statistics of pre-ordering on the training set, which includes the total count of each rule set and the number of sentences they were applied to. Both of our dependency systems outper-

<sup>1</sup><http://nlp.stanford.edu/software/segmenter.shtml>

Category	Count	Correct	Incorrect	Accuracy
plmod	42	26	16	61.9%
rcmod	89	49	40	55.1%
prep	54	36	18	66.7%
All	185	111	74	60.0%

Table 2: Accuracy of the dependency-based pre-ordering rules on a set of 200 sentences randomly selected from the development set.

formed WR07 slightly but were not significant at  $p = 0.05$ . However, both of them substantially decreased the total times about 60% (or 1,600,000) for pre-ordering rule applications on the training set, compared with WR07. In our opinion, the reason for the great decrease was that the dependency parse trees were more concise than the constituent parse trees in describing sentences and they could also describe the reordering at the sentence level in a finer way. In contrast, the constituent parse trees were more redundant and they needed more nodes to conduct long-distance reordering. In this case, the affect of the performance of the constituent parsers on pre-ordering is larger than that of the dependency ones so that the constituent parsers are likely to bring about more incorrect pre-orderings.

Similar to Wang et al. (2007), we carried out human evaluations to assess the accuracy of our dependency-based pre-ordering rules by employing the system “OUR DEP 2” in Table 1. The evaluation set contained 200 sentences randomly selected from the development set. Among them, 107 sentences contained at least one rule and the rules were applied 185 times totally. Since the accuracy check for dependency parse trees took great deal of time, we did not try to select error free (100% accurately parsed) sentences. A bilingual speaker of Chinese and English looked at an original Chinese phrase and the pre-ordered one with their corresponding English phrase and judged whether the pre-ordering obtained a Chinese phrase that had a closer word order to the English one. Table 2 shows the accuracies of three categories of our dependency-based pre-ordering rules. The overall accuracy of this rule set is 60.0%, which is almost at the same level as the WR07 rule set (62.1%), according to the similar evaluation (200 sentences and one annotator) conducted in Wang et al. (2007). Notice that some of the incorrect pre-orderings may be caused by erroneous parsing as also suggested by Wang et al. (2007). Through human evaluations, we found

that 19 out of the total 74 incorrect pre-orderings resulted from errors in parsing. Among them, 13 incorrect pre-orderings applied the rules of the rcmod category. The analysis suggests that we need to introduce constraints on the rule application of this category in the future.

#### 4 Conclusion

In this paper, we introduced a novel pre-ordering approach based on dependency parsing for a Chinese-English PBSMT system. The results showed that our approach achieved a BLEU score gain of 1.61. Moreover, our dependency-based pre-ordering rule set substantially decreased the time for applying pre-ordering rules about 60% compared with WR07, on the training set of 1M sentences pairs. The overall accuracy of our rule set is 60.0%, which is almost at the same level as the WR07 rule set. These results indicated that dependency parsing is more effective for conducting pre-ordering for Chinese-English PBSMT. Although our work focused on Chinese, the ideas can also be applied to other languages.

In the future, we attempt to create more efficient pre-ordering rules by exploiting the rich information in dependency structures.

#### Acknowledgments

We thank the anonymous reviewers for their valuable comments and suggestions. This work is supported in part by the International Science & Technology Cooperation Program of China (Grant No. 2014DFA11350) and Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China.

#### References

Bernd Bohnet. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceed-*

- ings of the 23rd International Conference on Computational Linguistics (COLING 2010).
- Daniel Cer, Christopher D. Manning, and Dan Jurafsky. 2013. Positive Diversity Tuning for Machine Translation System Combination. In *Proceedings of the Eighth Workshop on Statistical Machine Translation (WMT 2013)*.
- Pi-Chuan Chang, Huihsin Tseng, Dan Jurafsky, and Christopher D. Manning. 2009. Discriminative reordering with Chinese grammatical relations features. In *Proceedings of the HLT-NAACL Workshop on Syntax and Structure in Statistical Translation*, pages 51-59.
- Wanxiang Che, Valentin Spitzkovsky, and Ting Liu. 2012. A comparison of Chinese parsers for Stanford dependencies. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 11-16.
- Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 531-540.
- Dan Han, Katsuhito Sudoh, Xianchao Wu, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2012. Head Finalization reordering for Chinese-to-Japanese machine translation. In *Proceedings of SSST-6, Sixth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 57-66.
- Nizar Habash. 2007. Syntactic preprocessing for statistical machine translation. In *Proceedings of the 11th Machine Translation Summit (MT-Summit)*.
- Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, and Kevin Duh. 2010. Head Finalization: A simple reordering rule for SOV languages. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 250-257.
- Jason Katz-Brown, Slav Petrov, Ryan McDonald, Franz J. Och, David Talbot, Hiroshi Ichikawa, Masakazu Seno, and Hideto Kazawa. 2011. Training a parser for machine translation reordering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 183-192.
- Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 423-430.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177-180.
- Young-Suk Lee, Bing Zhao, and Xiaoqian Luo. 2010. Constituent reordering and syntax models for English-to-Japanese statistical machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 626-634.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311-318.
- Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 433-440.
- Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press.
- Chao Wang, Michael Collins, and Philipp Koehn. 2007. Chinese syntactic reordering for statistical machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 737-745.
- Xianchao Wu, Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2011. Extracting preordering rules from predicate-argument structures. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 29-37.
- Fei Xia and Michael McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of Coling 2004*, pages 508-514.
- Peng Xu, Jaeho Kang, Michael Ringgaard, and Franz J. Och. 2009. Using a dependency parser to improve SMT for subject-object-verb languages. In *Proceedings of HLT-NAACL*, pages 245-253.
- Jiajun Zhang, Chengqing Zong, and Shoushan Li. 2008. Sentence type based reordering model for statistical machine translation. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 1089-1096.
- Yuqi Zhang, Richard Zens, and Hermann Ney. 2011. Chunk-level reordering of source language sentences with automatically learned rules for statistical machine translation. In *HLT-NAACL Workshop on Syntax and Structure in Statistical Translation*, pages 1-8.