is "Left" or "Right", the parser then use MaxEnt with the same features to tag the label of relation. This strategy can tag the label according to the current states of the focused word pair. We divide the training instances according to the CPOSTAG of the focused word *n*, so that a classifier is constructed for each of distinct POS-tag of the word *n*.

2.3 Preprocessing

2.3.1 Preceding work

In our preceding work (Cheng, 2005a), we discussed three problems of our basic methods (adopt Nivre's algorithm with SVMs) and proposed three preprocessing methods to resolve these problems. The methods include: (1) using global features and a two-steps process to resolve the ambiguity between the parsing operations "Shift" and "Reduce". (2) using a root node finder and dividing the sentence at the root node to make use of the top-down information. (3) extracting the prepositional phrase (PP) to resolve the problem of identifying the boundary of PP.

We incorporated Nivre's method with these preprocessing methods for Chinese dependency analysis with Penn Chinese Treebank and Sinica Treebank (Chen et al., 2003). This was effective because of the properties of Chinese: First, there is no multi-root in Chinese Treebank. Second, the boundary of prepositional phrases is ambiguous. We found that these methods do not always improve the accuracy of all the languages in the shared task.

We have tried the method (1) in some languages to see if there is any improvement in the parser. We attempted to use global features and two-step analysis to resolve the ambiguity of the operations. In Chinese (Chen et al., 2003) and Danish (Kromann, 2003), this method can improve the parser performance. However, in other languages, such as Arabic (Hajič et al., 2004), this method decreased the performance. The reason is that the sentence in some languages is too long to use global features. In our preceding work, the global features include the information of all the un-analyzed words. However, for analyzing long sentences, the global features usually include some useless information and will confuse the two-step process. Therefore, we do not use this method in this shared task.

In the method (2), we construct an SVM-based root node finder to identify the root node and divided the sentence at the root node in the Chinese Treebank. This method is based on the properties of dependency structures "One and only one element is independent" and "An element cannot have modifiers lying on the other side of its own head". However, there are some languages that include multi-root sentences, such as Arabic, Czech, and Spanish (Civit and Martí, 2002), and it is difficult to divide the sentence at the roots. In multi-root sentences, deciding the head of the words between roots is difficult. Therefore, we do not use the method (2) in the share task.

The method (3) –namely PP chunker– can identify the boundary of PP in Chinese and resolve the ambiguity of PP boundary, but we cannot guarantee that to identify the boundary of PP can improve the parser in other languages. Even we do not understand construction of PP in all languages. Therefore, for the robustness in analyzing different languages, we do not use this method.

2.3.2 Neighboring dependency attachment tagger

In the bottom-up dependency parsing approach, the features and the strategies for parsing in early stage (the dependency between adjacent[2] words) is different from parsing in upper stage (the dependency between phrases). Parsing in upper stage needs the information at the phrases not at the words alone. The features and the strategies for parsing in early and upper stages should be separated into distinct. Therefore, we divide the neighboring dependency attachment (for early stage) and normal dependency attachment (for upper stage), and set the neighboring dependency attachment tagger as a preprocessor.

When the parser analyzes an input sentence, it extracts the neighboring dependency attachments first, then analyzes the sentence as described before. The results show that tagging the neighboring dependency word-pairs can improve 9 languages out of 12 scoring languages, although in some languages it degrades the performance a little. Potentially, there may be a number of ways for decomposing the parsing process, and the current method is just the simplest decomposition of the process. The best method of decomposition or dynamic changing of parsing models should be investigated as the future research.

[2] We extract all words that depend on the adjacent word (right or left).