SMT Helps Bitext Dependency Parsing

Wenliang Chen[†‡], **Jun'ichi Kazama**[‡], **Min Zhang**[†], **Yoshimasa Tsuruoka**[*‡],
Yujie Zhang[⋆‡], **Yiou Wang**[‡], **Kentaro Torisawa**[‡] **and Haizhou Li**[†]
[†]Human Language Technology, Institute for Infocomm Research, Singapore
[‡]National Institute of Information and Communications Technology (NICT), Japan
[*]School of Information Science, JAIST, Japan
[⋆]Beijing Jiaotong University, China
{wechen, mzhang, hli}@i2r.a-star.edu.sg
{kazama, torisawa, yujie, wangyiou}@nict.go.jp
tsuruoka@jaist.ac.jp

Abstract

We propose a method to improve the accuracy of parsing bilingual texts (bitexts) with the help of statistical machine translation (SMT) systems. Previous bitext parsing methods use human-annotated bilingual treebanks that are hard to obtain. Instead, our approach uses an auto-generated bilingual treebank to produce bilingual constraints. However, because the auto-generated bilingual treebank contains errors, the bilingual constraints are noisy. To overcome this problem, we use large-scale unannotated data to verify the constraints and design a set of effective bilingual features for parsing models based on the verified results. The experimental results show that our new parsers significantly outperform state-of-the-art baselines. Moreover, our approach is still able to provide improvement when we use a larger monolingual treebank that results in a much stronger baseline. Especially notable is that our approach can be used in a purely monolingual setting with the help of SMT.

1 Introduction

Recently there have been several studies aiming to improve the performance of parsing bilingual texts (bitexts) (Smith and Smith, 2004; Burkett and Klein, 2008; Huang et al., 2009; Zhao et al., 2009; Chen et al., 2010). In bitext parsing, we can use the information based on "bilingual constraints" (Burkett and Klein, 2008), which do not exist in monolingual sentences. More accurate bitext parsing results can be effectively used in the training of syntax-based machine translation systems (Liu and Huang, 2010).

Most previous studies rely on bilingual treebanks to provide bilingual constraints for bitext parsing. Burkett and Klein (2008) proposed joint models on bitexts to improve the performance on either or both sides. Their method uses bilingual treebanks that have human-annotated tree structures on both sides. Huang et al. (2009) presented a method to train a source-language parser by using the reordering information on words between the sentences on two sides. It uses another type of bilingual treebanks that have tree structures on the source sentences and their human-translated sentences. Chen et al. (2010) also used bilingual treebanks and made use of tree structures on the target side. However, the bilingual treebanks are hard to obtain, partly because of the high cost of human translation. Thus, in their experiments, they applied their methods to a small data set, the manually translated portion of the Chinese Treebank (CTB) which contains only about 3,000 sentences. On the other hand, many large-scale monolingual treebanks exist, such as the Penn English Treebank (PTB) (Marcus et al., 1993) (about 40,000 sentences in Version 3) and the latest version of CTB (over 50,000 sentences in Version 7).

In this paper, we propose a bitext parsing approach in which we produce the bilingual constraints on existing monolingual treebanks with the help of SMT systems. In other words, we aim to improve source-language parsing with the help of automatic translations.

In our approach, we first use an SMT system to translate the sentences of a source monolingual treebank into the target language. Then, the target sentences are parsed by a parser trained on a target monolingual treebank. We then obtain a bilingual treebank that has human annotated trees on the source side and auto-generated trees on the target side. Although the sentences and parse trees on the

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 73–83,
Edinburgh, Scotland, UK, July 27–31, 2011. ©2011 Association for Computational Linguistics

target side are not perfect, we expect that we can improve bitext parsing performance by using this newly auto-generated bilingual treebank. We build word alignment links automatically using a word alignment tool. Then we can produce a set of bilingual constraints between the two sides.

Because the translation, parsing, and word alignment are done automatically, the constraints are not reliable. To overcome this problem, we verify the constraints by using large-scale unannotated monolingual sentences and bilingual sentence pairs. Finally, we design a set of bilingual features based on the verified results for parsing models.

Our approach uses existing resources including monolingual treebanks to train monolingual parsers on both sides, bilingual unannotated data to train SMT systems and to extract bilingual subtrees, and target monolingual unannotated data to extract monolingual subtrees. In summary, we make the following contributions:

- We propose an approach that uses an auto-generated bilingual treebank rather than human-annotated bilingual treebanks used in previous studies (Burkett and Klein, 2008; Huang et al., 2009; Chen et al., 2010). The auto-generated bilingual treebank is built with the help of SMT systems.

- We verify the unreliable constraints by using the existing large-scale unannotated data and design a set of effective bilingual features over the verified results. Compared to Chen et al. (2010) that also used tree structures on the target side, our approach defines the features on the auto-translated sentences and auto-parsed trees, while theirs generates the features by some rules on the human-translated sentences.

- Our parser significantly outperforms state-of-the-art baseline systems on the standard test data of CTB containing about 3,000 sentences. Moreover, our approach continues to achieve improvement when we build our system using the latest version of CTB (over 50,000 sentences) that results in a much stronger baseline.

- We show the possibility that we can improve the performance even if the test set has no human translation. This means that our proposed approach can be used in a purely monolingual setting with the help of SMT. To our knowledge, this paper is the first one that demonstrates this widened applicability, unlike the previous studies that assumed that the parser is applied only on the bitexts made by humans.

Throughout this paper, we use Chinese as the source language and English as the target language. The rest of this paper is organized as follows. Section 2 introduces the motivation of this work. Section 3 briefly introduces the parsing model used in the experiments. Section 4 describes a set of bilingual features based on the bilingual constraints and Section 5 describes how to use large-scale unannotated data to verify the bilingual constraints and define another set of bilingual features based on the verified results. Section 6 explains the experimental results. Finally, in Section 7 we draw conclusions.

2 Motivation

Here, bitext parsing is the task of parsing source sentences with the help of their corresponding translations. Figure 1-(a) shows an example of the input of bitext parsing, where ROOT is an artificial root token inserted at the beginning and does not depend on any other token in the sentence, the dashed undirected links are word alignment links, and the directed links between words indicate that they have a dependency relation. Given such inputs, we build dependency trees for the source sentences. Figure 1-(b) shows the output of bitext parsing for the example in 1-(a).

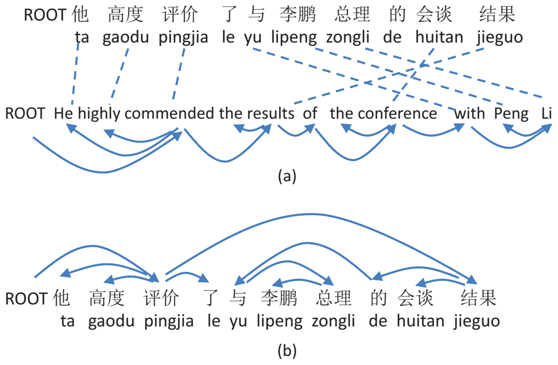
ROOT 他 高度 评价 了 与 李鹏 总理 的 会谈 结果
ta gaodu pingjia le yu lipeng zongli de huitan jieguo
ROOT He highly commended the results of the conference with Peng Li
(a)
ROOT 他 高度 评价 了 与 李鹏 总理 的 会谈 结果
ta gaodu pingjia le yu lipeng zongli de huitan jieguo
(b)

Figure 1: Input and output of our approach

In bitext parsing, some ambiguities exist on the source side, but they may be unambiguous on the target side. These differences are expected to help improve source-side parsing.

Suppose we have a Chinese sentence shown in Figure 2-(a). In this sentence, there is a nominalization case (Li and Thompson, 1997) in which the particle “的(de)/nominalizer” is placed after the verb compound “培育(peiyu)起来(qilai)/cultivate” to modify “技巧(jiqiao)/skill”. This nominalization is a relative clause, but does not have a clue about its boundary. That is, it is very hard to determine which word is the head of “技巧(jiqiao)/skill”. The head may be “发挥(fahui)/demonstrate” or “培育(peiyu)/cultivate”, as shown in Figure 2-(b) and -(c), where (b) is correct.

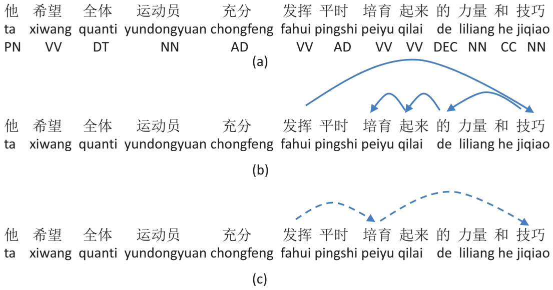
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
PN VV DT NN AD VV AD VV VV DEC NN CC NN
(a)
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
(b)
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
(c)

Figure 2: Example of an ambiguity on the Chinese side

In its English translation (Figure 3), word “that” is a clue indicating the relative clause which shows the relation between “skill” and “cultivate”, as shown in Figure 3. The figure shows that the translation can provide useful bilingual constraints. From the dependency tree on the target side, we find that the word “skill” corresponding to “技巧(jiqiao)/skill” depends on the word “demonstrate” corresponding to “发挥(fahui)/demonstrate”, while the word “cultivate” corresponding to “培育(peiyu)/cultivate” is a grandchild of “skill”. This is a positive evidence for supporting “发挥(fahui)/demonstrate” as being the head of “技巧(jiqiao)/skill”.

The above case uses the human translation on the target side. However, there are few human-annotated bilingual treebanks and the existing bilingual treebanks are usually small. In contrast, there are large-scale monolingual treebanks, e.g., the PTB and the latest version of CTB. So we want to use existing resources to generate a bilingual treebank with the help of SMT systems. We hope to improve source side parsing by using this newly built bilingual treebank.

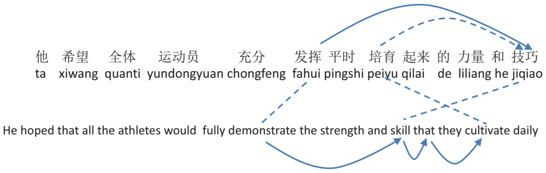
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
He hoped that all the athletes would fully demonstrate the strength and skill that they cultivate daily

Figure 3: Example of human translation

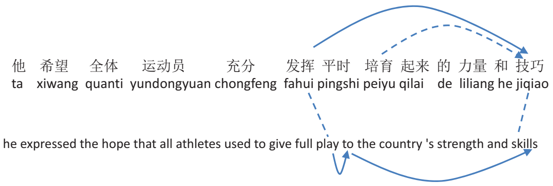
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
he expressed the hope that all athletes used to give full play to the country 's strength and skills

Figure 4: Example of Moses translation

Figure 4 shows an example of a translation using a Moses-based system, where the target sentence is parsed by a monolingual target parser. The translation contains some errors, but it does contain some correct parts that can be used for disambiguation. In the figure, the word “skills” corresponding to “技巧(jiqiao)/skill” is a grandchild of the word “play” corresponding to “发挥(fahui)/demonstrate”. This is a positive evidence for supporting “发挥(fahui)/demonstrate” as being the head of “技巧(jiqiao)/skill”.

From this example, although the sentences and parse trees on the target side are not perfect, we still can explore useful information to improve bitext parsing. In this paper, we focus on how to design a method to verify such unreliable bilingual constraints.

3 Parsing model

In this paper, we implement our approach based on graph-based parsing models (McDonald and Pereira, 2006; Carreras, 2007). Note that our approach can also be applied to transition-based parsing models (Nivre, 2003; Yamada and Matsumoto, 2003).

The graph-based parsing model is to search for the maximum spanning tree (MST) in a graph (McDonald and Pereira, 2006). The formulation defines the score of a dependency tree to be the sum of edge scores,

$$s(x, y) = \sum_{g \in y} score(w, x, g) = \sum_{g \in y} w \cdot f(x, g) \quad (1)$$

where x is an input sentence, y is a dependency tree for x, and g is a spanning subgraph of y. $f(x, g)$ can be based on arbitrary features of the subgraph and the input sequence x and the feature weight vector w are the parameters to be learned by using MIRA (Crammer and Singer, 2003) during training.

In our approach, we use two types of features for the parsing model. One is monolingual features based on the source sentences. The monolingual features include the first- and second- order features presented in McDonald and Pereira (2006) and the parent-child-grandchild features used in Carreras (2007). The other one is bilingual features (described in Sections 4 and 5) that consider the bilingual constraints.

We call the parser with the monolingual features on the source side Parser^s, and the parser with the monolingual features on the target side Parser^t.

4 Original bilingual features

In this paper, we generate two types of bilingual features, original and verified bilingual features. The original bilingual features (described in this section) are based on the bilingual constraints without being verified by large-scale unannotated data. And the verified bilingual features (described in Section 5) are based on the bilingual constraints verified by using large-scale unannotated data.

4.1 Auto-generated bilingual treebank

Assuming that we have monolingual treebanks on the source side, an SMT system that can translate the source sentences into the target language, and a Parser^t trained on the target monolingual treebank.

We first translate the sentences of the source monolingual treebank into the target language using the SMT system. Usually, SMT systems can output the word alignment links directly. If they can not, we perform word alignment using some publicly available tools, such as Giza++ (Och and Ney, 2003) or Berkeley Aligner (Liang et al., 2006; DeNero and Klein, 2007). The translated sentences are parsed by the Parser^t. Then, we have a newly auto-generated bilingual treebank.

4.2 Bilingual constraint functions

In this paper, we focus on the first- and second-order graph models (McDonald and Pereira, 2006; Carreras, 2007). Thus we produce the constraints for bigram (a single edge) and trigram (adjacent edges) dependencies in the graph model. For the trigram dependencies, we consider the parent-sibling and parent-child-grandchild structures described in McDonald and Pereira (2006) and Carreras (2007). We leave the third-order models (Koo and Collins, 2010) for a future study.

Suppose that we have a (candidate) dependency relation r_s that can be a bigram or trigram dependency. We examine whether the corresponding words of the source words of r_s have a dependency relation r_t in the target trees. We also consider the direction of the dependency relation. The corresponding word of the head should also be the head in r_t. We define a binary function for this bilingual constraint: $F_{bn}(r_{sn} : r_{tk})$, where n and k refers to the types of the dependencies (2 for bigram and 3 for trigram). For example, in $r_{s2} : r_{t3}$, r_{s2} is a bigram dependency on the source side and r_{t3} is a trigram dependency on the target side.

4.2.1 Bigram constraint function: F_{b2}

For r_{s2}, we consider two types of bilingual constraints. The first constraint function, denoted as $F_{b2}(r_{s2} : r_{t2})$, checks if the corresponding words also have a direct dependency relation r_{t2}. Figure 5 shows an example, where the source word “全体(quanti)” depends on “运动员(yundongyuan)” and word “all” corresponding to “全体(quanti)” depends on word “athletes” corresponding to “运动员(yundongyuan)”. In this case, $F_{b2}(r_{s2} : r_{t2}) = +$. However, when the source words are “他(ta)” and “希望(xiwang)”, this time their corresponding words “He” and “hope” do not have a direct dependency relation. In this case, $F_{b2}(r_{s2} : r_{t2}) = -$.

The second constraint function, denoted as $F_{b2}(r_{s2} : r_{t3})$, checks if the corresponding words form a parent-child-grandchild relation that often occurs in translation (Koehn et al., 2003). Figure 6 shows an example. The source word “技巧(jiqiao)” depends on “发挥(fahui)” while its corresponding word “skills” indirectly depends on “play” which corresponds to “发挥(fahui)” via “to”. In this case, $F_{b2}(r_{s2} : r_{t3}) = +$.

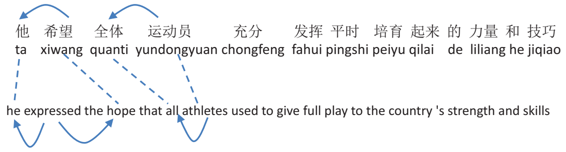
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
he expressed the hope that all athletes used to give full play to the country 's strength and skills

Figure 5: Example of bilingual constraints (2to2)

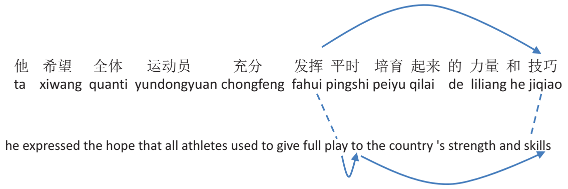
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
he expressed the hope that all athletes used to give full play to the country 's strength and skills

Figure 6: Example of bilingual constraints (2to3)

4.2.2 Trigram constraint function: F_{b3}

For a second-order relation on the source side, we consider one type of constraint. We have three source words that form a second-order relation and all of them have the corresponding words. We define function $F_{b3}(r_{s3} : r_{t3})$ for this constraint. The function checks if the corresponding words form a trigram dependencies structure. An example is shown in Figure 7. The source words "力量(liliang)", "和(he)", and "技巧(jiqiao)" form a parent-sibling structure, while their corresponding words "strength", "and", and "skills" also form a parent-sibling structure on the target side. In this case, function $F_{b3}(r_{s3} : r_{t3}) = +$.

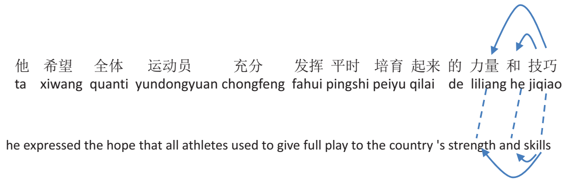
他 希望 全体 运动员 充分 发挥 平时 培育 起来 的 力量 和 技巧
ta xiwang quanti yundongyuan chongfeng fahui pingshi peiyu qilai de liliang he jiqiao
he expressed the hope that all athletes used to give full play to the country 's strength and skills

Figure 7: Example of bilingual constraints (3to3)

4.3 Bilingual reordering function: F_{ro}

Huang et al. (2009) proposed features based on reordering between languages for a shift-reduce parser. They define the features based on word-alignment information to verify whether the corresponding words form a contiguous span to resolve shift-reduce conflicts. We also implement similar features in our system. For example, in Figure 1-(a) the source span is [会谈(huitan), 结果(jieguo)], which maps onto [results, conference]. Because no word within this target span is aligned to a source word outside of the source span, this span is a contiguous span. In this case, function $F_{ro} = +$, otherwise $F_{ro} = -$.

4.4 Original bilingual features

We define original bilingual features based on the bilingual constraint functions and the bilingual reordering function.

Table 1 lists the original features, where Dir refers to the directions[1] of the source-side dependencies, F_{b2} can be $F_{b2}(r_{s2} : r_{t2})$ and $F_{b2}(r_{s2} : r_{t3})$, and F_{b3} is $F_{b3}(r_{s3} : r_{t3})$. Each line of the table defines a feature template that is a combination of functions.

First-order features	Second-order features
$\langle F_{ro} \rangle$	
$\langle F_{b2}, Dir \rangle$	$\langle F_{b3}, Dir \rangle$
$\langle F_{b2}, Dir, F_{ro} \rangle$	$\langle F_{b3}, Dir, F_{ro} \rangle$

Table 1: Original bilingual features

We use an example to show how to generate the original bilingual features in practice. In Figure 4, we want to define the bilingual features for the bigram dependency (r_{s2}) between "发挥(fahui)" and "技巧(jiqiao)". The corresponding words form a trigram relation r_{t3} in the target dependency tree. The direction of the bigram dependency is right. Then we have feature "$\langle F_{b2}(r_{s2} : r_{t3}) = +, RIGHT \rangle$" for the second first-order feature template in Table 1.

5 Verified bilingual features

However, because the bilingual treebank is generated automatically, using the bilingual constraints alone is not reliable. Therefore, in this section we verify the constraints by using large-scale unannotated data to overcome this problem. More specifically, r_{tk} of the constraint is verified by checking a list of target monolingual subtrees and $r_{sn} : r_{tk}$ is verified by checking a list of bilingual subtrees. The subtrees are extracted from the large-scale unannotated data. The basic idea is as follows: if the dependency structures of a bilingual constraint can be found in the list of the target monolingual subtrees

[1] For the second order features, Dir is the combination of the directions of two dependencies.

or bilingual subtrees, this constraint will probably be reliable.

We first parse the large-scale unannotated monolingual and bilingual data. Subsequently, we extract the monolingual and bilingual subtrees from the parsed data. We then verify the bilingual constraints using the extracted subtrees. Finally, we generate the bilingual features based on the verified results for the parsing models.

5.1 Verified constraint functions

5.1.1 Monolingual target subtrees

Chen et al. (2009) proposed a simple method to extract subtrees from large-scale monolingual data and used them as features to improve monolingual parsing. Following their method, we parse large unannotated data with the Parser^t and obtain the subtree list (ST_t) on the target side. We extract two types of subtrees: bigram (two words) subtree and trigram (three words) subtree.

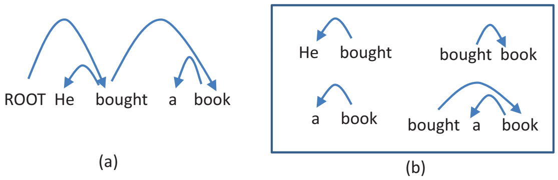

Figure 8: Example of monolingual subtree extraction

From the dependency tree in Figure 8-(a), we obtain the subtrees shown in Figure 8-(b) where the first three are bigram subtrees and the last one is a trigram subtree. After extraction, we obtain the subtree list ST_t that includes two sets, one for bigram subtrees, and the other one for trigram subtrees. We remove the subtrees occurring only once in the data. For each set, we assign labels to the extracted subtrees according to their frequencies by using the same method as that of Chen et al. (2009). If the frequency of a subtree is in the top 10% in the corresponding set, it is labeled HF. If the frequency is between the top 20% and 30%, it is labeled MF. We assign the label LF to the remaining subtrees. We use $Type(st_t)$ to refer to the label of a subtree, st_t.

5.1.2 Verified target constraint function: $F_{vt}(r_{tk})$

We use the extracted target subtrees to verify the r_{tk} of the bilingual constraints. In fact, r_{tk} is a candidate subtree. If the r_{tk} is included in ST_t, function $F_{vt}(r_{tk}) = Type(r_{tk})$, otherwise $F_{vt}(r_{tk}) = ZERO$. For example, in Figure 5 the bigram structure of "all" and "athletes" can form a bigram subtree that is included ST_t and its label is HF. In this case, $F_{vt}(r_{t2}) = HF$.

5.1.3 Bilingual subtrees

We extract bilingual subtrees from a bilingual corpus, which is parsed by the Parser^s and Parser^t on both sides. We extract three types of bilingual subtrees: bigram-bigram (st_{bi22}), bigram-trigram (st_{bi23}), and trigram-trigram (st_{bi33}) subtrees. For example, st_{bi22} consists of a bigram subtree on the source side and a bigram subtree on the target side.

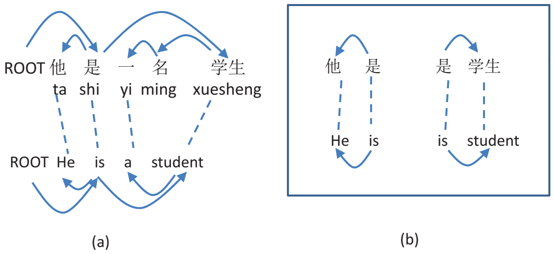

Figure 9: Example of bilingual subtree extraction

From the dependency tree in Figure 9-(a), we obtain the bilingual subtrees shown in Figure 9-(b). Figure 9-(b) shows the extracted bigram-bigram bilingual subtrees. After extraction, we obtain the bilingual subtrees ST_{bi}. We remove the subtrees occurring only once in the data.

5.1.4 Verified bilingual constraint function: $F_{vb}(r_{bink})$

We use the extracted bilingual subtrees to verify the $r_{sn} : r_{tk}$ (r_{bink} in short) of the bilingual constraints. r_{sn} and r_{tk} form a candidate bilingual subtree st_{bink}. If the st_{bink} is included in ST_{bi}, function $F_{vb}(r_{bink}) = +$, otherwise $F_{vb}(r_{bink}) = -$.

5.2 Verified bilingual features

Then, we define another set of bilingual features by combining the verified constraint functions. We call these bilingual features 'verified bilingual features'.

Table 2 lists the verified bilingual features used in our experiments, where each line defines a feature template that is a combination of functions.

We use an example to show how to generate the verified bilingual features in practice. In Figure 4, we want to define the verified features for the bigram dependency (r_{s2}) between "发挥(fahui)" and "技巧(jiqiao)". The corresponding words form a trigram relation r_{t3}. The direction of the bigram dependency is right. Suppose we can find r_{t3} in ST_t with label MF and can not find the candidate bilingual subtree in ST_{bi}. Then we have feature "$\langle F_{b2}(r_{s2} : r_{t3}) = +, F_{vt}(r_{t3}) = MF, RIGHT\rangle$" for the third first-order feature template and feature "$\langle F_{b2}(r_{s2} : r_{t3}) = +, F_{vb}(r_{bi23}) = -, RIGHT\rangle$" for the fifth in Table 2.

First-order features	Second-order features
$\langle F_{ro}\rangle$	
$\langle F_{b2}, F_{vt}(r_{tk})\rangle$	$\langle F_{b3}, F_{vt}(r_{tk})\rangle$
$\langle F_{b2}, F_{vt}(r_{tk}), Dir\rangle$	$\langle F_{b3}, F_{vt}(r_{tk}), Dir\rangle$
$\langle F_{b2}, F_{vb}(r_{bink})\rangle$	$\langle F_{b3}, F_{vb}(r_{bink})\rangle$
$\langle F_{b2}, F_{vb}(r_{bink}), Dir\rangle$	$\langle F_{b3}, F_{vb}(r_{bink}), Dir\rangle$
$\langle F_{b2}, F_{ro}, F_{vb}(r_{bink})\rangle$	

Table 2: Verified bilingual features

6 Experiments

We evaluated the proposed method on the translated portion of the Chinese Treebank V2 (referred to as CTB2$_{tp}$) (Bies et al., 2007), articles 1-325 of CTB, which have English translations with gold-standard parse trees. The tool "Penn2Malt"[2] was used to convert the data into dependency structures. Following the studies of Burkett and Klein (2008), Huang et al. (2009) and Chen et al. (2010), we used the exact same data split: 1-270 for training, 301-325 for development, and 271-300 for testing. Note that we did not use human translation on the English side of this bilingual treebank to train our new parsers. For testing, we used two settings: a test with human translation and another with auto-translation. To process unannotated data, we trained a first-order Parser^s on the training data.

To prove that the proposed method can work on larger monolingual treebanks, we also tested our methods on the CTB7 (LDC2010T07) that includes much more sentences than CTB2$_{tp}$. We used articles 301-325 for development, 271-300 for testing, and the other articles for training. That is, we evaluated the systems on the same test data as CTB2$_{tp}$. Table 3 shows the statistical information on the data sets.

	Train	Dev	Test
CTB2$_{tp}$	2,745	273	290
CTB7	50,747	273	290

Table 3: Number of sentences of data sets used

We built Chinese-to-English SMT systems based on Moses[3]. Minimum error rate training (MERT) with respect to BLEU score was used to tune the decoder's parameters. The translation model was created from the FBIS corpus (LDC2003E14). We used SRILM[4] to train a 5-gram language model. The language model was trained on the target side of the FBIS corpus and the Xinhua news in English Gigaword corpus (LDC2009T13). The development and test sets were from NIST MT08 evaluation campaign[5]. We then used the SMT systems to translate the training data of CTB2$_{tp}$ and CTB7.

To directly compare with the results of Huang et al. (2009) and Chen et al. (2010), we also used the same word alignment tool, Berkeley Aligner (Liang et al., 2006; DeNero and Klein, 2007), to perform word alignment for CTB2$_{tp}$ and CTB7. We trained a Berkeley Aligner on the FBIS corpus (LDC2003E14). We removed notoriously bad links in {a, an, the}×{的(de), 了(le)} following the work of Huang et al. (2009).

To train an English parser, we used the PTB (Marcus et al., 1993) in our experiments and the tool "Penn2Malt" to convert the data. We split the data into a training set (sections 2-21), a development set (section 22), and a test set (section 23). We trained first-order and second-order Parser^t on the training data. The unlabeled attachment score (UAS) of the second-order Parser^t was 91.92, indicating state-of-the-art accuracy on the test data. We used the second-order Parser^t to parse the auto-translated/human-made target sentences in the CTB

[2] http://w3.msi.vxu.se/˜nivre/research/Penn2Malt.html

[3] http://www.statmt.org/moses/

[4] http://www.speech.sri.com/projects/srilm/download.html

[5] http://www.itl.nist.gov/iad/mig//tests/mt/2008/

data.

To extract English subtrees, we used the BLLIP corpus (Charniak et al., 2000) that contains about 43 million words of WSJ texts. We used the MXPOST tagger (Ratnaparkhi, 1996) trained on training data to assign POS tags and used the first-order Parser^t to process the sentences of the BLLIP corpus. To extract bilingual subtrees, we used the FBIS corpus and an additional bilingual corpus containing 800,000 sentence pairs from the training data of NIST MT08 evaluation campaign. On the Chinese side, we used the morphological analyzer described in (Kruengkrai et al., 2009) trained on the training data of CTB$_{tp}$ to perform word segmentation and POS tagging and used the first-order Parser^s to parse all the sentences in the data. On the English side, we used the same procedure as we did for the BLLIP corpus. Word alignment was performed using the Berkeley Aligner.

We reported the parser quality by the UAS, i.e., the percentage of tokens (excluding all punctuation tokens) with correct HEADs.

6.1 Experimental settings

For baseline systems, we used the monolingual features mentioned in Section 3. We called these features basic features. To compare the results of (Burkett and Klein, 2008; Huang et al., 2009; Chen et al., 2010), we used the test data with human translation in the following three experiments. The target sentences were parsed by using the second-order Parser^t. We used PAG to refer to our parsers trained on the auto-generated bilingual treebank.

6.2 Training with CTB2$_{tp}$

	Order-1	Order-2
Baseline	84.35	87.20
PAG$_o$	84.71(+0.36)	87.85(+0.65)
PAG	85.37(+1.02)	88.49(+1.29)
ORACLE	85.79(+1.44)	88.87(+1.67)

Table 4: Results of training with CTB2$_{tp}$

First, we conducted the experiments on the standard data set of CTB2$_{tp}$, which was also used in other studies (Burkett and Klein, 2008; Huang et al., 2009; Chen et al., 2010). The results are given in Table 4, where Baseline refers to the system with the basic features, PAG$_o$ refers to that after adding the original bilingual features of Table 1 to Baseline, PAG refers to that after adding the verified bilingual features of Table 2 to Baseline, and ORACLE[6] refers to using human-translation for training data with adding the features of Table 1. We obtained an absolute improvement of 1.02 points for the first-order model and 1.29 points for the second-order model by adding the verified bilingual features. The improvements of the final systems (PAG) over the Baselines were significant in McNemar's Test ($p < 0.001$ for the first-order model and $p < 0.0001$ for the second-order model). If we used the original bilingual features (PAG$_o$), the system dropped 0.66 points for the first-order and 0.64 points for the second-order compared with system PAG. This indicated that the verified bilingual constraints did provide useful information for the parsing models.

We also found that PAG was about 0.3 points lower than ORACLE. The reason is mainly due to the imperfect translations, although we used the large-scale subtree lists to help verify the constraints. We tried adding the features of Table 2 to the ORACLE system, but the results were worse. These facts indicated that our approach obtained the benefits from the verified constraints, while using the bilingual constraints alone was enough for ORACLE.

6.3 Training with CTB7

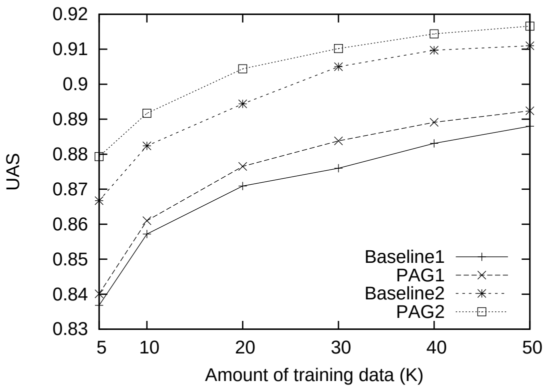

Figure 10: Results of using different sizes of training data

Here, we demonstrate that our approach is still able to provide improvement, even if we use larger

[6] Note that we also used the tool to perform the word alignment automatically.

	Baseline	D10	D20	D50	D100	GTran
BLEU	n/a	14.71	15.84	16.92	17.95	n/a
UAS	87.20	87.63	87.67	88.20	88.49	88.58

Table 5: Results of using different translations

training data that result in strong baseline systems. We incrementally increased the training sentences from the CTB7. Figure 10 shows the results of using different sizes of CTB7 training data, where the numbers of the x-axis refer to the sentence numbers of training data used, Baseline1 and Baseline2 refer to the first- and second-order baseline systems, and PAG1 and PAG2 refer to our first- and second-order systems. The figure indicated that our system always outperformed the baseline systems. For small data sizes, our system performed much better than the baselines. For example, when using 5,000 sentences, our second-order system provided a 1.26 points improvement over the second-order baseline. Finally, when we used all of the CTB7 training data, our system achieved 91.66 for the second-order model, while the baseline achieved 91.10.

6.4 With different settings of SMT systems

We investigated the effects of different settings of SMT systems. We randomly selected 10%, 20%, and 50% of FBIS to train the Moses systems and used them to translate CTB2$_{tp}$. The results are in Table 5, where D10, D20, D50, and D100 refer to the system with 10%, 20%, 50%, and 100% data respectively. For reference, we also used the Google-translate online system[7], indicated as GTran in the table, to translate the CTB2$_{tp}$.

From the table, we found that our system outperformed the Baseline even if we used only 10% of the FBIS corpus. The BLEU and UAS scores became higher, when we used more data of the FBIS corpus. And the gaps among the results of D50, D100, and GTran were small. This indicated that our approach was very robust to the noise produced by the SMT systems.

6.5 Testing with auto-translation

We also translated the test data into English using the Moses system and tested the parsers on the new test data. Table 6 shows the results. The results showed that PAG outperformed the baseline systems for both the first- and second-order models. This indicated that our approach can provide improvement in a purely monolingual setting with the help of SMT.

	Order-1	Order-2
Baseline	84.35	87.20
PAG	84.88(+0.53)	87.89(+0.69)

Table 6: Results of testing with auto-translation (training with CTB2$_{tp}$)

6.6 Comparison results

	With CTB2$_{tp}$		With CTB7	
Type	System	UAS	System	UAS
M	Baseline	87.20	Baseline	91.10
HA	Huang2009	86.3	n/a	
	Chen2010$_{BI}$	88.56		
	Chen2010$_{ALL}$	90.13		
AG	PAG	88.49	PAG	91.66
	PAG+ST$_s$	89.75		

Table 7: Comparison of our results with other previous reported systems. Type M denotes training on monolingual treebank. Types HA and AG denote training on human-annotated and auto-generated bilingual treebanks respectively.

We compared our results with the results reported previously for the same data. Table 7 lists the results, where Huang2009 refers to the result of Huang et al. (2009), Chen2010$_{BI}$ refers to the result of using bilingual features in Chen et al. (2010), and Chen2010$_{ALL}$ refers to the result of using all of the features in Chen et al. (2010). The results showed that our new parser achieved better accuracy than Huang2009 and comparable to Chen2010$_{BI}$. To achieve higher performance, we also added the source subtree features (Chen et al., 2009) to our system: PAG+ST$_s$. The new result is close to Chen2010$_{ALL}$. Compared with the approaches of

[7] http://translate.google.com/

Huang et al. (2009) and Chen et al. (2010), our approach used an auto-generated bilingual treebank while theirs used a human-annotated bilingual treebank. By using all of the training data of CTB7, we obtained a more powerful baseline that performed much better than the previous reported results. Our parser achieved 91.66, much higher accuracy than the others.

7 Conclusion

We have presented a simple yet effective approach to improve bitext parsing with the help of SMT systems. Although we trained our parser on an auto-generated bilingual treebank, we achieved an accuracy comparable to the systems trained on human-annotated bilingual treebanks on the standard test data. Moreover, our approach continued to provide improvement over the baseline systems when we used a much larger monolingual treebank (over 50,000 sentences) where target human translations are not available and very hard to construct. We also demonstrated that the proposed approach can be effective in a purely monolingual setting with the help of SMT.

Acknowledgments

This study was started when Wenliang Chen, Yujie Zhang, and Yoshimasa Tsuruoka were members of Language Infrastructure Group, National Institute of Information and Communications Technology (NICT), Japan. We would also thank the anonymous reviewers for their detailed comments, which have helped us to improve the quality of this work.

References

Ann Bies, Martha Palmer, Justin Mott, and Colin Warner. 2007. English Chinese Translation Treebank V 1.0, LDC2007T02. *Linguistic Data Consortium.*

David Burkett and Dan Klein. 2008. Two languages are better than one (for syntactic parsing). In *Proceedings of EMNLP 2008*, pages 877–886, Honolulu, Hawaii, October. Association for Computational Linguistics.

Xavier Carreras. 2007. Experiments with a higher-order projective dependency parser. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 957–961, Prague, Czech Republic, June. Association for Computational Linguistics.

Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, John Hale, and Mark Johnson. 2000. BLLIP 1987-89 WSJ Corpus Release 1, LDC2000T43. *Linguistic Data Consortium.*

Wenliang Chen, Jun'ichi Kazama, Kiyotaka Uchimoto, and Kentaro Torisawa. 2009. Improving dependency parsing with subtrees from auto-parsed data. In *Proceedings of EMNLP 2009*, pages 570–579, Singapore, August.

Wenliang Chen, Jun'ichi Kazama, and Kentaro Torisawa. 2010. Bitext dependency parsing with bilingual subtree constraints. In *Proceedings of ACL 2010*, pages 21–29, Uppsala, Sweden, July. Association for Computational Linguistics.

Koby Crammer and Yoram Singer. 2003. Ultraconservative online algorithms for multiclass problems. *J. Mach. Learn. Res.*, 3:951–991.

John DeNero and Dan Klein. 2007. Tailoring word alignments to syntactic machine translation. In *Proceedings of ACL 2007*, pages 17–24, Prague, Czech Republic, June. Association for Computational Linguistics.

Liang Huang, Wenbin Jiang, and Qun Liu. 2009. Bilingually-constrained (monolingual) shift-reduce parsing. In *Proceedings of EMNLP 2009*, pages 1222–1231, Singapore, August. Association for Computational Linguistics.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL 2003*, pages 48–54. Association for Computational Linguistics.

Terry Koo and Michael Collins. 2010. Efficient third-order dependency parsers. In *Proceedings of ACL 2010*, pages 1–11, Uppsala, Sweden, July. Association for Computational Linguistics.

Canasai Kruengkrai, Kiyotaka Uchimoto, Jun'ichi Kazama, Yiou Wang, Kentaro Torisawa, and Hitoshi Isahara. 2009. An error-driven word-character hybrid model for joint Chinese word segmentation and POS tagging. In *Proceedings of ACL-IJCNLP2009*, pages 513–521, Suntec, Singapore, August. Association for Computational Linguistics.

Charles N. Li and Sandra A. Thompson. 1997. *Mandarin Chinese - A Functional Reference Grammar*. University of California Press.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of NAACL 2006*, pages 104–111, New York City, USA, June. Association for Computational Linguistics.

Yang Liu and Liang Huang. 2010. Tree-based and forest-based translation. In *Tutorial Abstracts of ACL 2010*, page 2, Uppsala, Sweden, July. Association for Computational Linguistics.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguisticss*, 19(2):313–330.

Ryan McDonald and Fernando Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *Proceedings of EACL 2006*, pages 81–88.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of IWPT2003*, pages 149–160.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Proceedings of EMNLP 1996*, pages 133–142.

David A. Smith and Noah A. Smith. 2004. Bilingual parsing with factored estimation: Using English to parse Korean. In *Proceedings of EMNLP 2004*, pages 49–56.

Hiroyasu Yamada and Yuji Matsumoto. 2003. Statistical dependency analysis with support vector machines. In *Proceedings of IWPT 2003*, pages 195–206.

Hai Zhao, Yan Song, Chunyu Kit, and Guodong Zhou. 2009. Cross language dependency parsing using a bilingual lexicon. In *Proceedings of ACL-IJCNLP2009*, pages 55–63, Suntec, Singapore, August. Association for Computational Linguistics.