Learning to Parse Bilingual Sentences Using Bilingual Corpus and Monolingual CFG

Chung-Chi Huang[1] and Jason S. Chang[2]

[1]Dept. of Information Systems and Application/Taiwan International Graduate Program, National Tsing Hua University, HsinChu, Taiwan, u901571@alumni.nthu.edu.tw
[1]Taiwan International Graduate Program, Academia Sinica, Nankang, Taiwan
[2] Dept. of Computer Science, National Tsing Hua University, HsinChu, Taiwan, Jason.jschang@gmail.com

Abstract

We present a new method for learning to parse a bilingual sentence using Inversion Transduction Grammar trained on a parallel corpus and a monolingual treebank. The method produces a parse tree for a bilingual sentence, showing the shared syntactic structures of individual sentence and the differences of word order within a syntactic structure. The method involves estimating lexical translation probability based on a word-aligning strategy and inferring probabilities for CFG rules. At runtime, a bottom-up CYK-styled parser is employed to construct the most probable bilingual parse tree for any given sentence pair. We also describe an implementation of the proposed method. The experimental results indicate the proposed model produces word alignments better than those produced by Giza++, a state-of-the-art word alignment system, in terms of alignment error rate and F-measure. The bilingual parse trees produced for the parallel corpus can be exploited to extract bilingual phrases and train a decoder for statistical machine translation.

1. Introduction

1.1. Background

The amount of information available in English on the Internet has grown exponentially for the past few years. Although a myriad of data are at our disposal, non-native speakers often find it difficult to wade through all of it since they may not be familiar with the terms or idioms being used in the texts.

To ease the situation, a number of online machine translation (MT) systems such as SYSTRAN and Google Translate provide translation of source text on demand. Moreover, online dictionaries have mushroomed to provide access at any time and everywhere for second language learners.

1.2. Motivation

MT systems and bilingual dictionary are designed to provide the services for non-English speakers or to ease learning difficulties for second language learners. Both require a lexicon which can be derived from aligning words in a parallel corpus.

Furthermore, second language learners can benefit by learning from example sentences with translations. By looking at bilingual examples, we acquire knowledge of the usage and meaning of word in context. With word alignment result of a sentence pair, it is much easier to grab the essential concepts of unfamiliar foreign words in a sentence pair.

For instance, consider the English sentence "These factors will continue to play a positive role after its return" with its segmented Chinese translation "香港 回歸 後 這些 條件 將會 繼續 發揮 積極 作用" shown in Figure 1, where the solid dark lines are word alignment results of them and e, f stand for two sentences in two languages E, F respectively. If we don't know the usage of "play" in the sense of "perform," in this example sentence pair with the help of word alignment, we would quickly understand such meaning and learn useful expressions like "play … role" meaning "發揮 … 作用" in Chinese.

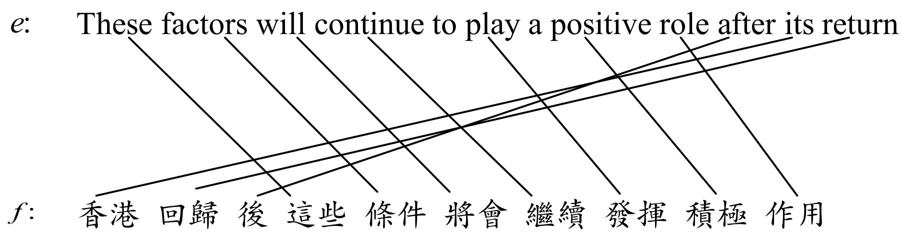

Figure 1. An example sentence pair.

Table 1. The word alignment of the example sentence pair.

i	j	e_i	f_j
1	4	These	這些
2	5	factors	條件
3	6	will	將會
4	7	continue	繼續
5	0	to	ε
6	8	play	發揮
7	0	a	ε
8	9	positive	積極
9	10	role	作用
10	3	after	後
11	1	its	香港
12	2	return	回歸

Table 1 shows the word alignment result of above example sentenece pair. In Table 1 we use 0, and ε to denote the corresponding translation does not exist for a particular word, that is, this word in one language is translated into no words in another and we use e_i, f_j to stand for the words at the position of i, j in sentence e, f respectively.

1.3. Bilingual Parsing

If we look more closely to the example sentence in Figure 1, we would notice that the beginning half "These factors will continue to play a positive role" is translated into the back of the Chinese sentence whileas the ending half "after its return" is translated into the beginning. This phenomenon is very common while translating one language into another. A simple observation is that if one language is SVO-structured and another SOV-structured, the "VO" part of the first language would constantly be reversely translated into "OV" of the second because of the reverse ordering of syntactic structures in "V" and "O" in these languages. We call it inverted word order during translation. More often than inverted cases, we have straight word order such as when "positive role" is translated into "積極 作用". It would occur more frequently if two languages have identical word orientation for a syntactic structure, such as adjectives modifying nouns in English and Chinese noun phrases.

In this paper, we propose a new method of learning to recognize straight and inverted phrases in bilingual parsing by using a parallel corpus and a monolingual treebank. The parallel text will be exploited to provide lexical translation information and project the syntactic information available in the source-language treebank onto the target language. This way we can leverage the monolingual treebank and avoid the difficult problem of inducing a bilingual grammar from scratch. We identify production rules derived from the treebank based on the part of speech information of the source text. This information is simultaneously projected to the target language by exploiting the cross-language lexical information produced by a word-aligning method. The relation of straight or inverted word orders between the syntax of the two languages at all phrase levels can be captured and modeled during the process. At runtime, these production rules are used to parse bilingual sentences, simultaneously determining the syntactic structures and word order relationships of languages involved.

Thus, the proposed model commits to common linguistic labels for words and phrases found in an English treebank, such as NN (noun), VB (verb), JJ (adjective), NP (noun phrase), VP (verb phrase), ADJP (adjective phrase), PP (prepositional phrase). Furthermore, we assume straight and inverted linguistic phenomena, when projected to the target language, should render a reasonable structural explanation of the target language. We extend ITG productions (Wu 1997) to carry out this process of projection. Take word-aligned sentences in Figure 1 for example. It is possible to match the part of speech information of the source language sentence against the right hand sides of the production rules induced from a tree bank and identify the instances of applying specific rules such as $NP \rightarrow JJ\ NN$; $JJ \rightarrow \text{"positive"}$ and $NN \rightarrow \text{"role."}$ Moreover, by exploiting the word alignment information, it is not difficult to infer that such syntactic structure is also present in the target language with similar rules such as $NP \rightarrow JJ\ NN$; JJ → ”積極,” and NN → ”作用.” By combining and tallying such information, we are likely to derive ITG productions such as $NP \rightarrow [JJ\ NN]$; JJ → “positive/積極” and NN → “role/作用.” Here, the square bracket pair, “[“ and “]” signifies that a straight synchronous nominal share between English and Mandarin Chinese. Similarly, we would also find out the inverted prepositional phrases like $PP \rightarrow \langle IN\ NP \rangle$; IN → “after/後” and NP → “its return/香港 回歸” where “<“ and “>” indicate cross-language inverted structure. See Figure 3 for more details. Additionally, the occurrence counts of these straight or inverted structures can be tallied and used in estimating the probabilistic parameters of the ITG model.

Intuitively, with rules like those shown in Figure 2 learned from a parallel corpus and a monolingual treebank, we should be able to extend a CYK-style parser to derive bilingual parse tree as shown in Figure 3, where the symbol $\star$ indicates word order of the subtrees in the target language is inverted. According to the theory of ITG, the probability of a bilingual parse tree consists of the lexical translation probability and the probability for the straight or inverted production rules involved.

$$
\begin{aligned}
&S \rightarrow \langle NP\ PP \rangle \\
&NP \rightarrow [NP\ VP] \\
&NP \rightarrow [DT\ NP] \\
&VP \rightarrow [VP\ VP] \\
&VP \rightarrow [TO\ VP] \\
&VP \rightarrow [VP\ NP] \\
&NP \rightarrow [JJ\ NN] \\
&PP \rightarrow \langle IN\ NP \rangle \\
&NP \rightarrow [PRP\$\ NN]
\end{aligned}
$$

Figure 2. Example grammar rules for the sentence pair.

The rest of the paper is organized as follows. We review the related work in the next section. In Section 3, we describe the steps for learning synchronous grammar rules in the form of ITG and the association probabilistic estimation. An implementation of the bottom-up CYK-styled bilingual parser based on ITG is also described in Section 3. Reports on experiments and discussions are covered in Section 4 and 5, respectively.

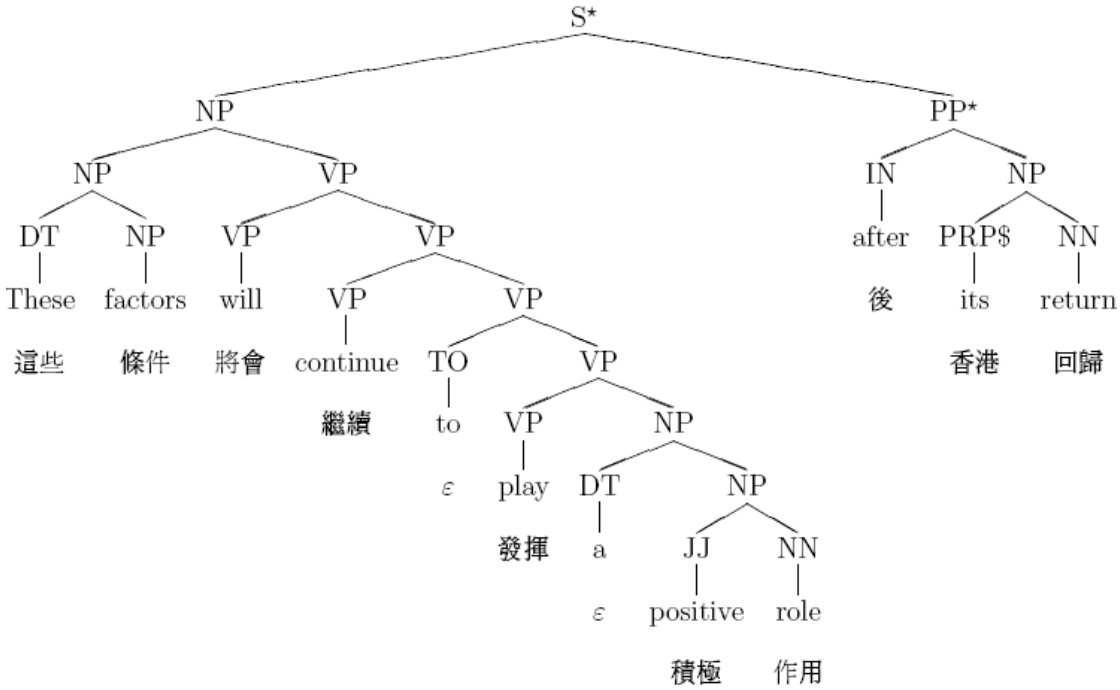

Figure 3. A bilingual parse tree for example sentence pair.

2. Related Works

A statistical translation model (STM) is a mathematical model in which process of human translation from one language into another is modeled statistically. Model parameters are estimated using a corpus of translation pairs with or without human supervision. STMs have been used in various researches and applications including statistical machine translation, word alignment of a sentence-aligned corpus and the automatic construction of a dictionary, just to name a few. For this point of view, a better STM cross language for processing is essential and fundamental for those applications.

Brown et al. (1988) first described a STM, or the alignment of sentence and word pairs in different languages. This and subsequent IBM models are based on noisy channel which converts or translates a sequence of words in one language into another. IBM model 1 can be trained using EM algorithm: starting with a uniform distribution among all translation candidate pairs and ending with convergent probabilities. While IBM model 1 does not utilize position information, the subsequent IBM models take positions into account when modeling for the translation process. (take an English-Chinese sentence pair for example, the first English word more likely translates into the first word in the Chinese sentence)

Another model called Hidden Markov model (HMM) is designed to capture localization effect in aligning the words in parallel texts. Vogel et al. (1996), motivated by the idea that words are not distributed arbitrarily over the sentence positions but tend to form clusters, presented a first-order HMM which makes the alignment probabilities explicitly dependent on the alignment position of the previous word. Nonetheless, Toutanova et al. (2002) pointed out that word order variations (large jumps) between languages seem to be a problem.

Neither IBM models nor HMMs explicitly utilize any linguistic information. However, other researchers have experimented with incorporation of part of speech (POS) information or context-specific features into STM. Exploiting POS tags of the two languages, Toutanova et al. (2002) introduced tag translation probabilities and tag sequences for jump probabilities to improve HMM-based word alignment models in modeling local word order differences. Cherry and Lin (2003) made use of dependency trees of a language to model features and constraints that are based on linguistic intuitions. In contrast, our model which uses POS information and tree structures from a treebank of a language to derive relation of syntax of two languages based on initial word alignments takes into consideration positions and linguistic characteristics such as word order and syntactic structures.

Wang (1998) enhanced the IBM models by introducing phrases, and Och et al. (1999) made use of templates to capture phrasal sequences in a sentence. While flat structures of languages beyond words are being used in above researches, often researchers attempted using nested structures. Those studies can be divided into two approaches according to whether they are linguistically syntax-based or not. Either ways, both approaches try to model structural differences between two languages.

Wu (1997) described an Inversion Transduction Grammar to model translation. However, only a lesser version, bracketing transduction grammar (BTG) with three structural labels *A,B,C* and a start symbol *S,* was experimented to perform bilingual parsing. Nevertheless, BTG accommodates a wide range of ordering variation between languages and imposes a realistic position distortion penalty. In other words, a system with structural-like, or hierarchical-like rules that specify the constituents and the order of the counterparts in both language is good at resolving the word alignment relations within a sentence pair. However, in their experiments, constituent categories are almost not differentiated, and thus their influences on ordering preferences of the counterparts are not taken into consideration. Consequently, very little syntax information is incorporated into the process of bilingual parsing. In contrast to Wu's experiment, we use regular context-free grammar rules in our experiments.

More recently, Yamada and Knight (2001) suggested the syntax differences in languages are really a better way to model translation. In their work, the English sentence goes through a parser to generate a full parse tree. Subtrees of each node are reordered, function words are inserted and finally the tree is linearized to produce the target sentence. The parse tree of an English sentence is generated independently from the target sentence. Although the monolingual parse might be correct, it may be difficult to project the structures onto the target language. Instead, our model has grammar rules that specify bilingual syntactic information including constituent labels and word ordering, which enables us to extend a CYK parser to parse bilingual sentences simultaneously.

Chiang (2005) introduced lexicalized labelless hierarchical bilingual phrase structure to model translation without any linguistic commitment. Since he does not assign any syntactic category to hierarchical phrase pairs, the rules he obtain are not generalized into linguistics-motivated constituents but anchored at certain words. These lexicalized rewrite rules specify the differences in hierarchical structure of two languages without generalization. Therefore, the size of the grammar tends to be very large (2.2M rules). The rules do not represent some general ideas of languages such as word classes like verb, noun, or adjective, but rather have to do with specific words. In any case, the word classes like verb, noun, and adjective and the phrase categories like verb phrase (*VP*), noun phrase (*NP*) and adjective phrase (*ADJP*) would provide a more general way to reflect the parallel and differences of languages. Chiang also posed the hypothesis that syntactic phrases are better for machine translation (MT) and predicted the future trend of MT is to move towards a more syntactically-motivated grammar. With that in mind, we exploit part-of-speech information and linguistic phrase categories to model the syntactic relation between two languages, which is designed to have a higher degree of generality, unlike Chiang's lexicalized labelless production rules.

In contrast to previous work in STM, the proposed method not only automatically identifies the hidden structural information of two languages but models variations of ordering counterparts within them. Moreover, a much-smaller set of flexible context-free grammar rules obtained from a very large-scale parallel corpus. Syntactic information indicated by those rules is exploited to parse bilingual sentences.

3. The Model

A promising method for learning to parse a bilingual sentence using Inversion Transduction Grammars is based on training on a monolingual treebank and a parallel corpus. We project part of speech information and syntactic structures from a treebank of source language onto target language based on initial word alignment results of a parallel corpus to obtain and estimate the probabilities for ITG rules. During the projection process, word order relationships (*straight* and *inverted*) of shared syntactic constructs between two languages are identified and modeled. At runtime, the derived ITG rules drive a CYK-style parser to construct bilingual parse trees and hopefully lead to better word alignment results at the leaf nodes.

3.1. Problem Statement

The model is aimed at statistically derived ITG rules with probability and making use of those rules for bilingual parsing and word alignments. We focus on the process of bilingual parsing which exploits the syntactic information such as shared syntactic structures and word order relationships in two languages using a parallel corpus and a monolingual treebank.

Problem Statement: Given a sentence-aligned corpus $\mathbf{C} = \{(r,e,f) \mid 1 \le r \le n\}$ where r is the record number of the aligned sentence pair (e,f) and n is the total number of sentence pairs in parallel corpus $\mathbf{C}$, and a grammar $\mathbf{G} = \{lhs \rightarrow rhs \mid lhs \rightarrow rhs \text{ is a grammar rule on } E \text{ side}\}$ derived from a source-language treebank, we extend $\mathbf{G}$ into ITG rewrite rules for bilingual parsing.

For the rest of this section, we describe our solution to this problem. First, we elaborate on our training process for learning synchronous context-free grammar rules in the form of probabilistic estimation for ITG rules in Section 3.2. Then, we describe the implementation of a bottom-up bilingual parsing algorithm based on ITG in Section 3.3.

3.2. Proposed Training Process

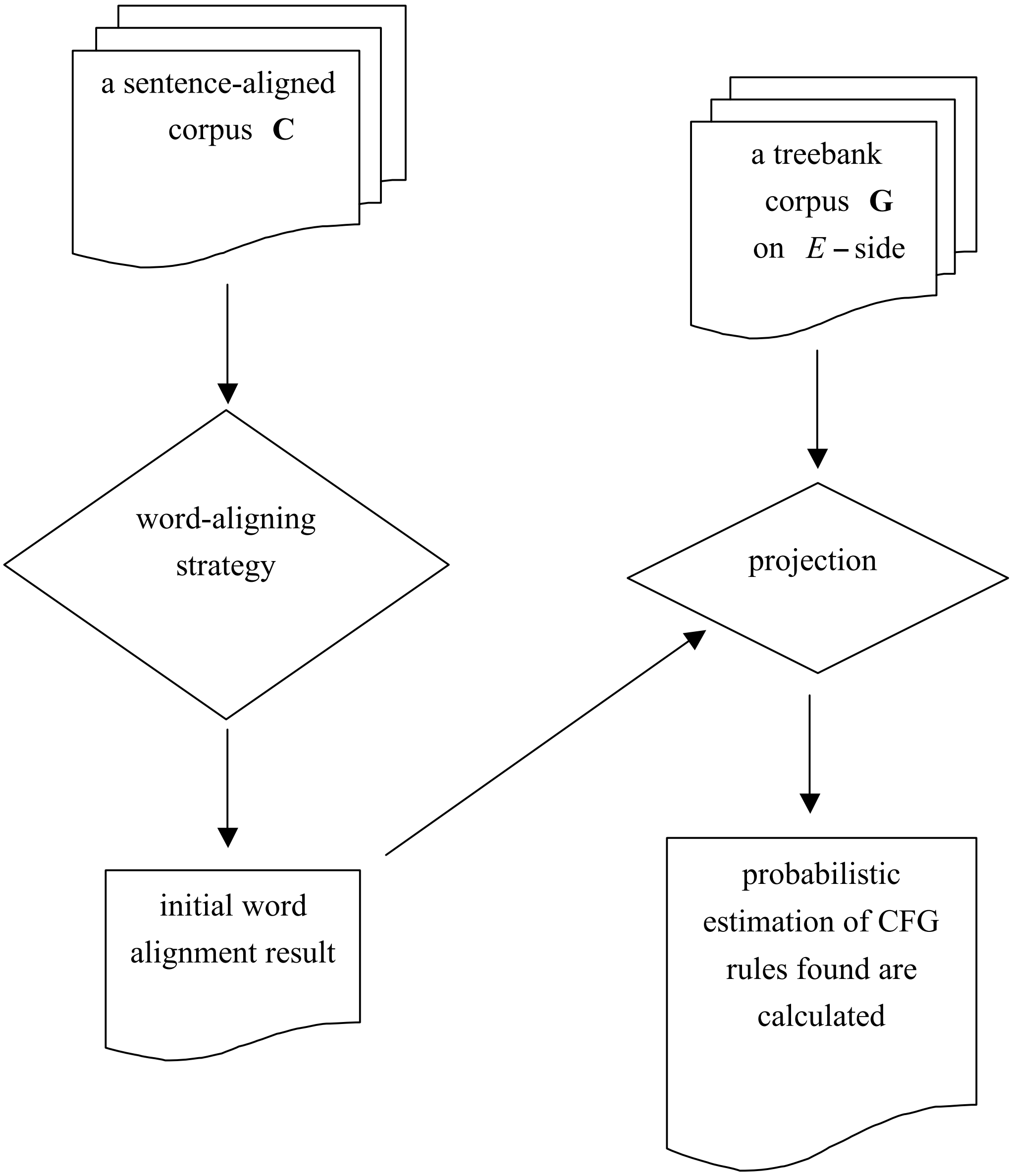

Figure 4: Flowchart of the proposed training process.

The training process can be illustrated using the flowchart in Figure 4.

Given a sentence-aligned corpus $\mathbf{C} = \{(r, e, f) \mid 1 \le r \le n, e \text{ and } f \text{ are an aligned sentence pair}\}$ where r is the record number of the sentence pair and n is the total number of sentence pairs in $\mathbf{C}$, a source-language grammar $\mathbf{G}$, we map part of speech information and syntactic structures of source language onto target language words using word alignment result. During the mapping process, we exploit occurrence of syntactic structures and the differences of word order of the right-hand-side constituents to estimate probabilities. The proposed training process is elaborated as follows.

Table 2. Outline of the training process.

(1)	Tag source-language sentences and segment target-language sentences (Section 3.2.1)
(2)	Apply a word-aligning strategy to obtain word alignment result (Section 3.2.2)
(3)	Apply the algorithm of projecting linguistic information of source language onto target language and estimating related probabilities of grammar rules found (Section 3.2.3)

Table 3. Lemmas and tags for English sentence of sentence pair 193.

position (i)	**lemma (e_i)**	**tag (t_i)**
1	these	*DT*
2	factor	*NNS*
3	will	*MD*
4	continue	*VB*
5	to	*TO*
6	play	*VB*
7	a	*DT*
8	positive	*JJ*
9	role	*NN*
10	after	*IN*
11	its	*PRP*$
12	return	*NN*

3.2.1 Tagging and Segmenting

In the first stage of the training process, for every sentence-aligned pair (e, f) in corpus $\mathbf{C}$, we tag sentence e using a POS tagger and generate $e = (e_1, e_2, \cdots, e_m)$ with tag sequence $(t_1, t_2, \cdots, t_m)$, where e_i stands for the i[th] word in e with m words and t_i stands for the POS tag of the word e_i. Further, we segment sentence f to obtain $\left(f_1, f_2, \cdots, f_n\right)$, where f_j stands for the j[th] word in f with n words.

Take sentence pair whose record number is 193 in Figure 1 for instance. Table 3 shows the lemmatized and tagged result of the English sentence, while Table 4 shows the segmentation result of the Chinese sentence.

Table 4. Segments for Chinese sentence of sentence pair 193.

position (j)	**segments (f_j)**
1	香港
2	回歸
3	後
4	這些
5	條件
6	將會
7	繼續
8	發揮
9	積極
10	作用

The POS information of sentence e will then be projected onto the target language based on word alignments described in next subsection.

3.2.2 Initial Word Alignments

In the second training stage, we obtain a word-aligning set **A** for corpus **C** by applying any existing word-level alignment method.

For notation convenience, we use 8-tuple $\left(r, i_1, i_2, j_1, j_2, L, rhs, rel\right)$ to represent that substring pair $\left(e_{i_1} \cdots e_{i_2}, f_{j_1} \cdots f_{j_2}\right)$ in sentence pair r has $L \rightarrow rhs$ as the derivation leading to the bilingual structure and rel as the cross-language word order relations (straight or inverted) of constituents of rhs. The right hand side, rhs, can be either a sequence of nonterminals or a single terminating bilingual word pair and the word order relation, rel, is either S (straight) or I (inverted). Followings are some examples using the 8-tuple representation. The tuple $\left(193, 1, 2, 4, 5, NP, DT\ NN, \mathrm{S}\right)$ denotes a straight bilingual noun phrase (these factors,這些 條件) in sentence 193. Similarly, the tuple $\left(193, 10, 12, 1, 3, PP, IN\ NP, \mathrm{I}\right)$ denotes an inverted prepositional phrase (after its return,香港 回歸後). The tuple (193,8,8,9,9,JJ,positive/積極,S) denotes a terminal bilingual adjective (positive,積極) which can be obtained from word alignment result.

Table 5. Some alignments after applying a word-aligning strategy.

# of sentence pair	i	j	e_i	f_j	t_i
406	10	5	in	在	*IN*
406	11	8	overseas	海外	*JJ*
406	12	18	Chinese	中國	*JJ*
406	13	10	community	社區	*NN*

Further take word alignments of the sentence pair specified in Table 5 for example. $\mathbf{A}$ would at least contain entries like (406,10,10,5,5,*IN*,in/在,S), (406,11,11,8,8,*JJ*,overseas/海外,S), (406,12,12,18,18,*JJ*,Chinese/中國,S) and (406,13,13,10,10,*NN*,community/社區,S).

3.2.3 Algorithm for Probability Estimation

In the final stage of the training process, we map the part of speech information and tree structures available in treebank of language E onto language F based on word alignment result.

We exploit following algorithm to identify syntactic structures of E and model the syntactic relation between E and F. The resulting ITG grammar will then be used in a bottom-up CYK parser for parsing bilingual sentences.

The algorithm begins with a set $\mathbf{H}$ initialized as word-aligning result $\mathbf{A}$. Then recursively select two elements from $\mathbf{H}$. If these two tuples have contiguous word sequence on source-language side and exhibit *straight* or *inverted* relation between source and target language during the mapping process, a new tuple representing these two is added into $\mathbf{H}$. In the end, we exploit the occurrence in $\mathbf{H}$ to estimate following probabilities: $\mathrm{P}\left(L \to \left[R_1\ R_2\right]\right)$, $\mathrm{P}\left(L \to \left\langle R_1\ R_2\right\rangle\right)$ and $\mathrm{P}\left(L \to t\right)$.

In this algorithm, we follow the notation described in section 3.2.1 and use $\left|\mathbf{W}\right|$ to stand for the number of entries in set $\mathbf{W}$, $\mathrm{count}\left(p;\mathbf{Q}\right)$ for the frequency of p in set $\mathbf{Q}$ and δ for the tolerance of *straight*/*inverted* phenomenon within source and target languages.

Algorithm for Probabilistic Estimation

H = A

For $\left(r, i_1, i_2, j_1, j_2, L, rhs, rel\right) \in \mathbf{H}, \left(\overline{r}, \overline{i_1}, \overline{i_2}, \overline{j_1}, \overline{j_2}, \overline{L}, \overline{rhs}, \overline{rel}\right) \in \mathbf{H}$ have not yet been considered

If ($i_2 = \overline{i_1} - 1$)

For every $L' \rightarrow L\ \overline{L} \in \mathbf{G}$

If ($j_2 + 1 \leq \overline{j_1} \leq j_2 + \delta$)

$$\mathbf{H} = \mathbf{H} \cup \left\{ \left(r, i_1, \overline{i_2}, j_1, \overline{j_2}, L', L\ \overline{L}, \mathrm{S} \right) \right\}$$

If ($\overline{j_2} + 1 \leq j_1 \leq \overline{j_2} + \delta$)

$$\mathbf{H} = \mathbf{H} \cup \left\{ \left(r, i_1, \overline{i_2}, \overline{j_1}, j_2, L', L\ \overline{L}, \mathrm{I} \right) \right\}$$

If ($\overline{i_2} = i_1 - 1$)

For every $L' \rightarrow \overline{L}\ L \in \mathbf{G}$

If ($\overline{j_2} + 1 \leq j_1 \leq \overline{j_2} + \delta$)

$$\mathbf{H} = \mathbf{H} \cup \left\{ \left(r, \overline{i_1}, i_2, \overline{j_1}, j_2, L', \overline{L}\ L, \mathrm{S} \right) \right\}$$

If ($j_2 + 1 \leq \overline{j_1} \leq j_2 + \delta$)

$$\mathbf{H} = \mathbf{H} \cup \left\{ \left(r, \overline{i_1}, i_2, j_1, \overline{j_2}, L', \overline{L}\ L, \mathrm{I} \right) \right\}$$

For $\left(r, i_1, i_2, j_1, j_2, L, rhs, rel \right) \in \mathbf{H}$

If ($rhs \neq t$)// t stands for terminating bilingual word pair

$$\mathrm{P}\left(L \rightarrow \left[R_1\ R_2 \right] \right) = \frac{\mathrm{count}\left(\left(*,*,*,*,*, L, R_1\ R_2, \mathrm{S} \right); \mathbf{H} \right)}{\left| \mathbf{H} \right|}$$

$$\mathrm{P}\left(L \rightarrow \left\langle R_1\ R_2 \right\rangle \right) = \frac{\mathrm{count}\left(\left(*,*,*,*,*, L, R_1\ R_2, \mathrm{I} \right); \mathbf{H} \right)}{\left| \mathbf{H} \right|}$$

Else

$$\mathrm{P}\left(L \rightarrow t \right) = \frac{\mathrm{count}\left(\left(*,*,*,*,*, L, t, \mathrm{S} \right); \mathbf{H} \right)}{\left| \mathbf{H} \right|}$$

Table 6. Some alignments by applying an aligning strategy on corpus $\mathbf{C}$.

# of sentence pair	i	j	e_i	f_j	t_i
1	1	1	solemn	莊嚴	*JJ*

1	2	2	ceremony	儀式	*NN*
1	3	3	mark	標誌	*VBZ*
1	4	4	handover	回歸	*NNS*
9	24	6	before	前	*IN*
9	25	5	midnight	午夜	*NN*
62	12	5	provisional	臨時	*JJ*
62	13	6	legislative	立法	*JJ*
62	14	7	council	會	*NN*
249	2	2	will	將	*MD*
249	3	3	strive	致力	*VB*

Consider the word alignment results in Table 6 as an example, the algorithm described above will identity syntactic structures and model syntax relations of languages. The overall projecting process is as follows.

Initially, for sentence pair 1, we have the following in $\mathbf{A}$.

(1,1,1,1,1,*JJ*,solemn/莊嚴,S)

(1,2,2,2,2,*NN*,ceremony/儀式,S)

(1,3,3,3,3,*VBZ*,mark/標誌,S)

(1,4,4,4,4,*NNS*,handover/回歸,S)

Table 7. Examples for the algorithm.

# of sentence pair	rule	entry
9	$PP \rightarrow IN\ NN$	$(9, 24, 25, 5, 6, PP, IN\ NN, \mathrm{I})$
62	$NP \rightarrow ADJP\ NN$	$(62, 12, 14, 5, 7, NP, ADJP\ NN, \mathrm{S})$
249	$VP \rightarrow MD\ VB$	$(249, 2, 3, 2, 3, VP, MD\ VB, \mathrm{S})$

After the first round, we have (1,1,2,1,2,*NP*,*JJ NN*,S), (1,3,4,3,4,*VP*,*VBZ NNS*,S). After the second round, we have (1,1,4,1,4,*S*,*NP VP*,S) where syntactic label S means simple declarative clause in linguistic sense.

Table 7 illustrates some derived grammar rules and entries inserted into $\mathbf{H}$ from sentence pair 9, 62 and 249.

3.3. Bottom-up Parsing

We then describe how we implement a bilingual parser which makes use of syntactic structures and preferences of word order within languages specified by automatically trained ITG rules.

We follow Wu's (1997) definition of $\delta_{stuv}(i)$ to denote the probability of the most likely parse tree with syntactic label i and containing substring pair $(e_{s+1}\ e_{s+2} \cdots e_t, f_{u+1}\ f_{u+2} \cdots f_v)$ in bilingual sentence (e, f).

3.3.1 Implementation

Given sentence $e = (e_1, e_2, \cdots, e_m)$ with tag sequence $(t_1, t_2, \cdots, t_m)$, its corresponding translation sentence $f = (f_1, f_2, \cdots, f_n)$, and a set of probabilities such as $\mathrm{P}(L \to t)$, $\mathrm{P}(L \to [R_1\ R_2])$ and $\mathrm{P}(L \to \langle R_1\ R_2 \rangle)$ associated with ITG, we utilize dynamic programming technique to find the most probable derivation to parse the bilingual sentence (e, f). Basically, we try to calculate the value of $\delta_{0m0n}(\overline{S})$ and backtrack by using following three steps, where $\overline{S}$ is the start symbol.

Step 1: Initial step

$$\delta_{i-1,i,j-1,j}(t_i) = \mathrm{P}(t_i \to e_i/f_j) \qquad \text{for } 1 \le i \le m, 1 \le j \le n$$

$$\delta_{i-1,i,j-1,j}(L) = \mathrm{P}(L \to e_i/f_j) \qquad \text{for } 1 \le i \le m, 1 \le j \le n, L \to t_i \in \mathbf{G}$$

$$\delta_{i-1,i,j,j}(t_i) = \mathrm{P}(t_i \to e_i/\varepsilon) \qquad \text{for } 1 \le i \le m, 0 \le j \le n$$

$$\delta_{i-1,i,j,j}(L) = \mathrm{P}(L \to e_i/\varepsilon) \qquad \text{for } 1 \le i \le m, 0 \le j \le n, L \to t_i \in \mathbf{G}$$

$$\delta_{i,i,j-1,j}(NN) = \mathrm{P}(NN \to \varepsilon/f_j) \qquad \text{for } 0 \le i \le m, 1 \le j \le n$$

Step 2: Recurrent step (bottom-up approach)

We proceed similar to Wu's algorithm. However, we observe that the length of the translation of a substring of source sentence should be bounded. We use the upper and lower bounds of lengths to prune search space and speed up computation. Consequently, $\delta_{stuv}^{[\ ]}(i), \delta_{stuv}^{\langle\ \rangle}(i)$ are calculated as below:

If $\frac{1}{ratio} \le \frac{t-s}{v-u} \le ratio$

$$\delta_{stuv}^{[\,]}(i) = \max_{\substack{j\in\mathbf{PJ} \\ k\in\mathbf{PK} \\ s\le S\le t \\ u\le U\le v \\ (S-s)(t-S)+(U-u)(v-U)\ne 0}} \left\{ \mathrm{P}\left(i \to \left[j\ k\right]\right) \times \delta_{sSuU}(j) \times \delta_{StUv}(k) \right\}$$

where **PJ** is the set consisting of possible syntactic labels for substring pair $\left(e_{s+1}\cdots e_S, f_{u+1}\cdots f_U\right)$ and **PK** is the set consisting of possible syntactic labels for substring pair $\left(e_{S+1}\cdots e_t, f_{U+1}\cdots f_v\right)$

Else

$$\delta_{stuv}^{[\,]}(i) = low\_probability$$

If $\frac{1}{ratio} \le \frac{t-s}{v-u} \le ratio$

$$\delta_{stuv}^{\langle\,\rangle}(i) = \max_{\substack{j\in\mathbf{PJ} \\ k\in\mathbf{PK} \\ s\le S\le t \\ u\le U\le v \\ (S-s)(t-S)+(U-u)(v-U)\ne 0}} \left\{ \mathrm{P}\left(i \to \left\langle j\ k\right\rangle\right) \times \delta_{sSUv}(j) \times \delta_{StuU}(k) \right\}$$

where **PJ** is the set consisting of possible syntactic labels for substring pair $\left(e_{s+1}\cdots e_S, f_{U+1}\cdots f_v\right)$ and **PK** is the set consisting of possible syntactic labels for substring pair $\left(e_{S+1}\cdots e_t, f_{u+1}\cdots f_U\right)$

Else

$$\delta_{stuv}^{\langle\,\rangle}(i) = low\_probability$$

Step 3: Reconstructing step

We exploit depth-first-traversal to construct the most probable bilingual parse tree for sentence pair (e, f).

3.3.2 Example Parse

Take sentence pair in Figure 1 for example.

At initial step, we would build the leaf nodes of the bilingual parse tree using probability like P(*DT* → these/這些), P(*NNS* → factors/條件), P(*NP* → factors/條件), ⋯, P(*IN* → after/後), P(*PP* → after/後), P(*PRP$* → its/香港), P(*NP* → its/香港) and etc.

At recurrent step, we find the most likely derivation of nodes using statistics derived so far. Take nodes in Figure 3 for instance. We will derive (these factors, 這些 條件) as a noun phrase using $NP \to [DT\ NP]$, an *inverted* prepositional phrase (after its return, 香港 回歸 後) using $PP \to \langle IN\ NP \rangle$, and a *straight* verb phrase (play a positive role, 發揮 積極 作用) using $VP \to [VP\ NP]$.

After reconstructing step, the most probable bilingual parse tree of the sentence pair is constructed. Figure 3 illustrates the tree structures derived for the example bilingual sentences.

4. Experiments

Our model is aimed at capturing shared syntactic structures and preferences in word order between two languages. The context-free grammar rules obtained in training process identity syntactic structures and model relations of syntax of languages involved. These rules can be exploited to produce better word-level alignments and most probable bilingual parse trees since syntactic information is taken into consideration.

In this section, we first present the details of training our model in Section 4.1. Then, we describe the evaluation metrics for the performance of the trained model in Section 4.2. The evaluation results are reported in Section 4.3.

4.1. Training Setting

We used the news portion of Hong Kong Parallel Text (Hong Kong news) distributed by Linguistic Data Consortium (LDC) as our sentence-aligned corpus $\mathbf{C}$. The corpus consists of 739,919 English and Chinese sentence pairs. English sentence is considered to be the source while Chinese sentence is the target. The average sentence length is 24.4 words for English and 21.5 words for Chinese. Table 8 and Table 9 show the statistics of number of sentences in this corpus according to sentence length. For monolingual treebank corpus $\mathbf{G}$, we made use of PTB section 23 production rules distributed by Andrew B. Clegg (http://textmining.cryst.bbk.ac.uk/acl05/). There are 2,184 distinct grammar rules. The statistics of $\mathbf{G}$ is shown in Table 10 while Table 11 illustrates some examples of grammar rules in $\mathbf{G}$.

Table 8. Statistics on English side.

sentence length	number of sentence	percentage
0~5	93,354	12.6%
6~10	118,513	16.0%
11~15	70,634	9.5%
16~20	66,431	9.0%
21~25	74,813	10.1%
26~30	71,902	9.7%
31~35	63,816	8.6%
36~	180,456	24.4%

Table 9. Statistics on Chinese side.

sentence length	number of sentence	percentage
0~5	146,957	19.9%
6~10	81,716	11.0%
11~15	72,870	9.8%
16~20	90,286	12.2%
21~25	84,802	11.4%
26~30	74,739	10.1%
31~35	57,347	7.7%
36~	131,202	17.7%

Table 10. Statistics of monolingual treebank.

# of constituents on right hand side	# of distinct grammars	percentage
1	106	4.85%
2	418	19.13%
3	752	34.43%
4	553	25.32%
5~	355	16.25%

Table 11. Example grammars in $\mathbf{G}$.

grammar rules	
$VP \rightarrow VB$	$NP \rightarrow DT\ ADJP\ NNS$
$ADJP \rightarrow RB\ JJ$	$PP \rightarrow RB\ IN\ NP$
$VP \rightarrow TO\ VP$	$VP \rightarrow MD\ ADVP\ VP$
$PP \rightarrow IN\ NP$	$ADVP \rightarrow ADVP\ CC\ ADVP$
$NP \rightarrow DT\ JJ\ NN$	

As for word alignment, we used bidirectional ranking (BDR) as the word-aligning strategy in training process, which means in a sentence pair, e_i and f_j will be aligned if $j = \arg\max_{i-sw \le q \le i+sw} dice\left(e_i, f_q\right)$, $i = \arg\max_{j-sw \le p \le j+sw} dice\left(e_p, f_j\right)$ and $dice\left(e_i, f_j\right) > \theta_{dice}$ where sw is the winow size (set to 7 in the experiment), θ_{dice} is the threshold for dice (set to 0.002) and $dice\left(\bar{e}, \bar{f}\right)$ is calculated as

$$\frac{2 \times \left|link(\bar{e}, \bar{f})\right|}{\left|link(\bar{e}, *)\right| + \left|link(*, \bar{f})\right|}$$

where * is the wildcard symbol and $\bar{e}, \bar{f}$ are words in language E, F respectively. Furthermore, in estimating ITG, we consider only fourgram on English side, that is, entries $\left(r, i_1, i_2, j_1, j_2, L, det\right)$ in $\mathbf{H}$ satisfy the criterion $i_2 - i_1 \le 3$. For the *straight* case to hold, the two Chinese fragments need to be contiguous or have a function word in-between while they need to be contiguous for the *inverted* case to hold.

Since the pieces have come to together, we follow the steps specified in Table 2 to learn ITG rules. Table 12 shows some of the grammar rules trained and associated estimations.

Table 12. Examples of grammar rules trained and their probabilities.

$L \to R_1\ R_2$	$\mathrm{P}\left(L \to \left[R_1\ R_2\right]\right)$	$\mathrm{P}\left(L \to \left\langle R_1\ R_2\right\rangle\right)$	$\mathrm{count}\left(L \to \left[R_1\ R_2\right]\right)$	$\mathrm{count}\left(L \to \left\langle R_1\ R_2\right\rangle\right)$
$S \to NP\ VP$	0.0107950	0.0009212	145,421	12,409
$VP \to VP\ NP$	0.0109561	0.0005481	147,591	7,383
$PP \to IN\ NP$	0.0031136	0.0007793	41,944	10,498
$VP \to VP\ VP$	0.0035528	0.0003922	47,860	5,283
$NP \to JJX\ NNX$	0.0108844	0.0006971	146,624	9,391
$NP \to ADJP\ NNX$	0.0148228	0.0008140	199,681	10,965

In Table 12 we notice that the adjective-noun structure has much more *straight* cases than *inverted*. In other words, adjectives modify nouns in much the same manner in English and Chinese. In general, the statistics suggests that Chinese, much like English, is SVO with only relatively small number of exceptional cases.

Another point worth mentioning is that the overwhelming predominance of *straight* over *inverted* is not observed in the rule of $PP \to IN\ NP$. For this grammar rule, the *straight* cases like “in August”, ”在 八月份” and the *inverted* cases such as “before midnight”, “午夜 前” are about the same order of magnitude. Consequently, it seems that there is no decisive preference of translation orientation for prepositional phrases.

4.2. Evaluation Metrics

We evaluated the trained ITG rules based on the performance of word alignment. We took the leaf nodes as word-level alignments and evaluate the proposed model in terms of agreement with human-annotated word alignments.

We used the metrics of alignment error rate (AER) proposed by Och and Ney (2000), in which the quality of a word alignment result $\mathbf{A} = \{(i, j)\}$, where i, j are positions of the sentence pair e, f respectively and $i, j \neq 0$, is evaluated using

$$precision = \frac{|\mathbf{A} \cap \mathbf{P}|}{|\mathbf{A}|}, \quad recall = \frac{|\mathbf{A} \cap \mathbf{S}|}{|\mathbf{S}|} \quad \text{and}$$

$$AER(\mathbf{S}, \mathbf{P}; \mathbf{A}) = 1 - \frac{|\mathbf{A} \cap \mathbf{S}| + |\mathbf{A} \cap \mathbf{P}|}{|\mathbf{A}| + |\mathbf{S}|} \quad ,$$

where $\mathbf{S}$ (sure) is the set which contains alignments that are not ambiguous and $\mathbf{P}$ (possible) is the set consisting of the alignments that might or might not exist $(\mathbf{S} \subseteq \mathbf{P})$. For that the human-annotated alignments may contain many-to-one and one-to-many relations. Furthermore, whether a word-level alignment is in $\mathbf{P}$ or $\mathbf{S}$ is determined by human experts who perform the annotation work.

4.3. Evaluation Result

For testing, we randomly selected 62 sentence pairs from the corpus of Hong Kong News. For the sake of time, we only selected sentence pairs in which the length of English and Chinese sentences does not exceed 15. From Table 8 and Table 9, we know the upper bound of 15 would cover approximately 40% of sentence pairs in HKN. We manual annotated the word alignment information in these bilingual sentences. The ratio of $|\mathbf{P}|$ and $|\mathbf{S}|$ of the test data is 1.2.

4.3.1 Baseline

We chose a freely-distributed word-aligning system, Giza++, as the baseline for evaluation. The adopted setting to run Giza++ is IBM model 4, the direction is from English to Chinese same as our model treating English as source language and the alignment units of Chinese are words not characters.

4.3.2 Word-level Evaluation

As preliminary evaluation, we examined whether syntactic consideration would lead to better word-level alignments. Figure 5 shows some alignments produced by the system and Giza++ and Table 13 displays evaluation results on alignments of the test data produced by both systems.

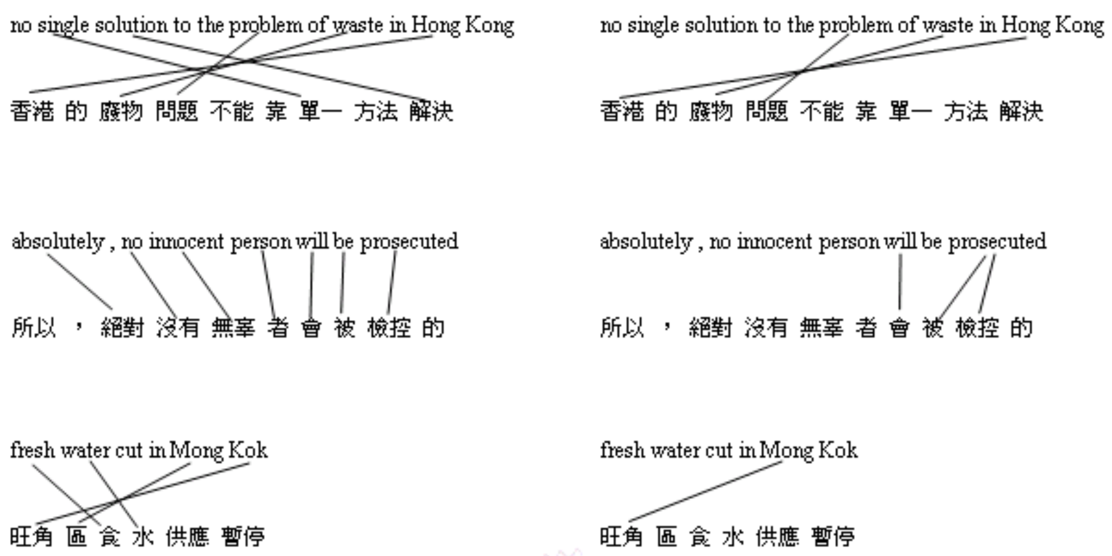

Figure 5. Alignments produced by our system (left) and Giza++ (right).

Table 13. Alignment results of the test data. Our system vs. Giza++.

	Recall	**Precision**	**AER**	**F-measure**
The proposed method	0.55	0.80	0.34	0.65
Giza++	0.37	0.87	0.48	0.52

Table 13 shows that although the precision is 87% for Giza++, the low recall leads to high alignment error rate and poor F-measure. However, our system with lower precision increased recall by 48.6%, which achieved a 29.2% alignment error reduction. From this experiment, we showed the proposed model with ITG rules allows for a wide range of ordering variations with a realistic position distortion penalty, which attributes to significantly better word alignment results.

Since the proposed model takes lexical and syntactic aspects of languages into consideration, the proposed method can be used to improve an existing word-aligning system that utilizes few linguistic information of languages. For that we evaluated the proposed method on top of the alignment results of Giza++, a freely-available state-of-the-art word alignment system. In other words, the **C** and **G** corpora are the same as the previous experiment but we adopted Giza++ as the word-aligning method in the training process.

Figure 6 shows some word alignment results produced by Giza++ with ITG and Giza++. Table 14 shows even better improvement than using the word alignment system along.

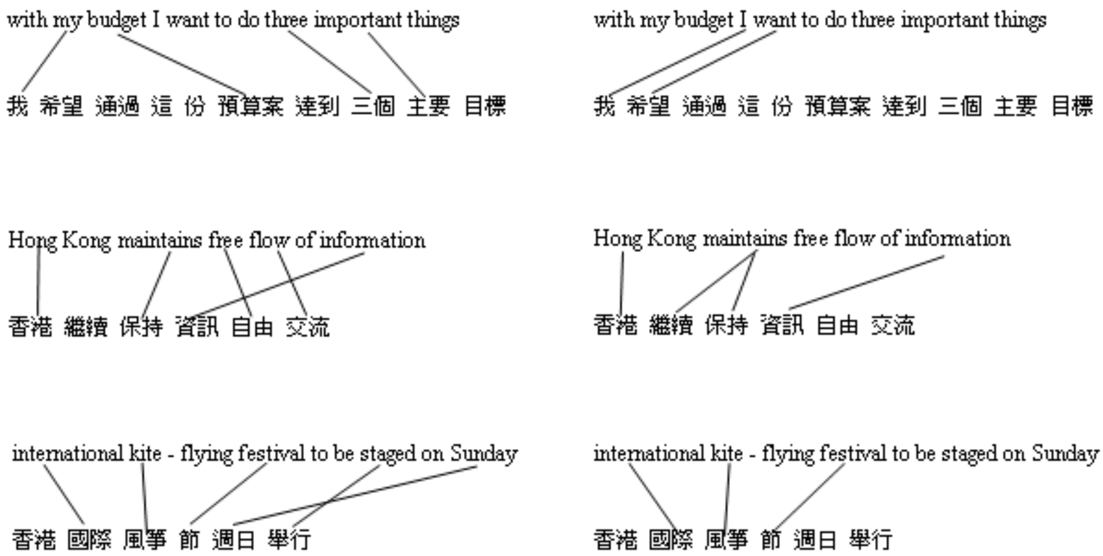

Figure 6. Alignments produced by Giza++ with ITG (left) and Giza++ (right).

Table 14. Alignment results of the test data. Giza++ with ITG vs. Giza++.

	Recall	**Precision**	**AER**	**F measure**
Giza++ with ITG	0.58	0.87	0.30	0.70
Giza++	0.37	0.87	0.48	0.52

The use of ITG results in significant improvement for recall and F-measure of Giza++ by 56.8% and 34.6% leading to substantial alignment error reduction (37.5%) while precision suffers only slightly (0.1%).

4.3.3 Phrase-level Evaluation

We further evaluated base phrases of the generated bilingual parse trees. We take into consideration the correctness of syntactic label and phrase alignment of a base phrase. Table 15 is how we rated a base phrase produced by our method concerning syntactic label and phrase alignment.

Table 15. Points of phrase-level evaluation.

syntactic label	**phrase alignment**	**point**
O	O	1.0
O	X	0.5
X	O	0.5
X	X	0.0

The first row in Table 15 means that if human judges assess the constituent label and alignment of the generated base phrase are both correct, it will be rated as *correct* (1 point). The second row means that if the syntactic label is correct but alignment is not quite right, human judges will rate the base phrase as *partially correct* (0.5 point). However, if the label is wrongly tagged but the phrase alignment is right, it will also be rated as *partially correct* (0.5 point). In the worse case, the label and alignment are not quite correct, 0 point is given to that base phrase.

The average score of the base phrases generated by Giza++ with ITG was 0.82, showing that our method produced satisfactory result in constituent label of base phrases and alignments in phrase level.

5. Conclusion and Future Work

Improvements of the proposed method and future researches have presented themselves along the way. Currently, we only focus on CFG with two right-hand-side constituents. Nonetheless, in linguistic sense, it is undesirable to divide the structure of $(NP\ CC\ NP)$ into $(NP\ CC)$ and (NP) or (NP) and $(CC\ NP)$ in that it is an indivisible syntactic-meaningful construct. Therefore, one of our future goals is to incorporate grammar rules with more constituents on the right hand side, such as $NP \to NP\ CC\ NP$, and their related probabilistic estimations into our model. Moreover, to make the structures of the bilingual parse trees more complete and rational, we would include a meaningful label for target-language words translated into no words in the source and grammar rules with the label in the future. It is also interesting to see how produced bilingual parse trees would influence the performance of the actual decoding process of machine translation and facilitate bilingual phrase extraction.

In conclusion, we have presented a robust method for learning ITG rules which specify the syntactic structures and relations of syntax of two languages involved. The proposed method exploits both lexical and syntax information to derive a structural model of the translation process. At runtime, a bottom-up CYK-styled implementation parses bilingual sentences simultaneously by exploiting trained ITG rules. Experiments show that our model consisting of grammar rules with linguistics-motivated labels and preferences of ordering counterparts in languages produces much more satisfying word alignment results compared with a state-of-the-art word-aligning system.

6. References

1. Andrew B. Clegg and Adrian Shepherd. 2005. "Evaluating and integrating Treebank parsers on a biomedical corpus." In *Association for Computational Linguistics Workshop on software 2005*.
2. Colin Cherry and Dekang Lin. 2003. A probability model to improve word alignment. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 88-95.
3. David Chiang. 2005. "A hierarchical phrase-based model for statistical machine translation." In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 263-270.
4. Yuan Ding and Martha Palmer. 2005. "Machine translation using probabilistic synchronous dependency insertion grammars." In *Proceedings of 43rd Annual Meetings of the ACL*, pages 541-548.
5. Wu Hua, Haifeng Wang, and Zhanyi Liu. 2005. "Alignment model adaptation for domain-specific word alignment." In *Proceedings of the 43rd Annual Meeting of the ACL,* pages 467-474.

6. I. Dan Melamed. 2003. "Multitext grammars and synchronous parsers." In Proceedings of the 2003 Meeting of the North American chapter of the Association for Computational Linguistics.
7. Franz Josef Och and Hermann Ney. 2000. "Improved statistical alignment models." In *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics (ACL-00)*, pages 440-447.
8. F Franz Josef Och, C. Tillmann, and H. Ney. 1999. "Improve alignment models for statistical machine translation." In 1999 *EMNLP*.
9. Kristina Toutanova, H. Tolga Ilhan and Christopher D. Manning. 2002. "Extentions to HMM-based statistical word alignment models." In *Proceedings of the Conference on Empirical Methods in Natural Processing Language*.
10. Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. "HMM-based word alignment in statistical translation." In *Proceedings of the 16th conference on Computational linguistics*, volume 2, pages 836-841
11. Wei Wang, Ming Zhou, Jin-Xia Huang, and Chang-Ning Huang. 2002. "Structure alignment using bilingual chunking." In *Proceedings of the 19th international conference on Computational linguistics*, volume 1, pages 1-7.
12. Wei Wang and Ming Zhou. "Improving word alignment models using structured monolingual corpora." In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing,* pages 198-205.
13. Ye-Yi Wang. 1998. "Grammar inference and statistical machine translation." Ph.D. thesis.
14. Dekai Wu. 1997. "Stochastic inversion transduction grammars and bilingual parsing of parallel corpora." *Computational Linguistics*, 23(3):377-403.
15. Kenji Yamada and Kevin Knight. 2001. "A syntax-based statistical translation model." In *Proceedings of the 39th Annual Conference of the Association for Computational Linguistics (ACL-01)*.
16. Hao Zhang and Daniel Gildea. 2004. "Syntax-based alignment: supervised or unsupervised?" In *Proceedings of the 20th International Conference on Computational Linguistics*.
17. Hao Zhang and Daniel Gildea. 2005. "Stochastic lexicalized inversion transduction grammar for alignment." In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 475-482.