Improved Sentence Alignment on Parallel Web Pages Using a Stochastic Tree Alignment Model

Lei Shi
Microsoft Research Asia
5F Sigma Center, 49 Zhichun Road, Beijing
100190, P. R. China
leishi@microsoft.com

Ming Zhou
Microsoft Research Asia
5F Sigma Center, 49 Zhichun Road, Beijing
100190, P. R. China
mingzhou@microsoft.com

Abstract

Parallel web pages are important source of training data for statistical machine translation. In this paper, we present a new approach to sentence alignment on parallel web pages. Parallel web pages tend to have parallel structures，and the structural correspondence can be indicative information for identifying parallel sentences. In our approach, the web page is represented as a tree, and a stochastic tree alignment model is used to exploit the structural correspondence for sentence alignment. Experiments show that this method significantly enhances alignment accuracy and robustness for parallel web pages which are much more diverse and noisy than standard parallel corpora such as "Hansard". With improved sentence alignment performance, web mining systems are able to acquire parallel sentences of higher quality from the web.

1 Introduction

Sentence-aligned parallel bilingual corpora have been essential resources for statistical machine translation (Brown et al. 1993), and many other multi-lingual natural language processing applications. The task of aligning parallel sentences has received considerable attention since the renaissance of data driven machine translation in late 1980s.

During the past decades, a number of methods have been proposed to address the sentence alignment problem. Although excellent performance was reported on clean corpora, they are less robust with presence of noise. A recent study by (Singh and Husain 2005) completed a systematic evaluation on different sentence aligners under various conditions. Their experiments showed that the performance of sentence aligners are sensitive to properties of the text, such as format complexity (presence of elements other than text), structural distance (a scale from literal to free translation), the amount of noise (text deletions or preprocessing errors) and typological distance between languages. Their performance varies on different type of texts and they all demonstrate marked performance degradation over noisy data. The results suggest that there is currently no universal solution to sentence alignment under all conditions, and different methods should be applied to different types of texts.

In this paper, we specifically address sentence alignment on parallel web pages. It has come to attention with the increasing trend of acquiring large-scale parallel data from the web. Currently, large-scale parallel data are not readily available for most language pairs and domains. But due to a sharply increasing number of bilingual web sites, web mining shows great promise as a solution to this knowledge bottleneck problem. Many systems (Ma 1999; Chen 2000; Yang 2002; Resnik 2003; Chen 2004) have been developed to discover parallel web pages, and sentence aligners are used to extract parallel sentences from the mined web corpora. Sentence alignment performance on parallel web pages, therefore, becomes an increasingly important issue for large-scale high-quality parallel data acquisition.

Compared with clean parallel corpora such as "Hansard" (Brown et al. 1993), which consists of

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 505–513,
Honolulu, October 2008.

French-English translations of political debates in the Canadian parliament, texts from the web are far more diverse and noisy. They are from many different domains and of various genres. Their translation may be non-literal or written in disparate language pairs. Noise is abundant with frequent insertions, deletions or non-translations. And there are many very short sentences of 1-3 words. Due to the characteristics of web corpora, direct application of conventional alignment methods without exploiting additional web document information acts as useful information to constrain the scope of search for parallel sentences.

The paper is organized as follows: In section 2, we briefly survey previous approaches to sentence alignment. In section 3, we present the stochastic tree alignment model, including parameter estimation and decoding. Then in section 4, we describe how to use the tree alignment model in sentence alignment. Benchmarks are shown in section 5, and the paper is concluded in section 6.

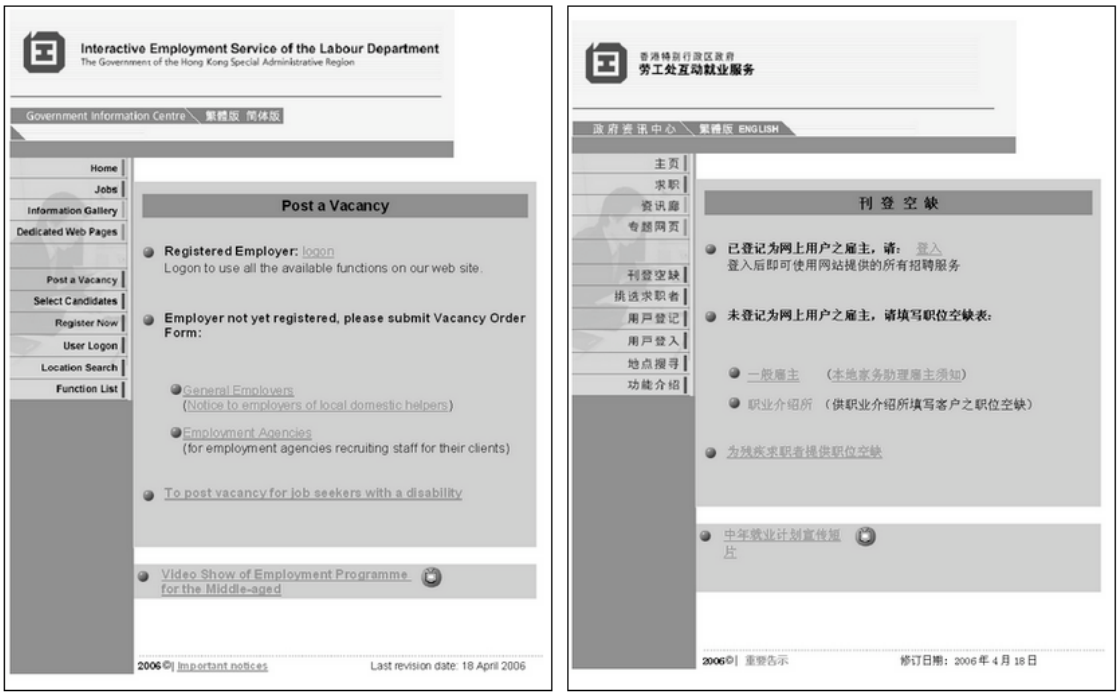

Figure 1. Example of parallel web pages

yields unsatisfactory alignment results.

Our approach to this problem is to make use of the structural parallelism between parallel web pages. Structural parallelism is the phenomenon, that when representing the same content in two different languages, authors have a very strong tendency to use the same document structure. As is shown in Figure 1, sentences located in similar position on both pages are more likely to be translations. Hence, correspondence in the web page structure is an informative indication of parallel sentences. In our approach, the web page is represented as a tree, and a stochastic tree alignment model is used to find the most probable alignment of the tree pair based on their structure and the texts in tree nodes. The tree alignment then

2 Sentence Alignment Models

Sentence alignment methods can be categorized into three major categories: the length-based, lexicon-based and hybrid method which combines the length-based model and lexicon-based model as complement to each other.

The length model was based on the intuition that the length of a translated sentence is likely to be similar to that of the source sentence. (Brown et. at. 1991) used word count as the sentence length, whereas (Gale and Church 1993) used character count. Dynamic programming is used to search the optimal sentence alignment. Both algorithms have achieved remarkably good results for language pairs like English-French and English-German

with an error rate of 4% on average. But they are not robust with respect to non-literal translations, deletions and disparate language pairs.

Unlike the length-based model, which totally ignores word identity, lexicon-based methods use lexical information to align parallel sentences. Kay's (Kay and Roscheisen 1993) approach is based on the idea that words that are translations of each other will have similar distribution in source and target texts. By adopting the IBM model 1, (Chen 1993) used word translation probabilities, which he showed gives better accuracy than the sentence length based method. Melamed (Melamed 1996) rather used word correspondence from a different perspective as geometric correspondence for sentence alignment.

The hybrid method combines the length model with the lexical method. (Simard and Plamondon 1996) used a two-pass approach, where the first pass performs length-based alignment at the character level as in (Gale and Church 1993) and the second pass uses IBM Model 1, following (Chen 1993). Moore's (Moore 2002) approach is similar to Simard's. The difference is that Moore used the data obtained in the first pass to train the IBM model in the second pass, so that his approach does not require a priori knowledge about the language pair. Instead of using a two-pass approach, (Zhao and Vogel 2002) combines the length model and the IBM model 1 in a unified framework under a maximum likelihood criterion. To make it more robust on noisy text, they developed a background model to handle text deletions.

To further improve sentence alignment accuracy and robustness, methods that make use of additional language or corpus specific information were developed. In Brown and Church's length-based aligner, they assume prior alignment on some corpus specific anchor points to constrain and keep the Viterbi search on track. (Wu 1994) implemented a length-based model for Chinese-English with language specific lexical clues to improve accuracy. (Simard et al. 1992) used cognates, which only exists in closely related language pairs. (Chuang and Yeh 2005) exploited the statistically ordered matching of punctuation marks in two languages to achieve high accuracy sentence alignment. In their web parallel data mining system, (Chen and Nie 2000) used HTML tags in the same way as cognates in (Simard et al. 1992) for aligning Chinese-English parallel sentences. Tree based alignment models have been successfully applied in machine translation (Wu 1997, Yamada & Knight 2001, Gildea 2003).

3 The Stochastic Tree Alignment Model

The structure of the HTML document is recursive, with HTML markup tags embedded within other markup tags. While converting an HTML document into the tree representation, such hierarchical order is maintained. Each node of the tree is labeled with their corresponding HTML tag (*e.g. body, title, img etc.*) and in labeling tree nodes, only markup tags are used and attribute value pairs are dropped. Among all markup tags in the HTML file, those of our most interest are tags containing content text, which is what we want to align. These tags are those surrounding a text chunk or have the attribute of "ALT". Comments, scripts and style specifications are not regarded as content text and hence are eliminated. Figure 2 illustrates the tree representation of an example HTML document.

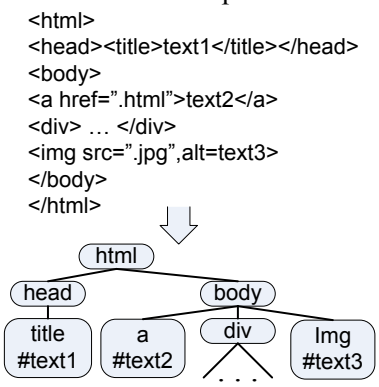

Figure. 2 An example HTML document and its tree representation

3.1 Tree Alignment Model

Given two trees, the tree alignment is the non-directional alignments of their nodes. A node in one tree can be aligned with at most one node in the other tree. It is valid for a node to be aligned with nothing (*NULL*) and such case is regarded as node deletion in tree alignment. To comply with the tree hierarchical structure, we constrain that the alignments keep the tree hierarchy invariant *i.e.* if node *A* is aligned with node *B*, then the children of *A* are either deleted or aligned with the children of *B*. Besides, to simplify the model training and decoding, the tree alignment model also keeps the

sequential order invariant, *i.e.* if node A is aligned with node B, then the left sibling nodes of A cannot be aligned with the right sibling nodes of B.

The stochastic tree alignment model assigns probabilities to tree alignments, based on the particular configuration of the alignment and model parameters. Then, the decoder is able to find the most probable (optimal) alignment of two trees. To facilitate the presentation of the tree alignment model, the following symbols are introduced: given a HTML document D, T^D denotes the corresponding tree; N_i^D denotes the i[th] node of T^D, and T_i^D denotes the sub-tree rooted at N_i^D. Especially, T_1^D is the root of the tree T^D. $T_{[i,j]}^D$ denotes the forest consisting of the sub-trees rooted at sibling nodes from T_i^D to T_j^D. $N_i^D.t$ denotes the text in the node N_i^D, and $N_i^D.t$ denotes the label (*i.e.* HTML tag) of the node N_i^D; $N_i^D.C_i$ denotes the j[th] child of the node N_i^D; $N_i^D.C_{[m,n]}$ denotes the consecutive sequence of N_i^D's children nodes from $N_i^D.C_m$ to $N_i^D.C_n$; the sub-tree rooted at $N_i^D.C_i$ is represented as $N_i^D.CT_i$ and the forest of the sub-trees rooted at N_i^D's children is represented as $N_i^D.CF$. To accommodate node deletion, $NULL$ is introduced to denote the empty node. Finally, the tree alignment is referred as A.

Given two HTML documents F (in French) and E (in English) represented as trees T^F and T^E, the tree alignment task is defined as finding the alignment A that maximizes the conditional probability $\Pr(A|T^F,T^E)$. Based on the Bayes' Rule, $\Pr(A|T^F,T^E) \propto \Pr(T^F,T^E|A)\Pr(A)$, where $\Pr(T^F,T^E|A)$ is the probability of synchronously generating T^F and T^E given the alignment A, and $\Pr(A)$ is the prior knowledge of the tree alignment. To simplify computation, we assume a uniform prior probability $\Pr(A)$. Hence, the tree alignment task is to find the A that maximizes the synchronous probability $\Pr(T^F,T^E|A)$.

Based on the hierarchical structure of the tree, in order to facilitate the presentation and computation of the tree alignment probabilistic model, the following alignment probabilities are defined in a hierarchically recursive manner:

$\Pr(T_m^F,T_i^E|A)$: The probability of synchronously generating sub-tree pair $\{T_m^F,T_i^E\}$ given the alignment A;

$\Pr(N_m^F,N_i^E|A)$: The probability of synchronously generating node pair $\{N_m^F,N_i^E\}$;

$\Pr(T_{[m,n]}^F,T_{[i,j]}^E|A)$: The probability of synchronously generating forest pairs $\{T_{[m,n]}^F,T_{[i,j]}^E\}$ given the alignment A.

From the definition, the tree pair generative probability $\Pr(T^F,T^E|A)$ equals to the root sub-tree pair generative probability $\Pr(T_1^F,T_1^E|A)$. The alignment of the sub-tree pair T_j^F and T_i^E may have the following configurations, based on which the tree pair generative probability $\Pr(T_j^F,T_i^E|A)$ can be calculated:

(1) If N_m^F is aligned with N_i^E, and the children of N_m^F are aligned with children of N_i^E (as is shown in Fig. 3a), then we have

$$\Pr(T_m^F,T_i^E|A) = \Pr(N_m^F,N_i^E)\Pr(N_m^F.CF,N_i^E.CF|A)$$

(2) If N_m^F is deleted, and the children of N_m^F is aligned with N_i^E (as shown in Fig. 3b), then we have

$$\Pr(T_m^F,T_i^E|A) = \Pr(N_m^F|NULL)\Pr(N_m^F.CF,T_i^E|A)$$

(3) If N_i^E is deleted, and N_m^F is aligned with children of N_i^E (as shown in Fig. 3c), then we have

$$\Pr(T_m^F,T_i^E|A) = \Pr(T_m^F,N_i^E.CF|A)\Pr(N_i^E|NULL)$$

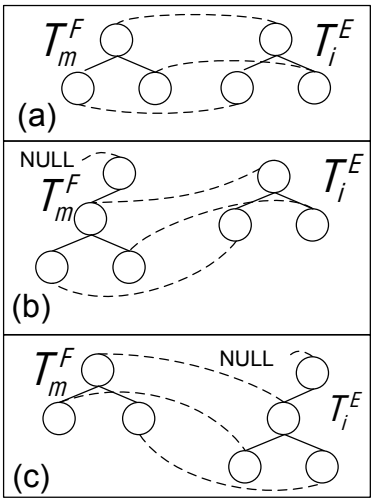

Figure. 3

The above equations involve forest pair generative probabilities. The alignment of the forest $T_{[m,n]}^F$ and $T_{[i,j]}^E$ may have the following configurations, based on which their forest pair generative probability $\Pr(T_{[m,n]}^F,T_{[i,j]}^E|A)$ can be calculated:

(4) If T_m^F is aligned with T_i^E, and $T_{[m+1,n]}^F$ is aligned with $T_{[i+1,j]}^E$ (as is shown in Fig. 4a), then

$$\Pr(T_{[m,n]}^F, T_{[i,j]}^E|A) = \Pr(T_m^F, T_i^E|A)\Pr(T_{[m+1,n]}^F, T_{[i+1,j]}^E|A)$$

(5) If N_m^F is deleted, and the forest rooted at N_m^F's children $N_m^F.CF$ is combined with $T_{[m+1,n]}^F$ for alignment with $T_{[i,j]}^E$, then

$$\Pr(T_{[m,n]}^F, T_{[i,j]}^E|A) = \Pr(N_m^F|NULL)\Pr(N_m^F.CF\ T_{[m+1,n]}^F, T_{[i,j]}^E|A)$$

(6) If N_i^E is deleted, and the forest rooted at N_i^E's children $N_i^E.CF$ is combined with $T_{[i,j]}^E$ for alignment with $T_{[m,n]}^F$, then

$$\Pr(T_{[m,n]}^F, T_{[i,j]}^E|A) = \Pr(N_i^E|NULL)\Pr(T_{[m,n]}^F, N_m^F.CF\ T_{[i+1,j]}^E|A)$$

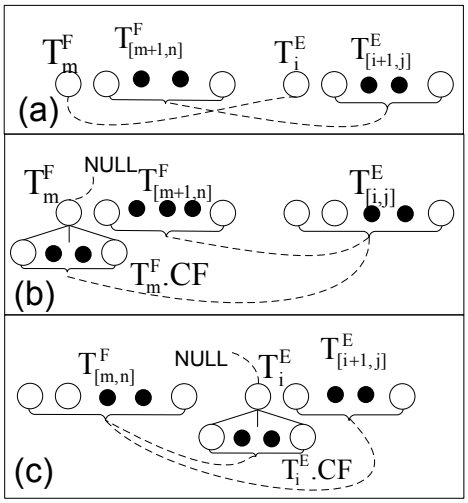

Figure. 4

Finally, the node pair probability is modeled as $\Pr(N_m^F, N_i^E) = \Pr(N_m^F.t, N_i^E.t)\Pr(N_m^F.l, N_i^E.l)$, where $\Pr(N_m^F.t, N_i^E.t)$ is the generative probability of the translationally equivalent text chunks in N_m^F and N_i^E, and $\Pr(N_m^F.l, N_i^E.l)$ is their HTML tag pair probability. The text chunk generative probability $\Pr(N_m^F.t, N_i^E.t)$ can be modeled in a variety of ways. The conventional length-based, lexicon-based or hybrid methods used for sentence alignment can be applied here. In the next subsection, we focus on how to estimate the tag pair probability $\Pr(N_m^F.l, N_i^E.l)$ from a set of parallel web pages. We expect pairs of the same or similar HTML tags to have high probabilities and the probabilities for pairs of disparate tags to be low.

3.2 Parameter Estimation Using Expectation-Maximization

One way to estimate the tag pair generative probability $\Pr(l, l')$ is to manually align nodes between parallel trees, and use the manually aligned trees as the training data for maximum likelihood estimation. However, this is a time-consuming and error-prone procedure. Instead, the Expectation Maximization (EM) (Dempster, Laird and Rubin 1977) algorithm is used to estimate the parameters $\Pr(l, l')$ on 5615 manually verified parallel web page pairs from 45 different bilingual web sites. The parameter estimation proceeds as follows:

1. Start with initial parameter values.
2. Expectation: estimate $\overline{\text{count}}(l, l')$ which is the expectation of aligning tag l with l'.
3. Maximization: update the parameters based to maximum likelihood estimation

$$\Pr(l, l') = \frac{\overline{count}(l, l')}{\sum_{l} \overline{count}(l, l')} \quad \text{and}$$

$$\Pr(l|NULL) = \frac{\overline{count}(NULL, l) + \overline{count}(l, NULL)}{\sum_{l}[\overline{count}(NULL, l') + \overline{count}(l', NULL)]}$$

4. Repeat step 2 and 3 until the parameters stabilize

In step 2, $\overline{\text{count}}(l, l')$ is the expected count of l being aligned with l' in the training corpus. By definition, $\overline{\text{count}}(l, l')$ is calculated as

$$\overline{\text{count}}(l, l') = \sum_{A} \Pr(A|T^F, T^E)\text{count}(l, l')$$

where $\text{count}(l, l')$ is the number of occurrence of l being aligned with l' in the tree alignment A.

To efficiently compute $\overline{\text{count}}(l, l')$ without enumerating the exponential number of A's in the above equation, we extended the inside-outside algorithm presented in (Lari and Young, 1990). The inside probability $\alpha(N_j^F, N_i^E)$ is defined as the

probability of generating sub-tree pair $\{T_j^F, T_i^E\}$ when N_i^E is aligned with N_j^F. It is estimated as:

$$\alpha(N_m^F, N_i^E) = \Pr(N_m^F, N_i^E)\alpha(N_m^F.CF, N_i^E.CF)$$

where $\alpha(N_m^F.CF, N_i^E.CF)$ is the inside probability for the forest pair $(N_m^F.CF, N_i^E.CF)$

$$\alpha(N_m^F.CF, N_i^F.CF) = \sum_A \Pr(N_m^F.CF, N_i^F.CF|A).$$

The inside probability can be estimated recursively according to the various alignment configurations presented in Figure 3 and Figure 4. The outside probability $\beta(N_j^F, N_i^E)$ is defined as the probability of generating the part of T^E and T^F excluding the sub-trees T_j^F and T_i^E, when N_i^E is aligned with N_j^F. It is estimated as:

$$\begin{aligned}\beta(N_m^F, N_i^E) = \sum_{p,q}[&\beta(a_{m,q}^F, a_{i,p}^E)\\ &\times\alpha(a_{m,q}^F.LCF(N_m^F), a_{i,p}^E.LCF(N_i^E))\\ &\times\alpha(a_{m,q}^F.RCF(N_m^F), a_{i,p}^E.RCF(N_i^E))\\ &\times\prod_{k<q}\Pr(a_{m,k}^F \mid NULL)\prod_{k<p}\Pr(a_{i,k}^E \mid NULL)]\end{aligned}$$

where $a_{m,q}^F$ is the q[th] ancestor of N_m^F, and $a_{i,q}^E$ is the p[th] ancestor of N_i^E. $a_{m,k}^F(k<q)$ is an ancestor of N_m^F and a decedent of $a_{m,q}^F$. Similarly $a_{i,k}^E(k<p$ is an ancestor of *NiE*, and a decedent of *ai,pE*. *a.LCF(N)* is the forest rooted at *a* and to the left of *N*, and *a.RCF(N). a.RCF(N)* is the forest rooted as *a* and to the right of *N*. Once inside and outside probabilities are computed, the expected counts can be calculated as

$$\overline{count}(l, l') = \sum_{\{T^F, T^E\}} \sum_{\substack{N_i^E.l=l' \\ N_m^F.l=l}} \frac{\alpha(N_m^F, N_i^E)\beta(N_m^F, N_i^E)}{\Pr(T^F, T^E)}$$

where $\Pr(T^F, T^E)$ is the generative probability of the tree pair $\{T^F, T^E\}$ over all possible alignment configurations. $\Pr(T^F, T^E)$ can be estimated using dynamic programming techniques that will be presented in the next sub-section. Furthermore, the expected count of tag deletion is estimated as:

$$\overline{\text{count}}(l, NULL) = \sum_i \text{count}(l, l') - \sum_{i \neq NULL} \overline{\text{count}}(l, l')$$

$$\overline{\text{count}}(NULL, l) = \sum_i \text{count}(l', l) - \sum_{i \neq NULL} \overline{\text{count}}(l', l)$$

3.3 Dynamic Programming for Decoding

An intuitive way to find the optimal tree alignment is to enumerate all alignments and pick the one with the highest probability. But it is intractable since the total number of alignments is exponential. Based on the observation that if two trees are optimally aligned, the alignment of their sub-trees must also be optimal, dynamic programming can be applied to find the optimal tree alignment using that of the sub-trees in a bottom-up manner. That is we first compute the optimal alignment probabilities of small trees and use them to compute that of the bigger tree by trying different alignment configurations. This procedure is recursive until the optimal alignment probability of the whole tree is obtained. The following is the pseudo-code of the bottom-up decoding algorithm:

```
for i=|T^E| to 1 (bottom-up) {
  for j=|T^F| to 1 (bottom-up) {
    Select and store optimal alignments of their children forests T_m^F.CF and T_i^E.CF by testing configurations 4-6;
    Select and store the optimal alignment of the sub-tree pair T_m^F and T_i^E by testing configurations 1-3;
    Store the optimal configuration}}
```

where $|T^F|$ and $|T^E|$ are the number of nodes in T^F and T^E. The decoding algorithm finds the optimal alignment and its probability for every sub-trees and forests. By replacing the selection operation with summing probabilities of all configurations, the sub-tree pair generative probability $\Pr(T^F, T^E)$ can be calculated along the way. The worst-case time complexity of the algorithm is $O(|T^F||T^E|(degr(T^F)+degr(T^E))^2)$, where the degree of a tree is defined as the largest degree of its nodes.

4 Sentence Alignment with Tree Alignment Model

Since the tree alignment model aligns parallel web pages at the tree node level instead of the sentence level, we integrate the tree alignment model with the sentence alignment model in a cascaded mode, in which the whole sentence alignment process is divided into two steps. In the first step, the tree alignment decoder finds the optimal alignment of the two trees. Nodes having texts should be aligned with nodes containing their translations. Then in the second step, the conventional sentence aligner is used to align sentences within text chunks in the

aligned nodes. In this step, various sentence alignment models can be applied, including the length-based model, the lexicon-based model and the hybrid model. Language or corpus specific information may also be used to further improve sentence alignment accuracy. The tree alignment acts as constraints that confine the scope of the search of sentence aligners.

5 Evaluation

To evaluate the effectiveness of exploiting web page document structure with the tree alignment model for improving sentence alignment accuracy, we compared the performance of three types of sentence alignment methods on parallel web pages.

The first type is to simply discard web page layout information. Web pages are converted to plain texts, and HTML tags are removed prior to performing sentence alignment. The second type is the baseline method of using web page document information. Instead of exploiting full HTML document structure, it follows Chen's approach (Chen and Nie 2000) which uses HTML tags in the same way as cognates used in (Simard et al. 1992). The third type is the combination of tree alignment model and conventional sentence models.

computer and literature. By manual annotation, 9,824 parallel sentence pairs are found. All sentence aligners run through the test parallel web pages, and each extracts a set of sentence pairs that it regards as parallel. The output pairs are matched with the annotated parallel sentences from the test corpus. Only exact matches of the sentence pairs are counted as correct.

Our evaluation metrics are precision (P), recall (R) and F-measure (F) defined as:

$$P = \frac{\#\text{ of correctly aligned sentence pairs}}{\#\text{ of total output pairs}}$$

$$R = \frac{\#\text{ of correctly aligned sentence pairs}}{\#\text{ of true parallel pairs}}$$

$$F = \frac{2 * P * R}{P + R}$$

Based on the results in table 1, we can see that both Type 2 and Type 3 aligners outperform conventional sentence alignment models. Leveraging HTML document information can enhance sentence alignment quality. Especially, by using the tree alignment model, Type 3 aligners achieve a significant increase of around 7% on both precision and recall. Compared with the tree alignment model, the improvement by the Type 2 aligners is marginal. A reason for this is that the tree alignment model not only exploits HTML tag similarities as in the Type 2 method, but also takes into account location of texts. In the tree alignment model, texts at similar locations in the tree hierarchical structure are more probable to be translations than those in disparate locations, even though they all have the same tag.

	Length			Lexicon			Hybrid		
	P	R	F	P	R	F	P	R	F
Type I	85.6%	72.8%	78.7%	83.1%	75.2%	78.9%	87.3%	76.4%	81.5%
Type II	86.3%	74.8%	80.1%	85.7%	77.0%	81.1%	88.1%	78.6%	83.1%
Type III	93.2%	79.3%	85.7%	92.9%	80.4%	86.2%	94.3%	83.1%	88.3%

Table 1. Performance comparison between different types of sentence alignment methods

Each type of the web page sentence aligner makes use of three conventional sentence alignment models, one is the length based model following (Brown 1991), one is the lexicon based model following (Chen 1993), and the other one is the hybrid model presented in (Zhao 2002). To be fair in performance comparisons, the text generative probability $\Pr(N^F.t, N^E.t)$ in tree node alignment is modeled in accordance with that in the sentence alignment model. All these sentence aligners are implemented to handle sentence bead types of "1-0", "0-1","1-1", "1-2","1-3","2-1" and "3-1".

The test corpus is 150 parallel web page pairs randomly drawn from 20 Chinese-English bilingual web sites on topics related to politics, sports,

We also evaluate the performance of the tree aligner. Since sentence alignment is performed within the text chunks of aligned nodes, tree alignment accuracy is very important for correct sentence alignment. We measure the alignment

accuracy on all nodes as well as that specifically on text nodes on the test corpus. The evaluation result is shown in table 2.

	total	correct	accuracy
all node alignment	18492	17966	97.2%
text node alignment	3646	3577	98.1%

Table 2. Tree Alignment Metrics

Benchmarks in Table 2 show that the tree alignment model yields very reliable results with high accuracy in aligning both text nodes and non-text nodes. After an analysis on text node alignment errors, we find that 79.7% of them have texts of very short length (no more than 4 words), which may not contain sufficient information to be identified as parallel.

6 Conclusions

In this paper, we present a new approach to sentence alignment on parallel web pages. Due to the diversity and noisy nature of web corpora, a stochastic tree alignment model is employed to exploit document structure in parallel web pages as useful information for identifying parallel sentences. The tree alignment model can be combined with various conventional sentence alignment models to extract parallel sentences from parallel web pages. Experimental results show that exploiting structural parallelism inherent in parallel web pages provides superior alignment performance over conventional sentence alignment methods and significant improvement (around 7% in both precision and recall) is achieved by using the stochastic tree alignment model. With improved sentence alignment performance, web parallel data mining systems are able to acquire parallel sentences of higher quality and quantity from the web.

References:

Brown, P. F., J. C. Lai and R. L. Mercer. 1991. *Aligning Sentences in Parallel Corpora.* Proceedings of ACL 1991.

Brown, P. E., S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer. 1993. *The Mathematics of Statistical Machine Translation: Parameter Estimation*, Computational Linguistics V19(2), 1993

Chen Jisong., Chau R. and C.-H. Yeh. 2004. *Discovering Parallel Text from the World Wide Web*, Proceedings of the second workshop on Australasian Information Security, Data Mining and Web Intelligence, and Software Internationalization.

Chen Jiang and Nie Jianyun. 2000. *Automatic construction of parallel English-Chinese corpus for cross-language information retrieval.* Proceedings of the sixth conference on applied natural language processing

Chen Stanley. 1993. *Aligning Sentences in Bilingual Corpora Using Lexical Information.* Proceedings of ACL 1993

Chuang T.C. and Yeh.K.C. 2005. *Aligning Parallel Bilingual Corpora Statistically with Punctuation Criteria.* Computational Linguistics and Chinese Language Processing. Vol. 10, 2005, pp. 95-122

Dempster, A., Laird, N., and Rubin, D. 1977. *Maximum likelihood from incomplete data via the EM algorithm.* Journal of the Royal Statistical Society, Series B, 39(1):1–38.

Gale W. A. and K. Church. 1993. *A Program for Aligning Sentences in Parallel Corpora*, Computational Linguistics, 19(1):75—102

Gildea. D. 2003. *Loosely Tree-Based Alignment for Machine Translation.* In Proceedings of ACL 2003

Kay Martin and Roscheisen Martin 1993. *Text Translation Alignment.* Computational Linguistics 19(1):121--142.

Lari K. and S. J. Young. 1990. *The estimation of stochastic context free grammars using the Inside-Outside algorithm,* Computer Speech and Language, 4:35—56

Ma, Xiaoyi and M. Liberman. 1999. *Bits: A Method for Bilingual Text Search over the Web.* Proceedings of Machine Translation Summit VII.

Melamed. I. Dan. 1996. *A Geometric Approach to Mapping Bitext Correspondence.* Proceedings of EMNLP 96

Moore Robert. C. 2002. *Fast and Accurate Sentence Alignment of Bilingual Corpora.* Proceedings of 5th Conference of the Association for Machine Translation in the Americas, pp. 135-244

Resnik, P. and N.A. Smith. 2003. *The Web as a Parallel Corpus.*Computational Linguistics, 29(3)

Simard, M. and Plamondon, P. 1996 *Bilingual Sentence Alignment: Balancing Robustness and Accuracy.* Proceedings of AMTA-96, Canada.

Simard, M., Foster, G. and Isabelle, P. 1992, *Using Cognates to Align Sentences in Bilingual Corpora.* Proceedings of the Fourth International Conference

on Theoretical and Methodological Issues in Machine translation (TMI92)

Singh, A. K. and Husain, S. (2005). *Comparison, selection and use of sentence alignment algorithms for new language pairs.* Proceedings of the ACL Workshop on Building and Using Parallel Texts.

Wu. Dekai. 1994. *Aligning a parallel English-Chinese corpus statistically with lexical criterias.* Proceedings of ACL 1994.

Wu. Dekai. "*Stochastic Inversion Transduction Grammar and Bilingual Parsing of Parallel Corpora*" Computational Linguistics, 23(3):374(1997)

Yamada H. and Knight K. 2001 *A Syntax based statistical translation model*. In Proceedings of ACL-01

Yang C. C., and Li K. W., *Mining English/Chinese Parallel Documents from the World Wide Web,* Proceedings of the International World Wide Web Conference, Honolulu, Hawaii, 2002.

Zhao Bin. and Stephan. Vogel. 2002. *Adaptive Parallel Sentences Mining From Web Bilingual News Collection.* 2002 IEEE International Conference on Data Mining. 745-748