Abstract - In this paper a new algorithm called Multi-Layer Filtering (MLF) is proposed for extracting bilingual alignment chunks automatically from a Chinese-English parallel corpus. Multiple layers are used to extract bilingual chunks according to different features of chunks in the bilingual corpus. And the alignment chunks are one-to-one corresponding with each other. The chunking and alignment algorithm doesn’t rely on the information from tagging, parsing, syntax analyzing or segmenting for Chinese corpus as most conventional algorithms do. Preliminary experimental results show that the algorithm achieves a good performance in chunking and alignment. Moreover, the translations generated by this algorithm are much better than the results generated by the baseline (word-based statistical machine translation).

Keywords: Alignment, chunking, multi-layer filtering, statistical machine translation.

1 Introduction

As is well known, with the rapid progress of internet technology, more and more machine-readable parallel bilingual corpora are available, which made statistical machine translation (SMT) become more and more important for its particular merits. The biggest advantages of the statistical methods are their trainability, coverage and robustness. They can automatically learn useful parameters and knowledge by employing well-studied algorithms and achieve good coverage if the training data are sufficiently representativelt.

The great majority of SMT Systems employ word-based alignment models based on the five word-based statistical models proposed by IBM in 1993 [1]. Subsequently several SMTS and various improved statistical methods have been proposed [2, 3, 4]. However, such systems still suffer from poor performance. One of the key reasons is that the word alignment models fundamentally rely on word-level translation. It is not appropriate when used in the language pairs which have great difference in structures such as Chinese and English.

Aiming to overcome the shortcomings of word-based methods, some alignment algorithms based on phrases, chunks or structures have been proposed [5, 6, 7, 8]. These modifications are advantageous and lead to more fluent translations since chunk-based translations capture local reordering phenomena. However, until now, most phrase or chunk alignment algorithms have been based on complex syntax information e.g. by incorporating parsing technology with crossing constraints or have been narrowly focused on certain special kinds of phrases. These methods have proven to yield poor performance when dealing with long sentences. Further, the methods heavily depend on the performance of associated tools, such as parsers, part-of-speech taggers, word segmenters for Chinese and so on.

In order to address these shortcomings effectively, we propose here a new algorithm called multi-layer filtering (MLF) for automatically aligning bilingual chunks.

This paper is organized as follows. In Section 2, our chunking and alignment algorithm is described in detail. Section 3 presents a series of experiments about bilingual chunk alignment in Chinese-English parallel text. Moreover, the chunk-to-chunk translations generated by this algorithm are compared with the baseline word alignment. Finally, Section 4 gives our conclusion of this new algorithm.

2 The multi-layer filtering algorithm

2.1 Overview

The purpose of the MLF algorithm is to discover one-to-one (including one-to-one, one-to-NULL and NULL-to-one three patterns) pairs of bilingual chunks in the untagged well-formed bilingual sentence pairs.

Our procedures can be summarized as follows: First, the most frequent monolingual chunks are filtered from the Chinese-English parallel texts. This technique allows us to obtain more accurate monolingual chunks and at the
same time helpfully makes long sentences shorter; Second, because of this shortening, the clustering algorithm which is used to cluster the similar words or structures will achieve better performance; Third, sequences of fragments which remain after clustering are simply combined into chunks which can participate in the alignment process. One-word fragments remaining in sentences are treated likewise; Finally, in order to guarantee that one Chinese chunk will correspond with one English chunk, only the best Chinese chunks (those with the highest co-occurrences with English chunks) are retained for use. This step seems justifiable, since translation output quality will not be seriously affected because most of these chunks aligned to one having the same or similar meaning. In the filtering steps, information concerning frequency, n-gram statistics and mutual information is employed in order to extract bilingual chunks. In the rest of this section, the algorithm will be described in detail.

2.2 Filtering the most frequent bilingual chunks

On the assumption that the most co-occurrent word lists may be a potential chunk, so these word lists are first filtered as initial monolingual chunks. The first filtering step proceeds as follows:

(1) As shown in formula (1), $D_k$ denotes the degree of cohesion of a chunk whose length is $k$. To some extent, the cohesion degree of a word lists reflects the probability of that word lists, so this measure is used to determine if a word lists is a plausible chunk. $D(w_1, w_2)$ (Here $k = 2$) is first used to compute the cohesion degree between two adjacent words in sentences, where,

$$D_k = D(w_1, w_2, \ldots, w_k) = (1 - \beta) \times MI(w_1, w_2, \ldots, w_k) + \beta \times P(w_1, w_2, \ldots, w_k)$$

$$MI(w_1, w_2, \ldots, w_k) = P(w_1, w_2, \ldots, w_k) / \log \left( \frac{P(w_1)P(w_2)\ldots P(w_k)}{P(w_1, w_2, \ldots, w_k)} \right)$$

Here $MI(w_1, w_2, \ldots, w_k)$ denotes the mutual information of the sequential words $w_1, w_2, \ldots, w_k$; $P(w_1, w_2, \ldots, w_k)$ denotes the probability of the sequential words $w_1, w_2, \ldots, w_k$; and $\beta$ is a coefficient between 0 and 1.

(2) After computing all the cohesion degrees between any two adjacent words in all sentences, tag the lowest $\mu$ values as anchor points within the sentences. Scan the sentences from the anchor points forward and backward in steps from 4 to 1 and keep the most frequent initial chunks in each step (where the maximum length of both Chinese and English chunks are 4, and $n$ is determined by formula (3)). The maximum length is defined as 4 because even in a very large training corpus chunks with length 5 are too infrequent. Moreover, the chunks to be obtained should conform to the following principles:

$$n = \int\left\lceil \frac{\text{length of a sentence}}{\text{the maximum length of a chunk}} \right\rceil$$

$$\mu = D_k / D_{k+1}$$

$$\nu = D_k / D_{k-1}$$

$$D_k = D_k \times \frac{\max(D_k)}{\max(D_2)}$$

Here $n$ is the number of segmenting the bilingual sentences; $\int\lceil x \rceil$ is the maximum integer of the result of the division operation; $D_k$ denotes the cohesion degree of the $i^{th}$ chunk whose length is $k$ in one sentence; $D'_k$ is the normalized cohesion degree for it is unfair to compare $D_k$ with different lengths in the same scale; $\mu$ and $\nu$ are both thresholds. By modifying them, we can obtain a different number of initial monolingual chunks. Moreover, on the basis of formulas (4) and (5), the length of a chunk is determined according to the maximum matching principle and these chunks will not be overlapped with each other. That is: if $\mu$ falls into the specified range, the length of the initial chunk should be $k$; and if $\nu$ falls into the specified range, the length of the chunk should be extended to $(k + 1)$.

Below, an example is given to explain the first filtering process in detail (See Figure 1).

![Figure 1. Cohesion degree between two adjacent words](image)

Here, “||” denotes the cohesion degree value between two adjacent words, and all cohesion degree values are shown on the scale of $10^{-3}$.

Based on the filtering steps (1) and (2), the entire set of initial monolingual chunks is found. In the Example above, 24 and 21 monolingual chunks are found in the Chinese and English sentence respectively. Excluding single words of length 1, the initial monolingual chunks and their cohesion degree are given as follows. (See Table 1 for Chinese and Table 2 for English.)

As shown in table1, we obtain such initial monolingual Chinese chunks (here $D_k > 1.0$) as: “什么样的”，“的房间”，“预定”，“什么”，“的房间”，“房间”。 According to the maximum matching principle, the Chinese sentence can be segmented as:

你 || 想 || 预定 || 什么样的 || 房间
Here, “&” is used as a conjunction symbol to combine individual words into chunks. Note that the initial chunks “的房间”, “什么样” and “的房间” are discarded even though they have large cohesion values. According to formula (4), we can determine that the division result of \( D(\text{什么样的}) \) and \( D(\text{什么样}) \) is greater than the division result of \( D(\text{的房间}) \) and \( D(\text{房间}) \) which means that the word “的” is more closely associated with “什么样的”, so “什么样的”的房间 is ultimately chosen as a chunk. Then according to the principle of that one chunk will not be overlapped with each other, so “的房间”, “什么样” and “的房间” are discarded.

2.3 Clustering the similar words and filtering the most frequent structures

Observing the corpus after the first filtering step, we find that there are many frequent structures distributed throughout it which are similar on inspection but different in detail, e.g. “at five o’clock” and “at six o’clock”. These structures may include word sequences with low frequency of occurrence (like “five” and “six”). Similar issues often arise in other structures such as the potentially good chunks “a single room” has been broken into several fragments (“a”, “single” and “room”) after the first filtering process. This segmentation would cause problems for the subsequent alignment process. So extracting the most frequent similar structures has turned out to be a very important step in the alignment of bilingual chunks.

Here we cluster similar words (like numbers) mainly according to the position vectors of their behavior relative to anchor words (This basic idea is due to the papers of [9, 10]). As the anchor words are the most common words, a great deal of information can be obtained. It is possible to analyze how the words to be clustered are distributed with respect to the anchor words. This information can compose a vector, and words with similar position vectors in relation to anchor words can be assumed to belong to similar word classes. (Here we have pre-defined two classes of words: 12 month names and 7 weekday names.) The clustering process is described in detail as follows:

(1) In the corpus resulting from the first filtering process, find the most frequent words as anchor words, here lists some most frequent words (See Table 3);

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{Chinese initial chunks} & \text{Initial chunks} & \text{D}_k & \text{D}'_k & \text{D}_k & \text{D}'_k \\
\hline
\text{什么样的} & 0.58 & 2.44 & 0.13 & 0.55 \\
\text{样的房间} & 0.21 & 0.84 & 0.13 & 0.30 \\
\text{样的房间} & 2.45 & 5.88 & 0.69 & 0.69 \\
\text{预定} & 2.39 & 2.39 & 7.80 & 7.80 \\
\text{么样} & 0.87 & 0.87 & 0.30 & 0.30 \\
\text{的房} & 1.27 & 1.27 & 4.52 & 4.52 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|c|c|c|c|c|c|}
\hline
\text{English initial chunks} & \text{Initial chunks} & \text{D}_k & \text{D}'_k & \text{D}_k & \text{D}'_k \\
\hline
\text{do you want to} & 0.13 & 0.90 & 0.086 & 0.60 \\
\text{what kind of} & 2.10 & 5.25 & 0.31 & 0.77 \\
\text{you want to} & 0.33 & 0.82 & 0.056 & 0.14 \\
\text{what kind} & 1.36 & 1.36 & 1.31 & 1.31 \\
\text{do you want to} & 10.07 & 10.07 & 0.61 & 0.61 \\
\text{want to} & 2.11 & 2.11 & 0.077 & 0.077 \\
\hline
\end{array}
\]

In the same way, these English monolingual chunks are obtained: “what kind of”, “room”, “do you”, “want to”, “reserve”. Based on Table 2, the English sentence is segmented as:

What&kind&of||room||do&you||want&to||reserve

In detail, e.g. “at five o’clock” and “at six o’clock”. These structures may include word sequences with low frequency of occurrence (like “five” and “six”). Similar issues often arise in other structures such as the potentially good chunks “a single room” has been broken into several fragments (“a”, “single” and “room”) after the first filtering process. This segmentation would cause problems for the subsequent alignment process. So extracting the most frequent similar structures has turned out to be a very important step in the alignment of bilingual chunks.

Table 1

<table>
<thead>
<tr>
<th>Chinese initial chunks &amp; their cohesion degree (10⁻³)</th>
</tr>
</thead>
<tbody>
<tr>
<td>初始片段</td>
</tr>
<tr>
<td>什么样的</td>
</tr>
<tr>
<td>样的房间</td>
</tr>
<tr>
<td>样的房间</td>
</tr>
<tr>
<td>预定</td>
</tr>
<tr>
<td>么样</td>
</tr>
<tr>
<td>的房</td>
</tr>
</tbody>
</table>

Table 2

<table>
<thead>
<tr>
<th>English initial chunks &amp; their cohesion degree (10⁻³)</th>
</tr>
</thead>
<tbody>
<tr>
<td>初始片段</td>
</tr>
<tr>
<td>do you want to</td>
</tr>
<tr>
<td>what kind of</td>
</tr>
<tr>
<td>you want to</td>
</tr>
<tr>
<td>what kind</td>
</tr>
<tr>
<td>do you want to</td>
</tr>
<tr>
<td>want to</td>
</tr>
</tbody>
</table>

2.3 Clustering the similar words and filtering the most frequent structures

Observing the corpus after the first filtering step, we find that there are many frequent structures distributed throughout it which are similar on inspection but different in detail, e.g. “at five o’clock” and “at six o’clock”. These structures may include word sequences with low frequency of occurrence (like “five” and “six”). Similar issues often arise in other structures such as the potentially good chunks “a single room” has been broken into several fragments (“a”, “single” and “room”) after the first filtering process. This segmentation would cause problems for the subsequent alignment process. So extracting the most frequent similar structures has turned out to be a very important step in the alignment of bilingual chunks.

Here we cluster similar words (like numbers) mainly according to the position vectors of their behavior relative to anchor words (This basic idea is due to the papers of [9, 10]). As the anchor words are the most common words, a great deal of information can be obtained. It is possible to analyze how the words to be clustered are distributed with respect to the anchor words. This information can compose a vector, and words with similar position vectors in relation to anchor words can be assumed to belong to similar word classes. (Here we have pre-defined two classes of words: 12 month names and 7 weekday names.) The clustering process is described in detail as follows:

(1) In the corpus resulting from the first filtering process, find the most frequent words as anchor words, here lists some most frequent words (See Table 3);

Table 3

<table>
<thead>
<tr>
<th>The selected anchor words and their rank in the filtered corpus.</th>
</tr>
</thead>
<tbody>
<tr>
<td>rank</td>
</tr>
<tr>
<td>word</td>
</tr>
<tr>
<td>rank</td>
</tr>
<tr>
<td>word</td>
</tr>
</tbody>
</table>

(2) Let the size of the window for observation be 5 (\( w_2, w_1, \text{anchor-word}, w_{w+1}, w_{w+2} \)), including the current anchor word position (such as “in” in position “0”) and the two adjacent positions to its right (position “-1” and “+1”) and left (position “-2” and “-1”). Then observe whether a candidate word (such as “the”) to be clustered falls within the window, if it does, we add all the occurrence times in each position;

\[
V_j = \sum_{k=1}^{N} \delta(w_j, w) \quad (7)
\]

\[
\delta(w_j, w) = \begin{cases} 
1 & w_j = w \\
0 & w_j \neq w 
\end{cases} \quad (8)
\]

Here let \( V_j \) denotes the position vector of a candidate word (such as “the”) relative to a given anchor word (such as “in”); \( N \) denotes the whole number of monolingual sentences; \( w_j \) denotes the word in position \( j \), \( w \) denotes
the candidate word (such as “the”).

We give an example of one position vector of the candidate word “the” relative to the anchor word “in” within the window size of 5. From the corpus, we can obtain the position vector is \{16, 1, 0, 415, 0\} (See Figure 2). From the figure, we may see that the word “the” always occur follow the word “in” in position “1”.

\[
V_j = V_j / \sum_{j=1}^{m} V_j
\]

(9)

Here the \(j\)th element of the vector \(V_j\) divided by the sum of all elements in \(V_j\) gives the \(j\)th element of the normalized vector \(V_j\).

![Figure 2. The position vector of the candidate word “the” relative to the anchor word “in” in the filtered corpus](image)

Figure 2. The position vector of the candidate word “the” relative to the anchor word “in” in the filtered corpus

(3)In order to compare vectors fairly, these vectors must be normalized. We normalize them according to formula (9) as follows:

From Figure 3 we may see that if the position vectors haven’t be normalized, the most likely two position vectors (“this” to “in” and “that” to “in”) will not be recognized in the following measure scale.

\[
D(V_X, V_Y) = \sqrt{\sum_{j=1}^{m} (V_{Xj} - V_{Yj})^2}
\]

(10)

Table 4 lists some classes discovered by the algorithm mentioned below when used on the sentences resulting from the first filtering process. The underlined italic words are the anchor words which are used to determine the position vectors of the words to be clustered relative to them. In the last row, it may be noticed that the words clustered in this class seem quite unrelated or random. The reason is that these words all frequently follow the word “in”. So the scaling position vectors of these words relative to the word “in” are very similar. The same effect occurs in many other classes: relevant words have been shadowed.

<table>
<thead>
<tr>
<th>Table 4: Word classes discovered after the first filtering corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>single double twin standard suite different quiet (a, room)</td>
</tr>
<tr>
<td>fried poached boiled scrambled (a, egg)</td>
</tr>
<tr>
<td>nine eight two about seven three four (at, o’clock)</td>
</tr>
<tr>
<td>morning afternoon evening night (tomorrow)</td>
</tr>
<tr>
<td>the mine this your our (in, room)</td>
</tr>
<tr>
<td>Piece cup slice glass bottle lot scoop (a, an, of)</td>
</tr>
<tr>
<td>America all fact Japan English yen dollars bed (in)</td>
</tr>
</tbody>
</table>

For all of the words in the same class, substitute with a particular symbol, and then consider this symbol as an ordinary word. Then filter the most frequent structures according to the method given in Sub-section 2.2.

2.4 Deal with the remnant fragment

The lengths of the sentences are quite short after the previous filtering steps, and it may even occur that only one or two individual or sequential words remain. We simply combine such word sequences as a chunk, or consider lone words as a chunk. Finally, these chunks too are filtered in the same way described in Sub-section 2.2.

2.5 Keeping one-to-one alignment

Our purpose is to find one-to-one chunk alignment (including one-to-one, one-to-NULL, NULL-to-one) on the assumption that the chunks to be aligned may occur almost equally in the corresponding parallel texts. Since these are very beneficial to the search process during translation. The merit of this treatment will be seen clearly
in the following language modeling and decoding steps.
The monolingual chunks are aligned with the bilingual cross-restriction: 
\[
\theta = 2 \times \frac{\text{Num}[\text{Co-occurrence}(\text{C}_\text{CHK}, \text{E}_\text{CHK})]}{\text{Num}(\text{C}_\text{CHK}) + \text{Num}(\text{E}_\text{CHK})}
\] (11)

Here, \(\text{Num}[\text{Co-occurrence}(\text{C}_\text{CHK}, \text{E}_\text{CHK})]\) denotes the number of times two chunks \((\text{C}_\text{CHK}, \text{E}_\text{CHK})\) co-occur; \(\text{Num}(\text{C}_\text{CHK})\) denotes the number of times the Chinese chunk \((\text{E}_\text{CHK})\) occur; \(\theta\) is a threshold used to determine whether the monolingual chunks \((\text{C}_\text{CHK}, \text{E}_\text{CHK})\) in the sentence \(k\) are aligned (Here \(k\) denotes the id number of the bilingual parallel sentence).

According to the cross-restriction \(\theta\), the following aligned bilingual chunks are obtained (See Table 5). From Table 5, the maximal value in each column is selected to mark the bilingual chunks to be aligned.

<table>
<thead>
<tr>
<th>(\theta)</th>
<th>what kind of</th>
<th>room</th>
<th>do you</th>
<th>want to</th>
<th>reserve</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.025</td>
<td>0.021</td>
<td>0.669</td>
<td>0.07</td>
<td>0.002</td>
</tr>
<tr>
<td></td>
<td>0.021</td>
<td>0.029</td>
<td>0.014</td>
<td>0.017</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>0.460</td>
<td>0.09</td>
<td>0.09</td>
<td>0.013</td>
<td>0.017</td>
</tr>
<tr>
<td></td>
<td>0.007</td>
<td>0.013</td>
<td>0.002</td>
<td>0.002</td>
<td>0.017</td>
</tr>
<tr>
<td></td>
<td>0.083</td>
<td>0.014</td>
<td>0.023</td>
<td>0.034</td>
<td>0.047</td>
</tr>
<tr>
<td></td>
<td>0.016</td>
<td></td>
<td>0.020</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.888</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.047</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3 Experiments

#### 3.1 Chunking and alignment accuracy

In our experiments, chunking and alignment accuracy are computed based on the MLF algorithm. 5,5000 pairs of Chinese-English spoken parallel sentences were used as the training data (See Table 6) and 400 sentence pairs were chosen randomly from this corpus as the test data. In order to compute the chunking and alignment accuracy, we manually partitioned the 400 pairs of sentences to obtain monolingual chunks and then manually aligned the corresponding bilingual chunks. Comparing the automatically obtained monolingual chunks and aligned bilingual chunks to chunks discovered manually, we compute their precision and recall. Since the performance is related to both precision and recall, the F-Measure [11] value is given as the final evaluation results. The results are shown in Table 7 and Table 8.

#### 3.2 Comparison of chunk-based translation to word-based translation

Two experiments have been carried out. One employed word-based SMT; one used chunk-based SMT. That is, we compared the chunk-to-chunk translations generated by our MLF with the baseline word alignment translations. The improvements in translation quality are dramatic.

In our experiments, the BLEU and NIST scores [12, 13] are used to evaluate our SMTS. The central idea of these two metrics is that the closer a machine translation is to a professional human translation, the better it is. The metrics compute a distance between a candidate (MT) translation and a human (reference) translation by finding the average n-gram similarity. Each output English translation is

\[
\text{precision} = \frac{N_p}{N_o} \times 100\% 
\] (12)
\[
\text{recall} = \frac{N_r}{N_o} \times 100\%
\] (13)
\[
F = \frac{(\beta^2 + 1) \times \text{precision} \times \text{recall}}{\beta^2 \times \text{precision} + \text{recall}}
\] (14)

Here, \(\beta=1; N_r\) denotes the number of chunks that were recognized correctly; \(N_p\) denotes the number of chunks that recognized automatically by the MLF algorithm, and \(N_a\) denotes the number of chunk answers given by humans. In computing the accuracy of chunking, the relevant chunks are the monolingual chunks of Chinese and English. In computing the accuracy of alignment, the chunks in question are the aligned bilingual chunks in the training corpus.

<table>
<thead>
<tr>
<th>Analytical cell</th>
<th>Sentences</th>
<th>Average Length</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word</td>
<td>5,5000</td>
<td>7.97</td>
<td>9935</td>
</tr>
<tr>
<td>Chunk</td>
<td>5,5000</td>
<td>5.66</td>
<td>13263</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The accuracy of chunking</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>77</td>
<td>65</td>
<td>0.70</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The accuracy of alignment</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>89</td>
<td>72</td>
<td>0.80</td>
<td></td>
</tr>
</tbody>
</table>
compared with three references. The evaluation results of these three experiments are presented in Table 9:

### Table 9
Comparison of word-based to chunk-based Systems

<table>
<thead>
<tr>
<th>Systems</th>
<th>BLEU</th>
<th>NIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word-based</td>
<td>0.259</td>
<td>2.661</td>
</tr>
<tr>
<td>Chunk-based</td>
<td>0.290</td>
<td>2.921</td>
</tr>
</tbody>
</table>

From the results in Table 9, it is clear that the new chunk-based SMTS outperforms the word-based algorithm greatly (with an improvement of nearly 10%).

## 4 Conclusions

Our chunking and alignment algorithm does not rely on the information from tagging, parsing or syntax analysis, and does not even require sentence segmentation. After the filtering steps, accurate bilingual alignment chunks and structures are extracted from a parallel corpus. Despite the simplicity of the MLF algorithm, its advantages are clear, it not only obtains accurate one-to-one alignment of chunks, but also greatly decreases search space and time complexity during translation. Our preliminary experiments show that the algorithms yield better performance than the baseline word alignment system.

## References


