

# 11-928 Master's Thesis

## Symmetric Probabilistic Alignment

Jae Dong Kim

April 23, 2006

### Abstract

The CMU Example-Based Machine Translation (EBMT) system has been deployed successfully in many projects for years. But even though a good alignment algorithm is essential since the CMU EBMT system uses parallel corpora, it has relatively less studied than other components of EBMT. For this reason, we developed a new alignment algorithm which uses statistical information drawn from parallel corpora and heuristics based on human linguistic knowledge. Unlike most alignment approaches in Statistical Machine Translation (SMT) systems, our alignment algorithm uses only bilingual dictionaries as statistical information trained from other systems, calculates alignment scores bi-directionally and aims at aligning up to 8 words long source fragments. In our experiments so far, it outperformed the old heuristic-based alignment algorithm in both alignment accuracy and translation accuracy in EBMT. Its performance was very close to the the state-of-the-art in SMT systems for which we picked IBM Model 4 for comparison, and a combination of our new method and IBM Model 4 performed best.

## 1 Introduction

A word or phrasal aligner is essential for the data-driven translation methods such as Example-Based Machine Translation and Statistical Machine Translation. It maps source words or phrases to target words or phrases in training time and this alignment information is used in actual translation time to find translations for the words or phrases in source sentences to be translated.

To translate a given source sentence, when the CMU EBMT system looks up its internal database of translation examples for the word sequences in the source sentence, it prefers the longest matches of word sequences to keep context as much as possible. To make this possible, we need an aligner that maps a source fragment to a target fragment which may be the translation of the source fragment. The target fragment can be either contiguous or non-contiguous. And we call a word sequence in the CMU EBMT system a fragment since a word sequence is different from what we call a phrase in the sense that it is not a unit of meaning.

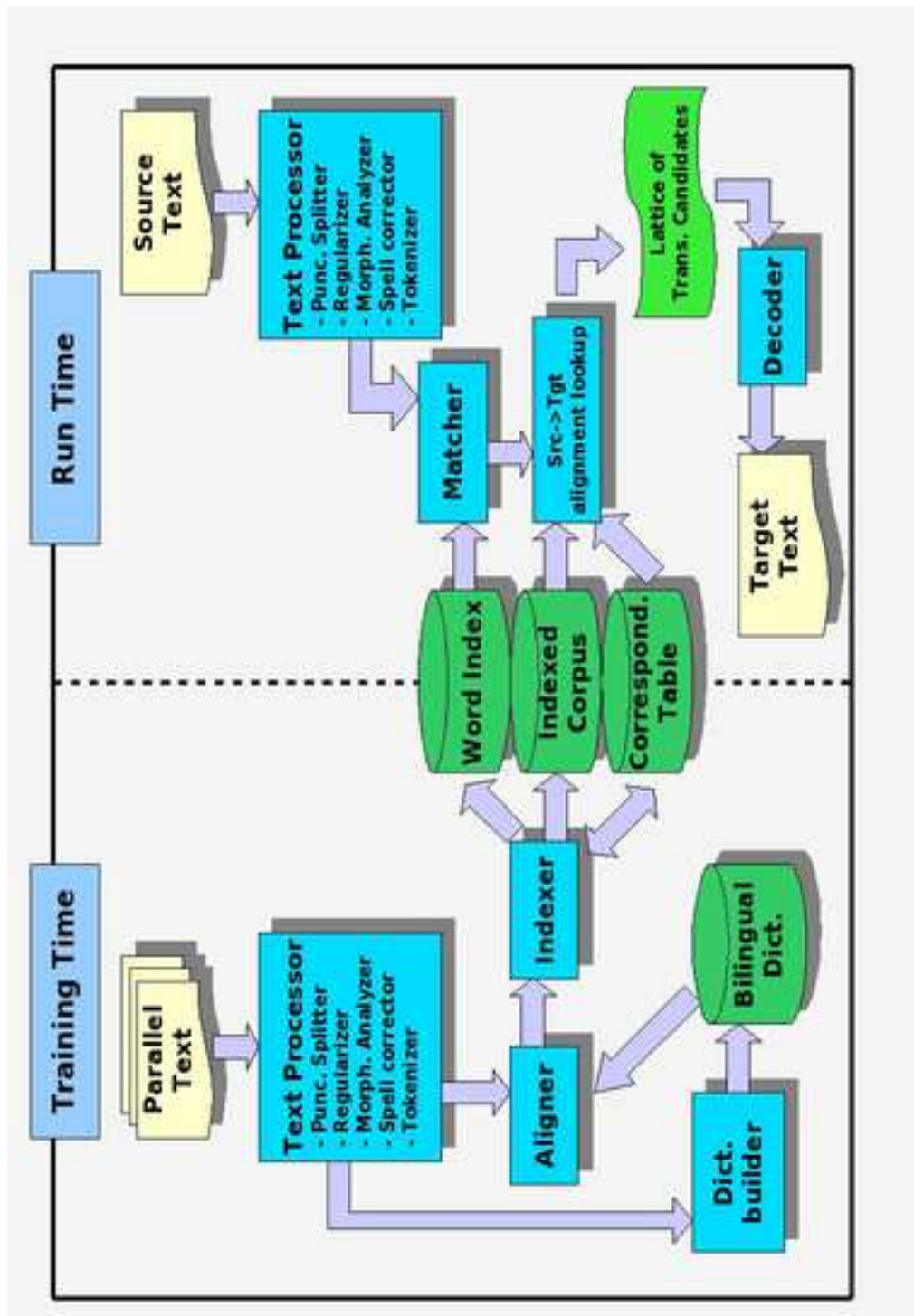


Figure 1: Aligner in the CMU EBMT system

Figure 1 shows where an aligner fits into the CMU EBMT system. In training time, the aligner receives sentence pairs and source fragments as input and finds target fragments in target sentences as translations for the source fragments. The EBMT system stores the aligned data into the internal database "Correspondence Table" and refers to it in actual translation time to retrieve translations for the fragments of a given source sentence to be translated.

## 1.1 The problem

The CMU EBMT system [3] uses a heuristic based alignment algorithm which is very fast but not as effective as the state-of-the-art alignment methods. It builds a correspondence table between individual words in both source and target sides based on a bilingual dictionary. And then, it examines the correspondence table to determine the best translation of a given matched chunk. First it finds shortest and longest possible translations of the matched chunk and then for every contiguous substring which includes the shortest one, a set of simple scoring functions is applied to calculate translation score, and the substring with the best score is chosen as the proper translation of the chunk.

This approach is very fast but doesn't make use of translation probabilities in the dictionary because it only checks the correspondence of the words. As a result, since it doesn't take the translation probabilities into account when it calculates the score, the likelihood of translations between source words and target words are not reflected. Thus, it cannot discriminate the translation pairs with different likelihood and the translation pairs with high and low probabilities affect the score equally.

On the other hand, Statistical Machine Translation research groups which also use parallel corpora have achieved very high performance in alignment trying to make full use of statistical information drawn from parallel corpora in various ways.

Since the alignment concept in both EBMT and SMT is not basically different, we are inspired to improve the CMU EBMT system's alignment algorithm using statistical information drawn from parallel corpora. Since we use bilingual dictionaries with translation probabilities, we can use any kind of statistical training application to build bilingual dictionaries with probabilities.

## 1.2 Review of literature

Many methods and algorithms for sentence and sub-sentential alignment have been developed by various machine translation groups. Some use heuristic-based methods, others work in pure statistical ways, and others exploit linguistic knowledge in alignment.

Some SMT researchers have used similarity functions between two languages [9] [6]. Variants of the Dice coefficient [4] have been used frequently to calculate similarity by obtaining a matrix including association scores between each pair of a source word and a target word at different positions for each sentence pair.

Brown et al [2], at the IBM T.J.Watson research center in the early 1990s, developed several alignment models for use with the EM algorithm, which are now commonly called IBM model 1 etc, intended to provide increasingly more accurate models of the translation process, but also increasingly less stable numerically. Model 1 assumes that for a source word position, all connections to target word positions are equally likely. It means all the possible alignments are equally likely. In Model 2, they have a more realistic assumption that the probability of a connection between a source position and a target position depends on the positions it connects and on the lengths of the two strings(i.e., a source string and the corresponding target string in a parallel corpus) The connection between positions is known as distortion. In Model 3, they introduced word fertility. They choose the number of French words which is connected to an English word first and then the remaining procedure is the same as Model 2. In Model 4, the connection probability of a French word and an English word depends on the positions of other French words connected to the English word in addition to the identities of the original words. Model 5 is very much like Mode 4 except that it is not deficient. A deficient model can choose the same target position repeatedly for the target words given different source words and it could result in too many empty target positions. In these models, parameter estimation is the key point to improve the performance and they used EM algorithms to estimate parameters. Since, EM algorithms converge to local maxima, they use the previous model's parameters as the initial parameter values to achieve better performance.

Vogel et al [11] have used HMM in alignment since the SMT alignment representation restricts a source word position to a target word position. They assumed a first-order dependence for the alignments and that the lexicon probability depends only on the word at a given position.

Yamada and Knight [12] used explicit syntactic information in re-ordering in target language in SMT and raised the prospect of training a SMT system using syntactic information for both languages.

Recently, the CMU EBMT group has developed a Symmetric Probabilistic Alignment (“SPA”) method which is based on optimizing phrasal alignments bidirectionally and uses statistical information [5]. The SPA differs from the above in that it aims at aligning fragments which is a normal translation unit in the CMU EBMT system and it uses only a bilingual statistical dictionary. It doesn't use additional statistical information nor uses syntactic information. This thesis is from my research in this group.

### 1.3 Aims of the thesis

In this thesis, we describe the algorithm of the SPA, the experimental settings and results. We report alignment accuracy to show how well the aligner works and the BLEU score through the CMU EBMT system to show how it contributed to the EBMT system. Finally, we describe our future work for further improvement.

## 1.4 Description of the remaining chapters

In the remaining chapters, we explain the basic idea of Symmetric Probabilistic Alignment and constraints applied to it and then show some experimental results.

In chapter 2, first we explain the basic idea of the SPA and then several constraints that we applied to the basic idea to improve the performance. In the last two sections of this chapter, we suggest, without experimental results, an SPA variant for non-contiguous alignment, and the design of an algorithm for merging human-constructed dictionary into an automatically generated statistical dictionary.

In chapter 3, we describe experimental settings and the performance for each setting. we describe the data and experimental procedure to calculate alignment accuracy and the data and experimental procedure to calculate the BLEU score when the method is used from within the EBMT system. Then we report experimental results and analysis.

In chapter 4, we summarize the experimental results and analysis. And we also explain the problems unsolved and future work to improve the SPA.

## 2 SPA

### 2.1 Notational Convention

In this paper, we try to use accepted notation for easier reading. But we use “s” and “t” to denote source and target language.

1. A word: A word is denoted by a lower case letter with a subscript letter for index. So the source word in position  $i$  would be denoted by  $s_i$  and the target word in position  $j$  would be denoted by  $t_j$ .
2. A fragment: A fragment is denoted by a lower case letter with a subscript and a superscript for the starting position and the ending position. So, the source fragment from position  $i$  to  $j$  would be denoted by  $s_i^j$  and the target fragment from position  $k$  to  $l$  would be denoted by  $t_k^l$ .  $s_i^i$  is the same as  $s_i$ . For an arbitrary fragment, we use just a lower case letter such as  $t$ .
3. A sentence: A sentence is denoted by a capital letter. So, a source sentence is  $S$  and a target sentence is  $T$ . We may add a subscript for index. When the length of  $S$  is  $K$ ,  $S = s_1^K$ .

### 2.2 Basic algorithm

In sub-sentential alignment, mappings are produced between words or phrases in the source language sentence and those words or phrases in the target language sentence that best express their meaning.

An alignment algorithm takes as input a bilingual corpus consisting of corresponding sentence pairs and strives to find the best possible alignment in the second for selected n-grams (sequences of n words) in the first language. The alignments are determined based on a number of factors, including a bilingual dictionary (preferably a probabilistic one), the position of the words, punctuation, invariants (such as numbers), and so forth.

For our baseline algorithm, we make the following simplifying assumptions, each of which we relax in the remainder of this thesis:

1. A fixed bilingual probabilistic dictionary is available.
2. Contiguous fragments (word sequences) of source language text are translated into contiguous fragments in the target language text.
3. Fragments are translated independently of surrounding context.

Our baseline algorithm is based on maximizing the probability of bi-directional translations of individual words between a selected n-gram in the source language and every possible n-gram in the corresponding paired target language sentence. The reason why we use the probability of bi-directional translations is that we are more convinced when both side’s fragments agree that the other sides’ fragments are their translations. For example, given a source fragment  $s_i^j$ , assume that the two target fragments  $t_k^l$  and  $t_n^o$  are equally probable ‘best’ translations of  $s_i^j$ . If we consider opposite directional translations and find that  $t_k^l$ ’s the most probable translation is  $s_i^j$  and  $t_n^o$ ’s the most probable translation is  $s_p^q$  ( $i \neq p$  or  $j \neq q$ ), we will choose  $t_k^l$  as the translation of  $s_i^j$ .

No positional preference assumptions are made, nor are any length preservation assumptions made. That is, an n-gram may translate to an m-gram, for any values of n or m bounded by the source and target sentence lengths, respectively. Finally, we introduce a small positive “smoothing value”  $\epsilon$  to avoid singularities (i.e. avoiding zero-probabilities for unknown words, or words never translated before in a way consistent with the dictionary).

Suppose that we are given a pair of aligned sentences  $S$  of length  $K$  and  $T$  of length  $L$  where a source sentence  $S$  is

$$S : s_1, \dots, s_{i+1}, \dots, s_{i+k}, \dots, s_K \tag{1}$$

and the corresponding target language sentence  $T$  is

$$T : t_1, \dots, t_{j+1}, \dots, t_{j+l}, \dots, t_L \tag{2}$$

and calculating the translation probabilities between a source fragment  $s_{i+1}^{i+k}$  and target fragments in  $\{t_{j+1}^{j+l}\}$ .

Then the fragment we try to obtain is the target fragment  $\bar{t}$  with the highest probability of all possible fragments of  $T$  to be a mutual translation with the

given source fragment, or

$$\bar{t} = \operatorname{argmax}_t \operatorname{Score}_t \quad (3)$$

$$= \operatorname{argmax}_t (p(s_{i+1}^{i+k} \leftrightarrow t_{j+1}^{j+l})) \quad (4)$$

$$= \operatorname{argmax}_t (p(s_{i+1}, \dots, s_{i+k} \leftrightarrow t_{j+1}, \dots, t_{j+l})) \quad (5)$$

$$= \operatorname{argmax}_t \left( \left( \prod_{p=1}^k \max_{q=1}^l p(t_{j+q} | s_{i+p}), \epsilon \right) \right)^{\frac{1}{k}} \quad (6)$$

$$\times \left( \prod_{q=1}^l \max_{p=1}^k p(s_{i+p} | t_{j+q}), \epsilon \right)^{\frac{1}{l}} \quad (7)$$

Here and in the following sections for algorithm description, we use  $t = t_{j+1}^{j+l}$  for the target candidate fragment  $t$ .

In above equation, (6) shows the unidirectional score calculation from source to target, and (7) shows the unidirectional score calculation from target to source. So, (6) and (7) together calculates the symmetric probabilistic alignment score.

All possible candidate target fragments can be checked in  $O(L^2)$ , where  $L$  is the target language length, because we will check  $L$  1-word-long fragments,  $L - 1$  2-word-long fragments, and so on.

### 2.3 Untranslated Word Penalty

In our basic algorithm, when we calculated a symmetric probabilistic alignment score, but didn't count how many words in the counter part fragment are actual translations for the given fragment words. But we prefer an alignment which has more actual translations in the counter part fragment. For example, for a given source fragment  $s = s_{i+1}^{i+k} = s_{i+1}, \dots, s_{i+k}$  and a given candidate target fragment  $t = t_{j+1}^{j+l} = t_{j+1}, \dots, t_{j+l}$ , if all source words in  $s$  are translated into a single target word in  $t$ , and if all target words in  $t$  are translated into a single source word in  $s$ , this alignment is not desirable and should be penalized.

So we will penalize alignment score according to the ratio of  $\frac{\#(\text{translations})}{|\text{fragment}|}$ . A modified formula would be

$$\begin{aligned} \operatorname{Score}_t &= P(s_{i+1}^{i+k} \leftrightarrow t_{j+1}^{j+l}) \quad (8) \\ &= P(s_{i+1}, \dots, s_{i+k} \leftrightarrow t_{j+1}, \dots, t_{j+l}) \\ &= \left( \prod_{p=1}^k \max_{q=1}^l p(t_{j+q} | s_{i+p}), \epsilon \right)^{\frac{1}{k}} \times (R_t)^\alpha \\ &\quad \times \left( \prod_{q=1}^l \max_{p=1}^k p(s_{i+p} | t_{j+q}), \epsilon \right)^{\frac{1}{l}} \times (R_s)^\alpha \end{aligned}$$

where  $R_f = \frac{\# \text{ of actual translation words in the fragment } f}{\# \text{ of potential translation words in the fragment } f}$ , and  $\alpha \geq 1$ . In this formula, when  $R_f$  is less than 1, it reduces  $\operatorname{Score}_t$  and, as a result, penalizes the

score. In the previous example,  $R_t = \frac{1}{7}$  and it obviously reduces  $Score_t$  when  $l > 1$ .

## 2.4 Length Penalty

The ratio of target fragment ( $n$ -gram) and source fragment ( $m$ -gram) lengths should be comparable to the length ratio of the target sentence and source sentence lengths, though certainly variation is possible. Therefore, we generate a penalty function to the alignment probability that increases with the discrepancy between the ratios as  $n/m$  is compared to the target/source sentence length ratio  $\frac{L}{K}$ .

Let the length of the source language fragment be  $k$  and the length of a target language fragment under consideration be  $l$ . And let the dynamic sentence length ratio be  $\frac{L}{K}$  given the source language sentence  $S$  and its corresponding target language sentence  $T$  in the section 2.2. The expected target fragment length is then given by  $\hat{l} = k \times \frac{L}{K}$ . Further defining an allowable length difference  $LD_{allowed}$ , our implementation calculates the length penalty  $LP_t$  as follows:

$$LD_{allowed} = LD_{constant} \times \frac{L}{|T|_{average}} \quad (9)$$

$$LP_t = \min\left(\left(\frac{|l - \hat{l}|}{LD_{allowed}}\right)^4, 1\right) \quad (10)$$

where  $|T|_{average}$  means *the average target sentence length in the training corpus*.

We wanted to ignore target candidate fragments which have larger difference than  $LD_{allowed}$  and to give bigger penalty to the  $LD_{allowed}$ -satisfying target candidate fragments as they have larger difference. For equation (10), the 4th power was the one that gave us the best experimental results among the powers from 2 through 6.

The score for a fragment including the penalty function is then:

$$Score_t \leftarrow Score_t \times (1 - LP_t) \quad (11)$$

Note that, as intended, the score is forced to 0 when the length difference  $|l - \hat{l}| > LD_{allowed}$ .

## 2.5 Distance Penalty

Closely related languages (such as French and English) tend to have more similar word orders than more distantly-related languages such as Korean and English. In the former case, this results in greater phrase order similarity and consequently similar phrase positions.

In such a close language pair, we introduce a distance penalty<sup>1</sup> to penalize the alignment score of any candidate target fragment as they are away from the

---

<sup>1</sup>Our distance penalty is conceptually different from the distortion penalty in SMT systems because it assumes that a target fragments should be in the proportional position to the source fragment position in a target sentence.

expected position range. Our distance penalty is calculated in the same way as in section 2.4. First, we calculate the expected center  $\hat{C}$  of the candidate target fragment using the center of the source fragment  $C_s$  and the dynamic sentence length ratio  $\frac{L}{K}$

$$\hat{C} = C_s \times \frac{L}{K} \quad (12)$$

Then we calculate  $DD_{allowed}$ , the dynamic allowed distance difference of the center, using a constant limit value  $DD_{constant}$  and the dynamic sentence length ratio  $\frac{L}{|T|_{average}}$  where  $|T|_{average}$  is the average target sentence length in the training corpus.

$$DD_{allowed} = DD_{constant} \times \frac{L}{|T|_{average}} \quad (13)$$

Given  $DD_{allowed}$ , we calculate the distance penalty  $DP_t$  as follows:

$$DP_{F_T} = \min\left(\left(\frac{|C_t - \hat{C}|}{DD_{allowed}}\right)^4, 1\right) \quad (14)$$

where  $C_t$  is the actual center of the target fragment  $t$  being processed.

As we did in section 2.4, we wanted to ignore target candidate fragments which have larger difference than  $DD_{allowed}$  and to give bigger penalty to the  $DD_{allowed}$ -satisfying target candidate fragments as they have larger difference. For equation (14), as in length penalty calculation, the 4th power was the one that gave us the best experimental results among the powers from 2 through 6.

The score for a fragment including the penalty function is then:

$$Score_t \leftarrow Score_t \times (1 - DP_t) \quad (15)$$

Note that, as intended, the score is forced to 0 when the length difference  $|C_t - \hat{C}| > DD_{allowed}$ .

It may in fact be possible to usefully apply the distance penalty to language pairs in which the language pairs have a very dissimilar word order, provided we can determine or estimate a positional mapping between the sentences in a pair, and then use the distance with respect to this mapping.

## 2.6 Anchor Context

If the words adjacent to the source fragment and the candidate target fragment are translations of each other, we expect that this alignment is more likely to be correct because adjacent source words are usually aligned to adjacent target words and in this case, an alignment of adjacent words adds supporting evidence to the alignment we are considering. We combine  $Score_t$  with the anchor context alignment score  $AnchorScore_t$  by a linear weighted combination in log space,

$$AnchorScore_t = P(s_i \leftrightarrow t_j) * P(s_{i+k+1} \leftrightarrow t_{j+l+1}) \quad (16)$$

$$Score_t \leftarrow (Score_t)^\lambda * (AnchorScore_t)^{1-\lambda} \quad (17)$$

Empirically, we found this combination gives the best score when  $\lambda = 0.8$  for both French-English and English-Chinese and it gave a better result than

$$Score_t \leftarrow \lambda * Score_t + (1 - \lambda) * AnchorScore_t \quad (18)$$

## 2.7 Merging Human Dictionaries

In this section and the next we consider research issues for which we have not yet completed the experimental setup and evaluation. One question is how an available human-constructed dictionary might be put to good use in alignment or in translation. A statistical dictionary only reflects the domain specific features of the parallel corpus on which it is trained, so it might lack generality in translations in general domains. Of course, in some situations it reflects the domain specific features and one might use it intentionally but our goal here is to make the dictionary general and cover all the translations as much as possible. The issues in combining a statistical dictionary and a human dictionary are what probabilities we give to human dictionary translation alternatives and how much weight we give to both side elements.

For the former one, for initial experiments, we try the same probability for the alternatives. That means if we have  $n$  translation alternatives for a given source word, the translation alternatives will have  $\frac{1}{n}$  as the translation probabilities. Another possible method is to give probabilities in decreasing order since usually they are listed in importance(or reference) decreasing order.

For the latter one, first, we test  $\lambda$  combination method. For source words which don't appear in the other dictionary, we just keep the probability. But for source words appearing in both dictionaries, we combine them using  $\lambda$ . That is,

$$p_{combined}(t_j|s_i) = \lambda \times p_{human}(t_j|s_i) + (1 - \lambda) \times p_{statistical}(t_j|s_i) \quad (19)$$

Another method we can try for the latter one is to give more weight to the alternatives which appear both in the statistical dictionary and in the human dictionary since they are voted as translations by both dictionaries.

## 2.8 Non-contiguous alignment

In many cases, a contiguous source fragment is not aligned to a contiguous target fragment but to a non-contiguous target fragment. For example, the English source fragment “not” may be aligned into a French target fragment “ne ... pas”, where often the “ne” and “pas” are not adjacent. Another example is the alignment of the English source fragment “lock the door” and a Spanish target fragment “cierre la puerta con llave” where “lock” is aligned to “cierre” and “con llave” (literally meaning ‘close with key’).

For cases like the above, we introduce non-contiguous alignment allowing at most one gap in the target fragment side. The basic heuristic we apply here is that we boost the alignment score when the gap and the outside of the source fragment have a translation relationship or the outside of the candidate target

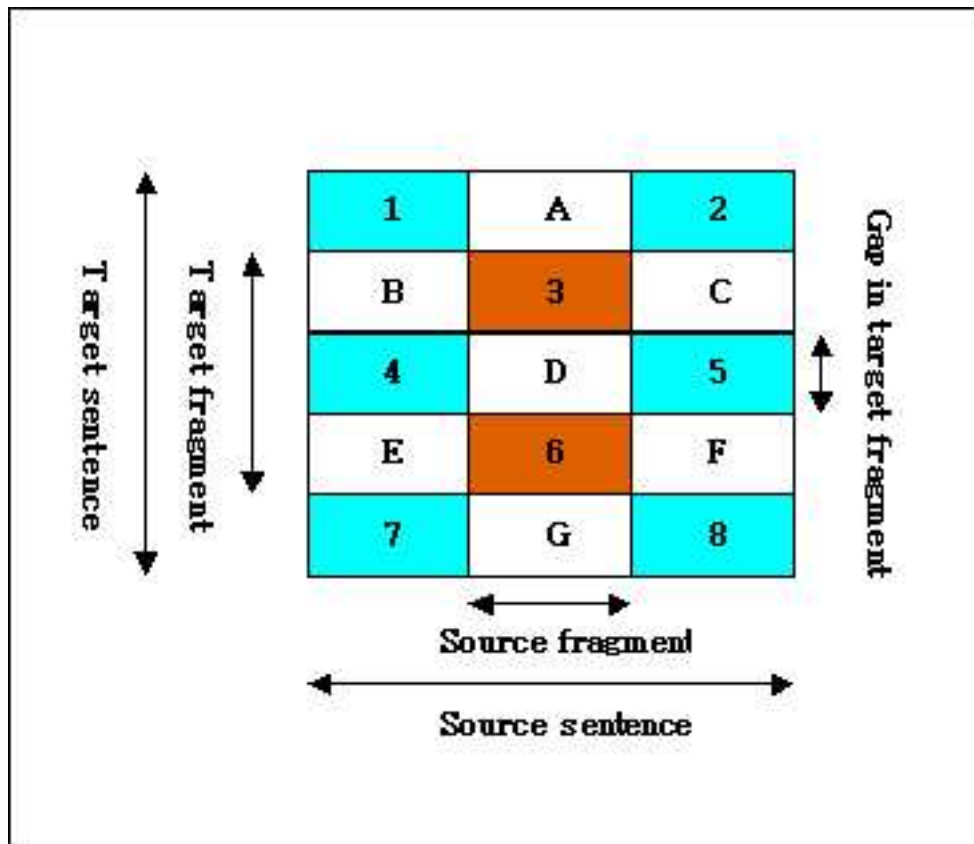


Figure 2: Non-contiguous alignment

fragment and the outside of the source fragment has translation relationship. But when the candidate target fragment and the outside of the source fragment has translation relationship, when the gap and the source fragment has translation relationship, or when the outside of the candidate target fragment and the source fragment has translation relationship, we penalize the score.

Figure 2 shows the boosting area and the penalizing area. The areas 3 and 6 mean non-contiguous alignment for the source and target fragments, the areas 1, 2, 4, 5, 7 and 8 are the boosting areas and the areas A, B, C, D, E, F and G are the penalizing areas.

One example of the formula for the alignment score could be like this:

$$Score_t \leftarrow \alpha \times ScoreF_t(i, i) \tag{20}$$

$$-\beta \times ScoreF_t(i, g) \tag{21}$$

$$-\gamma \times ScoreF_t(i, o) \tag{22}$$

$$+\delta \times ScoreF_t(o, o) \tag{23}$$

$$+\epsilon \times ScoreF_t(o, g) \tag{24}$$

$$-\zeta \times ScoreF_t(o, i) \tag{25}$$

where

$$ScoreF_t(i, g) = P(i \leftrightarrow g) \tag{26}$$

given a target fragment  $t$  and  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\delta$ ,  $\epsilon$ , and  $\zeta$  are all positive. Here the first parameter of  $ScoreF()$  is an area in the source sentence, and the second parameter is an area in the target sentence. The area labels  $i$ ,  $o$  and  $g$  represent *inside of the fragment*, *outside of the fragment in the sentence* and *the gap in the fragment* respectively. So, in figure 2,  $ScoreF_t(i, i)$  is the score for area 3 and 6,  $ScoreF_t(i, g)$  is the score for area D,  $ScoreF_t(i, o)$  is the score for area A and G,  $ScoreF_t(o, o)$  is the score for area 1, 2, 7 and 8,  $ScoreF_t(o, g)$  is the score for area 4 and 5,  $ScoreF_t(o, i)$  is the score for area B, C, E and F. For example, given a candidate target fragment  $t$ ,  $ScoreF_t(i, g)$  function calculates the alignment score between the source fragment and the gap in the target fragment.

But as we have seen in SPA so far, (20) is already implemented by the basic algorithm, (23) is implemented by adding the anchor context constraint. For an initial experiment, we may pick up (21) and (24) and implement them, which would show the effect of the gap consideration the most.

## 2.9 Implementation

Figure 3 shows the control flow diagram of the SPA. This diagram shows the input and output of the SPA and the order which the basic algorithm and the restrictions work in.

To explain the algorithm, given the input of a bilingual dictionary, a sentence pair and a source fragment,

- 1 load the dictionary and segment the target sentence into  $\frac{l \times (l+1)}{2}$  contiguous fragments when the target sentence is  $l$  words long.
- 2 select a target fragment.
- 3 calculate the basic score and apply untranslated word penalty to each unidirectional score.
- 4 apply length penalty, distance penalty and anchor context bonus.
- 5 if there are remaining target fragments to be considered, go to 2. Otherwise, output the target fragment which has the highest score.

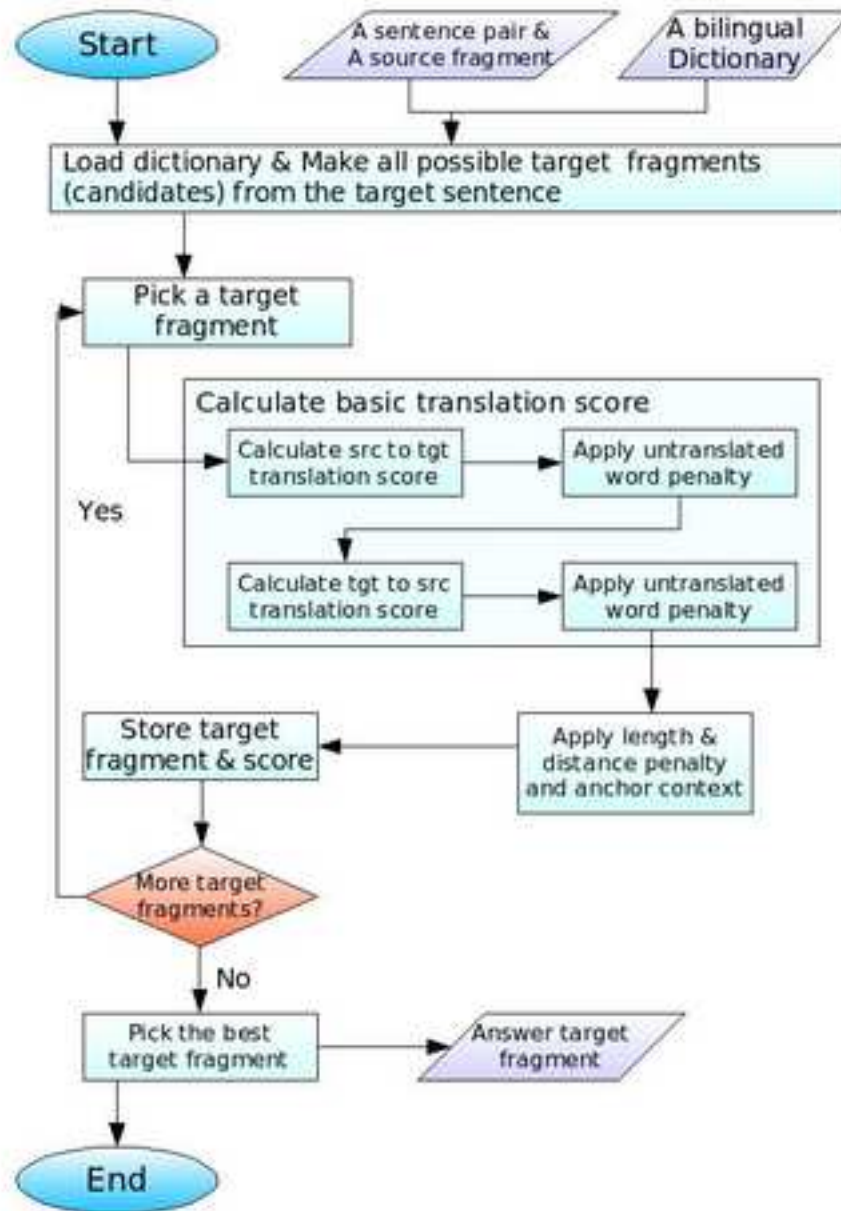


Figure 3: Symmetric Probabilistic Aligner

For efficiency, we improved above algorithm with followings.

- 1 Because floating point value manipulation is expensive, we calculate scores in log space. So, the translation probability values in the dictionary were transformed into log values before they were used.
- 2 Since  $\frac{l \times (l+1)}{2}$  target fragment are very many, we cut off candidates whose lengths are greater than 4 times the source fragment length or less than  $\frac{1}{4}$  times the source fragment length.
- 3 Since a lot of target fragments overlap, we use a dynamic programming technique for translation score calculation. For example,  $t_{j+1}^{j+l-1}$  and  $t_{j+1}^{j+l}$  overlap from  $t_{j+1}$  to  $t_{j+l-1}$ , and we reuse the unidirectional translation score from  $t_{j+1}^{j+l-1}$  to the source fragment when we calculate the unidirectional translation score from  $t_{j+1}^{j+l}$  to the source fragment. In this case, the unidirectional translation score from  $t_{j+1}^{j+l}$  to the source fragment is the larger unidirectional translation score of from  $t_{j+1}^{j+l-1}$  to the source fragment and from  $t_{j+l}$  to the source fragment. Given a source fragment and a target sentence, we built a 2-dimensional table which is indexed by a fragment's length and starting word position.

## 3 Experiments

### 3.1 Alignment Evaluation

#### 3.1.1 Data

We tested our alignment method on a set of French-English sentences taken from the Canadian Hansard corpus and on a set of English-Chinese sentences taken from Xinhua news agency. French and English are chosen as an easy pair because they have very similar word order while English and Chinese are chosen as a difficult pair because the word order difference and the sentence length difference are the most evident.

For French-English, we had 91 human aligned sentence pairs and from that, we generated 12466 3-8 words long contiguous source fragments.

For English-Chinese, we had 3 sets of 366 human aligned sentence pairs which have the same data but are aligned by different people (They are named ltao, xuwang and sandy according to the names of the human aligners). In addition to that, we had 20 more human aligned sentence pairs aligned by another person. So, for the alignment evaluation, we picked one of the three sets - ltao was picked in this experiment - and added it to the other 20 sentences to make a 386 human aligned sentence pair set and 27286 3-8 words long source fragments. And later we used the 3 sets to see how reliable human alignments are by evaluating each against the others.

For these experiments, we preprocessed the data. We segmented Chinese data into words, and expanded the contractions in French and English data. We separated punctuations in the data in all three languages.

To see the effect of merging a human dictionary into a statistical dictionary, we gathered French-English and English-French human dictionaries from TravLang web pages [10] and ARTFL [1] project web pages.

### 3.1.2 Evaluation Matrix

For the human aligned data, we compared the results of our algorithm to human alignments. Although the latter may not be perfect and sometimes are non-unique, they provide the only answer key available for repeatable tests. As metrics, we use *precision*, *recall* and  $F_1$  (the harmonic mean of precision and recall). Since *precision* and *recall* cannot be used alone to measure the performance of the alignment methods, we use  $F_1$  values to measure the performance and to compare the alignment methods. In other words, we use  $F_1$  to measure the performance in both terms of *precision* and *recall*.

We calculate precision, recall and  $F_1$  based on answer position overlaps. Let us suppose that the position sequence of our (machine) answer fragment is  $p_1, p_2, \dots, p_k$  and the position sequence of the correct answer (human) fragment is  $hp_1, hp_2, \dots, hp_l$ . Note that the correct (human) answer may be non-contiguous, but the combination of SPA and EBMT to date is only capable of using the best *contiguous* target  $m$ -gram alignment it can find. Given that  $o = \text{count}(p0_i)$  and  $p0_i$  is  $p_i$  which is not aligned to any in the human answer, we compute the recall  $R$  and precision  $P$  as follows:

$$R = \frac{\text{count}(hp_i \in \{p_i\})}{l} \quad (27)$$

$$P = \frac{\text{count}(p_i \in \{hp_i\})}{k - o} \quad (28)$$

To obtain an average alignment score for evaluation, we

- generated all the possible source language sentence fragments lengths 3 through 8 from the human aligned data.
- aligned those fragments by means of our algorithm, and
- calculated the metrics given above by comparison with the human-aligned answers.

### 3.1.3 Baselines

To have a better intuition of the alignment results we obtain for a given language pair (and corpus), we introduce the following as baselines: “random result”, “positional result”, and “oracle result”.

The “random result” is a randomly chosen target fragment regardless of the source fragment, constrained to be of a length corresponding to the source fragment normalized by the length ratio of the source and target sentences.

The “positional result” is a target fragment whose position in the target language most closely matches the position of the source fragment. We calculate the target fragment’s start and end position using source fragment’s start, end position and the length ratio of source sentence and target sentence. In particular, let the source sentence be of length  $n$  and the target sentence of length  $m$ , we expect source position  $i$  to correspond to target position  $j$  where  $j \simeq i \times \frac{m}{n}$ .

The “oracle result” is the best contiguous target fragment extracted from human alignments. To get the oracle result, first, we get human alignments for the sentence pairs which will be used to evaluate our algorithm. Then we choose a fragment which has the largest harmonic mean value among human alignment fragments and whole fragment. Notice that the human alignment may not be contiguous, therefore “oracle alignment” represents the best that our algorithm could possibly perform.

### 3.1.4 Comparison with the state-of-the-art alignment

We also included IBM Model 4 (“IBM4”) alignment accuracy to see what is the status of SPA compared to the state-of-the-art.

Finally to get advantages from both SPA and IBM4 we combine the results of SPA and IBM4. We set a threshold score for SPA and combined SPA and IBM4 results by substituting IBM4 results with SPA results which have higher alignment score than the threshold (“COMB”). For the significance test, we separated French-English human aligned data into 10 data sets of 9 sentences and English-Chinese human aligned data into 10 data sets of 36 sentences.

## 3.2 EBMT Performance

Since our goal is to develop a new alignment method to improve the CMU EBMT system’s performance, we evaluated the performance of the CMU EBMT system using SPA, IBM Model 4, and the original internal aligner of the system.

### 3.2.1 Data

For our EBMT experiments we used a subset of the IBM Hansard corpus available from the Linguistic Data Consortium. This corpus is divided into files of 10,000 sentence pairs (with an occasional garbled or missing line which has been removed prior to our use), of which we used only files 000 through 099.

The training data consisted of the first 20,000 sentence pairs – essentially files 000 and 001 – for EBMT and the first 700,000 English sentences for the language model. The development test (“Devtest”) set used for parameter tuning consisted of the first 100 sentences of file 040 and the evaluation test (“Test”) set consisted of ten segments of 100 sentences drawn from files 060 and 080. Segmenting the evaluation test set in this manner allowed us to perform statistical

significance tests. Another test (“2refTest”) set consists of 100 source sentences and 200 reference sentences. To see whether the performance is consistent we made another reference set for the 100 source sentences such that each source sentence has two reference sentences. The original 100 sentence pairs are mostly drawn from file 060.

### 3.2.2 Evaluation Methodology

To minimize the initial investment of effort for the EBMT evaluation, we performed a partial exploiting of the SPA and EBMT modules rather than fully incorporating SPA into the EBMT engine. In this partial integration, SPA is used to annotate the training corpus with alignments (both phrasal and word-to-word), and the annotations in the corpus override the EBMT engine’s internal aligner. Phrasal alignments are stored as-is, and whenever a partial match against the corpus is exactly equal to the source half of such an alignment, the target half is output as the candidate translation. The word-to-word alignments are used to build a correspondence table (overriding the one which would have been built in the absence of alignment annotations) and that table is consulted as usual to perform alignments of matches for which there is no phrasal alignment from SPA available. One drawback of this arrangement compared to a full integration is that we are unable to take advantage of non-contiguous alignments (however, such alignments would also require modifications to the decoder which have yet to be implemented).

This yields the following training regimens for the alignment methods. To test the old algorithm, we

1. built a statistical dictionary from the corpus
2. indexed the training text using that dictionary

To test performance with IBM Model 4 alignments, we

1. trained GIZA++ [7] on the training text
2. annotated the training corpus using Model 4
3. indexed the annotated corpus

To test performance with SPA, we

1. used GIZA++ to build a dictionary from the training text
2. ran the SPA aligner on the training text using that dictionary
3. indexed the annotated corpus generated by SPA

The differently-trained translation systems are then each evaluated on the test set using the BLEU metric [8].

In addition to testing the full IBM4, COMB and SPA algorithms with both phrasal and word-level alignments, we also tested the performance when using

Test/Answer	Recall	Prec.	$F_1$	Len(M)/Len(H)
ltao/xuawang	0.858758	0.980900	0.915774	0.875480
ltao/sandy	0.742688	0.982903	0.846076	0.755607
xuawang/ltao	0.896835	0.976533	0.934989	0.918387
xuawang/sandy	0.783359	0.987704	0.873742	0.793111
sandy/ltao	0.959004	0.950798	0.954884	1.008631
sandy/xuawang	0.968574	0.961476	0.965012	1.007382

Table 1: Human Answer Evaluation

Key	Description
random	Random results
positional	Results in proportional positions
oracle	The best possible contiguous results from human answer
SPA-single	SPA - unidirectional alignment (source to target)
SPA-basic	SPA - basic bi-directional alignment
SPA-anchor	SPA - basic + anchor bonus
SPA-len	SPA - basic + length penalty
SPA-dist	SPA - basic + distance penalty
SPA- $x_1-x_2..$	$x_n$ can be substituted with a,l,d and u. a: anchor bonus, l: length penalty, d: distance penalty, u: untranslated word penalty
IBM4-cont	IBM4 - considers the words between the smallest and the largest as the contiguous answer
IBM4-oracle	IBM4 - the best possible contiguous results
IBM4	IBM4 - non-contiguous results
COMB	combined results of the best SPA and IBM4

Table 2: Key to the following alignment evaluation tables

each algorithm restricted to generating pure word-level alignments for creating the correspondence table used by EBMT’s internal aligner (in lieu of having it generate that table itself from the bilingual lexicon it extracted from the training examples). The runs using the restriction to word-level alignments are identified as “IBM4-W”, “COMB-W” and “SPA-W”, respectively.

### 3.3 Results and Analysis

#### 3.3.1 Alignment Evaluation

As we already mentioned, given a set of parallel sentences, human alignments are not unique. This problem is related to how accurate our evaluations results are. To have a rough answer to this question, we evaluated human answers regarding them as machine answers and the others as human answers for the

Aligner	Recall	Prec.	$F_1$	Len(M)/Len(H)
random	0.321979	0.372175	0.345262	0.865128
positional	0.582254	0.576207	0.579215	1.010495
oracle	0.905602	0.861449	0.882974	1.051254
SPA-single	0.942574	0.355970	0.516776	2.647905
SPA-basic	0.869897	0.473884	0.613538	1.835674
SPA-anchor	0.792371	0.472218	0.591768	1.677975
SPA-len(7)	0.786690	0.610359	0.687396	1.288897
SPA-dist(10)	0.877859	0.467348	0.609967	1.878384
SPA-l-u	0.733485	0.693883	<i>0.713135</i>	1.057072
SPA-a-l	0.714582	0.569360	0.633758	1.255062
SPA-a-d	0.798097	0.471970	0.593162	1.690991
SPA-l-d	0.788104	0.603550	0.683590	1.305780
SPA-l-d-u	0.734956	0.684055	0.708592	1.074412
SPA-a-l-d	0.718347	0.568684	0.634814	1.263173
SPA-a-l-d-u	0.703414	0.598468	0.646711	1.175357
IBM4-cont	0.816726	0.604259	0.694608	1.351616
IBM4-oracle	0.727074	0.700257	0.713413	1.038295
IBM4	0.738995	0.807471	0.771717	0.915196
COMB	0.756338	0.804163	<b>0.779517</b>	0.940528

Table 3: English-Chinese: Best alignment results evaluation

same data set with our evaluation metrics. Table 1 shows the human answer evaluation results. In these tests,  $F_1$  varies from 0.846076 to 0.965012. This may give us a rough idea about what score we can aim to achieve. Of course, approaching those values doesn’t mean that the aligners are as good as human since the errors by the aligners might be linguistically serious while human errors are not.

For comparing the alignment accuracy, we chose the positional alignment as the base line – as this is the best we can do without any information about the words at all – and the oracle alignment as the goal. Table 3 through table 6 show the oracle result obtained by each alignment method. As previously mentioned, “positional” is the baseline for comparing alignment performance while “oracle” is the best possible selection of contiguous fragments.

Table 3 and table 5 show the best performance by each aligner and table 4 and table 6 show the possibility of improvement for SPA aligners. In table 4 and table 6, we reported the best of top 10 results of the SPA. This shows how closely we pulled the best results toward the top.

First, interesting to note is that the experiments support the hypothesis that a symmetric method performs better than a unidirectional method: SPA-basic outperformed SPA-single in both table 3 and table 5.

Second, table 5 and table 6 show the performance of SPA on French-English data. Here we observe that each penalty (length, distance, anchor and untrans-

Aligner	Recall	Prec.	$F_1$	Len(M)/Len(H)
SPA-single	0.986516	0.473891	0.640234	2.081735
SPA-basic	0.940464	0.620070	0.747377	1.516706
SPA-anchor	0.898028	0.674689	0.770500	1.331025
SPA-len(7)	0.888879	0.764484	0.822002	1.162718
SPA-dist(10)	0.947297	0.611117	0.742947	1.550108
SPA-l-u	0.876627	0.811159	<b>0.842624</b>	1.080709
SPA-a-l	0.862071	0.772312	0.814727	1.116220
SPA-a-d	0.903595	0.669246	0.768962	1.350167
SPA-l-d	0.888913	0.755690	0.816906	1.176293
SPA-a-l-d	0.861444	0.767694	0.811871	1.122118
SPA-a-l-d-u	0.857907	0.780515	0.817383	1.099155
COMB	0.763866	0.818049	0.790029	0.933766

Table 4: English-Chinese: Top 10 alignment results evaluation

lated word) helped SPA individually, and that in fact the highest score was obtained when all 4 were applied simultaneously. We think this was possible because French and English are very similar languages in the sentence structures.

Third, table 3 and table 4 show the performance of SPA on English-Chinese data. Here we observe that only two of the penalties (length, untranslated word) helped individually and the highest overall score was obtained when those two are applied simultaneously. We think this is because English and Chinese are very different languages in their sentence structures.

Fourth, in table 3, both IBM4 aligners and SPA aligners outperformed the baseline significantly. We evaluated IBM4 results in three ways: regarding the whole part between the smallest and the largest positions as a contiguous answer fragment ("IBM-cont"), regarding its best possible contiguous fragment as a contiguous answer fragment ("IBM-oracle") and considering it as it is ("IBM"). Overall the IBM Model 4 aligner showed the best performance and SPA-l-u approached to IBM-oracle. This means, for the contiguous alignment, IBM-oracle and SPA-l-u have almost the same performance. The best SPA in table 4 is better than IBM4 in table 3 which means that we have a room for SPA improvement and after the improvement, it is possible that it outperforms IBM4.

Finally, in table 5, both IBM4 and SPA also much outperformed the baseline. Here IBM4 results are much better than the best of SPA which is SPA-a-l-d-u but in table 6 we still have a room for SPA improvement and a possibility to outperform IBM. We need to investigate the SPA results to rank the best results in top 10 the highest.

Table 7 shows the results of the combined aligner ("COMB") for both language pairs, and the COMB outperforms IBM. We used  $e^{-11}$  and  $e^{-12}$  for French-English and English-Chinese thresholds respectively in probability space. (In our actual implementation, we use log space instead of probability space for

Aligner	Recall	Prec.	$F_1$	Len(M)/Len(H)
random	0.193939	0.238384	0.213877	0.813558
positional	0.668841	0.728991	0.697622	0.917489
oracle	0.980509	0.937717	0.958636	1.045634
SPA-single	0.880979	0.281680	0.426874	3.127586
SPA-basic	0.707808	0.712078	0.709936	0.994003
SPA-anchor	0.779839	0.672166	0.722010	1.160188
SPA-len(4)	0.699386	0.748196	0.722968	0.934764
SPA-dist(4)	0.770683	0.728987	0.749256	1.057198
SPA-a-l	0.752165	0.774997	0.763410	0.970540
SPA-a-d	0.810575	0.709604	0.756736	1.142292
SPA-l-d	0.752070	0.788768	0.769982	0.953473
SPA-a-l-d	0.783109	0.799521	0.791230	0.979472
SPA-a-l-d-u	0.781466	0.801407	<i>0.791311</i>	0.975118
IBM4-cont	0.852763	0.829349	0.840893	1.028232
IBM4-oracle	0.813175	0.914574	0.860899	0.889130
IBM4	0.777064	0.965592	<b>0.861130</b>	0.804753
COMB	0.781734	0.960679	0.862018	0.813731

Table 5: French-English: Best alignment results evaluation

computation efficiency and for this reason, we have such values as thresholds. And these values were obtained empirically.) Our significance test shows that for English-Chinese, the combined version significantly outperform IBM4 while for French-English, the difference is not very significant.

### 3.3.2 EBMT Performance

After separately and roughly tuning several key parameters in the EBMT system for each alignment algorithm in use, we obtained the scores shown in Table 8.

In the table, we observe SPA outperformed EBMT - the old aligner in the CMU EBMT system by a huge difference. For the Devtest, 2refTest and Test data set, SPA has 35%, 20% and 28% higher BLEU scores than EBMT, which is huge improvement.

We also observe that IBM4 and COMB significantly outperformed the EBMT but the performance differences among SPA, IBM4 and COMB are very small. For Devtest, 2refTest and Test the winners are different - COMB for Devtest, SPA for 2refTest and IBM4 for Test.

Our significance test shows that SPA, IBM4 and COMB perform significantly better than EBMT, but that the differences among SPA, IBM4 and COMB are not significant.

One thing interesting here is that for DevTest and 2refTest, the BLEU scores of translations using word level alignment of IBM4-W and COMB-W are better than those of translations using word and fragment level alignment of IBM4 and COMB. We suspect that this problem is caused because our current EBMT

Aligner	Recall	Prec.	$F_1$	Len(M)/Len(H)
SPA-single	0.968004	0.346001	0.509786	2.797687
SPA-basic	0.903788	0.820894	0.860349	1.100981
SPA-anchor	0.929404	0.843171	0.884190	1.102273
SPA-len(4)	0.882192	0.875378	0.878772	1.007784
SPA-dist(4)	0.938193	0.833794	0.882918	1.125209
SPA-a-l	0.909626	0.902600	0.906100	1.007784
SPA-a-d	0.943185	0.857410	0.898255	1.100040
SPA-l-d	0.915910	0.888186	0.901835	1.031215
SPA-a-l-d	0.923072	0.904490	0.913687	1.020544
SPA-a-l-d-u	0.922897	0.905394	<b>0.914061</b>	1.019332
COMB	0.794539	0.965081	0.871545	0.823288

Table 6: French-English: Top 10 alignment results evaluation

Test-set-id	COMB(en-cn)	IBM4(en-cn)	COMB(fr-en)	IBM4(fr-en)
0	0.650242	0.622334	0.873710	0.872495
1	0.750254	0.740113	0.869625	0.871524
2	0.741334	0.734779	0.823990	0.818237
3	0.738612	0.733213	0.877581	0.876985
4	0.787895	0.783484	0.899519	0.903400
5	0.836290	0.832842	0.865934	0.863540
6	0.793645	0.786877	0.816366	0.816886
7	0.796416	0.795579	0.820094	0.815675
8	0.768600	0.759885	0.890586	0.895273
9	0.804756	0.801179	0.868291	0.863771
P value	0.01		0.5	

Table 7: The significance test for COMB and IBM4

system doesn't take full advantage of non-contiguous alignment since we see this problem only with IBM4 and COMB which are non-contiguous aligners. This problem requires further investigation.

## 4 Conclusions

### 4.1 Conclusions

By the translation experiments, we have shown that the SPA improved the CMU EBMT system. When we compared the BLEU scores of the EBMT system using the SPA and the old aligner, the SPA significantly outperformed the old aligner for the development data set, the two reference test set and the test set. Our significance test showed that the performance difference between the SPA and

	<b>Devtest</b>	<b>2refTest</b>	<b>Test</b>
EBMT	0.1632	0.2400	0.13455
SPA	0.2214	<b>0.2896</b>	0.17287
IBM4	0.2197	0.2785	0.17549
COMB	<b>0.2240</b>	0.2815	0.17506
SPA-W	0.2153	0.2885	0.17219
IBM4-W	0.2180	0.2821	0.17826
COMB-W	0.2230	0.2884	<b>0.17983</b>

Table 8: French-English BLEU scores by aligners

the old aligner is very significant.

Through the CMU EBMT system, we also compared the SPA, IBM Model 4 and a combined aligner of the previous two. Our significance test showed that the differences among them are not significant.

We also measured and compared alignment accuracy for the aligners. We prepared human aligned data for English-Chinese and French-English pairs and measured precision, recall and  $F_1$  of alignment results on human alignment results. We had random, positional and oracle aligners and the accuracy order for them was random < positional < oracle. The SPA lay between positional and unidirectional alignment lay between the SPA and oracle. These results showed that a bidirectional score calculation method worked better than unidirectional method.

In the alignment accuracy evaluation, the SPA itself couldn't beat IBM4 but we saw two possibilities for that. First, when we picked the best results of top 10's from the SPA results, the SPA significantly outperformed IBM4 results and showed very close performance to oracle. Second, when we combined IBM4 results and the SPA results over a threshold, it outperformed IBM4. The performance difference in English-Chinese pair was very significant while the difference in French-English pair was not.

And we also showed a clue about how reliable the alignment accuracy numbers are. We had three human result sets for the same data set and evaluated each on the others. This showed that human aligners agree on alignments by 84.6% to 96.5%.

## 4.2 Future Work

In section 2.7, we explained how we will merge a human dictionary with an automatically generated statistical dictionary. We will try to make our EBMT system have higher data coverage in translations and work better on more general data by dictionary merge.

In section 2.8, we described a possible approach to a non-contiguous SPA. Since the assumption that the target fragments are contiguous is not realistic, we are going to modify our algorithm to produce non-contiguous target fragments.

In order to take full advantage of non-contiguous alignment, we recently updated our EBMT system to work with non-contiguous alignment results after our experiments for this thesis.

To improve our dictionary and alignment together, we will train them together iteratively with an EM-like method. Since the SPA is recently ready to use a phrasal dictionary, we will produce a phrasal dictionary from the SPA results, and use this dictionary for the SPA again. Repeating these two processes will lead to a better dictionary and alignment results. We will come up with an EM-like training method to achieve this.

The corpus used for our experiments was fairly small, in large part due to the computational expense of running the IBM Model 4 and the SPA – the SPA can produce word-level and phrasal alignments for approximately 10,000 sentence pairs per hour. We intend to repeat the experiments with a larger training corpus.

We have seen that the performance of aligners is very different according to language pairs. So intend to have experiments on various language pairs to make the SPA perform consistently well.

We also need to make the SPA faster. Since current SPA source code is experimental, it didn't have full algorithmic optimizations. Followings are obvious things to consider first. First, because considering all the contiguous target candidate fragments takes a lot of time, we need to shrink the candidate set based on linguistic knowledge or statistics. Second, we need to modify the SPA to compare words in numeric form.

Finally, since we haven't optimized the parameters for both the CMU EBMT system and GIZA++, we need to do that to get more reliable results.

## 5 Acknowledgment

I want to represent my special thanks to the CMU EBMT group members. Jaime Carbonell is my advisor and initiated the idea of the SPA, Peter Jansen also gave me a lot of advice and described the SPA, Ralf Brown who wrote and maintains the EBMT system gave me a lot of help in integrating SPA to the EBMT, and finally, Violetta Cavalli-Sforza also gave me a lot of useful advice and comments.

I also thank my wife Cathy and my daughter Janise who supported me to concentrate on and to finish this work.

## References

- [1] American and French Research on the Treasury of the French Language, ARTFL. Artfl project: Frenchenglish dictionary form, 2005. [http://humanities.uchicago.edu/orgs/ARTFL/forms\\_unrest/FR-ENG.html](http://humanities.uchicago.edu/orgs/ARTFL/forms_unrest/FR-ENG.html).

- [2] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer. The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, 19(2), 1993.
- [3] Ralf D. Brown. Generalized example-based machine translation: 7. word-level alignment, 2000. <http://www-2.cs.cmu.edu/~ralf/ebmt.html>.
- [4] Lee R. Dice. Measures of the amount of ecologic association between species. *Journal of Ecology*, 26:297–302, 1945.
- [5] Jae Dong Kim, Ralf D. Brown, Peter J. Jansen, and Jaime G. Carbonell. Symmetric Probabilistic Alignment for Example-Based Translation. In *Proceedings of the Tenth Workshop of the European Association for Machine Translation (EAMT-05)*, pages 153–159, May 2005.
- [6] I. Dan. Melamed. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249, 2000.
- [7] F. J. Och and H. Ney. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, pages 440–447, Hongkong, China, October 2000.
- [8] K. Papineni, S. Roukos, T. Ward, and W. Zhu. BLEU: A method for automatic evaluation of machine translation. In *ACL '02*, 2002.
- [9] Smadja, Frank, Kathleen R. Mckeown, and Vasileios Hatzivassiloglou. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38, 1996.
- [10] TravLang. French-english online dictionary, 2005. <http://dictionaries.travlang.com/FrenchEnglish/>.
- [11] S. Vogel, H. Ney, and C. Tillmann. Hmm-based word alignment in statistical translation. In *COLING '96: The 16th International Conference on Computational Linguistics*, pages pp. 836–841. ACL'96, 1996.
- [12] K. Yamada and K. Knight. A decoder for syntax-based statistical MT. In *ACL '02*, 2002.