Exploring Translation Similarities for Building a Better Sentence Aligner

Anil Kumar Singh and Samar Husain

Language Technologies Research Centre
International Institute of Information Technology
Hyderabad, India
anil@research.iiit.net, samar@research.iiit.net

Abstract. The approaches previously used for sentence alignment (sentence length, word correspondence and cognate matching) take into account different aspects of similarity between the source and the target language sentences. In this paper we discuss various aspects of similarity in translated texts that can be used for sentence alignment. We then describe a customizable method for combining several approaches that can exploit these aspects of similarity. This method also includes a novel way of using sentence length for alignment. Finally, the results of evaluation for this composite method (overall and at various stages) are presented. These results are compared with those for some previous approaches.

1 Introduction

A parallel corpus is a collection of sentences in two languages. The target language (TL) sentences are translations of the source language (TL) sentences. Sentence alignment means finding out which TL sentence is the translation of which SL sentence. An aligned parallel corpus is a collection of such pairs of sentences.

The translation of a sentence can be virtually anything at the surface (lexical and syntactic) level. Even at the semantic level, there may be significant differences. What we consider to be a translation may be an adaptation. To make the problem harder, there may be insertions, deletions, contractions, and expansions.

Many sentence alignment algorithms are available, based on different approaches. Most of them take into account one aspect of the translation process. For example, sentence length based methods rely on the similarity in the lengths of SL and TL sentences. Word correspondence based methods utilize the similarity in distribution of those SL and TL words which are translations of each other. Some methods are also based on a combination of two or more aspects of similarity [12].

There has been so much work on sentence alignment that the problem is often considered as solved. However, as shown by Singh and Husain [15], there is scope for improving the performance of sentence alignment, especially for real corpora which are noisy. We believe that it is possible to build a sentence alignment tool which combines several different approaches. Such a tool can take into account many aspects of the translation process (and similarity in SL and TL sentences). This paper discusses these aspects of similarity (sections 3 and 4) and then presents an attempt to build a customizable tool which tries to exploit many of these aspects. We present this attempt as an exploration in this direction rather than a complete solution.

The tool combines sentence length, word correspondence and cognate matching with some other approaches such as common word count, synonym intersection, and hypernym intersection (see section-4). Sentence length based approach is taken as the base, but we use many different customizable weighted sentence lengths (WSLs). These WSLs are weighted sums of word count, character count and *signature* (see section-4.1.1) of a sentence.

All the techniques used give alternative alignments, which are merged at different stages (section-6). The tool makes use of language resources such as bilingual dictionary and the WordNet, but it can work without them if they are not available for a particular language pair. This is because the results of alignment are available at various stages and the user can specify which stages are to be used.

2   Previous Work

Perhaps the best known methods of alignment are those based on sentence length. They rely on the intuition that the length of an SL sentence is likely to be similar to the length of its corresponding TL sentence. Length can be taken as character count [8] or word count [3]. Prior segmentation, either as alignment of paragraphs [8], or in the form of some already aligned sentences or anchors [3] was required by these methods as initially proposed. A modified version of Gale and Church [8] was used by Wu [17] for Chinese.

In one of the earliest attempts at sentence alignment, Kay and Roscheisen [9, 10] used word correspondence, which was based on the similarity of distributions of words in the two texts. IBM Model-1 developed by Brown et al [4] used word correspondence from the point of view of statistical machine translation. The method adopted by Melamed [11] was also based on word correspondence, but the correspondence was in terms of cognate words.

Some 'composite' algorithms (those which combine two or more methods) were also tried. In Simard's method [16], the first pass is based on alignment at the character level as in Church [6], which itself exploits the idea of cognates as proposed by Simard [14]. The second pass, following Chen [5], uses a statistical translation model (IBM Model-1). These two passes are said to combine the robustness of character based alignment with the accuracy of stochastic translation models.

Moore [12] combined sentence length and word correspondence (IBM Model-1) in two passes to come up with an algorithm that has the benefits of unsupervised learning. Moreover, it doesn't use any linguistic information or prior segmentation.

A different approach based on chunk matching was used by Bharati et al. [1]. But this method was too slow to be used for practical purposes and had the disadvantage of being completely dependent on language resources.

3 Similarities in Translations

There are several aspects of the translation process (and correspondingly of similarity) between SL and TL texts. We believe that under different conditions and for different language pairs, one or more of these may become more suitable for identifying translations, and therefore, also for sentence alignment. Thus, a robust and adaptable sentence alignment tool should exploit as many of these aspects of similarity as is practically possible.

If there are two (written) sentences in two different languages, one of which is a translation of the other, then the similarities between the two may be due to the following factors:

1. **Symbolic**
   (a) Common symbols, say, for numbers (Indian/Arabic numerals)
   (b) Proper names
   (c) Borrowed words
   (d) Phonetic and lexical correspondence
2. **Meta-syntactic**, one simple measure of which may be the number of verbs (or nouns)
3. **Syntactic**, such as the order or structure of syntactic constituents
4. **Meta-semantic**, such as the quantity of information, one simple measure of which is the sentence length
5. **Semantic**, i.e., the common meaning and the world knowledge

Sentence length based methods use meta-semantic similarity. Word correspondence based methods use symbolic similarity. Cognate matching also uses symbolic similarity. The method used by Bharati et al. [1] uses symbolic and syntactic similarity.

Symbols used in SL and TL may be different, but if we can find the correspondence between symbols, we might identify translations, or even do translation automatically by combining symbolic correspondence with syntactic correspondence. This is what the IBM models [2, 4] do.

Of course, the similarity that ultimately matters, as far as the results are concerned, is the semantic similarity, but perhaps it doesn't need to be considered at a deeper level for sentence alignment. Anyway, to consider it fully will mean solving all the NLP problems. And as our results show, it is very difficult to exploit this aspect of similarity because of the limitations of language resources, as well as due to ambiguity of words.

Our method tries to take into account all these similarities, except syntactic or meta-syntactic. But if POS taggers are available for both SL and TL, we expect that the number of nouns (or verbs) can, in fact, be used in a way similar to sentence length.

Semantic correspondence may not be reflected in the symbols, but we can try to infer such correspondence provided that a bilingual dictionary and a WordNet like resource are available.

4 Capturing Similarities for Sentence Alignment

In trying to capture different kinds of similarities in SL and TL texts, we have used several techniques. Each of these is explained in the following sub-sections.

4.1 Sentence Length

Instead of using a single measure of sentence length, we use a number of customizable weighted sentence lengths (WSLs). For calculating WSLs, we use three measures of sentence length. The first two are word count [3] and character count [8]. The third one is what we call the *signature* of a sentence.

Sentence Signature Sentence signature can be calculated by using a *signature function*. We use the sum of ASCII values of all the characters in a sentence as the signature function. This is, of course, the simplest hash function. In fact, both word count and character count can be seen as signatures of a sentence using different signature functions. For languages like Chinese, number of bytes can be seen as the sentence signature [7]. We can, thus, view sentence length as a special case of sentence signature.

If a bilingual dictionary is available, we can use a better form of signature, which we call signature with *substitution*. During substitution, the TL sentence is left as it is, but all the words in the SL sentence for which a TL meaning can be found are substituted by their meanings (only for calculating signatures). The idea is that after substitution, there will be many similar words in the SL and TL sentences if they are translations of each other. This is expected to bring the values of signatures of SL and TL sentences closer and increase the chances of their being aligned.

We did some experiments on the correlation between SL and TL word count, character count and signature. The experiment was done on a manually checked parallel corpus of 7000 sentences. The results showed that for English and Hindi, the correlation for signature (0.814) was almost equal to that for character count (0.816), and was better than for word count (0.783). Thus, signature as sum of ASCII values may not be better than character count, but it might be used as an additional feature which indicates the meta-semantic similarity between a sentence and its translation. It may be noted here that the signature without substitution (which doesn't need any language resources) still has a correlation

(0.767) only slightly less than that for word count. Thus, in the absence of language resources, signature without substitution can be used as the third measure of sentence length.

By combining these three measures of sentence length with weights, we can get a better measure. Moreover, we can use these weights for tuning the alignment tool.

Perhaps a better signature function can be found which makes use of other information like POS tags, etc. This is something we have not explored so far.

Weighted Sentence Length (WSL) WSL can be defined in terms of word count (wc), character count (cc), and signature (sig) as given below (w being the weight):

$$l = w_{wc}l_{wc} + w_{cc}l_{cc} + w_{sig}l_{sig} \tag{1}$$

such that:

$$w_{wc} + w_{cc} + w_{sig} = 1 \tag{2}$$

Note that each of l_{wc}, l_{cc}, and l_{sig} are normalized values. For example, to calculate l_{wc} of an SL sentence, we divide it by the length of the largest SL sentence, so that its value is between 0 and 1.

We assume that WSL will follow a Poisson distribution. The probability of a target sentence having a particular length, given the length of the source sentence, will then be [12]:

$$P(l_t|l_s) = exp(-l_s r)(l_s r)^{l_t}/(l_t)! \tag{3}$$

4.2   IBM Model-1 Based Word Correspondence (IWC)

Word Correspondence is based on the similarity in distribution of SL and TL words which are translations of each other. This approach was used by Moore [12] in the second pass of his algorithm. The training for IBM Model-1 [4] is provided by the tentative alignments obtained from the first pass.

4.3   Numeric, Phonetic, and Cognate Matching (NPC)

These will take care of sentences which have numbers, proper names and cognates or borrowed words. For NPC matching, words (or numbers) in SL and TL sentences are first compared directly, and then after normalization. Normalization means removing vowels and performing some string transformations to take care of the way proper nouns and cognates are written in SL and TL. The comparison itself is done in two stages. The first is simple string matching, and the second is using a dynamic time warping (DTW) algorithm [13].

4.4 Common Word Count (CWC)

It is reasonable to assume that some words will be translated literally. Using a bilingual dictionary, we can count such words. We do this after substitution, as for signature. Two bags of words are formed, one each for SL and TL, one with and one without substitution, respectively. The intersection of these two bags gives us the common word count. We get a normalized score for CWC on dividing this count by the length of the SL sentence. Of course, the performance of all the techniques which use a bilingual dictionary is limited by the dictionary coverage as well as by ambiguity of words.

4.5 Synonym Intersection (SNI)

A sentence and its translation may not be similar on the surface, but they will have some similarity at an abstract level. Synonyms from WordNet or a thesaurus can be used to measure such similarity. For doing this, we again form two bags. The bags in this case contain synonyms instead of words themselves. In our case, WordNet was available for the SL but not for the TL, and the bilingual dictionary is SL-TL. Therefore, to find synonym intersection, we do *reverse substitution*. This means that we leave the SL sentence as it is, but substitute all the words in the TL sentence with their SL equivalents. An SL-TL pair is likely to be a correct alignment if its synonym intersection is large. The score is again calculated by dividing this count by the length of the SL sentence.

4.6 Hypernym Intersection (HPI)

This is very similar to synonym intersection, except that we take hypernyms instead of synonyms. WordNet gives hypernyms at several levels, but the upper (general) levels are likely to be common even for words which are not related from the point of view of translation. Therefore, we take hypernyms from only the two lower (specific) levels.

5 'Compiling' or Preprocessing the Corpora

The SL and TL corpora are first 'compiled' or preprocessed. During compilation of the corpora, some tables are prepared. The basic units of the compiled corpora are Word Types. A Word Type contains the word string, the POS tag (if POS tagger is available), the index of the equivalent word in the other language (into the other word type table), the word signature, flags indicating whether the word is a corpus word or a substitute word or a synonym/hypernym word, or a combination of these. Some other information relevant to specific techniques may be optionally stored in Word Types. There is also an index for easy access to an entry.

For the corpus for which WordNet is available (SL in our case), there are Extended Word Types which also have synonym and hypernym indices. All

the words, including corpus words, substituted words, and hypernym words are stored as Word Types and accessed through indices in the tables. There is one Word Type Table each for SL and TL. The TL table contains Word Types, whereas the SL table contains Extended Word Types.

Sentences are formed in terms of indices into the word type tables. A sentence is a sequence of indices. Word count, character count, signature, etc. are stored for each sentence.

6   The Composite Algorithm

The basic framework of the composite algorithm is similar to Moore's [12]. There are two passes, DP based beam search with widening diagonal band, search pruning, sentence length method in the first pass, IBM Model-1 in the second pass, Poisson distribution for sentence length instead of Gaussian, and combination of the results of the two passes for the final alignment. However, our algorithm differs from Moore's in many ways since it uses many WSLs (instead of just word count) in the first pass and four other techniques in the second pass.

6.1   The First Pass

Moore used word count based alignment in the first pass. A probability cutoff of 0.99 was used by him to select tentative alignments out of the candidate pairs (or beads). The probabilities were calculated using equation-3.

In our algorithm, instead of relying on just the word count, we use a customizable number of weighted sentence lengths (WSLs). These are calculated using equation-1. The probabilities are still calculated using equation-3, but are based on WSLs, instead of word count. The number of WSLs and the weights for each of them can be specified by the user. We used 13 WSLs and a probability cutoff of 0.95.

All the WSLs give one possible alignment each (after applying the probability cutoff). Thus, if the user has specified n_{sl} number of WSLs, then we get n_{sl} possible alignment. These alignments are then merged, based on the criterion that only those aligned beads will be accepted which occur in a majority (but at least one third) of the alignments. For cases of conflict, a priority list is provided by the user. Thus, if the ninth SL sentence aligns with the tenth TL sentence according to three WSLs, but with the eleventh according to the rest of the four, then the latter will be accepted. But if three WSLs each support two choices (the seventh not having given any alignment for the ninth SL sentence), then the one that has higher priority WSLs will be accepted. If only one WSL aligned the ninth SL sentence with the tenth TL sentence, and the rest of the WSLs didn't align it with any TL sentence, then the merged WSL alignment will not have an entry for the ninth SL sentence. The merged alignment may be called the WSL or *meta-semantic alignment* and will be the output of the first pass.

Table 1. Results for Various Corpus Types (Corpus Size = 2500)

Type		Clean, Same Size				Noisy, Same Size				Noisy, Different Size			
		Wsl	Npc	Cwc	Sem	Wsl	Npc	Cwc	Sem	Wsl	Npc	Cwc	Sem
EMILLE	P	99.9	99.8	99.7	99.2	96.3	95.8	95.7	93.1	96.3	95.7	95.5	92.8
	R	93.2	93.5	93.9	93.5	83.5	84.1	84.8	84.1	81.7	82.0	82.8	82.0
	F	96.4	96.6	96.7	96.3	89.4	89.6	89.9	88.3	88.4	88.3	88.7	87.0
ERDC	P	99.9	99.9	99.9	99.9	94.7	94.6	94.3	91.6	95.4	95.2	95.2	92.5
	R	98.8	98.8	98.8	98.8	80.3	81.0	81.4	81.0	79.8	80.7	81.3	80.7
	F	99.3	99.4	99.4	99.4	86.9	87.3	87.4	86.0	86.9	87.4	87.7	86.2
India Today	P	99.1	96.4	96.4	95.3	92.0	87.9	88.0	85.1	92.4	88.6	88.6	86.1
	R	86.1	88.6	88.8	88.6	72.0	75.8	76.2	75.8	71.8	76.5	76.7	76.5
	F	92.2	92.3	92.5	91.8	80.8	81.4	81.7	80.2	80.8	82.1	82.2	81.0
P: Precision, *R:* Recall, *F:* F-Measure													

Table 2. Results for Various Corpus Sizes

Size		Clean, Same Size				Noisy, Same Size				Noisy, Different Size			
		Wsl	Npc	Cwc	Sem	Wsl	Npc	Cwc	Sem	Wsl	Npc	Cwc	Sem
500	P	99.8	99.8	99.8	99.4	94.1	94.2	93.6	88.9	93.8	93.9	93.7	88.43
	R	97.8	97.8	97.8	97.8	70.8	72.0	73.0	72.0	72.2	73.4	74.4	73.4
	F	98.8	98.8	98.8	98.6	80.8	81.6	82.0	79.6	81.6	82.4	82.9	80.2
1000	P	99.9	99.9	99.9	99.8	94.8	94.9	94.6	91.2	94.0	94.0	94.0	90.4
	R	99.2	99.2	99.2	99.2	76.9	77.7	78.3	77.7	76.1	77.4	77.9	77.4
	F	99.5	99.5	99.5	99.5	84.9	85.4	85.7	83.9	84.1	84.9	85.2	83.4
5000	P	100.0	100.0	100.0	99.9	96.4	96.3	96.0	93.6	96.0	95.9	95.8	94.0
	R	99.2	99.3	99.4	99.3	84.5	84.9	85.2	84.9	85.7	86.0	86.3	86.0
	F	99.7	99.7	99.7	99.6	90.0	90.2	90.3	89.0	90.6	90.7	90.8	89.8

6.2 The Second Pass

The tentative alignments obtained from the first pass are used for training the IBM model-1. Word correspondence based alignment is then done, as in Moore. The probability cutoff used in the second pass is 0.7 (Moore had used 0.5). In addition, we also use four other techniques in this pass, namely numeric, phonetic, and cognate matching (NPC), common word count (CWC), synonym intersection (SNI), and hypernym intersection (HPI). The cutoffs for scores obtained from these techniques are 1.0, 0.3, 5.0, and 7.0, respectively.

Thus, we get a total of five separate alignments in the second pass. The results of IBM model-1 based word correspondence are given the highest priority. The alignments given by NPC are then merged into it.

There are two divergent paths from here. In the first, CWC alignments are merged into the one obtained with word correspondence and NPC.

In the second, CWC, SNI, and HPI alignments are first merged together to give what we can call semantic alignments. This result is then merged with the one obtained with word correspondence and NPC. Note that the results of techniques which come later are given less priority while merging. For example, if

Table 3. Global Evaluation Measures: Comparison with Previous Approaches

		Previous Approaches				Composite Method			
		Brn	**GC**	**Mmd**	**Mre**	**Wsl**	**Npc**	**Cwc**	**Sem**
Clean,	L	92.6	93.4	81.4	80.8	96.8	96.9	97.0	96.7
Same Size	H	100.0	100.0	96.3	100.0	98.5	98.5	98.6	98.4
	P	98.4	98.7	90.3	95.1	99.8	99.3	99.3	98.9
	R	96.1	96.1	87.6	90.0	95.7	96.2	96.3	96.2
	F	97.2	97.3	88.9	92.4	97.7	97.7	97.8	97.5
Noisy,	L	73.1	75.8	44.1	72.6	84.4	84.9	85.1	83.4
Same Size	H	87.5	86.4	62.4	92.3	86.6	87.0	87.2	85.6
	P	82.7	84.1	53.8	92.2	94.7	94.0	93.7	90.6
	R	77.4	78.4	52.8	74.9	78.0	79.3	79.8	79.3
	F	79.8	81.1	53.3	82.5	85.5	85.9	86.2	84.5
Noisy,	L	74.7	76.4	46.2	71.3	84.3	85.0	85.3	83.6
Different	H	85.6	86.4	55.0	92.0	86.5	86.9	87.2	85.7
Size	P	83.4	84.9	51.2	91.5	94.6	94.0	93.8	90.7
	R	77.2	78.3	50.0	74.0	77.9	79.4	79.9	79.3
	F	80.1	81.4	50.6	81.6	85.4	86.0	86.3	84.6
Overall	L	81.1	82.4	55.4	80.0	86.6	87.1	87.3	85.8
	H	90.4	90.8	73.1	91.0	92.4	92.6	92.8	91.9
	P	88.2	89.2	65.1	92.9	96.4	95.7	95.6	93.4
	R	83.6	84.3	63.5	79.6	83.9	84.9	85.4	84.9
	F	85.7	86.6	64.6	85.5	89.5	89.9	90.1	88.9
L and *H:* Lower and higher limits of 95% confidence interval for F-measure									
P, *R*, and *F:* Average precision, recall, and F-measure									

conflicting alignments are given for a particular SL sentence by word correspondence plus NPC and CWC, then the one given by the former will be accepted. Since all the results at various stages are stored, the user can select any one of them for actual alignment, depending upon the performance at each stage during evaluation (ours or his own), or on the availability of language resources. The one which should be selected also depends on the trade-off between precision and recall, as the evaluation given later shows.

We use a pruned search for the four new techniques used in the second pass. This search uses the already available alignments given by the word correspondence method as the basic search path. Our search is within a certain window size from this path. For example, suppose that we are searching for the translation of the ninth SL sentence. Then we look into the alignment list given by word correspondence and find the entries at a certain distance (2 in our case) before and after the ninth sentence. Search is performed between the corresponding TL sentences for these two entries and TL sentence with the best score is selected.

7 Evaluation

7.1 Evaluation Data Sets

We have used the same evaluation scheme as used by Singh and Husain [15]. We evaluated our method on 18 different data sets consisting of three corpus types (EMILLE, ERDC, and India Today) and three corpus sizes (500, 1000, 5000). For each corpus type and size, there were three data sets: without noise, with noise (same SL and TL size), and with noise (different SL and TL size). The noise was 10% in the second case, and 5% (to SL) and 15% (to TL) in the third case. The measures of performance are also the same as used by Singh and Husain, i.e., local (precision, recall, and F-measure) and global measures (average precision, average recall, average F-measure, and confidence interval for F-measure).

7.2 Results

The results (tables 1 to 3) are presented for four stages of the composite method, all from the second pass:

- **Wsl**: Weighted sentence length plus word correspondence
- **Npc**: **Wsl** plus numeric, phonetic, and cognate matching
- **Cwc**: **Npc** plus common word count
- **Sem**: **Npc** plus semantic matching (common word count, common synonyms, and common hypernyms)

Global measures are compared with the results for four previous approaches as given in Singh and Husain:

- **Brn**: Word count, but with Poisson distribution
- **GC**: Character count, but with Poisson distribution
- **Mmd**: Melamed's geometric correspondence
- **Mre**: Moore's two pass method

On the whole, the results show that the first three stages (**Wsl**, **Npc**, and **Cwc**) led to an improvement over all the other approaches. However, in many cases, the precision decreased slightly as compared to Moore's method, though recall and F-measure increased.

Contrary to what we were expecting on the basis of our initial experiments, this systematic evaluation showed that synonym intersection and hypernym intersection decreased the performance, instead of increasing it.

8 Conclusion

In this paper, we first discussed various aspects of similarity in translated texts that can be used for sentence alignment. Then we described a method for combining many of these aspects in one alignment tool. A systematic evaluation was performed for this method and the results were presented. These results show an overall improvement over previous approaches.

8.1 Future Work

We will further tune the parameters (number of weighted sentence lengths and weights for each of them, as well various cutoffs) to improve the performance. We will also try to add meta-syntactic matching in terms of number of verbs, nouns, etc.

References

1. Akshar Bharati, V. Sriram, A. Vamshi Krishna, Rajeev Sangal, and Sushma Bendre. An algorithm for aligning sentences in bilingual corpora using lexical information. In *Proceedings of ICON-2002*, Mumbai, India, 2002.
2. Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. In *Computational Linguistics*, 1990.
3. Peter F. Brown, J. C. Lai, and Robert L. Mercer. Aligning sentences in parallel corpora. In *Proceedings of 29th Annual Meeting of the Association for Computational Linguistics*, pages 169–176, Berkeley, CA., 1991.
4. Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. Mathematics of statistical machine translation: Parameter estimation. In *Computational Linguistics*, pages 19(2):263–311, 1993.
5. Stanley F. Chen. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 9–16, Columbus, OH, 1993.
6. Kenneth W. Church. Char_align: A program for aligning parallel texts at the character level. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 1–8, Columbus, OH, 1993.
7. Kenneth W. Church and Patrick Hanks. Aligning parallel texts: Do methods developed for english-french generalize to asian languages? In *Proceedings of Rocling*, 1993b.
8. William A. Gale and Kenneth W. Church. A program for aligning sentences in bilingual corpora. In *Proceedings of 29th Annual Meeting of the Association for Computational Linguistics*, pages 177–184, Berkeley, CA., 1991.
9. Martin Kay. Text-translation alignment. In *ACH/ALLC '91: "Making Connections" Conference Handbook.*, Morristown, NJ, USA, 1991. Association for Computational Linguistics.
10. Martin Kay and Martin Rscheisen. Text-translation alignment. *Computational Linguistics*, 19(1):121–142, 1993.
11. I. Dan Melamed. A geometric approach to mapping bitext correspondence. In *IRCS Technical Report*. University of Pennsylvania, 1996.
12. Robert C. Moore. Fast and accurate sentence alignment of bilingual corpora. In *AMTA '02: Proceedings of the 5th Conference of the Association for Machine Translation in the Americas on Machine Translation: From Research to Real Users*, pages 135–144, London, UK, 2002. Springer-Verlag.
13. C. S. Myers. *A Comparative Performance Study of Several Dynamic Time Warping Algorithms for Speech Recognition*. PhD thesis, M.I.T., Cambridge, MA, Feb. 1980.
14. Michel Simard, George F. Foster, and Pierre Isabelle. Using cognates to align sentences in bilingual corpora. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, Montreal, Canada., 1992.

15. Anil Kumar Singh and Samar Husain. Comparison, selection and use of sentence alignment algorithms for new language pairs. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 99–106, Ann Arbor, Michigan, June 2005b. Association for Computational Linguistics.

16. Machine Translation staff. Bilingual sentence alignment: Balancing robustness and accuracy. *Machine Translation*, 13(1):59–80, 1998.

17. Dekai Wu. Aligning a parallel english-chinese corpus statistically with lexical criteria. In *Proceedings of 32nd Annual Meeting of the Association for Computational Linguistics*, pages 80–87, Las Cruces, NM, 1994.