Comparison, Selection and Use of Sentence Alignment Algorithms for New Language Pairs

Anil Kumar Singh
LTRC, IIIT
Gachibowli, Hyderabad
India - 500019
anil@research.iiit.net

Samar Husain
LTRC, IIIT
Gachibowli, Hyderabad
India - 500019
samar@iiit.net

Abstract

Several algorithms are available for sentence alignment, but there is a lack of systematic evaluation and comparison of these algorithms under different conditions. In most cases, the factors which can significantly affect the performance of a sentence alignment algorithm have not been considered while evaluating. We have used a method for evaluation that can give a better estimate about a sentence alignment algorithm's performance, so that the best one can be selected. We have compared four approaches using this method. These have mostly been tried on European language pairs. We have evaluated manually-checked and validated English-Hindi aligned parallel corpora under different conditions. We also suggest some guidelines on actual alignment.

1 Introduction

Aligned parallel corpora are collections of pairs of sentences where one sentence is a translation of the other. Sentence alignment means identifying which sentence in the target language (TL) is a translation of which one in the source language (SL). Such corpora are useful for statistical NLP, algorithms based on unsupervised learning, automatic creation of resources, and many other applications.

Over the last fifteen years, several algorithms have been proposed for sentence alignment. Their performance as reported is excellent (in most cases not less than 95%, and usually 98 to 99% and above). The evaluation is performed in terms of precision, and sometimes also recall. The figures are given for one or (less frequently) more corpus sizes. While this does give an indication of the performance of an algorithm, the variation in performance under varying conditions has not been considered in most cases. Very little information is given about the conditions under which evaluation was performed. This gives the impression that the algorithm will perform with the reported precision and recall under all conditions.

We have tested several algorithms under different conditions and our results show that the performance of a sentence alignment algorithm varies significantly, depending on the conditions of testing. Based on these results, we propose a method of evaluation that will give a better estimate of the performance of a sentence alignment algorithm and will allow a more meaningful comparison. Our view is that unless this is done, it will not be possible to pick up the best algorithm for certain set of conditions. Those who want to align parallel corpora may end up picking up a less suitable algorithm for their purposes. We have used the proposed method for comparing four algorithms under different conditions. Finally, we also suggest some guidelines for using these algorithms for actual alignment.

2 Sentence Alignment Methods

Sentence alignment approaches can be categorized as based on sentence length, word correspondence, and composite (where more than one approaches are combined), though other techniques, such as cog-

Proceedings of the ACL Workshop on Building and Using Parallel Texts, pages 99–106,
Ann Arbor, June 2005. ©Association for Computational Linguistics, 2005

nate matching (Simard et al., 1992) were also tried. Word correspondence was used by Kay (Kay, 1991; Kay and Roscheisen, 1993). It was based on the idea that words which are translations of each other will have similar distributions in the SL and TL texts. Sentence length methods were based on the intuition that the length of a translated sentence is likely to be similar to that of the source sentence. Brown, Lai and Mercer (Brown et al., 1991) used word count as the sentence length, whereas Gale and Church (Gale and Church, 1991) used character count. Brown, Lai and Mercer assumed prior alignment of paragraphs. Gale and Church relied on some previously aligned sentences as 'anchors'. Wu (Wu, 1994) also used lexical cues from corpus-specific bilingual lexicon for better alignment.

Word correspondence was further developed in IBM Model-1 (Brown et al., 1993) for statistical machine translation. Melamed (Melamed, 1996) also used word correspondence in a different (geometric correspondence) way for sentence alignment. Simard and Plamondon (Simard and Plamondon, 1998) used a composite method in which the first pass does alignment at the level of characters as in (Church, 1993) (itself based on cognate matching) and the second pass uses IBM Model-1, following Chen (Chen, 1993). The method used by Moore (Moore, 2002) also had two passes, the first one being based on sentence length (word count) and the second on IBM Model-1. Composite methods are used so that different approaches can compliment each other.

3 Factors in Performance

As stated above, the performance of a sentence alignment algorithm depends on some identifiable factors. We can even make predictions about whether the performance will increase or decrease. However, as the results given later show, the algorithms don't always behave in a predictable way. For example, one of the algorithms did worse rather than better on an 'easier' corpus. This variation in performance is quite significant and it cannot be ignored for actual alignment (table-1). Some of these factors have been indicated in earlier papers, but these were not taken into account while evaluating, nor were their effects studied.

Translation of a text can be fairly literal or it can be a recreation, with a whole range between these two extremes. Paragraphs and/or sentences can be dropped or added. In actual corpora, there can even be noise (sentences which are not translations at all and may not even be part of the actual text). This can happen due to fact that the texts have been extracted from some other format such as web pages. While translating, sentences can also be merged or split. Thus, the SL and TL corpora may differ in size.

All these factors affect the performance of an algorithm in terms of, say, precision, recall and F-measure. For example, we can expect the performance to worsen if there is an increase in additions, deletions, or noise. And if the texts were translated fairly literally, statistical algorithms are likely to perform better. However, our results show that this does not happen for all the algorithms.

The linguistic distance between SL and TL can also play a role in performance. The simplest measure of this distance is in terms of the distance on the family tree model. Other measures could be the number of cognate words or some measure based on syntactic features. For our purposes, it may not be necessary to have a quantitative measure of linguistic distance. The important point is that for languages that are distant, some algorithms may not perform too well, if they rely on some closeness between languages. For example, an algorithm based on cognates is likely to work better for English-French or English-German than for English-Hindi, because there are fewer cognates for English-Hindi. It won't be without a basis to say that Hindi is more distant from English than is German. English and German belong to the Indo-Germanic branch whereas Hindi belongs to the Indo-Aryan branch. There are many more cognates between English and German than between English and Hindi. Similarly, as compared to French, Hindi is also distant from English in terms of morphology. The *vibhaktis* of Hindi can adversely affect the performance of sentence length (especially word count) as well as word correspondence based algorithms. From the syntactic point of view, Hindi is a comparatively free word order language, but with a preference for the SOV (subject-object-verb) order, whereas English is more of a fixed word order and SVO type language. For sentence length and IBM model-1 based sentence alignment, this doesn't matter since they don't take the word order into account. However, Melamed's algorithm (Melamed, 1996), though it allows 'non-monotonic chains' (thus taking care of some difference in word order), is somewhat sensitive to the word order. As Melamed states, how it will fare with languages with more word variation than English and French is an open question.

Another aspect of the performance which may not seem important from NLP-research point of view, is its speed. Someone who has to use these algorithms for actual alignment of large corpora (say, more than 1000 sentences) will have to realize the importance of speed. Any algorithm which does worse than O(n) is bound to create problems for large sizes. Obviously, an algorithm that can align 5000 sentences in 1 hour is preferable to the one which takes three days, even if the latter is marginally more accurate. Similarly, the one which takes 2 minutes for 100 sentences, but 16 minutes for 200 sentences will be difficult to use for practical purposes. Actual corpora may be as large as a million sentences. As an estimate of the speed, we also give the runtimes for the various runs of all the four algorithms tested.

Some algorithms, like those based on cognate matching, may even be sensitive to the encoding or notation used for the text. One of the algorithms tested (Melamed, 1996) gave worse performance when we used a notation called ITRANS for the Hindi text, instead of the WX-notation.[1]

4 Evaluation in Previous Work

There have been attempts to systematically evaluate and compare word alignment algorithms (Och and Ney, 2003) but, surprisingly, there has been a lack of such evaluation for sentence alignment algorithms. One obvious problem is the lack of manually aligned and checked parallel corpora.

Two cases where a systematic evaluation was performed are the ARCADE project (Langlais et al., 1996) and Simard et al. (Simard et al., 1992). In the ARCADE project, six alignment systems were evaluated on several different text types. Simard et al. performed an evaluation on several corpus types and corpus sizes. They, also compared the performance of several (till then known) algorithms.

In most of the other cases, evaluation was performed on only one corpus type and one corpus size. In some cases, certain other factors were considered, but not very systematically. In other words, there wasn't an attempt to study the effect of various factors described earlier on the performance. In some cases, the size used for testing was too small. One other detail is that size was sometimes mentioned in terms of number of words, not number of sentences.

5 Evaluation Measures

We have used local (for each run) as well as global (over all the runs) measures of performance of an algorithm. These measures are:

- Precision (local and global)
- Recall (local and global)
- F-measure (local and global)
- 95% Confidence interval of F-measure (global)
- Runtime (local)

6 An Evaluation Scheme

Unless sentence alignment is correct, everything else that uses aligned parallel corpora, such as word alignment (for automatically creating bilingual dictionaries) or statistical machine translation will be less reliable. Therefore, it is important that the best algorithm is selected for sentence alignment. This requires that there should be a way to systematically evaluate and compare sentence alignment algorithms.

To take into account the above mentioned factors, we used an evaluation scheme which can give an estimate of the performance under different conditions. Under this scheme, we calculate the measures given in the previous section along the following dimensions:

- Corpus type
- Corpus size
- Difference in sizes of SL and TL corpora
- Noise

[1] In this notation, capitalization roughly means aspiration for consonants and longer length for vowels. In addition, 'w' represents 't' as in French *entre* and 'x' means something similar to 'd' in French *de*, hence the name of the notation.

We are also considering the corpus size as a factor in performance because the second pass in Moore's algorithm is based on IBM Model-1, which needs training. This training is provided at runtime by using the tentative alignments obtained from the first pass (a kind of unsupervised learning). This means that larger corpus sizes (enough training data) are likely to make word correspondence more effective. Even for sentence length methods, corpus size may play a role because they are based on the distribution of the length variable. The distribution assumption (whether Gaussian or Poisson) is likely to be more valid for larger corpus sizes.

The following algorithms/approaches were evaluated:

- **Brn**: Brown's sentence length (word count) based method, but with Poisson distribution
- **GC**: Church and Gale's sentence length (character count) based method, but with Poisson distribution
- **Mmd**: Melamed's geometric correspondence based method
- **Mre**: Moore's two-pass method (word count plus word correspondence)

For **Brn** and **GC** we used our own implementations. For **Mmd** we used the GMA alignment tool and for **Mre** we used Moore's implementation. Only 1-to-1 mappings were extracted from the output for calculating precision, recall and F-measure, since the test sets had only 1-to-1 alignments. English and Hindi stop lists and a bilingual lexicon were also supplied to the GMA tool. The parameter settings for this tool were kept the same as for English-Malay. For **Brn** and **GC**, the search method was based on the one used by Moore, i.e., searching within a growing diagonal band. Using this search method meant that no prior segmentation of the corpora was needed (Moore, 2002), either in terms of aligned paragraphs (Gale and Church, 1991), or some aligned sentences as anchors (Brown et al., 1991).

We would have liked to study the effect of linguistic distance more systematically, but we couldn't get equivalent manually-checked aligned parallel corpora for other pairs of languages. We have to rely on the reported results for other language pairs, but those results, as mentioned before, do not mention the conditions of testing which we are considering for our evaluation and, therefore, cannot be directly compared to our results for English-Hindi. Still, we did an experiment on the English-French test data (447 sentences) for the shared task in NAACL 2003 workshop on parallel texts (see table-1).

For all our experiments, the text in Hindi was in WX-notation.

In the following sub-sections we describe the details of the data sets that were prepared to study the variation in performance due to various factors.

6.1 Corpus Type

Three different types of corpora were used for the same language pair (English-Hindi) and size. These were EMILLE, ERDC and India Today. We took 2500 sentences from each of these, as this was the size of the smallest corpus.

6.1.1 EMILLE

EMILLE corpus was constructed by the EMILLE project (Enabling Minority Language Engineering), Lancaster University, UK, and the Central Institute of Indian Languages (CIIL), Mysore, India. It consists of monolingual, parallel and annotated corpora for fourteen South Asian languages. The parallel corpus part has a text (200000 words) in English and its translations in Hindi, Bengali, Punjabi, Gujarati and Urdu. The text is from many different domains like education, legal, health, social, and consumer markets. The documents are mostly in simple, formal language. The translations are quite literal and, therefore, we expected this corpus to be the 'easiest'.

6.1.2 ERDC

The ERDC corpus was prepared by Electronic Research and Development Centre, NOIDA, India. It also has text in different domains but it is an unaligned parallel corpus. A project is going on to prepare an aligned and manually checked version of this corpus. We have used a part of it that has already been aligned and manually checked. It was our opinion that the translations in this corpus are less literal and should be more difficult for sentence alignment than EMILLE. We used this corpus for studying the effect of corpus size, in addition to corpus type.

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

		Clean, Same Size				Noisy, Same Size				Noisy, Different Size			
Type		**Brn**	**GC**	**Mmd**	**Mre**	**Brn**	**GC**	**Mmd**	**Mre**	**Brn**	**GC**	**Mmd**	**Mre**
EMILLE	P	99.3	99.1	85.0	66.8	85.5	87.4	38.2	66.2	87.2	86.5	48.0	65.5
	R	96.0	93.0	80.0	63.2	80.4	80.0	36.2	58.0	81.2	79.1	46.5	57.4
	F	97.6	96.0	82.0	64.9	82.8	83.5	37.2	61.8	84.0	82.6	47.3	61.2
	T	23	23	261	45	47	44	363	64	25	25	413	47
ERDC	P	99.6	99.5	94.2	100.0	85.4	84.4	48.0	96.5	84.6	85.5	50.9	97.7
	R	99.0	99.1	92.7	97.0	81.7	80.6	46.7	78.9	80.5	81.3	49.8	79.1
	F	99.3	99.3	93.4	98.4	83.5	82.4	47.3	86.8	82.5	83.3	50.3	87.1
	T	31	29	1024	85	92	90	2268	124	55	52	3172	101
India Today	P	91.8	93.9	76.4	99.5	71.5	76.7	49.7	94.4	73.6	75.5	51.7	93.4
	R	81.0	83.0	70.6	81.5	61.0	65.5	47.6	67.5	62.4	64.4	50.1	62.6
	F	86.1	88.1	73.4	89.6	65.8	70.7	48.6	78.7	67.6	69.5	50.9	75.0
	T	32	32	755	91	96	101	2120	159	60	68	987	134
English-French	P	100.0	100.0	100.0	100.0	87.4	87.5	77.2	95.2	91.2	93.3	77.7	96.6
	R	100.0	99.3	100.0	99.3	85.5	84.3	81.7	84.6	83.2	83.7	82.6	83.0
P: Precision, *R:* Recall, *F:* F-Measure, *T:* Runtime (seconds)													

6.1.3 India Today

India Today is a magazine published in both English and Hindi. We used some parallel text collected from the Internet versions of this magazine. It consists of news reports or articles which appeared in both languages. We expected this corpus to be the most difficult because the translations are often more like adaptations. They may even be rewritings of the English reports or articles in Hindi. This corpus had 2500 sentences.

6.2 Corpus Size

To study the effect of corpus size, the sizes used were 500, 1000, 5000 and 10000. All these data sets were from ERDC corpus (which was expected to be neither very easy nor very difficult).

6.3 Noise and Difference in Sizes of SL and TL Corpora

To see the effect of noise and the difference in sizes of SL and TL corpora, we took three cases for each of the corpus types and sizes:

- Same size without noise
- Same size with noise
- Different size with noise

Three different data sets were prepared for each corpus type and for each corpus size. To obtain such data sets from the aligned, manually checked and validated corpora, we added noise to the corpora. The noise was in the form of sentences from some other unrelated corpus. The number of such sentences was 10% each of the corpus size in the second case and 5% to SL and 15% to the TL in the third case. The sentences were added at random positions in the SL and TL corpora and these positions were recorded so that we could automatically calculate precision, recall and F-measure even for data sets with noise, as we did for other data sets. Thus, each algorithm was tested on (3+4)(3) = 21 data sets.

7 A Limitation

One limitation of our work is that we are considering only 1-to-1 alignments. This is partly due to practical constraints, but also because 1-to-1 alignments are the ones that can be most easily and directly used for linguistic analysis as well as machine learning.

Since we had to prepare a large number of data sets of sizes up to 10000 sentences, manual checking was a major constraint. We had four options. The first was to take a raw unaligned corpus and manually align it. This option would have allowed consideration of 1-to-many, many-to-1, or partial

Table 2: Results for Various Corpus Sizes

		Clean, Same Size				Noisy, Same Size				Noisy, Different Size			
Size		**Brn**	**GC**	**Mmd**	**Mre**	**Brn**	**GC**	**Mmd**	**Mre**	**Brn**	**GC**	**Mmd**	**Mre**
500	P	99.2	99.2	93.9	99.8	75.4	78.2	57.4	94.3	83.5	87.2	45.4	92.4
	R	98.8	98.8	91.8	95.0	71.0	73.4	56.8	70.0	77.0	80.8	44.8	70.8
	F	99.0	99.0	92.8	97.3	73.1	75.7	57.1	80.4	80.1	83.9	45.1	80.2
	T	9	9	126	14	10	10	148	13	10	10	181	14
1000	P	99.3	99.6	96.4	100.0	84.6	84.6	67.8	96.8	82.2	84.0	47.3	95.1
	R	98.9	99.4	95.1	96.3	81.4	82.2	68.4	73.7	76.3	78.7	46.1	72.7
	F	99.1	99.5	95.7	98.1	83.0	83.4	68.1	83.7	79.1	81.2	46.7	82.4
	T	13	13	278	29	24	23	335	34	15	15	453	30
5000	P	99.8	99.8	93.2	99.9	88.5	88.6	56.1	98.5	85.9	86.6	57.6	97.8
	R	99.4	99.5	91.6	98.2	83.2	83.3	54.9	86.0	81.7	81.3	56.7	86.3
	F	99.6	99.7	92.4	99.1	85.7	85.9	55.4	91.8	83.7	83.9	57.2	91.7
	T	54	53	3481	186	199	185	5248	274	185	174	3639	275
10000	P	99.8	99.9	93.2	100.0	88.0	88.9	59.6	98.5	86.8	88.7	57.2	98.4
	R	99.4	99.6	91.4	98.6	82.9	83.7	58.9	89.9	81.3	82.8	56.2	89.2
	F	99.6	99.7	92.3	99.3	85.4	86.2	59.2	94.0	84.0	85.6	56.6	94.0
	T	102	96	4356	305	370	346	4477	467	345	322	4351	479

alignments. The second option was to pass the text through an alignment tool and then manually check the output for all kinds of alignment. The third option was to check only for 1-to-1 alignments from this output. The fourth option was to evaluate on much smaller sizes.

In terms of time and effort required, there is an order of difference between the first and the second and also between the second and the third option. It is much easier to manually check the output of an aligner for 1-to-1 alignments than to align a corpus from the scratch. We couldn't afford to use the first two options. The fourth option was affordable, but we decided to opt for a more thorough evaluation of 1-to-1 alignments, than for evaluation of all kinds of alignments for smaller sizes. Thus, our starting data sets had only 1-to-1 alignments.

In future, we might extend the evaluation to all kinds of alignments, since the manual alignment currently being done on ERDC corpus includes partial and 1-to-2 or 2-to-1 alignments. Incidentally, there are rarely any 2-to-1 alignments in English-Hindi corpus since two English sentences are rarely combined into one Hindi sentence (when translating from English to Hindi), whereas the reverse is quite possible.

8 Evaluation Results

The results for various corpus types are given in table-1, for corpus sizes in table-2, and the global measures in table-3. Among the four algorithms tested, Moore's (**Mre**) gives the best results (except for the EMILLE corpus). This is as expected, since **Mre** combines sentence length based method with word correspondence. The results for **Mmd** are the worst, but it should be noted that the results for **Mmd** reported in this paper may not be the best that can be obtained with it, because its performance depends on some parameters. Perhaps with better tuning for English-Hindi, it might perform better. Another expected outcome is that the results for **GC** (character count) are better than **Brn** (word count). One reason for this is that there are more of characters than words (Gale and Church, 1991).

Leaving aside the tuning aspect, the low performance of **Mmd** may be due to the fact that it relies on cognate matching, and there are fewer cognates between Hindi and English. It might also be due to the syntactic differences (word order) between Hindi and English. This could, perhaps be taken care of by increasing the maximum point dispersal threshold (relaxing the linearity constraint), as suggested by Melamed (Melamed, 1996).

The results of experiment on English-French (table-1) show that **Mmd** performs better for this language pair than for English-Hindi, but it still seems to be more sensitive to noise than the other three algorithms. **Mre** performed the best for English-French too.

With respect to speed, **Brn** and **GC** are the fastest, **Mre** is marginally slower, and **Mmd** is much slower.

The effects of the previously mentioned factors on performance have been summarized below.

8.1 Corpus Type

Brn, **GC**, and **Mmd** performed almost equally well for EMILLE and ERDC corpora, but not that well for India Today. However, surprisingly, **Mre** performed much worse for EMILLE than it did for the other two corpora. It could be because of the fact that the EMILLE has a lot of very short (1-3 words) sentences, and word correspondence (in the second pass) may not be that effective for such sentences. The results don't support our assumption that EMILLE is easier than ERDC, but India Today does turn out to be more difficult than the other two for all the test cases. This is understandable since the translations in this corpus are much less literal.

8.2 Corpus Size

Only in the case of **Mre**, the performance almost consistently increased with size. This is as expected since the second pass in **Mre** needs training from the results of the first pass. The corpus size has to be large for this training to be effective. There doesn't seem to be a clear relationship between size and performance for the other three algorithms.

8.3 Noise and Difference in Sizes of SL and TL Corpora

As expected, introducing noise led to a decrease in performance for all the algorithms (table-1 and table-2). However (barring EMILLE) **Mre** seems to become less sensitive to noise as the corpus size increases. This again could be due to the unsupervised learning aspect of **Mre**.

Making the SL and TL corpora differ in size tended to reduce the performance in most cases, but sometimes the performance marginally improved.

Table 3: Global Evaluation Measures

		Brn	**GC**	**Mmd**	**Mre**
Clean, Same Size	L	92.6	93.4	81.4	80.8
	H	100.0	100.0	96.3	100.0
	P	98.4	98.7	90.3	95.1
	R	96.1	96.1	87.6	90.0
	F	97.2	97.3	88.9	92.4
Noisy, Same Size	L	73.1	75.8	44.1	72.6
	H	87.5	86.4	62.4	92.3
	P	82.7	84.1	53.8	92.2
	R	77.4	78.4	52.8	74.9
	F	79.8	81.1	53.3	82.5
Noisy, Different Size	L	74.7	76.4	46.2	71.3
	H	85.6	86.4	55.0	92.0
	P	83.4	84.9	51.2	91.5
	R	77.2	78.3	50.0	74.0
	F	80.1	81.4	50.6	81.6
Overall	L	81.1	82.4	55.4	80.0
	H	90.4	90.8	73.1	91.0
	P	88.2	89.2	65.1	92.9
	R	83.6	84.3	63.5	79.6
	F	85.7	86.6	64.6	85.5
L and *H:* Lower and higher limits of 95% confidence interval for F-measure *P*, *R*, and *F:* Average precision, recall, and F-measure					

9 Some Notes on Actual Corpus Alignment

Based on the evaluation results and our experience while manually checking alignments, we make some observations below which could be useful to those who are planning to create aligned parallel corpora.

Contrary to what we believed, sentence length based algorithms turn out to be quite robust, but also contrary to the commonly held view, there is scope for improvement in the performance of these algorithms by combining them with other techniques as Moore has done. However, as the performance of **Mre** on EMILLE shows, these additional techniques might sometimes *decrease* the performance.

There is a tradeoff between precision and recall, just as between robustness and accuracy (Simard and Plamondon, 1998). If the corpus aligned automatically is to be used without manual checking, then we should opt for maximum precision. But if it's going to be manually checked before being used, then we should opt for maximum recall. It depends on the application too (Langlais et al., 1996), but if manual checking is to be done, we can as well try to get the maximum number of alignments, since some decrease in precision is not going to make manual checking much more difficult.

If the automatically aligned corpus is not to be checked manually, it becomes even more important to perform a systematic evaluation before aligning a corpus, otherwise the parallel corpus will not be reliable either for machine learning or for linguistic analysis.

10 Conclusion

We used a systematic evaluation method for selecting a sentence alignment algorithm with English and Hindi as the language pair. We tested four algorithms for different corpus types and sizes, for the same and different sizes of SL and TL corpora, as well as presence and absence of noise. The evaluation scheme we have described can be used for a more meaningful comparison of sentence alignment algorithms. The results of the evaluation show that the performance depends on various factors. The direction of this variation (increase or decrease) was as predicted in most of the cases, but some results were unexpected. We also presented some suggestions on using an algorithm for actual alignment.

References

Brown Peter F., Cocke John, Della Pietra Stephen A., Della Pietra Vincent J., Jelinek Frederick, Lafferty John D., Mercer Robert L., and Roossin Paul S. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics.*

Brown Peter F., Della Pietra Stephen A., Della Pietra Vincent J., and Mercer Robert L. 1993. Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Brown Peter F., Lai J. C. and Mercer Robert L. 1991. Aligning Sentences in Parallel Corpora. *Proceedings of 29th Annual Meeting of the Association for Computational Linguistics*, 169–176. Berkeley, CA.

Chen Stanley F. 1993. Aligning Sentences in Bilingual Corpora Using Lexical Information. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 9–16. Columbus, OH.

Church Kenneth W. 1993. Char_align: A Program for Aligning Parallel Texts at the Character Level. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 1–8. Columbus, OH.

Church Kenneth W. and Hanks Patrick. 1993b. Aligning Parallel Texts: Do Methods Developed for English-French Generalize to Asian Languages?. *Proceedings of Rocling.*

Gale William A. and Church Kenneth W. 1991. A Program for Aligning Sentences in Bilingual Corpora. *Proceedings of 29th Annual Meeting of the Association for Computational Linguistics*, 177–184. Berkeley, CA.

Kay Martin. 1991. Text-Translation Alignment. *ACH/ALLC '91: "Making Connections" Conference Handbook.* Tempe, Arizona.

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

Langlais Phillippe, Simard Michel, and Vronis Jean. 1996. Methods and Practical Issues in Evaluating Alignment Techniques. *Proceedings of 16th International Conference on Computational Linguistics (COLING-96).*

Melamed I. Dan. 1996. A Geometric Approach to Mapping Bitext Correspondence. *IRCS Technical Report, University of Pennsylvania*, 96–22.

Moore Robert C. 2002. Fast and Accurate Sentence Alignment of Bilingual Corpora. *Proceedings of AMTA*, 135–144.

Och Franz Joseph and Ney Hermann 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19-51.

Simard Michel, Foster George F., and Isabelle Pierre. 1992 Using Cognates to Align Sentences in Bilingual Corpora. *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation.* Montreal, Canada.

Simard Michel and Plamondon Pierre. 1998 Bilingual Sentence Alignment: Balancing Robustness and Accuracy. *Machine Translation*, 13(1):59–80.

Wu Dekai. 1994. Aligning a Parallel English-Chinese Corpus Statistically with Lexical Criteria. *Proceedings of 32nd Annual Meeting of the Association for Computational Linguistics*, 80–87. Las Cruces, NM.