A segmentation-independent method for sub-sentential alignment

Fabien Cromieres
Graduate school of informatics
Kyoto University
fabien.cromieres@gmail.com

Sadao Kurohasahi
Graduate school of informatics
Kyoto University
kuro@nlp.kuee.kyoto-u.ac.jp

Abstract

We present a knowledge-free method for the sub-sentential alignment of bilingual sentence pairs. This method is designed to be independent from the segmentation (e.g. in characters or in words) of the sentences to be aligned. This has several advantages, including the ability to handle non-segmented Japanese text without pre-processing, as well as the "natural" production of multi-words alignments.

1 Introduction

Sub-sentential alignment is an important step for the further use of parallel corpora for Data-driven Machine Translation. As has already been argued (Och et al., 1999), it is interesting to create these alignments in a knowledge-free way. Many algorithms for this have already been proposed. Some of them focus on word-to-word alignment ((Brown,97) or (Melamed,97)). Others allow the generation of phrase-level alignments, such as (Och et al., 1999). However, these phrase-level alignment algorithms still place their analyses at the word level; whether by first creating a word-to-word alignment or by computing correlation coefficients between pairs of individual words.

This is, in our opinion, an important limitation. Firstly, because it presupposes that we are able to segment the text we want to align into words, which is not trivial for many languages such as Japanese or Chinese. Secondly, there are many cases where, on the contrary, the “typographic word” is a unit too small for the alignment. For example, if we consider an idiomatic phrase, the meaning of the phrase may have no relation with the meaning of the individual words that compose it. For this reason, trying to do a word-to-word alignment of this phrase is likely to give bad results.

A method for segmentation-independent alignment has been proposed in (Cromieres,2006). This method works by aligning substrings of sentences directly, and so does not suffer from the limitations we mentioned.

In this paper, we present further improvements of this method, as well as additional experiments.

2 General principle

2.1 Notation and definitions

In the subsequent parts of this paper, a substring will denote indifferently a sequence of characters or a sequence of words (or actually a sequence of any typographic unit we might want to consider). The terms “elements” will be used instead of “word” or “characters” to denote the fundamental typographic unit we chose for a given language.

In general, the number of co-occurrences of two substrings s_1 and s_2 in a parallel corpus is the number of times they have appeared on the opposite sides of a bi-sentence in this corpus. It will be noted $N(s_1,s_2)$. In the cases where s_1 and s_2 appears several times in a single bi-sentence (n_1 and n_2 times respectively), we might count 1, n_1*n_2 or $min(n_1,n_2)$ co-occurrences. We will also note $N(s_1)$ the number of occurrences of s_1 in the corpus.

We will sometimes refer to the “semantic parts” of a sentence. We define intuitively a “semantic part” as a part of the sentence that convey a meaning by itself, and hence should have a proper translation in the opposite sentence.

2.2 Correlation of substrings

The basic idea of the method is that, instead of considering single elements such as words or characters, we are going to consider substrings of these elements.

Actually, for a given sentence pair, we will consider every possible substring on each sentence. For every substring in the left sentence, we look at every substring in the right sentence, and compute a correlation coefficient between them.

These correlation coefficients are computed with the help of a parallel corpus, used as training data (since the training is unsupervised, the training corpus may also be the corpus to be aligned). From this corpus, we extract the co-occurrence counts of pairs of substrings, as well as their individual occurrence counts. The correlation coefficients we can compute from this information are, for example, the Chi-Square statistic or the Dice coefficient.

These coefficients allow us to estimate the statistical correlation between substring pairs. We hope that there will be a kind of “statistical resonance” between substrings that are mutual translation, and that they will have a higher coefficient than non-related ones.

2.3 Counting substrings co-occurrences

Working with substring, and especially counting the co-occurrence of substrings in a corpus raise some technical issues.

When working with co-occurrence counts of words, it is usual to store these counts in a persistent data-structure such as a hash-table. However, the potential number of different substrings is much greater than the number of different words; and the potential number of substring pairs is hence even greater. Because of this, there is usually so much different values to be stored that it is impractical to use a persistent storage for most reasonably sized corpus.

We overcame this problem in the following way. Instead of storing the co-occurrence counts beforehand, we collect them “on the fly”, when they are needed for the computation of correlation coefficients. (Cromieres,2006) shows how it is possible to do this in a very efficient way with the help of the data-structure known as Suffix Array.

We will just give here a very brief explanation. For any substring, a Suffix Array allows us to know the list of the index of the sentences where this substring occurred. By comparing this index list between two substrings, we can easily obtain their co-occurrence counts.

3 Basic method

3.1 Algorithm

We use a greedy and somewhat brutal algorithm. As we mentioned before, for a given bi-sentence, we first compute a correlation coefficient between all possible substring pairs we can extract from this bi-sentence. Then we iterate through the following steps:

1) Align the 2 substrings with the highest correlation

2) Discard all substrings that intersect with the previously aligned substrings (so that no element can be aligned several times)

3) If there remains substring pairs with sufficiently high correlation, go to 1

The correlation threshold for determining whether or not to continue the algorithm at the step 3 depends on the correlation coefficient used as well as the precision/recall ratio we aim at.

3.2 The problem of the incomplete alignments

The algorithm as described before gives interesting results, but has a recurring problem: a tendency to link incomplete substrings. One of the reasons we found for this is what we call the “common substring problem”. This happen when a substring s_1 can be translated by two substrings s_2 and s_2’; s_2 and s_2’ having themselves a common substring. s_1 will then be linked to the common part of s_2 and s_2’.

For example, the English word “museum” has two Japanese equivalents: 博物館 and 美術館. In the BTEC corpus, the common part (館) will have a stronger association with “museum”, and so will be linked instead of the correct substring (博物館 or 美術館). Since it is quite common for phrases with similar meanings to share some common elements, this situation happens frequently.

The first solution we tried to solve this problem, was to introduce a bias in the correlation coefficient formulas, so that they would favour the alignment of longer substrings. While this reduced the problem, there was a lot of room for improvement.

4 Addition of an “extension” phase

4.1 Extending the alignments

To solve the problem of the “incompleteness” of certain alignments due to the common substring problem, we tried to improve it with an idea that is inspired from the ISA alignment algorithm (Zhang, Vogel, Waibel, 2003).

In the ISA algorithm, each part of a bi-sentence is represented on the horizontal and vertical axis of a grid. Each cell of the grid is thus at the intersection of two words (one in each sentence), and is filled with the value of a correlation coefficient between these words. On such a grid, a contiguous alignment is represented as a rectangle. The grid is then used to extend word-alignments to phrase-alignments by looking at areas that have similar values. A rectangle representing an alignment is extended to the right, the left, the top or the bottom if the values in these

	日	本	語	は	難	し	い
Japanese	**9.5**	**9.2**	**7.6**	0.1	0.1	0.1	0.1
is	0.1	0.1	0.1	**5.6**	**4.5**	**4.4**	**4.3**
difficult	0.1	0.1	0.1	**0.9**	**7.2**	**5.3**	**5.5**

Figure 1: an ISA-style grid for the sentence pair 日本語は難しい/Japanese is difficult

directions are superior to a certain fraction (for example 20%) of the maximum value inside the rectangle. Various heuristics we do not have the space to detail here are also involved.

We will use a similar method to try and improve our alignment algorithm. For every sentence pair, we produce a grid similar to the one used by ISA, the sentences being segmented with the chosen elements instead of words (see fig. 1)

We then modify our basic algorithm in the following way: Every time an alignment is produced at the step 1, we try to extend this alignment by looking in the grid if some neighboring cells have a value similar to the cells inside the rectangle representing the alignment. We then proceed to the second step.

However, a problem remains: correlation between elements is a very local information. Using it to extend alignments may lead to some problems, if the chosen segmentation element is too small (in the case of a character-segmentation, for example)

That is why, instead of filling the grid with “local” correlation between two elements, we will try to obtain a “global” correlation that take into account the whole context of each element.

4.2 Computing a global correlation

To obtain this “global correlation” between two elements e1 and e2, we compute a correlation between all of the substrings that contains e1 and e2. The “global correlation” will be a combination of all of these values.

For example, let us consider the sentence pair:

日本語は難しい / Japanese is difficult

(the English part being segmented in words and the Japanese one in characters).

To obtain the global correlation between 語 and Japanese, , we will compute the various substring correlations correl(語,**Japanese**), correl(本語は,**Japanese**), correl(日本語、**Japanese** is), correl(日本語は難しい, **Japanese** is difficult), …

When these correlations have been computed, there are several possibilities for combining them. We can, for example, sum them, or use their mean values. We chose to do a weighted sum, where the weight is fixed proportionally to the distance between the considered elements and the center of the substrings. The global correlation between two elements e1 and e2 is then given by:

$$CG(e1,e2) = \sum_{e1\in s1, e2\in s2} correl(s1,s2) * w(s1,s2,e1,e2)$$

where:

$$w(s1,s2,e1,e2) = \frac{(d(e1,ms1)+1)*(d(e2,ms2)+1)}{(|s1|/2+1)*(|s2|/2+1)}$$

with:

d(e1,ms1): the distance from e1 to the middle of s1

|*s1*|: the length of the substring s1

It may seem counter-intuitive that elements at the center of a substring receive less probability than elements at its extremities. The rationale for this is that an element positioned at the center of a “semantic part” will be contained by more substrings included in this semantic part than elements on the border. Hence, it is contained by more high-correlation substrings than elements on the extremities. The weights balance this phenomenon.

For example, in the sentence of the previous examples, 本 is included in 4 substrings that have a very high correlation with “Japanese”: 本, 日本, 本語 and 日本語, whereas 日 is only included in the 3 “high-correlation” substrings 日, 日本 and 日本語.

5 Evaluations

For evaluation purpose, we used around 280,000 sentence pairs from the Yumiuri Corpus, a Japanese-English News Corpus (Utiyama and Isahara). We also used a gold standard provided by NICT consisting of 30,000 aligned sentence pairs (Utimoto and al., 2004)

Evaluating the quality of alignments is notoriously difficult. We computed the recall and precision character by character in the following way:

For every Japanese character and for every English word linked together in the gold standard, we count a “correct link” if they are also linked in the alignment and a “missed link” if they are not. For every link that appears in the alignment and not in the gold standard, we count an “incorrect link". It should be noted that alignments spanning several words or characters link all their elements together. So, if "by the end of the year" and "年内" are aligned, we this represent 12 links.

Recall and precision are then computed in the usual way for every sentence:

$$\mathrm{Re}\,call = \frac{\#corrects}{\#corrects + \#missed}$$

$$\mathrm{Pr}\,ecision = \frac{\#corrects}{\#corrects + \#incorrects}$$

The final recall and precision are an average over all the sentences in the gold standard.

For comparison, we segmented the data using the program JUMAN and created alignments using the GIZA++ program. (Och and Ney, 2003). 10 iterations of each of the model 1, 2, 3 and 4 were used.

We used two versions of our alignment program: one that make use of the basic algorithm with a biased correlation coefficient (such as described in section 3), and another that use the "extend" method described in section 4. For the extend method, we also modified the correlation threshold to obtain a higher precision (see 3.1): this is the Extend-Precision method.

The results may seem very low, but this is by a large measure a consequence of our method of evaluation. On one hand, our gold standard contains only one alignment for every sentence (when several are usually valid), and on the other hand, the evaluation methodology punish strongly any deviation from this gold standard (because each single character has to be correctly aligned). These results are still a good basis for comparison.

	Precision	Recall	F
Basic method	54.2	20.8	30.1
Extend method	**55.6**	**24.1**	**33.6**
Extend-Precision	**62.5**	**18.1**	**28.1**
GIZA+Juman	59.9	17.0	26.4

The figures obtained are interesting. Firstly, they show that the "extend" phase we added to the basic algorithm really improve the performances.

Secondly, we obtain better results than the combination of Giza and a segmentation tool. This superiority should however be interpreted: Giza only produces one-to-many alignments. Even if, because of the pre-segmentation, they are actually many-characters to many-words alignments, they still span only a few characters. On the other hand, our gold standard contains many very long alignments. This is probably why the recall of Giza is so low. If we had used a gold-standard with more fine-grained alignments, the results may have been different. That is why we will need to confirm these results in further experiments.

6 Conclusion

In this paper we presented an algorithm for segmentation-independent alignment. Among its advantages is the possibility of aligning non-segmented text, with results comparable (and possibly better) to other knowledge-free methods making use of a pre-segmentation. This show that the statistical information contained in a non-segmented parallel corpus is sufficient to compensate the absence of pre-segmentation.

7 Acknowledgement

We would like to thank Toshioki Nakazawa for kindly providing us his scripts for the automatic evaluation of the alignments.

References

Ralph Brown. 1997. *Automated Dictionary Extraction for Knowledge-Free Example Based Translation*, Proceedings of TMI97, , pp. 111-118, Santa-Fe, July 1997.

Dan Melamed. 1997. *A Word-to-Word Model of Translational Equivalence*, Proceedings of ACL 97, Madrid, Spain, 1997.

Franz Joseph Och, Christophe Tillmann, Hermann Ney. 1999. *Improved Alignment Models for Statistical Machine Translation*. Proceedings of EMNLP-VLC, pp 20-28, University Of Maryland,

Masao Utiyama and Hitoshi Isahara. 2003. *Reliable Measures for Aligning Japanese-English News Articles and Sentences.* ACL-2003, pp. 72--79.

Franz Josef Och, Hermann Ney. 2003. *A Systematic Comparison of Various Statistical Alignment Models*, Computational Linguistics, volume 29, number 1, pp. 19-51 March 2003

Fabien Cromieres. 2006. *Sub-Sentential Alignment Using Substring Co-Occurrence Count*s. Colling-ACL-2006

Kiyotaka Uchimoto, Yujie Zhang, Kiyoshi Sudo, Masaki Murata, Satoshi Sekine and Hitoshi Isahara. 2004. *Multilingual Aligned Parallel Treebank Corpus Reflecting Contextual Information and Its Application*s. Proceedings of the MLR2004 pp 63-70

Ying Zhang, Stephan Vogel,Alex Waibel. 2003. *Integrated Phrase Segmentation and Alignment algorithm for Statistical Machine Translation*, Proceedings of ICNLP-KE, Beijing,China., October 2003