Hierarchical Sub-sentential Alignment with Anymalign

Adrien Lardilleux
LIMSI-CNRS
Orsay, France
adrien.lardilleux@limsi.fr

François Yvon
LIMSI-CNRS/University Paris Sud
Orsay, France
francois.yvon@limsi.fr

Yves Lepage
Waseda University, IPS
Waseda, Japan
yves.lepage@waseda.jp

Abstract

We present a sub-sentential alignment algorithm that relies on association scores between words or phrases. This algorithm is inspired by previous work on alignment by recursive binary segmentation and on document clustering. We evaluate the resulting alignments on machine translation tasks and show that we can obtain state-of-the-art results, with gains up to more than 4 BLEU points compared to previous work, with a method that is simple, independent of the size of the corpus to be aligned, and directly computes symmetric alignments. This work also provides new insights regarding the use of "heuristic" alignment scores in statistical machine translation.

1 Introduction

Sub-sentential alignment consists in identifying translation units in sentence-aligned parallel corpora, i.e. in texts in which each sentence has been matched with its translation. This task constitutes the first step in the process of training most data-driven machine translation (MT) systems (statistical or example-based). The most prominent approach nowadays is phrase-based statistical machine translation (SMT), where the core model is a translation table derived from sub-sentential mappings. This table consists in a pre-computed list of phrase[1] pairs, where each (*source*, *target*) pair is associated with a certain number of scores loosely reflecting the likelihood that *source* translates to *target*.

The problem of identifying sub-sentential mappings from parallel texts, e.g. between isolated words or n-grams of words, is well-known, and numerous proposals have been put forward to perform this task. Those methods roughly fall into two main categories. On the one hand, the *probabilistic* approach, introduced by Brown et al. (1988), considers the problem of identifying *links* between words or groups of words in parallel sentences. This approach consists in defining a probabilistic model of the parallel corpus, the parameters of which are estimated by a global maximization process which simultaneously considers all possible associations in the corpus. The goal is to determine the best set of alignment links between all source and target words of every parallel sentence pair. The most famous representatives in this category are the IBM models (Brown et al., 1993) for aligning isolated words, which have given rise to an impressive series of variants and amendments (see e.g. (Vogel et al., 1996; Wu, 1997; Deng and Byrne, 2005; Liang et al., 2006; Fraser and Marcu, 2007; Ganchev et al., 2008), to cite a few). Generalizing word alignment models to phrase alignment proves to be a much more difficult problem, and in the view of work of Marcu and Wong (2002) and Vogel (2005), such alignments are generally produced by heuristically combining asymmetric 1–*n* word alignments ("oriented") in both directions (Koehn et al., 2003; DeNero and Klein, 2007). Once the set of alignment links is constituted, it is possible to assign scores to each pair of segments extracted.

On the other hand, *associative* approaches (also called *heuristic* by Och and Ney (2003)), were introduced by Gale and Church (1991). They do not rely on an alignment model: in order to detect translations, they rely on independence statistical measures such as, for instance, Dice coefficient, mutual information (Gale and Church, 1991; Fung and Church, 1994), or likelihood ratio (Dunning, 1993)—see also more recent work by Melamed (2000) and by Moore (2005). Computations are generally limited to a list of association candidates precomputed using patterns and filters, for instance, by focusing exclusively on the most frequent word n-grams. In this approach, a local maximisation process is used, where each sentence is processed

[1] In this context, a phrase is a sequence of words and does not necessarily correspond to a syntactic phrase.

independently. Alignment links can then be computed, using for instance the greedy algorithm proposed by Melamed (2000) *(competitive linking)*.

The probabilistic approach is the most widely used, mainly due to its tight integration with SMT, of which it constitutes a cornerstone since the introduction of IBM models (Brown et al., 1993). The two approaches have shown complementary strengths and weaknesses, as acknowledged by e.g. Johnson et al. (2007), where phrase associations extracted from word alignments are filtered out according to statistical association measures.

Anymalign, introduced in (Lardilleux and Lepage, 2009; Lardilleux et al., 2011a), aims at extracting sub-sentential associations, addressing a number of issues that are often overlooked. It can process any number of languages simultaneously, it does not make any distinction between source and target, is amenable to massive parallelism, scales easily, and is very simple to implement. Anymalign's association scores have proven to produce better results than state-of-the-art methods on bilingual lexicon constitution tasks (evaluation performed by comparing word associations with reference dictionaries). However, Anymalign's phrase tables are not as good as those obtained with standard methods (evaluation performed with standard MT metrics) (Lardilleux et al., 2011b).

One possible explanation for these contrasted results is that, Anymalign does not compute any alignment at the word or at the phrase level; instead, it directly computes translation tables along with their associated scores. Those tables have very different profiles than those obtained with probabilistic methods, mainly in terms of their n-gram distribution (Luo et al., 2011). In particular, despite recent improvements (Lardilleux et al., 2011b), the quantity of long n-grams produced remains relatively small compared with Moses's translation tables.

In this paper, we complement Anymalign with a simple alignment algorithm, so as to better understand its current limitations. The resulting alignments improve Anymalign's phrase tables to a point where they can be used to obtain state-of-the art results. In passing, we also propose a computationally cheap way to compute ITG alignments based on arbitrary word level association scores.

The rest of this paper is organized as follows: Section 2 describes the alignment method in detail, Section 3 presents an evaluation on machine translation tasks and an analysis of the results, and Section 4 concludes and discusses further prospects.

2 Description of the Method

In a nutshell, out method segments pairs of parallel sentences in two parts, linking the two resulting target segments with their proper translation amongst the two source segments (monotonous or inverted translation), and repeats this process recursively on the segment pairs thus obtained.

This work is strongly inspired by that of Wu (1997) and Deng et al. (2006). The former introduces *inversion transduction grammars*, which generate synchronized binary parse trees in source and target languages. This formalism models both variable-length associations at leaf (terminal) nodes, and reorderings (inversions) at any level of the parse tree. As we are only interested in computing alignment based on arbitrary lexical association scores, we will dispense here from using the full apparatus of stochastic grammars, yielding algorithms that are computationally much cheaper. The latter uses a similar concept, where more or less coarse bi-segments are extracted from non-sentence-aligned parallel texts by iteratively recursively applying a *top-down* binary segmentation algorithm. We reproduce the same approach here at the sentence level, using different local association scores.

2.1 Alignment Matrix

Our starting point are (1) a sentence-aligned bitext; and (2) a function w measuring the strength of the translation link between any *source* and *target* pair of words. Several definitions of w are possible; it is nevertheless natural to define it endogenously from word occurrences in the bitext. The scores we will first use will be obtained using Anymalign's output. We will see later that they lead to better results than scores obtained using other standard measures.

In the following, the score $w(s,t)$ between a source word s and a target word t is defined as the product of the two translation probabilities $\mathrm{p}(s|t) \times \mathrm{p}(t|s)$, produced by Anymalign:

$$\begin{aligned} w(s,c) &= \mathrm{p}(s|t) \times \mathrm{p}(t|s) \\ &= \frac{\sum_{n=1}^{N} [\![(s,t) \in (S_n,T_n)]\!] k_n}{\sum_{n'=1}^{N} [\![s \in S_{n'}]\!] k_{n'}} \times \frac{\sum_{n=1}^{N} [\![(s,t) \in (S_n,T_n)]\!] k_n}{\sum_{n'=1}^{N} [\![t \in T_{n'}]\!] k_{n'}} \\ &= \frac{\left(\sum_{n=1}^{N} [\![(s,t) \in (S_n,T_n)]\!] k_n\right)^2}{\left(\sum_{n'=1}^{N} [\![s \in S_{n'}]\!] k_{n'}\right) \times \left(\sum_{n'=1}^{N} [\![t \in T_{n'}]\!] k_{n'}\right)} \end{aligned}$$

where:

- $[\![x]\!] = 1$ if x is true, 0 otherwise;
- N is the number of entries (source–target phrase pairs) in Anymalign's translation table;
- S_n (resp. T_n) is the source (resp. target) part of an entry in the translation table;
- k_n is the count associated to the pair (S_n, T_n) in the translation table. This figure is not by itself

S_n	C_n	k_n
pays	countries	151,190
pays	*country*	17,717
pays tiers	third countries	10,865
les *pays*	countries	6,284
mon *pays*	my *country*	4,057
ces *pays*	these countries	3,742
pays .	*country* .	2,007
état	*country*	122

$$\begin{aligned} w(pays, country) &= \mathrm{p}(pays|country) \times \mathrm{p}(country|pays) \\ &= \frac{17{,}717 + 4{,}057 + 2{,}007}{151{,}190 + 17{,}717 + 10{,}865 + 6{,}284 + 4{,}057 + 3{,}742 + 2{,}007} \\ &\quad \times \frac{17{,}717 + 4{,}057 + 2{,}007}{17{,}717 + 4{,}057 + 2{,}007 + 122} \\ &\simeq 0.121 \end{aligned}$$

Figure 1: Computing a score between source word *pays* and target word *country* from a subset of a translation table produced by Anymalign with the French and English parts of the Europarl corpus (Koehn, 2005).

> an indicator of the quality of the entry; it is just the number of times the translation pair has been produced by Anymalign (see (Lardilleux et al., 2011a) for details).

This computation is illustrated on Figure 1.

What we do here is tantamount to a very simplified version of the algorithm that is used to train standard translation models: starting with lexical associations, we derive by heuristic means an optimal (Viterbi) alignment, from which the translation tables are finally computed. Our procedure is much simpler, though, as we do not iterate the procedure (like in EM training) and directly manipulate symmetric representations at the phrase level.

2.2 Segmentation Criterion

The segmentation criterion described hereafter is inspired by the work of Zha et al. (2001) on document clustering. Their problem consists in computing the optimal joint clustering of a bipartite graph representing occurrences of terms inside a set of documents. We adapt it to the search of the best alignment between words of a source sentence and those of a target sentence.

To this end, we consider a pair of sentences (S,T) from the parallel corpus, where the source sentence S is made up of I source words and the target sentence T is made up of J target words: $S = [s_1 \dots s_I]$ and $T = [t_1 \dots t_J]$. Moreover, we consider "split" indices x and y which define a binary segmentation of the source and target sentences (the "." symbol refers to the concatenation of word strings):

$$\begin{aligned} S = A.\bar{A} \quad &\text{with} \quad A = [s_1 \dots s_{x-1}] \text{ and } \bar{A} = [s_x \dots s_I] \\ T = B.\bar{B} \quad &\text{with} \quad B = [t_1 \dots t_{y-1}] \text{ and } \bar{B} = [t_y \dots t_J] \end{aligned}$$

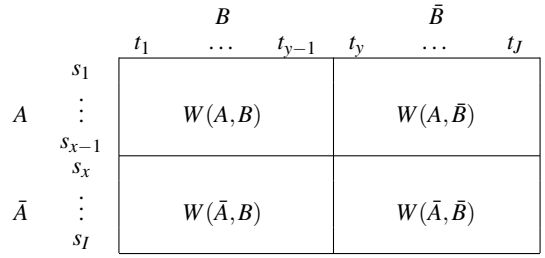

Figure 2: Schematic representation of the segmentation of a pair of sentences $S = A.\bar{A}$ and $T = B.\bar{B}$.

The choice of x and y will be guided by the sum W of the association scores between each source and target words of a block $(X,Y) \in \{A,\bar{A}\} \times \{B,\bar{B}\}$:

$$W(X,Y) = \sum_{s \in X, t \in Y} w(s,t)$$

These notations are summarized in Fig. 2.

Then, we define the total score of a segmentation:

$$\mathrm{cut}(X,Y) = W(X,\bar{Y}) + W(\bar{X},Y)$$

Note that $\mathrm{cut}(X,Y) = \mathrm{cut}(\bar{X},\bar{Y})$. In our case, a low value indicates that the association scores between the words of X and that of $\bar{Y}$ on the one hand, and between the words of $\bar{X}$ and that of Y on the other hand, are low; in other words, those two blocks are unlikely to correspond to good translations, contrarily to (X,Y) and $(\bar{X},\bar{Y})$. We would thus like to identify the pair (x,y) that leads to the lowest possible value of $\mathrm{cut}(X,Y)$.

As pointed out by Zha et al. (2001), this quantity tends to produce unbalanced segments (document clusters in their case) because of the absence of normalisation, which warrants its replacement by:

$$\mathrm{Ncut}(X,Y) = \frac{\mathrm{cut}(X,Y)}{\mathrm{cut}(X,Y) + 2 \times W(X,Y)} + \frac{\mathrm{cut}(\bar{X},\bar{Y})}{\mathrm{cut}(\bar{X},\bar{Y}) + 2 \times W(\bar{X},\bar{Y})}$$

This variant adds a density constraint on (X,Y) and $(\bar{X},\bar{Y})$, which is partially satisfied by the introduction of the denominators in the above expression. Its values are in the range $[0,2]$.

Our problem eventually consists in determining the pair (x,y) that minimizes Ncut. Although efficient search methods exist and are commonly used in graph theory, our "graphs" (pairs of sentences) are small in practice: about 30 words per sentence in average in the Europarl corpus used in the following experiments. We thus content ourselves with determining the best segmentation through an exhaustive enumeration.

2.3 Alignment Algorithm

We can now recursively segment and align a pair of sentences. At each step, we test every possible pair (x,y) of indices in order to determine

```
procedure align(S,T) :
    if length(S) = 1 or length(T) = 1 :
        link each word of S to each word of T
        stop procedure
    minNcut = 2
    (X,Y) = (S,T)
    for each (i,j) ∈ {2...I} × {2...J} :
        if Ncut(A,B) < minNcut :
            minNcut = Ncut(A,B)
            (X,Y) = (A,B)
        if Ncut(A,B̄) < minNcut :
            minNcut = Ncut(A,B̄)
            (X,Y) = (A,B̄)
    align(X,Y)
    align(X̄,Ȳ)
```

Figure 3: Recursive alignment algorithm.

the lowest Ncut. The worst case happens when the matrix is cut in the most unbalanced possible way; the complexity of the algorithm is thus cubic ($O(I \times J \times \min(I,J))$) in the length of the input sentences. Using a greedy strategy only delivers suboptimal solutions, yet it does so much faster than exact ITG parsing, which is cubic in the product $I \times J$ (Wu, 1997). For a given pair (x,y), two values are computed: one corresponds to a monotonous alignment ($\text{Ncut}(A,B)$) and the other one to an inversion of the two segments ($\text{Ncut}(A,\bar{B})$). We then apply the process recursively on each of the two segment pairs that correspond to the minimal Ncut. It ends when one of the segments contains only one word and produces 1–n or n–1 alignments. In this approach, all words are aligned. By considering different stopping criteria, eg. based on thresholds on Ncut, variants of the algorithm are readily obtained, which enable to balance the granularity of the alignment with its precision, by choosing to build larger and safer blocks (m–n alignments) instead of smaller and less sure ones. We leave this for future work. Figure 3 presents the complete algorithm, and Fig. 4 illustrates the process on two actual examples. In the following, we refer to this algorithm under the name of "Cutnalign."

The algorithm itself is independent of the size of the parallel corpus to align, because each sentence pair is processed independently. Aligning a corpus can thus easily be made parallel: the total running time is divided by the number of available processors. Another advantage is that the alignments produced are symmetric during the whole process, contrary to more widely spread models such as IBM models that produce better result when run in both translation directions and their outputs combined using heuristics.

3 Evaluation

3.1 Description of Experiments

Our alignment method is evaluated within a phrase-based SMT system. We use the Moses toolkit (Koehn et al., 2007), and data extracted from the Europarl corpus (Koehn, 2005), in three languages: Finnish–English (agglutinating language–isolating language), French–Spanish, and Portuguese-Spanish (very close languages). For each pair, we use a training set made up of 350,000 sentence pairs (avg.: 30 words/sentence in English), and development and test sets made up of 2,000 sentence pairs each. The systems are optimized with MERT (Och, 2003). Unless otherwise specified, a lexicalized reordering model is used. Translations are evaluated using BLEU (Papineni et al., 2002) and TER[2] (Snover et al., 2006).

Five approaches are compared:

MGIZA++ (Gao and Vogel, 2008), implements the IBM models (Brown et al., 1993) and the HMM of Vogel et al. (1996). Integrated to Moses, it remains the reference in the domain. It is run with default settings: 5 iterations of IBM1, HMM, IBM3, and IBM4, in both directions (source to target and target to source). The alignments are then made symmetric and a translation table is produced from the alignments using Moses tools (*grow-diag-final-and* heuristic for phrase pair extraction).

Anymalign (Lardilleux et al., 2011a), used to directly build the translation tables. As this tool can be stopped at any time, its running time is set so that it runs for the same duration as MGIZA++. The same experiment is repeated by varying the length of output phrases from 1 to 4 (see (Lardilleux et al., 2011b) for details). In the following, we refer to it under the names "Anymalign-1" to "Anymalign-4." The reordering model used in this configuration is a simple distance-based model, because Anymalign alone cannot provide the information required for a lexicalized reordering model.

Anymalign + Cutnalign: we apply the algorithm described in previous section to each of the four translation tables produced by Anymalign-1 to Anymalign-4. Although every intermediary segmentation step (all possible rectangles in Fig. 4) actually corresponds to a phrase pair that could be extracted and fit in a phrase-table, in our experiments, we only rely on *terminal alignment points*, that are then passed to the Moses toolkit to build new translation tables (using again the *grow-diag-final-and*

[2]Contrary to BLEU, lower scores are better.

	the	level	of	budgetary	implementation	;
le	**0.037**	ε	0.001	ε	ε	ε
niveau	ε	**0.591**	ε	ε	ε	ε
d’	ε	ε	**0.003**	ε	ε	ε
exécution	ε	ε	ε	ε	**0.060**	ε
budgétaire	ε	ε	ε	**0.659**	ε	ε
;	ε	ε	ε	ε	ε	**0.287**

	finally	,	what	our	fellow	citizens	are	demanding	is	the	right	to	information	.
enfin	**0.607**	0.001	ε	ε	0	ε	ε	0	ε	ε	ε	ε	ε	ε
,	0.001	**0.445**	ε	ε	ε	ε	ε	ε	ε	0.001	ε	0.001	ε	0.001
c’	ε	ε	0.001	ε	ε	ε	ε	0	**0.036**	0.001	ε	ε	ε	ε
est	ε	ε	0.001	ε	ε	ε	ε	0	**0.223**	0.016	ε	0.001	ε	0.001
un	ε	ε	ε	ε	ε	ε	ε	ε	0.005	ε	ε	ε	ε	ε
droit	ε	ε	ε	ε	ε	ε	ε	0	ε	ε	**0.084**	ε	ε	ε
à	ε	ε	ε	ε	ε	ε	0.001	ε	0.001	0.003	0.001	**0.018**	ε	ε
l’	ε	ε	ε	ε	ε	ε	ε	ε	0.002	0.009	ε	**0.002**	ε	ε
information	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	**0.499**	ε
que	ε	ε	0.002	ε	ε	ε	0.001	ε	0.002	0.001	ε	0.001	ε	ε
réclament	0	0	ε	ε	ε	ε	ε	**0.152**	ε	ε	0	0	0	ε
nos	ε	ε	ε	**0.171**	0.004	0.001	ε	ε	ε	ε	ε	ε	ε	ε
concitoyens	0	ε	ε	0.001	**0.323**	**0.009**	ε	ε	ε	ε	0	ε	0	ε
.	ε	ε	ε	ε	ε	ε	ε	ε	0.001	0.001	ε	ε	ε	**0.954**

Figure 4: Two examples of segmentation-alignment. The number in each cell corresponds to the value of the function w, with $0 < \varepsilon \leq 0.001$. A null value indicates that the two words never appear together in the translation table. Alignment points retained by the algorithm, i.e. at maximum level of recursion, are in boldface. In the first example, the translation is monotonous except for the name/adjective inversion (*exécution budgétaire/budgetary implementation*), therefore most alignment links are along the diagonal. The second example, more complex, attests for the inversion of propositions inside the sentence.

heuristic). This approach yields more phrase pairs as it allows to extract together segments on both sides of a split point, e.g. *le niveau/the level*.

Simple probabilities + Cutnalign: the purpose of this configuration is to evaluate the choice of w, rather than the algorithm itself. To this end, we use a very simple association score: the probability that a source word and a target word are translations of one another (product of the two translation probabilities), where this probability is computed from their co-occurrence counts over the training corpus. The definition of w is thus the same as in Sec. 2.1, with two minor differences: (1) counts are directly computed over the training bitext; and (2) $k_n = 1, \forall n$.

Anymalign + Cutnalign / MGIZA++: This is a combination of the MGIZA++ and Anymalign+Cutnalign approaches. We do this by taking the union of the two alignment sets. In pratice, we simply concatenate the two alignment files produced by the aligners, and duplicate the training bicorpus so that we end up with a new, twice as large, training bicorpus and alignment file, from which the phrase table is extracted.

In terms of runtime, although Cutnalign is currently implemented in a high-level programming language (Python) and its complexity is cubic in the length of the sentence pairs to process, the fact that each sentence pair can be aligned independently makes it amenable to massive parallelism if numerous CPUs are available.

3.2 Results

Results are in Table 1. For each task, using the basic version of Anymalign yields worse scores than MGIZA++-based system, even though extending the phrase length reduces this gap by roughly a half, except for the Finnish–English pair. Those results are in line with (Lardilleux et al., 2011b).

Cutnalign leads to significant gains in all configurations: from 1.6 to 4.6 BLEU points (fr–en, Anymalign-1 + Cutnalign), with an average gain of 2.6 BLEU and 2.7 TER points. Anymalign + Cutnalign is still 1.1 to 1.6 BLEU points below in Finnish–English relatively to MGIZA++ but produces results of comparable quality in French–English and Portuguese–Spanish.

The “simple probabilities + Cutnalign” configuration produces intermediary quality results, generally between “basic” Anymalign and Anymalign + Cutnalign. This shows that the function w has a significant impact on the behavior of the alignment method. Assuming the function used in these experiments is one of the simplest possible, there is ample room here for improvements. Merging both phrase tables is almost always the best strategy, at

Task	System	BLEU (%)	TER (%)	Entries (millions)	Length of entries	Links	Length of extracted blocks
fi–en	MGIZA++	22.27	62.92	22.2	3.24	26	1.16
	Anymalign-1	18.68	67.30	11.8	1.87		
	Anymalign-2	17.86	68.60	4.4	2.09		
	Anymalign-3	18.06	68.13	3.0	2.32		
	Anymalign-4	18.06	68.53	2.1	2.42		
	Anymalign-1 + Cutnalign	21.14	63.74	7.7	3.26	62	1.45
	Anymalign-2 + Cutnalign	21.14	64.69	7.5	3.27	69	1.48
	Anymalign-3 + Cutnalign	20.83	64.18	7.3	3.29	73	1.50
	Anymalign-4 + Cutnalign	20.64	64.52	7.1	3.29	78	1.53
	Simple prob. + Cutnalign	19.09	67.09	5.5	3.23	74	1.78
	Anymalign-1 + Cutnalign / MGIZA++	22.66	62.45	27.0	3.24	44	1.30
	Anymalign-2 + Cutnalign / MGIZA++	22.68	62.91	26.9	3.24	47	1.31
	Anymalign-3 + Cutnalign / MGIZA++	22.73	62.82	26.8	3.24	49	1.32
	Anymalign-4 + Cutnalign / MGIZA++	**22.78**	**62.11**	26.7	3.24	52	1.33
fr–en	MGIZA++	29.65	55.25	25.6	4.29	31	1.17
	Anymalign-1	25.10	59.36	6.1	1.27		
	Anymalign-2	26.60	58.16	6.3	1.99		
	Anymalign-3	27.02	57.96	3.9	2.29		
	Anymalign-4	26.85	58.00	2.6	2.42		
	Anymalign-1 + Cutnalign	29.65	55.22	12.9	4.21	50	1.49
	Anymalign-2 + Cutnalign	29.69	55.44	13.1	4.22	48	1.48
	Anymalign-3 + Cutnalign	29.26	55.49	13.0	4.23	50	1.49
	Anymalign-4 + Cutnalign	29.16	55.46	12.8	4.23	52	1.51
	Simple prob. + Cutnalign	27.97	56.85	10.2	3.95	54	1.62
	Anymalign-1 + Cutnalign / MGIZA++	30.02	**54.81**	31.9	4.24	41	1.32
	Anymalign-2 + Cutnalign / MGIZA++	29.91	54.88	31.9	4.24	40	1.32
	Anymalign-3 + Cutnalign / MGIZA++	**30.22**	54.94	31.9	4.24	41	1.32
	Anymalign-4 + Cutnalign / MGIZA++	29.91	54.87	31.8	4.24	42	1.33
pt–es	MGIZA++	38.53	48.46	32.2	4.30	30	1.09
	Anymalign-1	35.20	50.89	5.7	1.26		
	Anymalign-2	36.80	49.60	5.9	1.99		
	Anymalign-3	36.82	49.67	3.7	2.26		
	Anymalign-4	36.96	49.80	2.4	2.37		
	Anymalign-1 + Cutnalign	37.35	49.55	17.9	4.30	50	1.32
	Anymalign-2 + Cutnalign	**38.96**	**48.04**	18.0	4.30	48	1.32
	Anymalign-3 + Cutnalign	38.55	48.40	17.7	4.31	50	1.33
	Anymalign-4 + Cutnalign	38.56	48.37	17.3	4.31	54	1.35
	Simple prob. + Cutnalign	37.71	49.04	13.9	4.09	50	1.41
	Anymalign-1 + Cutnalign / MGIZA++	38.77	48.12	37.7	4.25	40	1.20
	Anymalign-2 + Cutnalign / MGIZA++	38.69	48.39	37.9	4.25	39	1.20
	Anymalign-3 + Cutnalign / MGIZA++	38.94	48.12	37.8	4.25	40	1.20
	Anymalign-4 + Cutnalign / MGIZA++	38.82	48.18	37.8	4.25	42	1.21

Table 1: Summary of results obtained in our experiments. The first two columns (BLEU and TER) report performance in machine translation. The two middle columns diplay various characteristics of the translation tables: the number of entries and their length in words. The last two columns present characteristics of the alignments prior to the production of the translation table: average number of alignment links per training sentence pair and average length of the source part of minimal blocks extracted (translations of the phrases that are consistent with word alignments).

the most of much larger models.

3.3 Analysis of Alignments

One motivation for proposing this new alignment method is that Anymalign still lacks the ability to extract long n-gram translations in sufficient quantity. In this section, we study some characteristics of the alignments thus produced (see Table 1).

Regarding translation tables first, we observe that those obtained from Cutnalign contain many more entries than those produced by Anymalign alone[3] (three times more in average), except for Anymalign-1 in Finnish–English. Nevertheless, they are still much smaller than tables obtained from MGIZA++, as they contain twice less entries in average. In addition, the average length of those entries is almost equal to that of those in MGIZA++'s translation tables, while those produced by Anymalign are much shorter: producing a translation table from alignment links allows to make up for the lack of long n-grams as desired.

Secondly, we study the alignment links themselves. The column "Links" of Table 1 shows that our method produces more alignment links than MGIZA++: between 1.5 and 3 times more, depending on the task. The last column gives the main reason: alignment blocks extracted by our method, i.e. rectangles obtained at maximal recursion depth, are always longer than minimum blocks obtained from MGIZA++'s alignments (+ 26% in average). Since we systematically align all source words with all target words in such a rectangle, and since all words of a sentence pair are therefore necessarily aligned, the total number of alignments produced is naturally high. This also explains the fact that the number of entries in our translation tables is always much lower than those obtained from MGIZA++, as the latter produces 0–1 alignments that are at the origin of numerous phrases extracted during the constitution of the table by Moses (*grow-diag-final-and* heuristic by default) (Ayan and Dorr, 2006). Despite this, alignments produced by our method lead to state-of-the-art scores in two machine translation tasks over three in our experiments.

4 Conclusion

We have presented a sub-sentential alignment method based on a recursive binary segmentation process of the alignment matrix between a source sentence and its translation. Inspired by work on alignment by Wu (1997) and Deng et al. (2006) and work on document clustering by Zha et al. (2001), we have shown that despite its simplicity, this method leads to state-of-the-art results in two tasks over three in our experiments. When fed with Anymalign's scores, it yields significant gains (up to 4.6 BLEU points in French–English) in comparison with Anymalign alone. These experiments confirm that Anymalign's main handicap concerns the translation of long n-grams. A complementary alignment step, strictly speaking, is thus desired in order to improve its results in machine translation. The alignment method proposed here is simple, symmetric with respect to the translation direction, and the use of local computations makes it scale up easily. Many improvements are possible, amongst which the use of early stopping criteria during segmentation of the alignment matrix so as to trade alignment granularity for confidence; the use more sophisticated metrics for scoring blocks, or the exploration of richer (e.g. ternary) segmentation schemes, enabling to account for more complex linguistic constructs.

[3]These tables were produced by running Anymalign for an identical amount of time in all configurations, which explains why larger values of the length parameter lead to smaller tables—see details in (Lardilleux et al., 2011b).

References

Ayan, Necip Fazil and Bonnie J. Dorr. 2006. Going beyond AER: An extensive analysis of word alignments and their impact on MT. In *Proc. of Coling/ACL'06*, pages 9–16, Sydney, Australia.

Brown, Peter, John Cocke, Stephen Della Pietra, Vincent Della Pietra, Fredrick Jelinek, Robert Mercer, and Paul Roossin. 1988. A statistical approach to language translation. In *Proc. of Coling'88*, pages 71–76, Budapest.

Brown, Peter, Stephen Della Pietra, Vincent Della Pietra, and Robert Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Dagan, Ido and Ken Church. 1994. Termight: identifying and translating technical terminology. In *Proc. of the 4th conference on Applied natural language processing*, pages 34–40, Stuttgart.

DeNero, John and Dan Klein. 2007. Tailoring word alignments to syntactic machine translation. In *Proc. of ACL'07*, pages 17–24, Prague.

Deng, Yonggang and William Byrne. 2005. HMM word and phrase alignment for statistical machine translation. In *Proc. of HLT/EMNLP'05*, pages 169–176, Vancouver, British Columbia, Canada, October.

Deng, Yonggang, Shankar Kumar, and William Byrne. 2006. Segmentation and alignment of parallel text for statistical machine translation. *Natural Language Engineering*, 13(3):235–260.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Fraser, Alexander and Daniel Marcu. 2007. Getting the structure right for word alignment: LEAF. In *Proc. of EMNLP/CoNLL'07)*, pages 51–60, Prague.

Fung, Pascale and Kenneth Church. 1994. K-vec: A new approach for aligning parallel texts. In *Proc. of Coling'94*, volume 2, pages 1096–1102, Kyōto.

Fung, Pascale and Lo Yuen Yee. 1998. An IR approach for translating new words from nonparallel, comparable texts. In *Proc. of Coling/ACL'98*, volume 1, pages 414–420, Montreal.

Gale, William and Kenneth Church. 1991. Identifying word correspondences in parallel texts. In *Proc. of the 4th DARPA workshop on Speech and Natural Language*, pages 152–157, Pacific Grove.

Ganchev, Kuzman, João Graça, and Ben Taskar. 2008. Better alignments = better translations? In *Proc. of ACL'08*, pages 986–993, Columbus, Ohio.

Gao, Qin and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus (Ohio, USA).

Gaussier, Éric and Jean-Marc Langé. 1995. Modèles statistiques pour l'extraction de lexiques bilingues. *Traitement Automatique des Langues*, 36(1-2):133–155.

Johnson, Howard, Joel Martin, George Foster, and Roland Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *Proc. of EMNLP/CoNLL'07*, pages 967–975, Prague.

Koehn, Philipp, Franz Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. of HLT/NAACL'03)*, pages 48–54, Edmonton.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of ACL'07*, pages 177–180, Prague.

Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proc. of MT Summit X)*, pages 79–86, Phuket.

Lardilleux, Adrien and Yves Lepage. 2009. Sampling-based multilingual alignment. In *Proc. of RANLP*, pages 214–218, Borovets.

Lardilleux, Adrien, Yves Lepage, and François Yvon. 2011a. The contribution of low frequencies to multilingual sub-sentential alignment: a differential associative approach. *International Journal of Advanced Intelligence*, 3(2):189–217.

Lardilleux, Adrien, François Yvon, and Yves Lepage. 2011b. Généralisation de l'alignement sous-phrastique par échantillonnage. In *Proc. of TALN 2011*, volume 1, pages 507–518, Montpellier, France.

Liang, Percy, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proc. of the HLT/NAACL'06*, pages 104–111, New York City.

Luo, Juan, Adrien Lardilleux, and Yves Lepage. 2011. Improving sampling-based alignment by investigating the distribution of n-grams in phrase translation tables. In *Proc. of PACLIC 25*, pages 150–159, Singapour.

Marcu, Daniel and Daniel Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. of EMNLP'02*, pages 133–139, Philadelphia.

Melamed, Dan. 2000. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249.

Moore, Robert. 2004. On log-likelihood-ratios and the significance of rare events. In *Proc. of EMNLP'04*, pages 333–340, Barcelona.

Moore, Robert. 2005. Association-based bilingual word alignment. In *Proc. of the ACL Workshop on Building and Using Parallel Texts*, pages 1–8, Ann Arbor.

Och, Franz and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.

Och, Franz. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL'03*, pages 160–167, Sapporo.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. of ACL'02*, pages 311–318, Philadelphia.

Smadja, Frank, Vasileios Hatzivassiloglou, and Kathleen McKeown. 1996. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proc. of AMTA'06)*, pages 223–231, Cambridge, August.

Vogel, Stephan, Hermann Ney, and Christoph Tillman. 1996. Hmm-based word alignment in statistical translation. In *Proc. of Coling'96*, pages 836–841, Copenhague.

Vogel, Stephan. 2005. PESA: Phrase pair extraction as sentence splitting. In *Proc. of MT Summit X*, pages 251–258, Phuket.

Wu, Dekai. 1997. Stochastic inversion transduction grammar and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–404.

Zha, Hongyuan, Xiaofeng He, Chris Ding, Horst Simon, and Ming Gu. 2001. Bipartite graph partitioning and data clustering. In *Proc. of the 10th international conference on Information and knowledge management*, pages 25–32, Atlanta.