Alignement sous-phrastique hiérarchique avec Anymalign

Adrien Lardilleux[1] François Yvon[1,2] Yves Lepage[3]
(1) LIMSI-CNRS
(2) Université Paris-Sud
(3) Université Waseda, Japon
adrien.lardilleux@limsi.fr, francois.yvon@limsi.fr

RÉSUMÉ

Nous présentons un algorithme d'alignement sous-phrastique permettant d'aligner très facilement un couple de phrases à partir d'une matrice d'alignement pré-remplie. Cet algorithme s'inspire de travaux antérieurs sur l'alignement par segmentation binaire récursive ainsi que de travaux sur le clustering de documents. Nous évaluons les alignements produits sur des tâches de traduction automatique et montrons qu'il est possible d'atteindre des résultats du niveau de l'état de l'art, affichant des gains très conséquents allant jusqu'à plus de 4 points BLEU par rapport à nos travaux antérieurs, à l'aide une méthode très simple, indépendante de la taille du corpus à traiter, et produisant directement des alignements symétriques. En utilisant cette méthode en tant qu'extension à l'outil d'extraction de traductions Anymalign, nos expériences nous permettent de cerner certaines limitations de ce dernier et de définir des pistes pour son amélioration.

ABSTRACT

Hierarchical sub-sentential alignment with Anymalign

We present a sub-sentential alignment algorithm that aligns sentence pairs from an existing alignment matrix in a very easy way. This algorithm is inspired by previous work on alignment by recursive binary segmentation and on document clustering. We evaluate the alignments produced 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 our previous work, with a method that is very simple, independent of the size of the corpus to be aligned, and can directly produce symmetric alignments. When using this method as an extension of the translation extraction tool Anymalign, our experiments allow us to determine some of its limitations and to define possible leads for further improvements.

MOTS-CLÉS : corpus parallèle ; alignement sous-phrastique ; traduction automatique statistique.

KEYWORDS: parallel corpus ; sub-sentential alignment ; statistical machine translation.

Actes de la conférence conjointe JEP-TALN-RECITAL 2012, volume 2: TALN, pages 113–126,
Grenoble, 4 au 8 juin 2012. ©2012 ATALA & AFCP

1 Introduction

L'alignement sous-phrastique consiste à identifier des traductions d'unités textuelles à partir d'un corpus parallèle aligné en phrases, c'est-à-dire dont les phrases ont été préalablement mises en correspondance avec leur traduction. Cette tâche constitue la première étape du processus d'entraînement de la plupart des systèmes de traduction automatique fondée sur les données (traduction statistique ou par l'exemple). L'approche la plus répandue est actuellement la traduction automatique statistique par segments (n-grammes de mots), où le modèle central prend la forme d'une table de traductions, obtenue à partir de correspondances sous-phrastiques. Cette table consiste en une liste pré-calculée de couples de segments (*source*, *cible*), à chacun desquels est associé un certain nombre de scores reflétant la probabilité que *source* se traduise par *cible*.

Le problème de l'identification d'associations sous-phrastiques à partir de textes parallèles, entre mots isolés ou n-grammes de mots par exemple, est bien connu, et de nombreuses propositions ont été faites pour le résoudre. On peut grossièrement classer ces méthodes en deux catégories. La première, l'approche *probabiliste*, introduite par Brown *et al.* (1988), considère le problème d'identifier des *liens* entre mots ou groupes de mots dans des phrases parallèles. Cette approche consiste à définir un modèle probabiliste du texte parallèle, dont les paramètres sont estimés par un processus de maximisation global qui considère toutes les associations possibles du corpus en même temps. Le but est de déterminer le meilleur ensemble de liens d'alignement entre les mots source et cible de chaque couple de phrases parallèles. Les plus connus dans cette catégorie sont les modèles IBM (Brown *et al.*, 1993), permettant d'aligner des mots isolés, et qui ont donné lieu à une impressionnante liste de variantes et d'améliorations (voir par exemple les travaux de Vogel *et al.* (1996); Wu (1997); Deng et Byrne (2005); Liang *et al.* (2006); Fraser et Marcu (2007); Ganchev *et al.* (2008), pour ne citer qu'eux). La généralisation des modèles d'alignement de mots à l'alignement de segments s'avère être un problème bien plus difficile, et au vu des déficiences des propositions de Marcu et Wong (2002) et Vogel (2005), de tels alignements sont généralement produits en combinant des alignements de mots 1–n asymétriques (« orientés ») dans les deux directions à l'aide d'heuristiques (Koehn *et al.*, 2003; DeNero et Klein, 2007). Une fois l'ensemble de ces liens d'alignement constitué, il est possible d'attribuer des scores à chacun des couples de segments extraits.

La seconde approche, *associative* (qualifiée d'*heuristique* par Och et Ney (2003)), a été introduite par Gale et Church (1991). Celle-ci ne nécessite pas de modèle d'alignement : pour détecter des traductions, elle repose sur des mesures d'indépendance statistique telles que, par exemple, le coefficient de Dice, l'information mutuelle (Gale et Church, 1991; Fung et Church, 1994), ou le rapport de vraisemblance (Dunning, 1993) — voir aussi les travaux plus récents de Melamed (2000) et Moore (2005). On limite généralement les tests à une liste d'associations candidates pré-calculée à partir de motifs et de filtres, en se concentrant par exemple uniquement sur les n-grammes de mots les plus fréquents. Dans cette approche, on utilise un processus de maximisation locale, où chaque segment est traité indépendamment des autres. Cette approche permet généralement d'extraire directement des couples de traductions. Dans ce courant, on trouve par exemple les travaux de Gale et Church (1991), qui ont été depuis étendus aux corpus non strictement parallèles (Fung et Church, 1994; Fung et Yee, 1998), de Dagan et Church (1994); Gaussier et Langé (1995); Smadja *et al.* (1996) pour apprendre des associations de segments ou de termes, ou encore des travaux ayant recours à diverses mesures d'association, telles que le G^2 (Gale et Church, 1991) ou le ϕ^2 (Dunning, 1993; Moore, 2004, 2005). Dans

un second temps, on peut induire des liens d'alignement à la façon des méthodes probabilistes, comme l'a proposé Melamed (2000) avec le *competitive linking*.

L'approche probabiliste est la plus répandue, principalement du fait de sa bonne intégration avec la traduction automatique statistique, dont elle constitue un fondement depuis l'introduction des modèles IBM (Brown *et al.*, 1993). Les deux approches présentent des forces et faiblesses complémentaires, comme l'ont montré par exemple les travaux de Johnson *et al.* (2007), où les associations extraites à partir d'alignements de mots sont ensuite filtrées selon des mesures d'association.

Nous avons récemment proposé une méthode d'extraction de traductions de segments sous-phrastiques (Lardilleux *et al.*, 2011a), nommée *Anymalign*, qui s'attaque à un certain nombre de problèmes souvent négligés dans le domaine. En particulier, cette méthode permet le traitement d'un nombre quelconque de langues simultanément, ne fait aucune distinction entre source et cible, est massivement parallélisable, passe facilement à l'échelle, et est très simple à implémenter. Cette méthode, qui s'inscrit dans le courant des méthodes associatives, est meilleure que l'état de l'art sur des tâches de constitution de lexiques bilingues. Les résultats obtenus lorsqu'on l'utilise pour construire des modèles de traductions statistiques s'avèrent toutefois inférieurs aux méthodes standard (Lardilleux *et al.*, 2011b).

Une des hypothèses que nous avons précédemment émises pour expliquer ces résultats contrastés est qu'Anymalign ne comporte pas de phase d'alignement à proprement parler. Cette méthode ne produit donc pas de *liens* à la manière des méthodes probabilistes, mais directement des tables de traductions avec leurs scores associés. Ces tables ont des profils très différents de celles extraites à partir d'alignements produits par les méthodes probabilistes, principalement en termes de distribution des n-grammes (Luo *et al.*, 2011). En particulier, malgré de récentes améliorations (Lardilleux *et al.*, 2011b), la quantité de traductions de longs n-grammes est relativement faible comparée aux tables de traductions obtenues à partir des méthodes probabilistes. Dans cet article, nous proposons une extension à notre méthode lui permettant de produire des liens d'alignement, à la manière des approches probabilistes, tout en conservant le caractère local de la recherche des traductions propre aux approches associatives. Notre but principal n'est pas ici de proposer une nouvelle méthode d'alignement destinée à améliorer les outils de l'état de l'art, mais d'essayer de mieux comprendre les limitations actuelles d'Anymalign, en l'utilisant ici de manière non plus directe, mais détournée, pour construire le modèle de traduction. La méthode pour construire des alignements, très simple, est donc indépendante d'Anymalign et pourrait être remplacée par tout autre procédé équivalent.

Cet article est organisé comme suit : la section 2 présente en détail chacune des étapes qui compose notre méthode d'alignement, la section 3 présente une évaluation de la méthode sur des tâches de traduction automatique et une analyse des résultats obtenus, et la section 4 conclut ces travaux.

2 Description de la méthode

En un mot, notre méthode consiste à segmenter chaque couple de phrases d'un corpus parallèle de façon binaire, déterminer parmi les deux segments cible obtenus lequel est la bonne traduction de chacun des deux segments source (traduction monotone ou inversée), et recommencer

récursivement sur chacun des deux couples de segments obtenus.

Ces travaux s'inspirent fortement de ceux de Wu (1997) et Deng *et al.* (2006). Les premiers présentent des grammaires de transduction inversibles où les parties source et cible d'un couple de phrases alignées sont analysées simultanément selon un arbre de dérivation binaire dont la particularité est de permettre l'inversion des constituants d'une langue à l'autre à n'importe quel niveau de l'arbre (approche *bottom-up*). On retrouve un concept similaire dans les seconds, où on extrait des bi-segments plus ou moins grossiers à partir de textes parallèles non préalablement alignés en phrases en appliquant une segmentation binaire de façon itérative selon le principe « diviser pour régner » (approche *top-down*).

Nos travaux se rapprochent davantage de ces derniers en ce sens que nous ne nous intéressons qu'à une procédure simple ne reposant que sur des décomptes au niveau lexical, plutôt que sur une grammaire telle qu'utilisée par Wu. Néanmoins, alors que Deng *et al.* produisent des alignements de segments plus ou moins grossiers à partir d'un bi-texte non préalablement aligné en phrases, dans le but de simplifier des tâches subséquentes d'alignement sous-phrastique par exemple, notre but est plus classiquement d'aligner directement le grain le plus fin possible, ici le mot typographique, à partir de textes préalablement alignés en phrases. Le critère que nous utilisons pour décider de la segmentation d'un couple de phrases est adapté en conséquence.

2.1 Matrice d'alignement

Notre point de départ se compose :
– d'un bi-texte préalablement aligné en phrases ;
– d'une fonction w associant à chaque couple de mots (*source*, *cible*) du bi-texte un score reflétant la force du lien de traduction entre *source* et *cible*.

Plusieurs définitions de w sont possibles ; il est néanmoins naturel de la définir de façon endogène à partir des occurrences des mots sur l'ensemble du bi-texte. En ce qui nous concerne, les scores que nous utiliserons seront dans un premier temps obtenus à partir des sorties d'Anymalign. Nous verrons par la suite que ceux-ci mènent à de meilleurs résultats que d'autres scores obtenus à partir de modèles plus répandus, principalement du fait de la grande redondance des sorties d'Anymalign, qui permet de renforcer les scores de traductions se produisant dans des contextes variés.

Par la suite donc, le score $w(s,c)$ entre un mot source s et un mot cible c sera défini comme le produit des deux probabilités de traduction orientées $\mathrm{p}(s|c) \times \mathrm{p}(c|s)$, celles-ci étant calculées à partir des décomptes associés aux traductions produites par Anymalign :

$$\begin{aligned} w(s,c) &= \mathrm{p}(s|c) \times \mathrm{p}(c|s) \\ &= \frac{\sum_{n=1}^{N} [\![(s,c) \in (S_n, C_n)]\!] k_n}{\sum_{n'=1}^{N} [\![s \in S_{n'}]\!] k_{n'}} \times \frac{\sum_{n=1}^{N} [\![(s,c) \in (S_n, C_n)]\!] k_n}{\sum_{n'=1}^{N} [\![c \in C_{n'}]\!] k_{n'}} \\ &= \frac{\left(\sum_{n=1}^{N} [\![(s,c) \in (S_n, C_n)]\!] k_n\right)^2}{\left(\sum_{n'=1}^{N} [\![s \in S_{n'}]\!] k_{n'}\right) \times \left(\sum_{n'=1}^{N} [\![c \in C_{n'}]\!] k_{n'}\right)} \end{aligned}$$

avec :
– $[\![x]\!] = 1$ si x est vrai, 0 sinon ;

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

$$
\begin{aligned}
w(pays, country) &= \mathrm{p}(pays|country) \times \mathrm{p}(country|pays) \\
&= \frac{17\,717 + 4\,057 + 2\,007}{151\,190 + 17\,717 + 10\,865 + 6\,284 + 4\,057 + 3\,742 + 2\,007} \\
&\quad \times \frac{17\,717 + 4\,057 + 2\,007}{17\,717 + 4\,057 + 2\,007 + 122} \\
&\approx 0,121
\end{aligned}
$$

FIG. 1 – Exemple de calcul de score entre le mot source *pays* et le mot cible *country* sur un sous-ensemble d'une table de traductions produite par Anymalign à partir des parties française et anglaise du corpus parallèle Europarl (Koehn, 2005).

– N le nombre d'entrées (couples de segments source–cible) dans la table de traductions produite par Anymalign ;
– S_n (resp. C_n) le segment source (resp. cible) d'une entrée de la table de traductions ;
– k_n le décompte associé au couple (S_n, C_n) dans la table de traductions. Ce nombre n'est pas en soi un indicateur de la qualité de l'entrée ; il s'agit simplement du nombre de fois où le couple a été produit par Anymalign (voir détails dans (Lardilleux *et al.*, 2011a)).

La figure 1 donne un exemple.

En pratique, ce que nous faisons ici revient à partir d'une table de traductions pour aller vers des liens d'alignements — pour retourner ultimement vers une nouvelle table de traductions. Cela va à rebours des usages du domaine, qui construisent la table de traductions à partir de l'ensemble des liens d'alignements calculés sur un corpus parallèle. Cette particularité ouvre de nouvelles pistes pour l'amélioration de la qualité des liens d'alignements et d'une table de traductions, l'amélioration des uns pouvant avoir des répercussions sur l'autre, et vice-versa, de façon itérative, à la manière des approches probabilistes reposant par exemple sur l'algorithme Espérance Maximisation. Cela sort néanmoins du cadre de cet article, et nous nous consacrons pour l'instant au passage de la table de traductions vers les liens d'alignements.

2.2 Critère de segmentation

Le critère de segmentation décrit ci-après est issu des travaux de Zha *et al.* (2001) sur le clustering de documents. Leur problème consiste à partitionner de façon optimale un graphe biparti représentant les occurrences d'un ensemble de termes au sein d'un ensemble de documents. Nous le transposons à la recherche du meilleur alignement entre l'ensemble des mots d'une

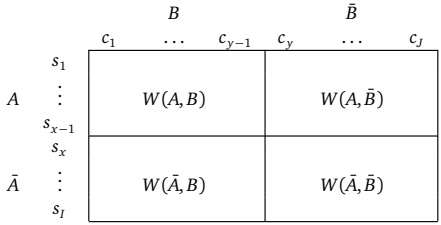

FIG. 2 – Représentation schématique de la segmentation d'un couple de phrases $S = A\,.\,\bar{A}$ et $C = B\,.\,\bar{B}$.

phrase source et l'ensemble des mots d'une phrase cible.

Pour cela, nous considérons un couple de phrases (S,C) du corpus parallèle, où la phrase source S est constituée de I mots source et la phrase cible C est constituée de J mots cible : $S = [s_1 \dots s_I]$ et $C = [c_1 \dots c_J]$. Nous considérons par ailleurs des indices de coupure x et y définissant une segmentation binaire des phrases source et cible (le symbole « . » désigne la concaténation de chaînes de mots) :

$$\begin{array}{lllll} S = A\,.\,\bar{A} & \text{avec} & A = [s_1 \dots s_{x-1}] & \text{et} & \bar{A} = [s_x \dots s_I] \\ C = B\,.\,\bar{B} & \text{avec} & B = [c_1 \dots s_{y-1}] & \text{et} & \bar{B} = [c_y \dots c_J] \end{array}$$

Le choix de x et y sera guidé par la somme W des scores d'association entre chacun des mots source et cible d'un couple de segments $(X,Y) \in \{A,\bar{A}\} \times \{B,\bar{B}\}$:

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

On retrouve l'ensemble des notations utilisées dans la figure 2, qui donne une représentation schématique de la segmentation d'un couple de phrases.

On définit alors :

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

Notons que $\text{cut}(X,Y) = \text{cut}(\bar{X},\bar{Y})$. Dans notre cas, une valeur faible indique que les scores d'association entre les mots de X et $\bar{Y}$ d'une part, et entre ceux de $\bar{X}$ et Y d'autre part, sont faibles également, autrement dit que ces deux couples de segments ont peu de chances d'être de bonnes traductions, (X,Y) et $(\bar{X},\bar{Y})$ constituant alors *éventuellement* de bonnes traductions. Idéalement donc, nous désirons déterminer le couple (x,y) qui mène à la plus petite valeur de $\text{cut}(X,Y)$ possible. Zha *et al.* (2001) pointent néanmoins le fait que cette quantité tend à produire des segments (clusters de documents dans leur cas) déséquilibrés du fait de l'absence de normalisation, et en proposent par conséquent une version normalisée :

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

Cette variante permet de rajouter une contrainte de densité sur (X,Y) et $(\bar{X},\bar{Y})$, ce qui est partiellement satisfait par l'introduction des dénominateurs dans l'expression ci-dessus. Sa valeur est comprise entre 0 et 2.

```
procédure aligner(S, C) :
    si longueur(S) = 1 ou longueur(C) = 1 :
        lier chacun des mots de S avec chacun des mots de C
        arrêt procédure
    minNcut = 2
    (X, Y) = (S, C)
    pour chaque (i, j) ∈ {2...I} × {2...J} :
        si Ncut(A, B) < minNcut :
            minNcut = Ncut(A, B)
            (X, Y) = (A, B)
        si Ncut(A, B̄) < minNcut :
            minNcut = Ncut(A, B̄)
            (X, Y) = (A, B̄)
    aligner(X, Y)
    aligner(X̄, Ȳ)
```

FIG. 3 – Algorithme d'alignement récursif.

Notre problème consiste finalement à déterminer le couple (x, y) qui minimise Ncut. Bien que des méthodes de recherche performantes existent et sont couramment utilisées en théorie des graphes, nos « graphes » (couples de phrases) sont petits en pratique : environ 30 mots par phrase en moyenne dans le corpus Europarl que nous utilisons pour la suite de nos expériences. Nous nous contentons donc par la suite de déterminer la meilleure segmentation en testant toutes les coupures possibles.

2.3 Algorithme d'alignement

À partir du critère défini précédemment, nous pouvons segmenter et aligner un couple de phrases de façon récursive. À chaque étape, nous testons tous les couples (x, y) possibles afin de déterminer le plus faible Ncut. Le pire des cas se produit lorsque la matrice est coupée de la façon la plus déséquilibrée possible ; la complexité de l'algorithme est donc cubique (de l'ordre de $I \times J \times \min(I, J)$). Pour un couple (x, y) donné, nous calculons deux valeurs : l'une correspondant à un alignement monotone (Ncut(A, B)) et l'autre à une inversion des deux segments (Ncut$(A, \bar{B})$). Le processus est alors appliqué sur chacun des couples de segments correspondant au Ncut minimal. Il s'arrête lorsqu'un segment ne comporte qu'un seul mot : les alignements produits sont tous de multiplicité 1–n ou n–1, et il en résulte que tous les mots sont nécessairement alignés. Des variantes où le processus récursif s'arrête plus tôt sont envisageables, en fixant un seuil sur Ncut par exemple, auquel cas les alignements produits seraient de multiplicité m–n. Nous gardons cette possibilité pour des recherches futures.

La figure 3 présente l'algorithme complet, et la figure 4 illustre le processus sur deux exemples réels. Dans la suite de l'article, nous ferons référence à cet algorithme sous le nom « Cutnalign ».

L'algorithme en lui-même est indépendant de la taille du corpus parallèle à aligner, car chaque couple de phrases est traité indépendemment des autres. On peut donc très facilement paralléliser l'alignement d'un corpus : le temps d'alignement total est divisé par le nombre de processeurs à disposition. Un autre avantage est que les alignements produits sont symétriques tout au long du processus, contrairement à des modèles plus répandus comme les modèles IBM qui produisent de

	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	0	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
,	0,001	**0,445**	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0,001	ϵ	0,001	ϵ	0,001
c'	ϵ	ϵ	0,001	ϵ	ϵ	ϵ	0	**0,036**	0,001	ϵ	ϵ	ϵ	ϵ
est	ϵ	ϵ	0,001	ϵ	ϵ	ϵ	0	**0,223**	0,016	ϵ	0,001	ϵ	0,001
un	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0,005	ϵ	ϵ	ϵ	ϵ	ϵ
droit	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0	ϵ	ϵ	**0,084**	ϵ	ϵ	ϵ
à	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	0,002	0,009	ϵ	**0,002**	ϵ	ϵ
information	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	**0,499**	ϵ
que	ϵ	ϵ	0,002	ϵ	ϵ	0,001	ϵ	0,002	0,001	ϵ	0,001	ϵ	ϵ
réclament	0	0	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
concitoyens	0	ϵ	ϵ	0,001	**0,323**	**0,009**	ϵ	ϵ	ϵ	0	ϵ	0	ϵ
.	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	0,001	0,001	ϵ	ϵ	ϵ	ϵ	**0,954**

FIG. 4 – Deux exemples de segmentation-alignement. Les nombres dans les cellules correspondent à la valeur de la fonction w, avec ϵ une valeur non nulle inférieure à 0,001. Une valeur de 0 indique que les deux mots n'apparaissent jamais ensemble dans la table de traductions. Les points d'alignement retenus par l'algorithme, correspondant au niveau de récursion maximal, sont indiqués en gras. Dans le premier cas, la traduction est monotone à l'exception de l'inversion de l'ordre du nom et de l'adjectif (*exécution budgétaire/budgetary implementation*), la plupart des liens d'alignement se situent donc sur la diagonale. Le second cas, plus complexe, rend compte de l'inversion de l'ordre de propositions au sein de la phrase.

meilleurs résultats lorsqu'exécutés dans les deux sens de traduction puis leurs sorties combinées à l'aide d'heuristiques.

3 Évaluation

3.1 Description des expériences

Nous évaluons notre méthode d'alignement en tant que premier module d'un système de traduction automatique statistique par segments. Nous utilisons pour cela le système de traduction Moses (Koehn *et al.*, 2007), et des données constituées d'un échantillon du corpus parallèle Europarl (Koehn, 2005), couvrant trois couples de langues : finnois–anglais (langue agglutinante–langue isolante), français–anglais, et portugais–espagnol (langues très proches). Pour chacun, nous utilisons un jeu d'entraînement de 350 000 couples de phrases (30 mots par phrase en moyenne en anglais), et des jeux de développement et de test de 2 000 couples de phrases chacun. L'optimisation des systèmes est réalisée à l'aide de la procédure MERT (Och, 2003). Sauf mention contraire, un modèle de réordonnancement lexicalisé est utilisé.

Nous comparons quatre approches :

MGIZA++ (Gao et Vogel, 2008), implémentant les modèles IBM (Brown *et al.*, 1993) et le modèle caché de Markov de Vogel *et al.* (1996). Intégré à Moses, il s'agit toujours de la référence du domaine. Nous l'utilisons avec ses paramètres par défaut, en enchaînant 5 itérations de chacun des modèles IBM1, HMM, IBM3 et IBM4. Une table de traductions est ensuite produite à partir des alignements à l'aide des outils de Moses.

Anymalign (Lardilleux *et al.*, 2011a), produisant directement des tables de traductions. Cet outil pouvant être arrêté à tout moment, nous fixons son temps d'exécution de façon à ce qu'il soit exécuté pendant la même durée que MGIZA++. Nous répétons la même expérience en faisant varier son paramètre « -i », permettant de contrôler la longueur des segments qu'il produit en sortie, de 1 à 4 (voir détails dans (Lardilleux *et al.*, 2011b)). Nous y faisons référence par la suite sous les noms « Anymalign-1 » à « Anymalign-4 ». Le modèle de réordonnancement utilisé dans cette configuration n'est qu'un simple modèle basé sur la distance entre mots, car Anymalign seul ne peut fournir l'information nécessaire à un modèle de réordonnancement lexicalisé.

Anymalign + Cutnalign : nous appliquons l'algorithme décrit dans la section précédente à chacune des quatre tables de traductions produites par Anymalign-1 à Anymalign-4. Les alignements obtenus sont utilisés pour construire de nouvelles tables de traductions à l'aide du jeu d'outils de Moses.

Simples probabilités + Cutnalign : cette configuration permet d'évaluer non pas l'algorithme proposé précédemment, mais le choix de la fonction w, qui sert de base à l'algorithme. Nous utilisons pour cela un score d'association très simple : la probabilité qu'un mot source et un mot cible soient traductions l'un de l'autre (produit des deux probabilités de traduction), cette probabilité étant calculée à partir de leurs occurrences dans le corpus d'entraînement. La définition de w est donc ici la même qu'à la section 2.1, à deux différences près :
- les décomptes ne sont pas effectués sur une table de traductions produite par Anymalign, mais directement sur le bi-texte d'entraînement ;
- $k_n = 1, \forall n$.

Les traductions sont évaluées selon les mesures BLEU (Papineni *et al.*, 2002) et TER (Snover *et al.*, 2006, contrairement à BLEU, des scores faibles sont meilleurs).

3.2 Résultats

Les résultats sont présentés dans le tableau 1. Sur chacune des trois tâches, Anymalign (version « de base ») est plus ou moins en retrait par rapport à MGIZA++. L'utilisation du paramètre « -i » permet de réduire cet écart de moitié environ, à l'exception notable du couple finnois–anglais (langue agglutinante–langue isolante), ce qui est conforme aux résultats présentés dans (Lardilleux *et al.*, 2011b).

L'ajout de Cutnalign mène à un gain considérable dans toutes les configurations : de 1,6 à 4,6 points BLEU (fr–en, Anymalign-1 + Cutnalign), avec un gain moyen de 2,6 points BLEU et 2,7 points TER. Anymalign+Cutnalign est toujours en retrait de 1,1 à 1,6 point BLEU en finnois–anglais par rapport à MGIZA++, mais produit des résultats de même qualité, voire meilleurs mais de façon non significative, en français–anglais et portugais–espagnol.

L'approche « simples probabilités + Cutnalign » produit des résultats de qualité intermédiaire,

Tâche	Système	BLEU (%)	TER (%)	Entrées (millions)	Long. des entrées	Liens	Long. des blocs extraits
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
	Simples prob. + Cutnalign	19,09	67,09	5,5	3,23	74	1,78
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
	Simples prob. + Cutnalign	27,97	56,85	10,2	3,95	54	1,62
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
	Simples prob. + Cutnalign	37,71	49,04	13,9	4,09	50	1,41

TAB. 1 – Récapitulatif des résultats obtenus dans nos expériences. Les deux premières colonnes de nombres donnent les scores obtenus en traduction automatique. Les deux colonnes du milieu présentent les caractéristiques des tables de traductions : nombre d'entrées et longueur de celles-ci en nombre de mots. Les deux dernières colonnes présentent les caractéristiques des alignements avant production de la table de traductions : nombre moyen de liens d'alignements par couple de phrases d'entraînement et longueur moyenne de la partie source des blocs minimaux extraits (après détermination des segments alignés cohérents avec les liens d'alignement).

généralement entre Anymalign « de base » et Anymalign + Cutnalign. Cela montre que le choix de la fonction w a une grande influence sur le comportement de la méthode d'alignement que nous avons proposée. En admettant que la fonction définie dans ces expériences est une des plus simples qui soient, nous pouvons anticiper que de nombreuses améliorations sont possibles, comme le montrent les résultats obtenus en initiant la méthode à partir des tables de traductions d'Anymalign.

3.3 Regard sur les alignements

Comme précisé en introduction, l'une des raisons pour laquelle nous avons proposé cette méthode d'alignement est que, malgré de récentes améliorations, Anymalign peine toujours à extraire suffisamment de traductions de longs n-grammes. Dans cette section, nous étudions quelques caractéristiques des alignements produits par la méthode que nous avons proposée. Elles sont présentées dans le tableau 1.

En ce qui concerne les tables de traductions d'abord, on constate que celles qui sont obtenues à partir de Cutnalign contiennent un nombre beaucoup plus important d'entrées que les tables correspondantes produites par Anymalign seul[1] (trois fois plus en moyenne), à l'exception notable d'Anymalign-1 en finnois–anglais. Elles sont néanmoins toujours beaucoup plus petites que les tables obtenues à partir de MGIZA++ et contiennent deux fois moins d'entrées en moyenne. La longueur moyenne de ces entrées est en outre quasiment égale à celles des tables de traductions de MGIZA++, alors que celles produites par Anymalign sont beaucoup plus courtes : la production d'une table de traductions à partir de liens d'alignement permet bien de combler le manque de longs n-grammes comme nous le désirions.

Dans un second temps, nous étudions plus en détail les liens d'alignement à proprement parler, tels qu'ils sont avant la production des tables de traductions. La colonne « Liens » du tableau 1 montre que le nombre de liens d'alignement produits par notre méthode est bien supérieur à celui de ceux produits par MGIZA++ : entre 1,5 et 3 fois plus selon la tâche. La dernière colonne en donne la principale raison : les blocs d'alignement extraits par notre méthode, c'est-à-dire les rectangles obtenus au niveau de récursion maximal, sont toujours plus longs que les blocs minimums obtenus à partir des alignements de MGIZA++ (+ 26 % en moyenne). Comme nous alignons systématiquement tous les mots source avec tous les mots cible d'un tel rectangle, et tous les mots d'un couple de phrases étant par conséquent nécessairement alignés, le nombre total de liens produits est naturellement élevé. Cela explique également le fait que le nombre d'entrées dans les tables de traductions est toujours beaucoup plus faible que dans celles obtenues à partir de MGIZA++, ce dernier produisant des alignements de multiplicité 0–1 qui sont à l'origine de l'extraction de très nombreux segments lors de la constitution de la table par Moses (heuristique *grow-diag-final-and* par défaut) (Ayan et Dorr, 2006). Malgré cela, les alignements produits par notre méthode permettent d'atteindre des scores identiques à l'état de l'art dans deux tâches de traduction automatique sur trois dans nos expériences.

[1] Ces tables ont été produites en exécutant Anymalign pendant un temps identique dans les quatre configurations, ce qui explique pourquoi de plus grandes valeurs de l'option « -i » mènent à de plus petites tables — voir détails dans (Lardilleux *et al.*, 2011b).

4 Conclusion

Nous avons présenté une méthode d'alignement sous-phrastique fondée sur un découpage récursif binaire de la matrice d'alignement entre une phrase source et sa traduction. Inspirée des travaux de Wu (1997) et Deng *et al.* (2006) sur l'alignement et de Zha *et al.* (2001) sur le clustering de documents, nous avons montré qu'en dépit de sa simplicité, cette méthode produit des résultats du niveau de l'état de l'art dans deux tâches sur trois dans nos expériences. Couplée à Anymalign, elle permet des gains conséquents (jusqu'à 4,6 points BLEU en français–anglais) par rapport à l'utilisation d'Anymalign seul. Nos expériences ont confirmé que le principal handicap d'Anymalign concerne bien les traductions de longs n-grammes. Une étape complémentaire d'alignement au sens strict du terme se révèle donc souhaitable pour améliorer ses résultats en traduction automatique, car elle permet de combler la plupart de ses manques en termes de traductions de longs segments. La méthode d'alignement proposée ici est relativement simple, symétrique du point de vue du sens de la traduction, et le caractère local du calcul des alignements lui permet de passer facilement à l'échelle. Dans l'optique d'améliorer les alignements, de multiples enrichissements de la méthode sont possibles, comme par exemple l'intégration des valeurs seuils lors de la recherche du meilleur découpage de la matrice afin d'arrêter le processus d'alignement à des blocs plus larges et plus sûrs, ou encore l'examen d'un découpage ternaire plutôt que binaire afin de rendre compte de constructions linguistiques plus complexes générant des constituants non connexes.

Remerciements

Ces travaux ont été financés par le projet Cap Digital SAMAR.

Références

Ayan, N. F. et Dorr, B. J. (2006). Going beyond AER : an extensive analysis of word alignments and their impact on MT. *In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 9–16, Sydney, Australie.

Brown, P., Cocke, J., Della Pietra, S., Della Pietra, V., Jelinek, F., Mercer, R. et Roossin, P. (1988). A statistical approach to language translation. *In Proceedings of the 12th International Conference on Computational Linguistics (Coling'88)*, pages 71–76, Budapest.

Brown, P., Della Pietra, S., Della Pietra, V. et Mercer, R. (1993). The mathematics of statistical machine translation : Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Dagan, I. et Church, K. (1994). Termight : identifying and translating technical terminology. *In Proceedings of the fourth conference on Applied natural language processing*, pages 34–40, Stuttgart.

DeNero, J. et Klein, D. (2007). Tailoring word alignments to syntactic machine translation. *In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07)*, pages 17–24, Prague.

DENG, Y. et BYRNE, W. (2005). HMM word and phrase alignment for statistical machine translation. *In Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pages 169–176, Vancouver, British Columbia, Canada.

DENG, Y., KUMAR, S. et BYRNE, W. (2006). Segmentation and alignment of parallel text for statistical machine translation. *Natural Language Engineering*, 13(3):235–260.

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

FRASER, A. et MARCU, D. (2007). Getting the structure right for word alignment : LEAF. *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 51–60, Prague.

FUNG, P. et CHURCH, K. (1994). K-vec : A new approach for aligning parallel texts. *In Proceedings of the 15th International Conference on Computational Linguistics (Coling'94)*, volume 2, pages 1096–1102, Kyōto.

FUNG, P. et YEE, L. Y. (1998). An IR approach for translating new words from nonparallel, comparable texts. *In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, volume 1, pages 414–420, Montreal.

GALE, W. et CHURCH, K. (1991). Identifying word correspondences in parallel texts. *In Proceedings of the fourth DARPA workshop on Speech and Natural Language*, pages 152–157, Pacific Grove.

GANCHEV, K., GRAÇA, J. et TASKAR, B. (2008). Better alignments = better translations ? *In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies (ACL-08 : HLT)*, pages 986–993, Columbus, Ohio.

GAO, Q. et VOGEL, S. (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, E. et LANGÉ, J.-M. (1995). Modèles statistiques pour l'extraction de lexiques bilingues. *Traitement Automatique des Langues*, 36(1-2):133–155.

JOHNSON, H., MARTIN, J., FOSTER, G. et KUHN, R. (2007). Improving translation quality by discarding most of the phrasetable. *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 967–975, Prague.

KOEHN, P. (2005). Europarl : A parallel corpus for statistical machine translation. *In Proceedings of the tenth Machine Translation Summit (MT Summit X)*, pages 79–86, Phuket.

KOEHN, P., HOANG, H., BIRCH, A., CALLISON-BURCH, C., FEDERICO, M., BERTOLDI, N., COWAN, B., SHEN, W., MORAN, C., ZENS, R., DYER, C., BOJAR, O., CONSTANTIN, A. et HERBST, E. (2007). Moses : Open source toolkit for statistical machine translation. *In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007)*, pages 177–180, Prague.

KOEHN, P., OCH, F. et MARCU, D. (2003). Statistical phrase-based translation. *In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*, pages 48–54, Edmonton.

LARDILLEUX, A., LEPAGE, Y. et YVON, F. (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, A., YVON, F. et LEPAGE, Y. (2011b). Généralisation de l'alignement sous-phrastique par échantillonnage. *In Actes de la 18e conférence sur le Traitement Automatique des Langues Naturelles (TALN 2011)*, volume 1, pages 507–518, Montpellier.

LIANG, P., TASKAR, B. et KLEIN, D. (2006). Alignment by agreement. *In Proceedings of the Human Language Technology Conference of the NAACL*, pages 104–111, New York City.

LUO, J., LARDILLEUX, A. et LEPAGE, Y. (2011). Improving sampling-based alignment by investigating the distribution of n-grams in phrase translation tables. *In Proceedings of the 25th Pacific Asia Conference on Language, Information and Computation (PACLIC 25)*, pages 150–159, Singapour.

MARCU, D. et WONG, D. (2002). A phrase-based, joint probability model for statistical machine translation. *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 133–139, Philadelphie.

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

MOORE, R. (2004). On log-likelihood-ratios and the significance of rare events. *In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 333–340, Barcelona.

MOORE, R. (2005). Association-based bilingual word alignment. *In Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 1–8, Ann Arbor.

OCH, F. (2003). Minimum error rate training in statistical machine translation. *In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 160–167, Sapporo.

OCH, F. et NEY, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.

PAPINENI, K., ROUKOS, S., WARD, T. et ZHU, W.-J. (2002). BLEU : a method for automatic evaluation of machine translation. *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, pages 311–318, Philadelphie.

SMADJA, F., HATZIVASSILOGLOU, V. et MCKEOWN, K. (1996). Translating collocations for bilingual lexicons : A statistical approach. *Computational Linguistics*, 22(1):1–38.

SNOVER, M., DORR, B., SCHWARTZ, R., MICCIULLA, L. et MAKHOUL, J. (2006). A study of translation edit rate with targeted human annotation. *In Proceedings of the 7th Conference of the Association for Machine Translation of the Americas (AMTA 2006)*, pages 223–231, Cambridge.

VOGEL, S. (2005). PESA : Phrase pair extraction as sentence splitting. *In Proceedings of the tenth Machine Translation Summit (MT Summit X)*, pages 251–258, Phuket.

VOGEL, S., NEY, H. et TILLMAN, C. (1996). Hmm-based word alignment in statistical translation. *In Proceedings of the 16th International Conference on Computational Linguistics (Coling'96)*, pages 836–841, Copenhague.

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

ZHA, H., HE, X., DING, C., SIMON, H. et GU, M. (2001). Bipartite graph partitioning and data clustering. *In Proceedings of the tenth international conference on Information and knowledge management*, pages 25–32, Atlanta.