

De la traduction probabiliste aux mémoires de traduction (ou l'inverse)

Philippe Langlais et Michel Simard

RALI

Département d'Informatique et de Recherche Opérationnelle

Université de Montréal

CP. 6128 Succursale Centre-Ville

H3C3J7 Montréal, Québec, Canada

{felipe,simardm}@iro.umontreal.ca

Mots-clefs – Keywords

Traduction automatique, mémoire de traduction sous-phrastique, alignement sous-phrastique
Automatic translation, translation memories, word alignment

Résumé - Abstract

En dépit des travaux réalisés cette dernière décennie dans le cadre général de la traduction probabiliste, nous sommes toujours bien loin du jour où un engin de traduction automatique (probabiliste ou pas) sera capable de répondre pleinement aux besoins d'un traducteur professionnel. Dans une étude récente (Langlais, 2002), nous avons montré comment un engin de traduction probabiliste pouvait bénéficier de ressources terminologiques extérieures. Dans cette étude, nous montrons que les techniques de traduction probabiliste peuvent être utilisées pour extraire des informations sous-phrastiques d'une mémoire de traduction. Ces informations peuvent à leur tour s'avérer utiles à un engin de traduction probabiliste. Nous rapportons des résultats sur un corpus de test de taille importante en utilisant la mémoire de traduction d'un concordancier bilingue commercial.

Despite the exciting work accomplished over the past decade in the field of Statistical Machine Translation (SMT), we are still far from the point of being able to say that machine translation fully meets the needs of real-life users. In a previous study (Langlais, 2002), we have shown how a SMT engine could benefit from terminological resources, especially when translating texts very different from those used to train the system. In the present paper, we discuss the opening of SMT to examples automatically extracted from a Translation Memory (TM). We report results on a fair-sized translation task using the database of a commercial bilingual concordancer.

1 Introduction

Le fait que la traduction automatique ne soit viable que dans des contextes applicatifs très précis (domaine limité, tâche répétitive) est une évidence que les tentatives récentes, notamment dans le domaine effervescent de la traduction probabiliste, n'ont pas réussi à démentir. Ce constat indique que des approches d'aide à la traduction sont à privilégier (du moins pour le moment) à une approche complètement automatique. C'est dans ce contexte que des systèmes de mémoire de traduction comme ceux commercialisés par les firmes Trados (*Translator's Workbench*), IBM (*Translation Manager/2*), Atril (*Déjà-Vu*) et Star AG (*Transit*) ont vu le jour et connaissent un certain succès commercial.

Ces systèmes opèrent à un niveau de granularité qui correspond grossièrement à celui de la phrase et ne sont donc utilisables que dans des contextes également très précis comme la révision de documents. Des systèmes travaillant à un niveau sous-phrastique existent cependant sur le marché. C'est par exemple le cas des produits *Déjà-Vu* et *MultiTrans* (MultiCorpora R&D inc.). Il est cependant important de souligner, que ces outils ne s'appuient sur aucune technique d'alignement sous-phrastique et font au contraire appel au bon sens des traducteurs qui ont à charge d'alimenter leur(s) base(s) avec des alignements (sous-phrastiques ou pas) pertinents. À notre connaissance, et au-delà des arguments commerciaux avancés par les différents concepteurs de logiciels de mémoire de traduction, il n'existe pas d'étude qui permet d'établir de manière concluante que de tels outils augmentent la productivité des utilisateurs (traducteurs ou cabinet de traduction).

Dans une étude passée (Langlais & Simard, 2001), nous avons montré qu'une mémoire de traduction systématiquement interrogée à un niveau sous-phrastique permettait d'obtenir une couverture (source) d'un texte à traduire de loin supérieure (de 70% à 95% selon la nature du document) à celle que l'on peut obtenir si l'on interroge la même mémoire à un niveau phrastique. Des couvertures comparables ont été décrites dans le cadre du système Pangloss (Brown, 1996).

Nous avons également montré que ce taux n'est qu'un indicateur très biaisé de la **réutilisabilité d'une mémoire**. D'autres étapes — non triviales dans le cas d'une mémoire sous-phrastique — telles que la sélection du matériel source à aller rechercher dans la mémoire, l'identification de son correspondant cible ainsi que les choix de présentation du matériel ramené au traducteur sont en effet autant de facteurs qui conditionnent l'utilisabilité d'une mémoire. En simulant plusieurs scénarios, nous avons montré dans (Langlais & Simard, 2001) qu'en gros moins d'un tiers du matériel ramené d'une mémoire pouvait être utile à la construction de moins d'un quart du matériel cible à produire. Ces taux étaient mesurés sur deux bitextes de référence de 100 paires de phrases chacun.

Dans cette étude, nous proposons une nouvelle évaluation — sur un corpus de plus grande taille — d'un système d'interrogation de mémoire de traduction (SIM) dont les composants ont été améliorés par rapport à (Langlais & Simard, 2001). Nous nous intéressons de plus à vérifier dans quelle mesure un moteur de traduction probabiliste ne pourrait pas bénéficier d'exemples extraits automatiquement d'une mémoire.

Dans la section 2, nous présentons l'architecture générale qui sous-tend les expériences que nous avons menées et décrivons la mémoire de traduction que nous avons utilisée. Dans la section 3, nous décrivons les expériences que nous avons réalisées et qui valident l'idée qu'une mémoire de traduction et un moteur de traduction peuvent faire bon ménage. Nous discutons nos expériences dans la section 4.

2 Architecture du système

Nous clarifions dès à présent que dans cette étude nous dissocions la ressource constituée de traductions déjà disponibles — et que nous appelons la mémoire de traduction — de son exploitation. La mémoire que nous utilisons ici est un énorme bitexte anglais-français de plus de 100 millions de mots par langue aligné au niveau de la phrase. Ce matériel contient tous les textes parlementaires canadiens publiés entre avril 1986 et décembre 2001 et constitue le cœur de la mémoire à laquelle ont accès les utilisateurs du système commercial `TSRali.com`¹. Une vue d'ensemble de l'architecture est offerte en figure 1.

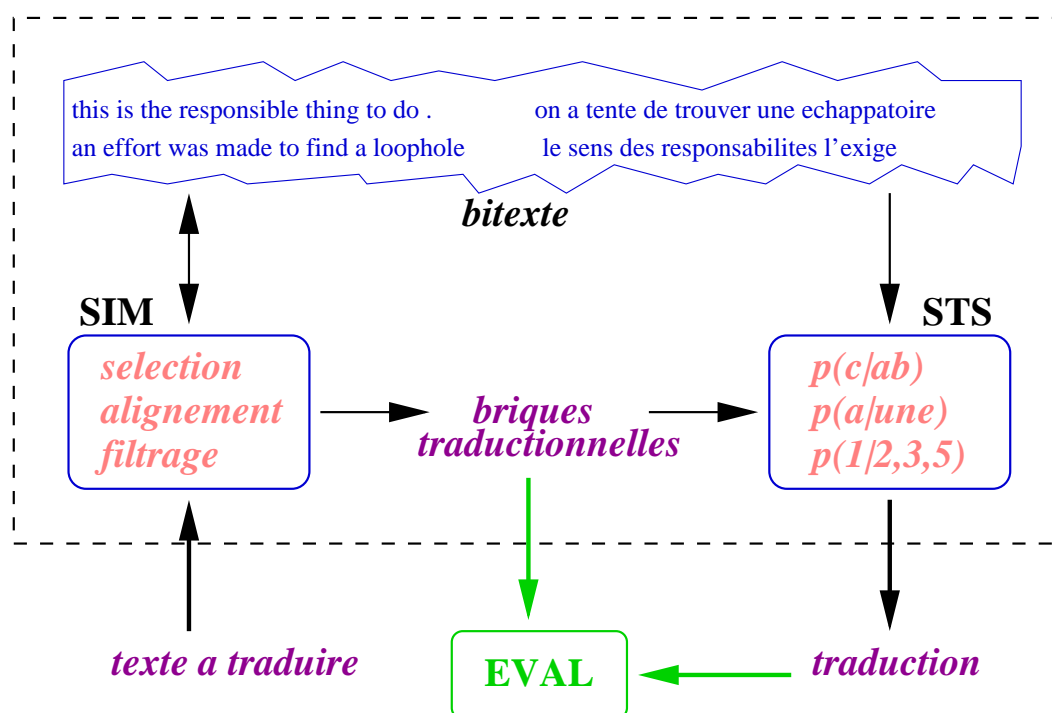


Figure 1: Architecture utilisée dans cette étude. Voir le texte pour des explications.

2.1 Le moteur de traduction (STS)

Le système de traduction statistique utilisé ici est un système “canal bruité” mettant en jeu un modèle de langue (modèle trigramme interpolé), et un modèle de traduction de type IBM2 (Brown *et al.*, 1993). Les modèles embarqués ont été entraînés sur un sous-ensemble d'environ 1.6 million de paires de phrases de la mémoire. Le décodage est effectué par programmation dynamique selon une amélioration de la méthode décrite par Nießen et al. (1998). L'algorithme de recherche proposé s'accommode de plus de contraintes “flottantes” exprimées sous la forme de lexiques bilingues. Le décodeur garantit alors qu'une des traductions possibles de chaque unité source du lexique qui se trouve dans la phrase à traduire sera proposée dans la traduction. Le choix de l'unité cible (dans le cas où le lexique amende plusieurs traductions) et de sa position dans la traduction est déterminé de manière à optimiser sur la phrase, les prédictions des modèles de langue et de traduction. La figure 2 montre un exemple de traduction produite par le décodeur lorsqu'on lui fournit un lexique bilingue (*smt + mt*).

¹Consulter la page <http://www.tsrali.com> pour plus d'information.

2.2 Le système d'interrogation de la mémoire de traduction (SIM)

Le système d'interrogation de la mémoire est basé sur trois opérations automatiques dont la précision conditionne le succès de l'architecture au complet.

sélection consiste à sélectionner dans le matériel à traduire des séquences de mots avec lesquelles interroger la mémoire. Nous appelons ces séquences les *requêtes*.

Dans cette étude, le texte à traduire est tout d'abord découpé en *chunks* à l'aide d'une cascade de modèles markoviens qui étiquette chaque mot par un marqueur de début, ou d'intermédiaire de groupe (voir (Osborne, 2000) pour la description de la cascade de distributions modélisées en pratique). Toute séquence de chunks présente dans la mémoire est alors une requête valide. Cette étape de segmentation en groupes simples n'est pas indispensable, mais nous pensons qu'elle permet de limiter les requêtes à des groupes de mots dont la traduction dans le texte cible est plus facile à localiser.

Dans l'exemple de la figure 2, 15 requêtes sont valides; leur fréquence d'apparition dans la mémoire est indiquée entre crochets; au total 564 couples de phrases sont sélectionnés dans la mémoire.

alignement consiste à identifier dans les paires de phrases de la mémoire qui contiennent les requêtes, leur traduction. Nous appelons les couples requête/matériel cible extrait de la mémoire les *briques traductionnelles*.

Aligner des phrases au niveau des mots est une tâche difficile. Dans le projet ARCADE, qui était dédié à l'évaluation d'alignements bilingues, il a été montré que les meilleurs systèmes testés obtenaient une performance de l'ordre de 75% (précision et rappel) dans une tâche simplifiée de localisation (dans un bitexte) des traductions de 60 mots choisis (Véronis & Langlais, 2000).

Nous avons expérimenté différentes stratégies d'alignement qui sont décrites et analysées dans (Simard, 2003). La technique d'alignement que nous avons retenue ici est une version simplifiée des grammaires stochastiques proposées par Wu (1997) qui consiste à détecter de manière descendante des points de coupures binaires dans la paire de phrase à aligner, jusqu'à isoler (lorsque c'est possible) le segment dont on cherche la traduction. Le choix du découpage est fait en consultant le modèle de traduction.

Dans notre exemple, 355 briques traductionnelles différentes ont ainsi été identifiées, parmi lesquelles 316 (89%) ont été rencontrées une seule fois dans la mémoire.

filtrage consiste à sélectionner les briques traductionnelles qui seront finalement proposées au traducteur ou au moteur de traduction. Nous avons expérimenté différentes stratégies de filtrage (dont le détail n'est pas pertinent ici) comme des seuillages (sur la probabilité des associations ou sur leur fréquence dans la mémoire), ou encore la sélection des requêtes de manière à maximiser (par exemple) la couverture de la phrase à traduire.

Dans notre exemple, des 355 briques traductionnelles identifiées, seulement 8 sont finalement conservées.

2.3 Module d'évaluation (EVAL)

Ce module vise deux objectifs; d'une part de mesurer le taux de récupérabilité des briques traductionnelles extraites de la mémoire, d'autre part de vérifier si un système de traduction

probabiliste peut bénéficier du matériel extrait automatiquement d'une mémoire à l'aide des mêmes modèles que le moteur utilise. Dans les deux cas, nous utilisons un bitexte de référence dont la partie cible (une traduction produite par un humain) est le texte que nous cherchons à reproduire (traduction *oracle*).

2.3.1 Taux de couverture et de précision

Les deux taux suivants sont indépendants du moteur de traduction et tentent de mesurer directement la réutilisabilité des briques traductionnelles extraites de la mémoire.

Le taux de couverture source (resp. cible) est le pourcentage de mots sources (resp. cibles) du bitexte de référence qui sont couverts par le matériel source (resp. cible) des briques traductionnelles. Ces taux sont mesurés sur les briques traductionnelles entières issues de l'étape de filtrage.

Dans l'exemple de la figure 2, les trois requêtes retenues (*for the translation, immediately without waiting* et *will the minister table the report*) couvrent douze des treize mots de la phrase à traduire, la couverture source est donc de 12/13 (92.3%). De la même manière, les briques cibles (*la traduction* et *sans attendre*) couvrent quatre des douze mots de la traduction oracle; la couverture cible est donc de 4/12 (33%).

Taux de précision et de rappel. Dès lors que l'on s'intéresse à la couverture cible, nous avons montré dans (Langlais & Simard, 2001) qu'il était préférable de mesurer des taux de précision et de rappel. Il est en effet possible de maximiser la couverture cible en gardant toutes les traductions potentielles de toutes les requêtes valides, et ce au détriment d'un utilisateur qui se verrait submergé de matériel cible.

Dans un contexte d'évaluation automatique, cela signifie qu'il nous faut faire des hypothèses quant à l'usage que ferait un utilisateur du matériel cible qu'on lui propose. Dans cette étude, nous avons imaginé qu'un utilisateur (fictif) produit la traduction oracle en copiant/collant tout ou partie du matériel cible qui lui est proposé. Nous définissons alors la précision P_c comme de rapport du nombre de mots collés dans la traduction sur le nombre de mots cibles des briques traductionnelles. Le taux de rappel R_c est quant à lui mesuré par le ratio du nombre de mots collés sur le nombre de mots cibles de la traduction oracle. Là encore, il est important d'observer que les taux ainsi mesurés dépendent grandement de la façon avec laquelle on admet qu'un utilisateur sélectionnera des mots dans une brique traductionnelle. Nous admettons ici qu'il copiera au plus une séquence de mots d'au moins c mots de chaque brique traductionnelle. $c = 0$ désigne le scénario où le traducteur ne fera que copier/coller des briques cibles entières. Ces deux taux peuvent être résumés en un seul (la f -mesure) qui est leur moyenne harmonique.

Dans notre exemple, si l'on admet que l'utilisateur ne fait que des copier/coller de briques cibles entières ($c = 0$), alors 11 mots (ceux des briques dont la partie cible est en gras) sont copiés sur un total de 30 mots cibles proposés, d'où une précision P_0 de 11/30 (soit 36.6%). Le rappel R_0 est alors de 11/13 (soit 84.6%).

2.3.2 Mesures automatiques de la qualité d'une traduction

Une manière indirecte de mesurer la réutilisabilité des briques traductionnelles est de vérifier leur apport à la qualité d'une traduction. Bien qu'indirecte, cette évaluation présente l'avantage

input :

source will the minister table the report immediately without waiting for the translation ?
 oracle le ministre va il déposer immédiatement le rapport sans attendre la traduction ?

requêtes :

will the minister table [64], **will the minister table the report [3]**, the minister table [98], the minister table the report [4], the report [100], the report immediately [1], immediately without [34], immediately without waiting [1], **immediately without waiting for [1]**, without waiting [62], without waiting for [49], waiting for [98], waiting for the translation [3], **for the translation [6]**, the translation [40]

briques traductionnelles :

for the translation	la traduction
for the translation	pour entendre la traduction
for the translation	pour la traduction de
for the translation	pour qu'
immediately without waiting	sans attendre
will the minister table the report	le comité
will the minister table the report	le ministre déposera il le rapport sur
will the minister table the report	le ministre va il déposer le rapport

output :

smt le ministre de déposer le rapport sans attendre la suite de la traduction ?
 smt+mt **le ministre va il déposer le rapport sans attendre la traduction ?**

alignements :

REF: le min va il dép imm le rap san att la tra ?
 SMT: le min de dép le rap san att la suite de la tra ?
 SMT+: le min va il dép le rap san att la tra ?

Figure 2: Illustration sur une session de traduction simple des principales étapes de l'architecture décrite. Voir le texte pour des explications.

de pouvoir s'appuyer sur des métriques existantes. Nous comparons ici la qualité des traductions produites par notre moteur de traduction avec et sans brique traductionnelle à l'aide de trois mesures automatiques qui sont couramment utilisées dans la littérature.

BLEU est une mesure qui comptabilise de manière pondérée le nombre d'unigrammes, de bigrammes, de trigrammes et de quadigrammes qu'une traduction à évaluer partage avec une ou plusieurs traductions oracles. Les valeurs que peuvent prendre cette mesure sont entre 0 et 1; une valeur de 1 indiquant une bonne traduction. Papineni et al. (2002) rapportent que cette mesure est corrélée à des jugements produits par des humains. C'est la mesure qui est utilisée dans les évaluations NIST. Pour des questions de lisibilité nous multiplions par 100 les scores obtenus par le programme `mt-eval` disponible sur le site de NIST².

Deux traductions sont rapportées dans l'exemple de la figure 2. La première (*smt*) est produite par le moteur probabiliste seul et obtient un score BLEU de 39.07; la seconde (*smt + mt*), est

²<http://www.nist.gov/speech/tests/mt/doc/index.htm>

obtenue par le moteur de traduction alimenté par les briques traductionnelles et son score BLEU est de 76.77.

WER est une version normalisée de ce que l'on appelle la distance d'édition (le nombre d'opérations d'édition minimum qu'il faut appliquer à la traduction candidate pour obtenir la traduction oracle). Les valeurs de WER sont exprimées entre 0 et 100; une valeur de 0 indique une traduction identique à la traduction oracle.

Dans notre exemple, le moteur de traduction seul propose une traduction dont le WER est de 37.5, alors que la traduction qu'il propose lorsque les briques traductionnelles sont disponibles est de 7.7. Les alignements respectifs à la traduction oracle sont indiqués à même la figure 2.

SER est le pourcentage de traductions qui ne sont pas identiques *verbatim* à la traduction oracle. Cette mesure est bien sûr très sévère (une traduction pouvant être bonne sans pour autant être celle produite par l'oracle). Dans notre exemple, aucune des deux traductions proposées n'est identique à la traduction oracle, et le SER sur ce corpus d'une phrase est donc de 100 dans les deux cas.

3 Expériences

Nous avons extrait du Hansard un passage de mars 2002 (une période non couverte par la mémoire de traduction) constitué de 1646 phrases de longueur source (resp. cible) moyenne de 23 (resp. 20) mots. Dans cette expérience, nous avons choisi la langue anglaise comme langue source car la traduction oracle a été produite dans cette direction.

Il existe de nombreux paramètres qui sont ajustables dans l'architecture que nous avons présentée. Nous pouvons en particulier sélectionner la nature des requêtes faites à la mémoire de même que la quantité de briques traductionnelles ramenées (en tenant compte de différents critères de sélection). Nous avons testé dans cette expérience 462 instances de l'architecture dont les détails ne sont pas pertinents ici.

3.1 Couverture des briques traductionnelles

En terme de couverture source, les variantes testées diffèrent sensiblement selon la quantité de filtres appliqués de 68% à 80%, la moyenne entre tous ces systèmes étant de 74.7% (écart-type de 3.9%). Les différences les plus importantes sont de manière naturelle observées sur les taux de couverture cible. Les configurations les plus filtrantes (celles où par exemple on décide de ne retenir qu'une brique traductionnelle par requête) montrent des taux de couverture cible de l'ordre de 10%. Les configurations les plus permissives plafonnent quant à elles à 68%. La moyenne des configurations est de 28.4% (écart type de 12.4%). Ces observations sont encore une fois cohérentes avec les mesures faites dans (Langlais & Simard, 2001).

La table 1 montre parmi l'ensemble des configurations pour chaque scénario, les taux de précision, de rappel et de f-mesure que l'on obtient en maximisant ou bien la f-mesure (deuxième colonne), ou bien la précision (troisième colonne). Les configurations qui correspondent à ces mesures sont toutes des configurations où une seule brique traductionnelle a été conservée

par requête; ce sont également les configurations qui montrent des taux de couverture cible minimum; ce qui confirme que les taux de couverture ne sont pas de bons indicateurs de la récupérabilité d'une mémoire.

On observe également que si plus de la moitié des briques traductionnelles proposées à un utilisateur permet de reconstruire un peu moins de la moitié d'une traduction oracle en sélectionnant des séquences d'un mot ou plus dans ce matériel ($c = 1$), ces taux de récupérabilité diminuent avec la taille des segments que l'utilisateur s'autorise à copier. À l'extrême, un utilisateur qui ne pourrait que cliquer sur les briques proposées pour construire sa traduction ($c = 0$) ne pourrait utiliser qu'un sixième environ des briques proposées pour construire à peu près un cinquième de sa traduction. Ces chiffres sont plus faibles que ceux que nous avons mesurés dans (Langlais & Simard, 2001) sur le matériel Hansard testé à ce moment. La taille plus grande du corpus de test utilisé ici et le fait que ce soit un texte d'une période non couverte par la mémoire sont des explications possibles de cette différence.

c	meilleure F_c			meilleure P_c	
	P_c	R_c	F_c	P_c	R_c
0	14.1	19.9	16.5	18.6	10.5
1	56.9	42.3	48.6	57.1	32.0
2	34.4	25.6	29.3	34.4	25.6
3	24.7	18.4	21.0	24.7	18.4
4	17.2	12.8	14.7	17.2	12.8

Table 1: Taux de précision et rappel en fonction du scénario utilisateur. La deuxième colonne montre les meilleures f-mesures que l'on obtient parmi toutes les versions testées; la troisième colonne montre les configurations avec la meilleure précision.

3.2 Qualité de la traduction

Nous rapportons en table 2 les résultats de traduction obtenus par notre décodeur avec et sans les briques traductionnelles en fonction de la longueur maximale des phrases considérées³. On note clairement ici la corrélation entre les différentes métriques et la longueur maximale des phrases traduites: traduire une phrase plus courte est en moyenne plus facile. On observe également l'amélioration systématique que les briques traductionnelles ont sur la qualité (telle que mesurée) des traductions produites, et ce, quelque soit la longueur maximale des phrases considérées. L'amélioration relative de score BLEU mesurée pour l'ensemble des phrases d'au plus 30 mots est de 46.8%. Ce résultat est très intéressant et confirme les observations faites par Marcu (2001), et ce, même si les protocoles expérimentaux entre les deux études sont assez différents. En particulier, il est important de souligner que dans nos expériences, les modèles de traduction et de langue ont été entraînés sur une petite portion de la mémoire utilisée ici et qu'il n'est pas du tout certain que nous aurions observé de telles améliorations si les modèles avaient été entraînés sur l'ensemble de la mémoire. Il convient cependant d'ajouter que l'entraînement d'un modèle de traduction sur des quantités de textes supérieures à celles utilisées dans cette étude n'est pas une tâche simple⁴ et qu'il existe à notre avis de nombreuses situations où il n'est

³Seules les phrases d'au plus 30 mots ont été traduites ici.

⁴Des contraintes pratiques mais bien réelles, notamment de mémoire vive, font que l'entraînement de gros modèles nécessite entre autres choses une version parallèle de l'algorithme d'entraînement.

L	$nb.$	SMT + MT				SMT			ratio
		BLEU	WER	SER	perfect	BLEU	WER	SER	
5	199	28.94	52.57	83.42	33	18.07	58.06	91.96	60.1
10	397	25.04	55.82	89.42	42	18.84	59.39	95.46	32.9
15	646	24.00	59.56	93.03	45	17.53	62.39	97.21	36.9
20	890	24.93	60.51	94.72	47	16.87	63.66	97.98	47.8
25	1026	23.65	62.24	95.42	47	15.98	64.64	98.25	48.0
30	1126	23.46	63.01	95.83	47	15.98	65.07	98.40	46.8

Table 2: Mesures de la qualité des traductions produites par le décodeur seul (*SMT*) et par le décodeur aidé des briques traductionnelles (*SMT + MT*). *ratio* indique l'amélioration relative du score BLEU.

pas envisageable d'entraîner "régulièrement" de nouveaux modèles, alors qu'il est assez simple de mettre à jour une mémoire de traduction.

4 Discussions

Nous avons proposé deux types d'évaluation de la récupérabilité d'une mémoire de traduction sous-phrastique. Nous avons montré sur un corpus de test de taille importante que les taux de précision et de rappel des briques traductionnelles extraites de la mémoire étaient loin d'approcher les taux de couvertures sources et cibles que l'on rapporte habituellement. Il est important de noter cependant que le fait de n'utiliser qu'une traduction de référence ne nous permet que de mesurer une borne inférieure de récupérabilité d'une mémoire. Nous montrons enfin, que les briques traductionnelles aident un engin de traduction probabiliste à proposer des traductions de meilleure qualité (dans la limite de la pertinence des métriques utilisées ici).

L'idée d'utiliser des exemples extraits d'une mémoire pour générer ensuite une traduction n'est pas nouvelle et sous-tend en fait tout système de traduction à base d'exemples. Le système *Pangloss* (Frederking *et al.*, 1994; Brown, 1996) utilisait par exemple un modèle de langue pour assembler les exemples extraits de différentes ressources et différents systèmes de traduction. L'utilisation d'un moteur de traduction probabiliste pour assembler des exemples est cependant moins populaire.

Nous voyons cependant plusieurs avantages à utiliser un moteur de traduction probabiliste pour assembler des exemples. Premièrement, il convient de rappeler que la réalisation d'un moteur probabiliste est une opération relativement aisée. Des *packages* simples à utiliser sont en effet disponibles pour l'entraînement des modèles de langue (Clarkson & Rosenfeld, 1997) et de traduction (Och & Ney, 2000) et un décodeur peut également être téléchargé (Germann *et al.*, 2001). Deuxièmement, la relative simplicité d'un décodeur probabiliste rend assez simple l'intégration d'informations externes aux modèles probabilistes (Langlais, 2002), ce qui n'est pas nécessairement le cas d'autres approches. Outre ces aspects pratiques, nous trouvons intéressante l'idée avancée par Marcu (2001) qui mentionne que dans l'état actuel des technologies d'alignement (probabilistes ou non), il reste plus simple d'analyser une traduction que de la générer. Dans notre contexte, et peut-être paradoxalement, cela signifie qu'il est peut-être plus productif d'extraire des exemples d'une mémoire à l'aide de modèles probabilistes pour ensuite produire une traduction à l'aide d'un moteur faisant usage de ces extraits

et de ces mêmes modèles probabilistes, que d'utiliser directement cet engin de traduction. Les expériences décrites ici semblent confirmer la véracité de cette hypothèse bien que des expérimentations plus poussées restent nécessaires à sa validation.

Références

- BROWN P. F., PIETRA S. A. D., PIETRA V. J. D. & MERCER R. L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, **19**(2), 263–311.
- BROWN R. (1996). Example-based machine translation in the pangloss system. In *Proceedings of International Conference on Computational Linguistics (COLING)*, p. 169–174, Copenhagen, Denmark.
- CLARKSON P. & ROSENFELD R. (1997). Statistical language modeling using the CMU-cambridge toolkit. In *Proc. Eurospeech '97*, p. 2707–2710, Rhodes, Greece.
- FREDERKING R., NIRENBURG S., FARWELL D., HELMREICH S., HOVY E., KNIGHT K., BEALE S., DOMASHNEV C., ATTARDO D., GRANNES D. & BROWN R. (1994). Integrating translations from multiple sources within the pangloss mark iii machine translation. In *Proceedings of the first conference of the Association for Machine Translation in the Americas (AMTA)*, Columbia, MD.
- GERMANN U., JAHR M., KNIGHT K., MARCU D. & YAMADA K. (2001). Fast decoding and optimal decoding for machine translation. In *Proceedings of the 39th Annual Meeting of the ACL*, Toulouse, France.
- LANGLAIS P. (2002). Ressources terminologiques et traduction probabiliste: premiers pas positifs vers un système adaptatif. In *9e Conférence Annuelle sur le Traitement Automatique des Langues Naturelles (TALN)*, p. 43–52, Nancy, France.
- LANGLAIS P. & SIMARD M. (2001). Récupération d'unités sous-phrastiques dans une mémoire de traduction. In *8e conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, p. 243–252, Tours, France.
- MARCU D. (2001). Towards a unified approach to memory- and statistical-based machine translation. In *Proceedings of the 39th Annual Meeting of the ACL*, p. 378–385, Toulouse, France.
- NIESSEN S., VOGEL S., NEY H. & TILLMANN C. (1998). A dp based search algorithm for statistical machine translation. In *Proceedings of the 36th Annual Meeting of the ACL and 17th COLING*, p. 960–966, Montréal, Canada.
- OCH F. J. & NEY H. (2000). Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the ACL*, p. 440–447, Hongkong, China.
- OSBORNE M. (2000). Shallow parsing as part-of-speech tagging. In *Proceedings of the 4th Computational Natural Language Learning Workshop (CoNLL)*, Lisbon, Portugal.
- PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the ACL*, p. 311–318, Philadelphia, Pennsylvania, USA.
- SIMARD M. (2003). *Mémoire de traduction sous-phrastique*. PhD thesis, Université de Montréal.
- VÉRONIS J. & LANGLAIS P. (2000). *Evaluation of parallel text alignment systems: The ARCADE project*, volume 13, chapter 19, p. 369–388. Parallel Text Processing, Kluwer.
- WU D. (1997). Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, **23**(3), 377–404.