Extraction de corpus parallèle pour la traduction automatique depuis et vers une langue peu dotée

Thi Ngoc Diep Do

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR EN SCIENCES DÉLIVRÉ PAR L'UNIVERSITÉ DE GRENOBLE

Spécialité : **MSTII / INFORMATIQUE**

Arrêté ministériel : 7 août 2006

Présentée par

Thi-Ngoc-Diep DO

Thèse dirigée par **Laurent BESACIER** et
codirigée par **Eric CASTELLI**

préparée au sein du **Laboratoire d'Informatique de Grenoble**
et du **Centre de Recherche International Multimédia, Information, Communication et Applications – Hanoi, Vietnam**
dans **l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

Extraction de corpus parallèle pour la traduction automatique depuis et vers une langue peu dotée

Thèse soutenue publiquement le **20 Décembre 2011**,
devant le jury composé de :

M. Christian BOITET
Professeur à l'Université Joseph Fourier - Grenoble, Président
M. Holger SCHWENK
Professeur à l'Université du Maine, Rapporteur
M. Kamel SMAÏLI
Professeur à l'Université Nancy 2, Rapporteur
M. Georges LINARÈS
Professeur à l'Université d'Avignon et des Pays de Vaucluse, Membre
M. Laurent BESACIER
Professeur à l'Université Joseph Fourier - Grenoble, Membre
M. Eric CASTELLI
Chargé de Recherche (HDR) - CNRS, Membre

Remerciements

Je voudrais tout d'abord présenter tous mes remerciements, ainsi que toute ma gratitude, à Monsieur Laurent BESACIER et Monsieur Eric CASTELLI pour avoir accepté d'être mes deux directeurs de thèse. Un grand remerciement très chaleureusement à Monsieur Laurent BESACIER, qui m'a guidé tout au long de ces années de thèse, pour ses conseils, ses aides et ses encouragements précieux. Un grand remerciement également à Monsieur Eric CASTELLI pour tous ses critiques et ses conseils sur mes travaux de recherche.

J'adresse mes remerciements à Madame Brigitte PLATEAU, directrice du Laboratoire d'Informatique de Grenoble, pour m'avoir accepté dans l'équipe GETALP. Un grand remerciement également à Monsieur NGUYEN Trong Giang, l'ancien directeur du Centre MICA, et Madame PHAM Thi Ngoc Yen, directrice du Centre MICA, pour m'avoir accueilli dans l'équipe TIM et pour leur aide dévouée.

Je tiens à remercier Monsieur Christian BOITET pour avoir accepté d'être le président du jury. Je voudrais remercier aussi à Monsieur Holger SCHWENK et Monsieur Kamel SMAÏLI pour avoir accepté d'être les rapporteurs de ma thèse. Un grand merci à Monsieur Georges LINARÈS d'avoir accepté de participer à ce jury.

Je remercie également tous les membres de l'équipe GETALP du LIG et l'équipe TIM du MICA pour leur accueil et leur sympathie. Je remercie mes amis à MICA et mes amis à Grenoble avec qui j'ai partagé de grands moments au cours de ma thèse.

Je pense à ma famille qui m'a apporté un soutien important, non seulement sur l'aspect sentimental, mais également par les encouragements dont j'avais besoin pour mener à bien ce travail.

Un grand merci à tous.

Résumé

Les systèmes de traduction automatique obtiennent aujourd'hui de bons résultats sur certains couples de langues comme anglais – français, anglais – chinois, anglais – espagnol, etc. Les approches de traduction empiriques, particulièrement l'approche de traduction automatique probabiliste, nous permettent de construire rapidement un système de traduction si des corpus de données adéquats sont disponibles. En effet, la traduction automatique probabiliste est fondée sur l'apprentissage de modèles à partir de grands corpus parallèles bilingues pour les langues source et cible. Toutefois, la recherche sur la traduction automatique pour des paires de langues dites « peu dotées » doit faire face au défi du manque de données.

Nous avons ainsi abordé le problème d'acquisition d'un grand corpus de textes bilingues parallèles pour construire un système de traduction automatique probabiliste. L'originalité de notre travail réside dans le fait que nous nous concentrons sur les langues peu dotées, où des corpus de textes bilingues parallèles sont inexistants dans la plupart des cas. Ce manuscrit présente notre méthodologie d'extraction d'un corpus d'apprentissage parallèle à partir d'un corpus comparable, une ressource de données plus riche et diversifiée sur Internet. Nous proposons trois méthodes d'extraction. La première méthode suit l'approche de recherche classique qui utilise des caractéristiques générales des documents ainsi que des informations lexicales du document pour extraire à la fois les documents comparables et les phrases parallèles. Cependant, cette méthode requiert des données supplémentaires sur la paire de langues. La deuxième méthode est une méthode entièrement non supervisée qui ne requiert aucune donnée supplémentaire à l'entrée, et peut être appliquée pour n'importe quelle paire de langues, même des paires de langues peu dotées. La dernière méthode est une extension de la deuxième méthode, elle utilise une troisième langue pour améliorer les processus d'extraction de deux paires de langues. Les méthodes proposées sont validées par des expériences appliquées sur la langue peu dotée vietnamienne et les langues française et anglaise.

Mots clés : langues peu dotées, traduction automatique probabiliste, extraction de données parallèles, corpus comparable, méthode non supervisée, triangulation, alignement, métriques d'évaluation

Abstract

Nowadays, machine translation has reached good results when applied to several language pairs such as English – French, English – Chinese, English – Spanish, etc. Empirical translation, particularly statistical machine translation allows us to build quickly a translation system if adequate data is available because statistical machine translation is based on models trained from large parallel bilingual corpora in source and target languages. However, research on machine translation for under-resourced language pairs always faces the lack of training data.

Thus, we have addressed the problem of retrieving a large parallel bilingual text corpus to build a statistical machine translation system. The originality of our work lies in the fact that we focus on under-resourced languages for which parallel bilingual corpora do not exist in most cases. This manuscript presents our methodology for extracting a parallel corpus from a comparable corpus, a richer and more diverse data resource over the Web. We propose three methods of extraction. The first method follows the classical approach using general characteristics of documents as well as lexical information of the document to retrieve both parallel documents and parallel sentence pairs. However, this method requires additional data of the language pair. The second method is a completely unsupervised method that does not require additional data and can be applied to any language pair, even to under-resourced language pairs. The last method deals with the extension of the second method using a third language to improve the extraction process (triangulation). The proposed methods are validated by a number of experiments applied on the under-resourced Vietnamese language and to the English and French languages.

Key words: under resourced languages, statistical machine translation, mining parallel data, comparable corpus, unsupervised method, triangulation, alignments, evaluation metrics

Table des matières

Liste des figures

Liste des tableaux

Introduction

La traduction automatique (TA) est dans une période de croissance forte, reflétée par le grand nombre de groupes de recherche et par le grand nombre de produits d'application sur le marché. L'une des raisons de cet essor est la naissance de l'approche de traduction « empirique ». L'approche de traduction traditionnelle, l'approche « experte », nécessite l'intervention de spécialistes de la langue et elle est coûteuse en temps et ressources humaines. La naissance de l'approche de traduction empirique, particulièrement l'approche de traduction automatique probabiliste, avec des métriques d'évaluation automatique et des boîtes à outils disponibles, nous permet de construire rapidement un système de traduction dans un temps relativement réduit pour n'importe quelle paire de langues à partir du moment où un corpus parallèle de grande taille est disponible. La traduction automatique probabiliste considère le problème de la traduction comme un problème d'apprentissage automatique avec de grandes bases de données d'apprentissage (les corpus parallèles). Evidemment, comme d'autres approches basées sur les données, la précision de ces systèmes dépend essentiellement de la quantité, de la qualité et de l'adéquation au domaine des données d'apprentissage utilisées.

Les corpus de textes bilingues et parallèles sont des ressources indispensables pour la traduction automatique probabiliste. Un corpus parallèle se compose de textes bilingues alignés au niveau des phrases. Le système de traduction automatique probabiliste utilise ce corpus parallèle comme matériau d'apprentissage. L'augmentation de la quantité de corpus parallèle peut conduire à une meilleure qualité. [Och 2006] suppose que le modèle de traduction est formé idéalement à partir de centaines de millions de mots, et pour obtenir une qualité de traduction raisonnable, nous avons besoin d'un corpus de millions de mots. L'auteur montre aussi que la qualité du meilleur système de traduction entraîné sur une grande quantité de données peut être significativement supérieure à la qualité d'un système de traduction traditionnel basé sur l'approche experte.

D'un autre côté, l'acquisition d'un grand corpus parallèle de haute qualité pour certaines paires de langues dans un certain domaine nécessite beaucoup de temps et d'efforts.

La base de données *Ethnologue*[1] recense plus de 6900 langues vivantes connues dans le monde. Pour quelques langues telles que l'anglais, le chinois, l'arabe, et quelques langues d'Europe (le français, l'allemand, l'italien, etc.), les ressources de données sont abondantes ou au moins sont existantes et leur quantité augmente. Pour ces langues, les données bilingues sont recueillies souvent à partir de documents des institutions multinationales telles que les Nations Unies, l'Union européenne ou à partir de documents gouvernementaux de pays multilingues, etc. Plusieurs corpus parallèles pour ces paires de langues sont construits. Par exemple le corpus *Hansard des débats du parlement du Canada*[2] (1,3 million de paires de phrases parallèles anglaises – françaises) ; le corpus *Europarl*[3] des actes du parlement européen (en 11 langues, 1,8

[1] http://www.ethnologue.com
[2] Version en 2001 : http://www.isi.edu/natural-language/download/hansard/
[3] Version v6 : http://www.statmt.org/europarl/

million de paires de phrases anglaises – françaises) [Koehn 2005] ; le corpus *UN Parallel Text*[1] du consortium LDC[2] (trois langues : l'anglais, le français et l'espagnol) [Graff 1994] (LDC fournit aussi d'autres corpus parallèles tels que anglais–chinois mandarin, anglais–tchèque, anglais–arabe, anglais–hongrois) ; le corpus JRC-Acquis[3] [Steinberger 2006] (22 langues, près de 50 millions de mots par langue), etc. Ces corpus ont été utilisés largement en recherche depuis de nombreuses années, spécialement dans plusieurs campagnes d'évaluation (telles que WMT (*Workshop on Statistical Machine Translation*), IWSLT (*International Workshop on Spoken Language Translation*), etc.).

Malheureusement, ces corpus parallèles sont disponibles pour certaines paires de langues mais pas pour les autres, que nous nommerons « langues peu dotées ». En général, ce terme fait référence à une langue présentant certaines, sinon toutes les caractéristiques suivantes :

- manque d'un système d'écriture unique ou une d'orthographe stable,
- présence limitée sur le Web,
- manque d'experts linguistes,
- manque de ressources pour le TALN (traitement automatique de la langue naturelle) telles que les données linguistiques, les corpus monolingues et bilingues, les dictionnaires électroniques, les thésaurus, les analyseurs morphologiques, les analyseurs syntaxiques, les étiqueteurs, etc.

Dans le cadre de cette thèse, nous nous concentrons sur les langues qui possèdent peu de ressources informatiques (ressources électroniques) disponibles pour l'implémentation des technologies du TALN.

Ainsi, pour la traduction probabiliste depuis ou vers une langue peu dotée, le corpus d'apprentissage parallèle n'existe pas toujours, ou bien il n'existe qu'avec une quantité faible de données, qui n'est pas suffisante pour apprendre des modèles probabilistes robustes.

Les stratégies pour la traduction probabiliste avec une quantité limitée de données d'apprentissage attirent de plus en plus l'attention. Des informations supplémentaires sont utilisées, telles que des informations morphosyntaxiques : la transformation de la structure de la phrase, les modèles hiérarchiques [Niessen 2004], le réarrangement des noms et adjectifs, la transformation d'un mot en sa forme de surface, etc. [Popovic 2006].

D'autres recherches proposent d'améliorer la qualité de traduction d'une langue peu dotée par une autre langue similaire ayant des ressources abondantes. Par exemple, un système de traduction malais–anglais peut aider à améliorer la qualité d'un système de traduction indonésien–anglais grâce à la similarité de l'ordre des mots et de la syntaxe, aux mots communs (etc.) entre le malais et l'indonésien [Nakov 2009].

Utiliser un corpus monolingue disponible est une autre piste intéressante. Un système de traduction de la langue source vers la langue cible (le système de référence) est utilisé pour traduire un corpus monolingue de la langue source, et former de nouvelles paires de phrases parallèles synthétisées source–cible, qui sont ensuite rajoutées au corpus d'apprentissage et le système de référence est ré-entraîné [Schwenk 2008].

Comme les corpus de textes bilingues parallèles sont des ressources indispensables non seulement pour la TA probabiliste, mais aussi pour d'autres applications du TALN, nous nous concentrons sur la construction des corpus parallèles pour des langues peu dotées et insistons sur une ressource de données beaucoup plus riche et diversifiée : les corpus comparables. Les corpus

[1] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC94T4A
[2] http://www.ldc.upenn.edu/
[3] http://langtech.jrc.it/JRC-Acquis.html

comparables contiennent des textes qui ne sont pas strictement parallèles, mais liés. Un exemple typique de textes comparables peut être deux articles de nouvelles journalistiques dans deux langues différentes qui relatent le même événement. Ils sont souvent produits indépendamment, mais ils sont susceptibles de contenir des données parallèles qui expriment le même contenu. Ce type de données peut être trouvé en grande quantité sur le Web, où il existe dans plusieurs paires de langues. Dans de nombreux domaines, il est continuellement mis à jour et enrichi. Comme le Web continue de se diversifier et de croître, ces ressources de données pour les langues peu dotées sont de plus en plus disponibles.

L'objectif de notre travail est de construire rapidement un système de traduction automatique depuis et vers une langue peu dotée. Donc nous travaillons sur l'extraction d'un corpus d'apprentissage parallèle à partir d'un corpus comparable. Plus concrètement, l'originalité de notre travail réside dans le fait que nous nous concentrons sur la construction de corpus pour des langues peu dotées, pour lesquelles les méthodes d'extraction existant sont difficiles à réaliser : au point de départ, nous n'avons qu'un seul site web multilingue de cette langue peu dotée, nous n'avons pas de données parallèles disponibles ou de données supplémentaires. De plus, nos méthodes d'extraction sont proposées afin d'être appliquées indépendamment de la paire de langues.

Cette thèse se compose de deux grandes parties : l'état de l'art (chapitres 1, 2 et 3) et nos contributions (chapitres 4, 5, 6 et 7).

Dans le chapitre 1, nous passons en revue les études théoriques sur la traduction automatique. Un bref historique de la TA est présenté. Des architectures différentes des systèmes de TA sont citées : l'architecture linguistique et l'architecture computationnelle. Correspondant à chaque architecture, les approches diverses pour construire le système de traduction sont présentées, telles que l'approche de traduction directe, l'approche à transfert, l'approche de traduction par interlingue ; des méthodes expertes ou empiriques, ou des méthodes hybrides.

Le chapitre 2 détaille l'approche de la traduction automatique probabiliste, celle sur laquelle nous nous concentrons. C'est une des approches de TA les plus étudiées dans le monde à ce jour. Le problème de la traduction est divisé en trois sous-problèmes : la modélisation, l'apprentissage et le décodage. Nous présentons deux approches de modélisation par les modèles génératifs et les modèles discriminants. Les méthodes d'estimation des paramètres du modèle pour deux types de modélisation sont présentées aussi. Après cette étape d'apprentissage, le problème du décodage est abordé avec les métriques d'évaluation automatique utilisées aujourd'hui. Quelques exemples de systèmes de référence existants aujourd'hui terminent le chapitre.

Dans le chapitre 3, nous parlons de la langue vietnamienne, une langue peu dotée (présence limitée sur le Web, manque de ressources pour le TALN) que nous choisissons pour appliquer les méthodes proposées. Après une introduction générale au vietnamien, et la description fondamentale du lexique et de la grammaire vietnamienne, nous présentons la situation du développement des systèmes de traduction depuis et vers la langue vietnamienne aujourd'hui, et la nécessité de construire des corpus parallèles pour la langue vietnamienne.

La deuxième partie commence par exposer nos contributions sur l'extraction de corpus parallèles à partir d'un corpus comparable. Dans le chapitre 4, nous présentons plus précisément nos objectifs. Les divers niveaux de parallélisme d'un corpus sont aussi décrits. Les trois chapitres suivants présentent les méthodes proposées.

Dans le chapitre 5, nous proposons la première méthode qui concerne l'utilisation d'informations d'alignement pour extraire à la fois les documents comparables et les phrases parallèles. Cette méthode requiert des données supplémentaires sur la paire de langues (telles qu'un dictionnaire bilingue, une liste de mots d'arrêt, etc.). Lorsque ces données supplémentaires sont disponibles, nous pouvons appliquer cette méthode pour toute paire de langues.

Le chapitre 6 présente notre deuxième méthode d'extraction, une méthode non supervisée. L'originalité de cette méthode est qu'elle ne requiert aucune donnée supplémentaire contrairement à la première méthode, donc elle est bien adaptée au cas des langues peu dotées où les données initiales ne sont pas forcément disponibles. Cette méthode peut être appliquée pour toute paire de langues, à partir d'un corpus comparable seulement, et aucune donnée supplémentaire n'est requise sur les deux langues considérées.

La dernière méthode présentée dans le chapitre 7 aborde l'utilisation d'une troisième langue dans l'extraction du corpus comparable bilingue. Nous revisitons ici l'approche de triangulation, introduite pour la traduction automatique, pour le processus d'extraction de données parallèles. Nous proposons d'utiliser une troisième langue dans le processus d'extraction des données parallèles à partir d'un corpus comparable.

Dans chaque chapitre, les applications de notre méthodologie sur les paires de langue vietnamien – français et vietnamien – anglais sont présentées. Les résultats associés, les ressources obtenues et les expériences sur le système de traduction automatique, sont également présentés. Le document se termine par les conclusions et les perspectives.

Partie I. État de l'art

Chapitre 1 : Introduction à la traduction automatique

La traduction automatique (TA) désigne des systèmes informatisés qui peuvent produire des traductions avec ou sans l'assistance humaine. Il s'agit d'un sous-domaine de la linguistique computationnelle qui tente d'automatiser le processus de traduction d'une langue naturelle source (texte ou parole) à une autre langue cible, en utilisant l'ordinateur. Le défi dans la TA réside dans la façon de programmer un ordinateur pour comprendre un texte comme un homme le fait et aussi pour créer un nouveau texte dans la langue cible comme il serait écrit par un humain.

Ici, nous distinguons la TA de la traduction assistée par l'ordinateur TAO (*machine-aided human translation*) où le but est d'aider un humain à effectuer une tache de traduction à l'aide de dictionnaires électroniques en ligne, de bases de données terminologiques, de mémoires de traduction, etc.

Le problème de la TA est complexe et il occupe de nombreux chercheurs depuis le début de la deuxième moitié du vingtième siècle.

1.1. Bref historique de la TA

Les premiers brevets pour « Les machines à traduire » ont été déposés dans le milieu des années 1930, avant même que l'ordinateur ne soit inventé. *Georges Artsrouni*, un français d'origine arménienne, a proposé un mécanisme utilisant une bande de papier perforé pour sélectionner la traduction d'un mot source à partir d'un dictionnaire bilingue. *Peter Troyanskii*, un Russe, a utilisé trois étapes dans le processus de traduction : la phrase source est analysée par un humain en des « symboles logiques », ces symboles sont traduits par la machine en utilisant un dictionnaire bilingue, et un autre homme génère la phrase cible à partir des traductions de ces symboles. Et dans les années 50, ces mécanismes étaient bien connus [Hutchins 2001].

Les premières propositions pour la TA à l'aide de l'ordinateur ont été formulées en 1949 par *Warren Weaver* et la première démonstration d'un système de TA s'est tenue en 1954 et est connue sous le nom de l'« *Expérience de Georgetown – IBM* ». Le système lui-même avait un vocabulaire de 250 mots, six règles de grammaire, et pouvait traduire en anglais seulement une soixantaine de phrases russes soigneusement choisies, principalement dans le domaine de la chimie. La démonstration a été largement relatée dans les journaux et elle a suscité un grand

intérêt public. Les travaux de recherche dans les années 60 en Union Soviétique et aux Etats-Unis se sont concentrés principalement sur des documents scientifiques et techniques russes–anglais.

La recherche en TA a été freinée vers 1966, suite à la sortie du rapport de l'ALPAC (*Automatic Language Processing Advisory Committee*), le comité consultatif sur le traitement automatique des langues du gouvernement des Etats-Unis. Celui-ci a conclu que la TA était plus chère, moins précise et plus lente que la traduction humaine, et que la TA n'était pas susceptible d'atteindre dans l'avenir la qualité d'un traducteur humain. La publication du rapport eut un impact profond sur la recherche en TA et celle-ci, au moins aux Etats-Unis, a presque été totalement abandonnée pendant presque deux décennies. Au Canada, en France et en Allemagne, cependant, la recherche a toujours continué.

Dans les années 70, le système *Systran*[1] a été développé et utilisé par la commission européenne. Le système *TAUM-Meteo* développé à l'Université de Montréal a été installé pour traduire des prévisions météorologiques anglais–français dans les deux sens (de l'anglais vers le français et du français vers l'anglais). Dans les années 80, la diversité et le nombre de systèmes de TA a fortement augmenté : les travaux se sont typiquement focalisés sur une traduction experte utilisant des analyseurs syntaxiques et sémantiques.

Dès le début des années 90, la puissance de calcul a augmenté et plusieurs recherches ont été initiées en modélisation statistique pour la traduction. Les systèmes fondés sur des méthodes probabilistes ont été développés au sein d'*IBM*. D'autres méthodes fondées sur un grand nombre d'exemples de traduction ont été étudiées. Ces nouvelles approches *empiriques* consistaient toutes deux à traiter de grands corpus de textes au lieu de définir des règles syntaxiques et sémantiques. Pendant les années 90, suite aux progrès des systèmes de reconnaissance automatique et de synthèse de la parole, des recherches en traduction de parole ont vu le jour.

Aujourd'hui, il existe de nombreux outils en ligne permettant de traduire d'une langue vers une autre, tels que le système *Systran*, *Google translate*[2] et *Babelfish*[3]. Bien qu'il n'y ait pas de système qui assure une traduction de qualité pour un domaine large, ces systèmes fournissent une sortie utile pour de l'accès à l'information, par exemple.

1.2. Architectures des systèmes de TA

En général, le processus de TA de texte consiste en trois étapes fondamentales : 1) *l'analyse* : analyser le texte source en des représentations intermédiaires en langue source, 2) *le transfert* : transférer ces représentations intermédiaires vers des représentations intermédiaires en langue cible, et 3) *la génération* : générer le nouveau texte en langue cible à partir des représentations intermédiaires en langue cible. Dans [Boitet 2008], l'auteur différencie *l'architecture linguistique* d'un système de TA, caractérisée par les représentations intermédiaires qu'il utilise durant le processus de traduction, de son *architecture computationnelle*, caractérisée par les méthodes de calcul et les ressources utilisées lors des diverses phases transformant une représentation en une autre lors de la traduction.

[1] http://www.systransoft.com ; http://www.systran.fr
[2] http://translate.google.com
[3] http://fr.babelfish.yahoo.com

1.2.1. Architectures linguistiques

Le « *triangle de Vauquois* » proposé par [Vauquois 1968] pour la traduction décrit les différentes architectures linguistiques possibles d'un système de TA. Chaque chemin dans le triangle correspond à une architecture linguistique.

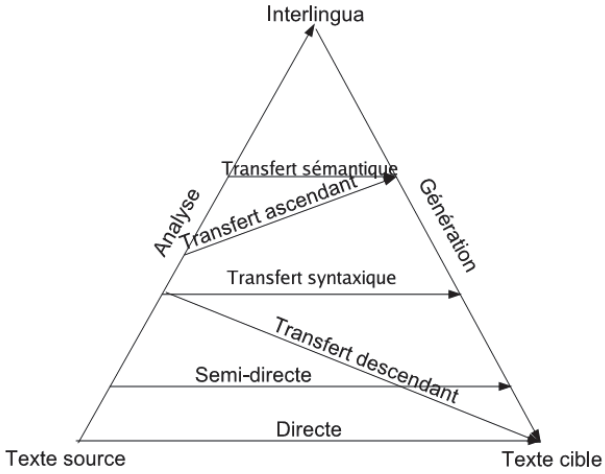


Figure 1-1 : Triangle de Vauquois, représentation des différentes architectures linguistiques (src [Nikoulina 2010])

1.2.1.1 Système de traduction direct

Dans ce type de système de traduction, nous n'utilisons pas de représentation intermédiaire. En d'autres termes, il n'y a pas d'étape d'analyse de la phrase source en représentations intermédiaires ni d'étape de génération de la traduction à partir de ces représentations intermédiaires. La phrase source est segmentée souvent par mots et le système exécute la traduction mot à mot. L'étape de transfert utilise une table bilingue qui va spécifier pour chaque mot ou séquence de mots source un ensemble de règles de traduction et de réarrangement qui permettent de traduire et réordonner les mots dans la phrase de traduction finale.

Les systèmes de traduction directe sont des systèmes bilingues et unidirectionnels : ils traitent une seule paire de langues et dans une seule direction de traduction. Par ailleurs, le manque d'analyse de la phrase source et de connaissance de la construction de la phrase cible est un défaut majeur de ces systèmes [Lavecchia 2010].

Dans les systèmes de traduction semi-directe, la phase source peut être segmentée ou analysée au niveau morphèmes. Après l'étape de transfert lexical, une phase de génération permet d'obtenir une séquence en langue cible.

Cette stratégie peut être utilisée dans certains cas d'application avec un vocabulaire limité. Les systèmes de traduction russe–anglais et anglais–russe dans leurs premières années étaient des systèmes de traduction semi-directe. Et les systèmes de TA probabiliste fondés sur les séquences utilisent l'architecture directe (ou semi-directe) de traduction [Nikoulina 2010] en sont un autre exemple.

1.2.1.2 Système à transfert

Les systèmes de traduction à transfert sont plus complexes que les systèmes directs. Contrairement aux systèmes de traduction directe, le processus de traduction à transfert consiste en trois étapes : l'étape d'analyse, l'étape de transfert, l'étape de génération.

Les systèmes de traduction à transfert nécessitent des connaissances pour analyser les phrases source et cible en des représentations intermédiaires et pour lier ces représentations entre deux

langues. Ces représentations intermédiaires peuvent être à différents niveaux : morphèmes, morphèmes+étiquettes syntaxiques, arbres de dépendances, représentations sémantiques, etc.

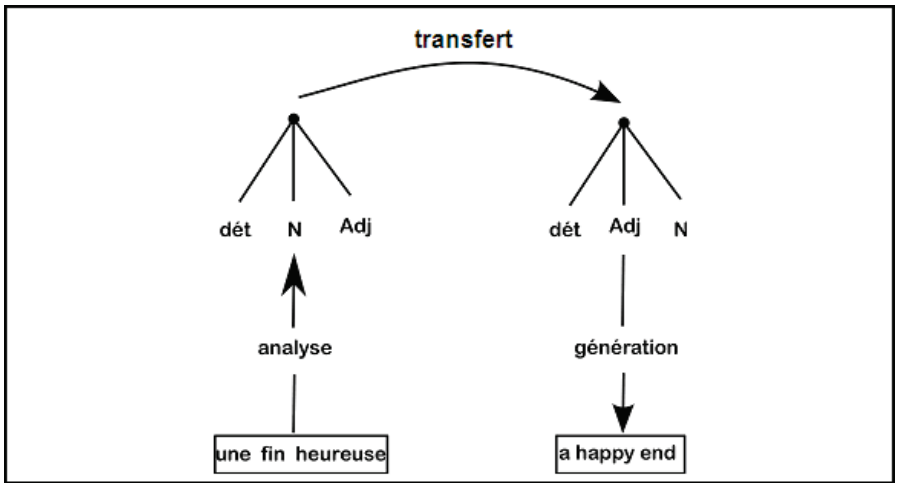


Figure 1-2 : Le système à transfert : un exemple de la traduction d'un groupe nominal en français « une fin heureuse » vers une phrase en anglais (src [Lavecchia 2010])

Parfois, le transfert est effectué à deux niveaux différents. Par exemple, un système à transfert descendant passe de la représentation syntaxique de la phrase source à la représentation morphologique de la phrase cible. Et un système à transfert ascendant passe de la représentation syntaxique de la phrase source à la représentation sémantique de la phrase cible, etc. [Nikoulina 2010].

Les systèmes à transfert comprennent généralement trois dictionnaires : deux dictionnaires de la langue source et de la langue cible contenant des renseignements détaillés sur la morphologie, la grammaire et la sémantique ; et un dictionnaire (ou table) bilingue contenant les règles de transfert entre les représentations intermédiaires de la langue source et celles de la langue cible. Donc la tâche de construction de dictionnaire de transfert est la tâche la plus difficile. Cependant, les systèmes de traduction à transfert peuvent être adaptés pour traduire dans les deux sens de traduction en inversant les règles de transfert.

Des nombreux systèmes de TA fondés sur cette approche sont disponibles aujourd'hui, tels que *Systran*, utilisé par *Babelfish, Reverso, The Translator (Toshiba)*, etc. Le système de traduction *Systran* est un exemple de système à transfert descendant.

1.2.1.3 Système de traduction par interlingue

Cette approche décrit le cas idéal d'une représentation intermédiaire indépendante des langues source et cible : la représentation pivot. Dans ce cas, la traduction de n'importe quelle langue source vers n'importe quelle langue cible est le processus qui consiste à « enconvertir » la phrase source vers la représentation pivot puis à « déconvertir » la phrase cible à partir de cette représentation pivot. Le transfert des représentations de la langue source vers celles de la langue cible n'a plus lieu d'être. L'avantage de cette méthode est la possibilité de l'appliquer dans un environnent multilingue. Pour couvrir tous les sens de traduction entre *n* langues, nous n'avons besoin que de *n* modules d'enconvertion et *n* modules de déconvertion.

Cependant, la complexité de cette méthode réside dans l'obligation de construire un vocabulaire pivot pour représenter tous les concepts possibles de toutes les langues et les liens des concepts entre deux langues. La construction peut être basée sur une langue artificielle « logique », sur une langue auxiliaire « naturelle » (comme l'anglais ou l'espéranto), sur un ensemble de concepts primitifs communs à toutes les langues, ou sur un vocabulaire « universel ». UNL

(*Universal Networking Language*) est un exemple de langage pivot [Uchida 2001] en cours de développement.

1.2.2. Architectures computationnelles

L'architecture computationnelle est caractérisée par les méthodes de calcul et les ressources utilisées dans le processus de traduction. Les architectures linguistiques et computationnelles des systèmes de TA sont indépendantes. Des phases et modules d'architectures linguistiques diverses peuvent être construits en utilisant des méthodes de calcul « expertes » ou « empiriques ».

1.2.2.1 Méthodes « expertes »

Les méthodes « expertes » font appel à la programmation basée sur des modèles de calcul abstraits donnant lieu à des langages spécialisés pour la programmation linguistique. Dans ces méthodes, nous pouvons utiliser les formalismes de grammaires pour modéliser les langues et les structures de données (comme LFG (*Lexical-Functional Grammars*), HPSG (*Head-driven Phrase Structure Grammar*), ou TAG (*Tree Adjoining Grammar*), etc.) ou utiliser des structures d'automates sous la forme d'automates finis ou de transducteurs finis, etc.

Pour la programmation, un langage de haut niveau (comme Lisp ou Prolog, etc.) avec des structures de données et de contrôle plus adaptées à la programmation linguistique peut être utilisé. Le plus souvent, les règles d'analyse, de transfert et de génération sont définies sur la base de connaissances approfondies en linguistique d'experts humains [Boitet 2008].

Beaucoup de systèmes à transfert se trouvent être développés dans le cadre de la TA experte ; par ailleurs, des systèmes à interlingue sont en majorité des systèmes de TA experte.

1.2.2.2 Méthodes « empiriques » : la TA fondée sur les exemples et la TA probabiliste

Les méthodes expertes, comme leur nom l'indique, nécessitent l'intervention de spécialistes de la langue et sont donc coûteuses en temps et ressources humaines, pour leur développement. Avec la disponibilité croissante de données textuelles en grande quantité, les approches empiriques, pour lesquelles des corpus parallèles bilingues de grande taille sont utilisés (l'ensemble de textes en une langue et leur traduction dans une autre langue), se sont imposées. Les systèmes empiriques de traduction sont généralement très dépendants du domaine. Ils donnent de bons résultats lorsqu'ils traduisent des textes semblables aux textes d'apprentissage, mais se dégradent rapidement sur d'autres types de textes.

Nous pouvons distinguer deux approches en TA empirique : la TA fondée sur les exemples (EBMT, *Example-Based Machine Translation*) et la TA probabiliste (SMT, *Statistical Machine Translation*). Dans cette section, nous présentons en bref les systèmes de TA fondés sur les exemples, et dans le chapitre suivant, nous présenterons en détail l'approche de TA probabiliste.

L'approche de TA fondée sur les exemples a été proposée par [Nagao 1984]. L'idée de base de cette approche est de réutiliser des exemples de traductions existantes comme base pour la nouvelle traduction.

D'abord, une base d'exemples est collectée et contient un ensemble de phrases en langue source associées à leur traduction en langue cible. Le processus de traduction d'une phrase source est basé sur la correspondance entre cette phrase et la base d'exemples. Le processus de TA fondée sur les exemples se décompose en trois étapes :

(1) Trouver les correspondances : pour chaque phrase source à traduire, les phrases «exemple» les plus proches dans la base d'exemples sont sélectionnées. La fonction la plus importante dans

ce type de système est de savoir comment trouver la similitude de la phrase à traduire et une phrase « exemple ». Il existe plusieurs méthodes de sélection basées sur la capacité de remplacer des mots correspondants entre deux phrases, des listes de synonymes, ou basées sur la similarité au niveau syntaxique ou sémantique, ou encore basées sur des informations statistiques.

Si aucune phrase « exemple » correspondant avec la phrase source entière n'est pas trouvée, nous segmentons la phrase source à traduire en plusieurs fragments, dont des fragments similaires correspondants existent dans la base d'exemples.

(2) Aligner les phrases : la phrase à traduire et les phrases exemples diffèrent sur un ou plusieurs mots. L'alignement est utilisé pour identifier les parties des phrases qui peuvent être réutilisées dans la traduction de la phrase source. Les autres parties qui diffèrent entre la phrase source et les phrases exemples peuvent être traduites en utilisant un dictionnaire bilingue.

(3) Combiner : dans cette dernière étape, les traductions des parties réutilisées et non réutilisées sont assemblées d'une manière logique pour créer la traduction de la phrase source entière. La traduction est extraite à partir de véritables exemples et la qualité de traduction est garantie.

Par exemple, la Figure 1-3 représente un exemple simple de traduction anglais – japonais pris à partir des travaux de [Sato 1990]. Pour traduire la phrase anglaise « *he buys a book on international politics* », nous pouvons utiliser les phrases exemples « *he buys a notebook* » et « *i read a book on international politics* ».

Figure 1-3 : Un exemple simple de traduction anglais – japonais fondée sur les exemples (L'exemple est pris de [Sato 1990])

La Figure 1-4, quant à elle, représente une déduction simple pour trouver la meilleure traduction d'un mot basée sur la similitude sémantique (autre exemple sorti des travaux de [Nagao 1984]). Pour trouver la meilleure traduction du mot « *eat* » dans la phrase « *he eats potatoes* » et «*sulphuric acid eats iron* » nous pouvons faire une déduction à partir des phrases exemplaires « *a man eats vegetables* » et « *acid eats metal* ».

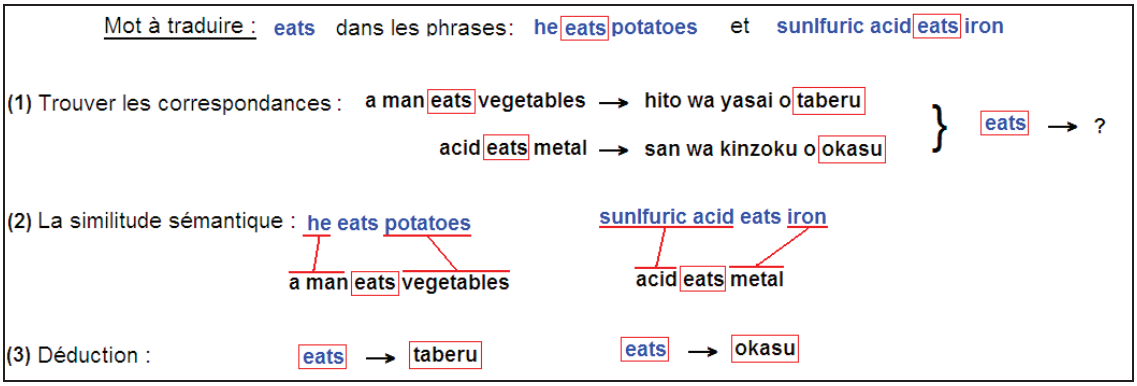


Figure 1-4 : Un exemple simple de déduction dans la traduction fondée sur les exemples (L'exemple est pris de [Nagao 1984])

L'approche de la TA probabiliste est présentée dans le chapitre suivant ; en effet, comme elle est utilisée dans ce travail de thèse, nous y consacrons un chapitre complet.

1.2.2.3 TA hybride

L'approche hybride est un sujet de recherche très actif ces dernières années avec plusieurs types d'hybridation. Les méthodes empiriques peuvent être intégrées dans le processus de traduction par les méthodes expertes, et réciproquement.

L'intégration peut être simple ou complexe. Par exemple, un système de traduction expert peut utiliser une terminologie extraite empiriquement à partir de corpus parallèles bilingues en même temps que des dictionnaires créés par des experts. Des hypothèses d'un système de traduction expert peuvent être reclassées avec des probabilités lexicales, un modèle de langue, etc. [Oepen 2007]. Inversement, des hypothèses issues d'un système de traduction empirique peuvent être triées à l'aide d'informations linguistiques sources ou cibles [Och 2003b], [Shen 2004].

Par ailleurs, certains modules du système de traduction expert peuvent être remplacés par des modules empiriques. Par exemple, le module de désambiguïsation lexicale et le module d'extraction de règles de transfert peuvent être construits empiriquement en utilisant un corpus parallèle [Sanchez-Martinez 2008]. Il est aussi possible de prétraiter un corpus parallèle, et d'apprendre les alignements pour un système de traduction empirique à l'aide d'analyseurs linguistiques experts [Ma 2008].

Plus complexe, l'intégration des connaissances expertes dans le processus de décodage du système empirique peut impliquer un changement de l'architecture linguistique (par exemple le transfert descendant et le transfert ascendant présentés dans la section 1.2.1.2).

Chapitre 2 : Approche de la traduction automatique probabiliste

2.1. Introduction

La traduction automatique (TA) probabiliste considère le problème de la traduction comme un problème d'apprentissage automatique. Un algorithme d'apprentissage extrait des informations statistiques à partir de grandes bases de données de textes déjà traduits (les corpus parallèles). Le système est alors capable de traduire des phrases inédites. L'approche probabiliste permet notamment d'obtenir de multiples hypothèses de traduction que nous différencions entre elles par des probabilités.

Les premières idées de la TA probabiliste ont été introduites par *Warren Weaver* en 1949. Le mémorandum écrit par Warren Weaver en 1949 est peut-être la publication la plus influente dans les premiers jours de la TA probabiliste. Weaver avait d'abord cité la possibilité d'utiliser l'ordinateur pour traduire en 1947. Ses propositions concernent l'applicabilité de méthodes cryptographiques à la traduction, l'existence de régularités linguistiques universelles et logiques entre les langues.

La TA probabiliste a été réintroduite en 1991 par des chercheurs du centre de recherche d'IBM et elle a contribué au regain d'intérêt pour la TA de ces dernières années. Aujourd'hui, la TA probabiliste est une des méthodes de TA les plus étudiées dans le monde.

L'intérêt pour la TA probabiliste de ces dernières années est dû à plusieurs facteurs :

- de plus en plus de données de texte traduites deviennent disponibles et accessibles sur le Web, et cela pour de nombreuses langues,
- l'évolution des performances des machines a permis d'implémenter des systèmes probabilistes appris sur des données volumineuses et manipulant des milliards de valeurs statistiques ;
- à condition qu'un corpus bilingue soit disponible, l'approche de TA probabiliste peut être adaptée à n'importe quelle paire de langues, tandis que les autres approches de traduction exigent l'élaboration de règles linguistiques qui peuvent être coûteuses et qui le plus souvent ne sont pas communes entre les langues ;
- l'élaboration de boîtes à outils disponibles et gratuites de TA probabiliste facilite l'entrée de nouveaux chercheurs dans le domaine de la TA.

Aujourd'hui, avec une boîte à outils disponible et des textes parallèles suffisants, nous pouvons construire un système de traduction pour une nouvelle paire de langues dans un temps relativement réduit, et appliquer cette démarche à un grand nombre de paires de langues. La précision de ces systèmes dépend essentiellement de la quantité, de la qualité et de l'adéquation au domaine des textes parallèles utilisés.

2.1.1. Description formelle

Formellement, la tâche de traduction consiste à transformer une séquence de mots dans une langue source *F* présentée à l'entrée en une séquence équivalente de mots dans une autre langue cible *E*. Nous notons une séquence de *I* mots dans la langue source *F* comme $f_1 f_2 ... f_I$, ou $f_1^I \in F$, et une séquence de *J* mots dans la langue cible *E* comme $e_1 e_2 ... e_J$, ou $e_1^J \in E$.

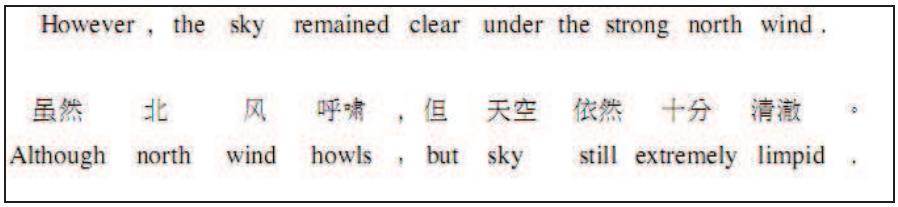


Figure 2-1 : Exemple d'une phrase anglaise et de sa traduction chinoise (avec les mots équivalents) (src [Lopez 2007])

Nous assignons à chaque paire de phrases (*e*, *f*) une probabilité P(*e*|*f*), qui est considérée comme la probabilité que le système de traduction produise la phrase cible *e* lorsque la séquence *f* est présentée à l'entrée. Nous espérons que P(*e*|*f*) est faible avec des paires de phrases peu probables (par exemple « *Le président Lincoln était un bon avocat* » et « *In the morning I brush my teeth*») et élevée avec des paires plus probables (par exemple « *Le président Lincoln était un bon avocat*» et « *President Lincoln was a good lawyer* »).

Nous prenons comme hypothèse que chaque phrase source *f* peut avoir comme traduction n'importe quelle phrase *e* dans la langue cible. Parmi les traductions possibles, comment trouvons-nous la meilleure traduction ? Supposons que le système de traduction produit *e'* et que *L(e,e',f)* est le coût (la perte) pour produire la traduction correcte *e* de *f*, nous pouvons reformuler le problème comme étant celle de trouver la meilleure traduction (désignée par *ê)* où la perte attendue est minimale. Nous avons la règle de décision avec la perte minimale [Och 2005] :

$$\hat{e}(f) = \arg\min_e \sum_{e'} L(e, e', f) \cdot P(e \mid f) \qquad \textbf{(2-1)}$$

Supposons que la fonction de perte est la binaire, c'est-à-dire que *L(e,e',f)=0* si *e'* et *e* sont identiques, et sinon *L(e,e',f)=1*. Nous obtenons ainsi la règle de décision du maximum à posteriori (MAP) et une traduction optimale *ê* d'une phrase source *f* sera définie comme suit :

$$\hat{e}(f) = \arg\max_e P(e \mid f) \qquad \textbf{(2-2)}$$

À partir de cette règle, nous pouvons dire que le processus de TA probabiliste d'une phrase source consiste à trouver une phrase cible la plus probable selon la probabilité *P*(*e*|*f*). Calculer la probabilité réelle est impossible, donc nous allons approximer un score qui correspond à *p(e|f)* en utilisant un modèle qui est appris à partir d'un corpus parallèle. Il y a donc trois problèmes que nous devons résoudre afin de pouvoir construire un système de TA probabiliste :

1. *Modélisation* : comment pouvons-nous calculer la probabilité *p*(*e*|*f*) ? Il semble impossible d'énumérer et de stocker la probabilité *p(e|f)* de toutes les paires de phrases source et cible. Nous devons définir un modèle mathématique qui permet d'assigner une probabilité *p*(*e*|*f*) à chaque paire (*f*,*e*). Normalement, la phrase source est réécrite en traduisant et réordonnant des mots, jusqu'à ce que tous les mots dans la phrase source

soient remplacés par les mots dans la langue cible. Chaque mot source est souvent ambigu entre plusieurs traductions. De plus, les mots et leurs traductions équivalentes n'apparaissent pas dans le même ordre dans les deux phrases. Donc il faut modéliser ces phénomènes dans le modèle mathématique. Nous l'appelons l'étape de modélisation.

2. *Apprentissage* : avec ce modèle mathématique, comment pouvons-nous estimer les paramètres du modèle qui permet d'estimer $p(e|f)$. Pour cela, nous ne pouvons collecter qu'un corpus, appelé corpus d'apprentissage, de taille limitée. Nous devons utiliser une méthode d'estimation pour assigner la valeur $p(e|f)$ à chaque paire de phrases (f,e) dans le corpus d'apprentissage. Nous appelons cette étape l'estimation des paramètres.
3. *Décodage* : finalement, pour une phrase en entrée f, nous devons chercher une phrase cible e qui est conforme avec le modèle donné et attribue à cette phrase la probabilité $p(e|f)$ la plus élevée. Nous l'appelons l'étape de décodage.

2.2. Modélisation

Nous revenons au problème consistant à modéliser la probabilité $p(e|f)$ par un modèle mathématique. En TA probabiliste, deux classes de modèles sont proposées : les modèles génératifs et les modèles discriminants.

2.2.1. Les modèles génératifs

Répétons que le processus de TA probabiliste d'une phrase source f consiste à trouver la phrase cible e la plus probable selon le modèle de probabilité $p(e|f)$ comme décrit dans l'équation (2-2). D'après le théorème de Bayes :

$$p(e \mid f) = \frac{p(e) \cdot p(f \mid e)}{p(f)} \quad \textbf{(2-3)}$$

Le terme $p(f)$ ne dépend pas de e, il n'a donc aucune influence sur le calcul de la fonction *argmax*. L'équation (2-2) est reformulée dans l'équation (2-4) [Brown 1990]

$$\hat{e}(f) = \arg\max_e \frac{p(e) \cdot p(f \mid e)}{p(f)} = \arg\max_e p(e) \cdot p(f \mid e) \quad \textbf{(2-4)}$$

La modélisation de $p(e|f)$ est reportée sur deux modèles indépendants $p(e)$ et $p(f|e)$. En TA probabiliste, $p(e)$ est appelé le modèle de langue dont les paramètres sont appris sur un corpus monolingue en langue cible E, et $p(f|e)$ est le modèle de traduction dont les paramètres sont appris sur un corpus d'apprentissage parallèle aligné. Le modèle de traduction détermine si la phrase f est une traduction probable de la phrase e. Il peut être vu comme un dictionnaire bilingue dans lequel chaque entrée met en relation de traduction un groupe de mots source avec un groupe de mots cible. Chaque association entre un groupe de mots et sa traduction se voit attribuer une probabilité. Le modèle de langue assigne des probabilités aux suites de mots dans la langue cible. Il détermine si la phrase e est bien formée dans la langue cible.

[Brown 1993] a présenté les modèles IBM qui sont connus aujourd'hui comme les premiers modèles mathématiques d'un système de traduction basé sur les mots (voir en détails dans la section 2.2.1.2). Alors que notre objectif est de découvrir la phrase e pour une phrase f donnée, nous voyons qu'il faut en fait faire l'inverse après application de la formule de Bayes. Cette transformation est à l'origine des modèles d'IBM qui sont décrits comme des modèles cible à source, ce qui veut dire que le système va retrouver la phrase source f à partir de la phrase cible e donnée. Les modèles d'IBM reprennent le principe du modèle du « canal bruité de transmission»

(*noisy channel model*) appliqué largement dans la reconnaissance automatique de la parole. Le modèle de canal bruité est décrit dans la Figure 2-2.

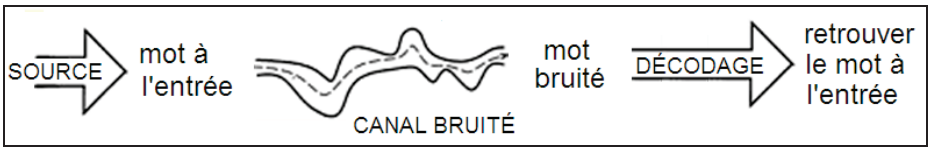


Figure 2-2 : Le modèle de canal bruité de transmission

Un modèle de source produit un mot à l'entrée du canal de transmission. À cause du bruit de canal, le mot reçu après le canal est différent du mot en entrée. Le but du modèle de décodage est de retrouver le mot à l'entrée à partir du mot reçu et des connaissances sur le canal bruité.

Dans le cas de modèles d'IBM, la source est le modèle *p(e)*. Il produit la phrase cible *e* qui est ensuite transformée par le modèle du canal *p(f|e)* pour produire la phrase source *f*. Si nous avons la phrase *f*, nous pouvons déduire la phrase *e* en utilisant les connaissances des modèles *p(e)* et p*(f|e)*.

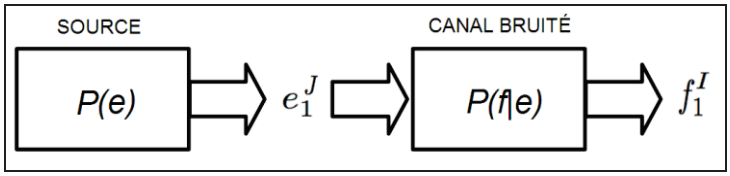


Figure 2-3 : Le modèle de canal bruité du modèle d'IBM

2.2.1.1 Modèle de langue

Le modèle de langue est utilisé largement dans le domaine du traitement automatique des langues naturelles, avec plusieurs applications sur la reconnaissance de l'écriture, la correction d'orthographe, la reconnaissance optique de caractères, la reconnaissance automatique de la parole, la TA, etc.

La plupart des modèles de langue dans les systèmes de traduction sont basés sur les modèles *n-grammes*. La probabilité *p(e)* d'une phrase *e* dans la langue cible peut être décomposée par :

$$p(e) = p(e_1 e_2 ... e_J) = \prod_{j=1}^{J} p(e_j \mid e_1 e_2 ... e_{j-1}) \quad \textbf{(2-5)}$$

Les modèles n-grammes font l'hypothèse que le mot e_i de *e* n'est dépendant que des *n-1* mots le précédant. La probabilité *p(e)* est réécrite comme suit :

$$p(e) = p(e_1 e_2 ... e_J) = \prod_{j=1}^{J} p(e_j \mid e_{j-(n-1)} ... e_{j-1}) \quad \textbf{(2-6)}$$

Les paramètres du modèle sont appris sur un grand corpus de données monolingue. Plusieurs techniques de lissage sont proposées qui permettent d'estimer mieux les probabilités quand il y a insuffisamment de données, ou permettent d'estimer les probabilités des n-grammes non observés [Chen 1998], [Rosenfeld 2000], etc. Récemment, pour résoudre ces problèmes, une nouvelle représentation de mots dans un espace/domaine continu a été proposée. Pour le faire, un réseau de neurones multicouches complètement connecté est utilisé. L'entrée du réseau est la séquence de mots « $e_{j-(n-1)} ... e_{j-1}$ ». Ces mots sont projetés dans un espace continu (la couche de projection) et le processus d'estimation est réalisé dans cet espace. Les sorties sont les probabilités $p(e_j=i \mid e_{j-(n-1)} ... e_{j-1})$ pour tous les mots dans le vocabulaire ($i \in$ vocabulaire). Plus de détails peuvent être trouvés dans les travaux de [Schwenk 2006, 2007, 2010]. En résumé, un modèle perfectionné de langue est bénéfique à la traduction automatique probabiliste [Eck 2004], [Kirchhoff 2005], [Schwenk 2006].

2.2.1.2 Modèle de traduction à base de mots

La correspondance entre une phrase et sa traduction (Figure 2-1) peut être décomposée en un certain nombre de plus petites correspondances au niveau du mot ou du groupe de mots. En fonction de l'unité fondamentale de traduction choisie, nous avons deux types de modélisation du modèle de traduction : le modèle de traduction à base de mots et le modèle de traduction à base de groupes de mots.

Les premiers modèles probabilistes pour la TA ont été introduits par IBM. [Brown 1993] a défini cinq modèles de traduction à base de mots avec une complexité croissante et les appelés les « modèles IBM ». Parce que les modèles IBM sont des modèles cible à source, dans la suite de cette section, la phrase source est notée $e_1e_2...e_I$, ou e_1^I et la phrase cible est $f_1f_2...f_J$, ou f_1^J.

Les correspondances entre des mots (ou des groupes de mots) d'une phrase donnée et ceux de sa traduction sont appelées l'alignement. Nous désignons par le terme général « alignement » d'une paire de phrases, l'ensemble de ces correspondances. Brown fait l'hypothèse que l'alignement est asymétrique. C'est-à-dire chaque mot de la phrase cible est connecté à exactement un et un seul mot de la phrase source. Les mots de la phrase source ne sont pas contraints, chaque mot de la phrase source peut correspondre à un nombre de mots de la phrase cible, défini par sa *fertilité*. En outre, un mot spécial, « le mot NULL », peut être utilisé lorsqu'un ou plusieurs mots d'une phrase n'ont pas de correspondance dans l'autre phrase (lorsque ces mots sont omis dans la traduction).

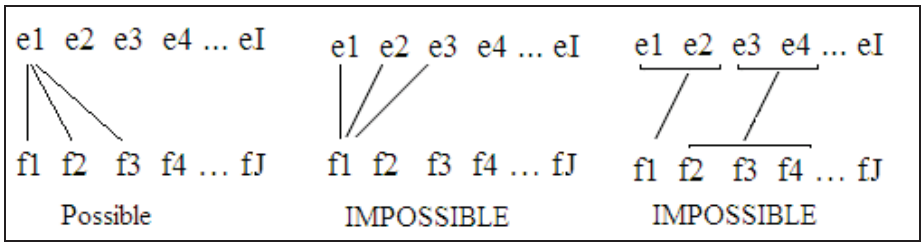


Figure 2-4 : La contrainte sur l'alignement en mots de [Brown 1993]

Notons l'alignement entre les phrases source et cible par A. $A = a_1^J = a_1a_2 \ldots a_J$. Chaque élément a_j présente une correspondance et sa valeur dans l'intervalle $\{0, 1, \ldots, I\}$. La valeur de $a_j=i$ indique que le mot f_j correspond au mot e_i. $a_j = 0$ si f_j est aligné avec le mot NULL e_0 de la phrase e. Considérant l'ensemble des alignements possibles, la vraisemblance d'une traduction (f,e) est définie par :

$$p(f \mid e) = \sum_a p(f, a \mid e) \qquad \textbf{(2-7)}$$

où a est un alignement possible entre f et e. La somme s'effectue sur tous les alignements possibles.

Les cinq modèles IBM diffèrent dans l'estimation de la probabilité $p(f|e)$. Chaque modèle s'appuie sur les paramètres estimés par le modèle précédent.

Pour les modèles IBM-1 et -2, pour générer la phrase f, tout d'abord nous déterminons la longueur J de la phrase f. Ensuite, pour chaque position $j = 1..J$ nous sélectionnons la position a_j du mot e_i qui lui sera associée. Cette position dépend de e, de J, et des mots déjà alignés (a_1^{j-1} et f_1^{j-1}), enfin nous produisons le mot cible f_j en fonction de a_1^j , f_1^{j-1} , J et e. La probabilité $p(f,a|e)$ est définie comme dans l'équation (2-8).

$$p(f, a \mid e) = p(J \mid e) \prod_{j=1}^{J} p(a_j \mid a_1^{j-1}, f_1^{j-1}, J, e)\, p(f_j \mid a_1^j, f_1^{j-1}, J, e) \qquad \textbf{(2-8)}$$

Le modèle « IBM-1 » suppose que chaque mot f_j peut être connecté à n'importe quel mot e_i avec la même probabilité et $p(J|e)$ est une constante $= \varepsilon$. Le modèle de traduction repose donc sur une seule loi lexicale de probabilité de traduction de e_{aj} en f_j notée par $t(f_j|e_{aj})$.

$$p(f,a \mid e) = \frac{\varepsilon}{(I+1)^J} \prod_{j=1}^{J} t(f_j \mid e_{a_j}) \quad \textbf{(2-9)}$$

Après plusieurs transformations mathématiques, il est démontré que $p(f|e)$ peut être calculée de manière exacte par la formule suivante :

$$p(f \mid e) = \frac{\varepsilon}{(I+1)^J} \prod_{j=1}^{J} \sum_{i=0}^{I} t(f_j \mid e_i) \quad \textbf{(2-10)}$$

Le modèle « IBM-2 » : en plus de la loi de traduction lexicale $t(f|e)$, le modèle IBM-2 introduit une loi d'alignement ou de distorsion qui détermine la position du mot source aligné avec le mot cible. Cette probabilité dépend de la position du mot cible et de la longueur des phrases :

$$p(f,a \mid e) = \varepsilon \prod_{j=1}^{J} t(f_j \mid e_{a_j}) a(a_j \mid j,J,I) \quad \textbf{(2-11)}$$

Après plusieurs transformations mathématiques, nous obtenons :

$$p(f \mid e) = \varepsilon \prod_{j=1}^{J} \sum_{i=0}^{I} t(f_j \mid e_i) a(i \mid j,J,I) \quad \textbf{(2-12)}$$

où $\sum_{i=0}^{I} a(i \mid j,J,I) = 1$.

Le modèle « HMM » de [Vogel 1996] est similaire au modèle IBM-2 avec une loi lexicale $t(f|e)$ et une loi de distorsion. La forme de la loi de distorsion du modèle IBM-2 est une loi d'ordre 0 avec la forme $a(a_j|j,J,I)$ tandis que celle du modèle HMM est une loi d'ordre 1 de forme $a(a_j|a_{j-1},I)$. La loi de distorsion du modèle HMM dépend de l'alignement du mot cible précédent et de la longueur de la phrase source.

Pour les modèles **IBM-3, -4,** et **-5**, le processus de transformation d'une phrase source vers la phrase cible contient trois étapes :

1. En général, le nombre de mots de la phrase *e* est différent de celui de la phrase f à cause des mots composés, de la morphologie et des expressions idiomatiques. Pour représenter ce phénomène, chaque mot e_i choisit un nombre de mots ϕ_i dans la phrase *f* qu'il va générer. Ce nombre ϕ_i est appelé la *fertilité* du mot e_i. Lorsque nous rencontrons le mot e_i dans la phrase *e*, nous produisons ϕ_i copies de e_j dans la phrase *f*.
2. Pour ϕ_i copies de e_i, nous produisons ϕ_i mots dans la phrase *f*.
3. Les mots dans la phrase *f* sont réordonnés pour obtenir la phrase *f* finale.

La Figure 2-5 visualise un exemple qui transforme une phrase anglaise vers une phrase chinoise à base de mots.

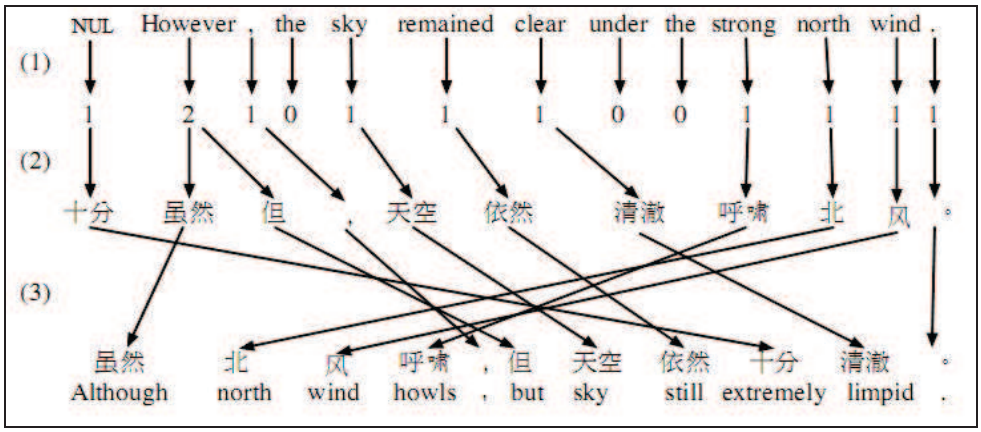


Figure 2-5 : Trois étapes générales pour transformer une phrase anglaise vers une phrase chinoise avec le modèle à base de mots
(src [Lopez 2007])

Le modèle « IBM-3 » intègre donc en plus une loi de fertilité $n(\phi_i|e_i)$ pour chaque position $i \in [1,I]$ dans la phrase source. ϕ_i est le nombre de mots cibles alignés avec e_i. Le modèle IBM-3 autorise aussi l'alignement des mots cibles avec le mot source NULL e_0. Nous appelons ces mots cibles des « mots faux ». La probabilité de ce nouveau mot cible connecté au mot source réel est p_0 et la probabilité de générer un mot faux est $p_1=1-p_0$. Le modèle de distorsion pour les mots réels du modèle IBM-3 est comme celui du modèle IBM-2.

Enfin, la probabilité $p(f,a|e)$ est calculée par la formule suivante :

$$p(f,a|e) = \binom{J-\phi_0}{\phi_0} p_0^{J-2\phi_0}(1-p_0)^{\phi_0}\frac{1}{\phi_0!}\times\prod_{i=0}^{I}\phi_i! \times \prod_{i=1}^{I} n(\phi_i|e_i)\times\prod_{j=1}^{J} t(f_j|e_{a_i})\times\prod_{j:a_j\neq 0}^{J} a(j|a_j,I,J) \quad \textbf{(2-13)}$$

la probabilité de créer ϕ_0 mots faux et les insérer dans la phrase cible

loi de fertilité — loi de traduction lexicale — loi d'alignement

Le modèle « IBM-4 » (équation (2-14)) ne diffère du modèle IBM-3 que par son modèle de distorsion. Il définit deux lois de distorsion. La première loi $d_{=1}(\Delta j|E, F)$ permet de positionner le premier mot cible f généré par un mot source e_i. La deuxième loi $d_{>1}(\Delta j|F)$ permet de positionner les derniers ϕ_{i-1} mots cibles f générées par le même mot source e_i. La distorsion dépend de la classe du mot cible f et de la classe du dernier mot source précédant e_i (par exemple les adjectifs anglais précèdent les noms).

$$p(f,a|e) = \prod_{i=1}^{I} n(\phi_i|e_i)\times t(f_{\tau_{i,1}}|e_i)\,d_{=1}(\tau_{i,1}-\overline{\tau_{i-1}}\,|\,E_{i-1},F(f_{\tau_{i,1}})) \times \prod_{k=2}^{\phi_i} t(f_{\tau_{i,k}}|e_i)\,d_{>1}(\tau_{i,k}-\tau_{i,k-1}\,|\,F(f_{\tau_{i,k}})) \times \binom{J-\phi_0}{\phi_0} p_0^{J-2\phi_0}(1-p_0)^{\phi_0}\frac{1}{\phi_0!}\times\prod_{k=1}^{\phi_0} t(f_{\tau_{0,k}}|e_0) \quad \textbf{(2-14)}$$

loi de fertilité

loi de traduction lexicale et loi de positionnement pour le premier mot cible $f_{\tau_{i,1}}$ généré par un mot source e_i

loi de traduction lexicale et loi de positionnement pour les derniers mots cibles $f_{\tau_{i,k}}$ générés par un mot source e_i

la probabilité de créer ϕ_0 mots faux — loi de traduction lexicale e_0

Le modèle « IBM-5 » est identique au modèle IBM-4 mais il corrige la déficience du modèle IBM-4. Les lois de distorsion du modèle IBM-4 ne prennent pas en compte les positions cibles déjà couvertes. En plus, le modèle IBM-4 assigne des probabilités aux objets qui ne sont pas vraiment des séquences de mots cible. Le modèle IBM-5 élimine ces déficiences, il donne des résultats beaucoup plus satisfaisants. Plus de détails et d'explications sur les modèles IBM se trouvent dans [Brown 1993].

2.2.1.3 Modèle de traduction par groupe de mots

En réalité, la traduction de certains mots dépend d'autres mots dans la même phrase où une séquence consécutive de mots est traduite comme une unité. Or, avec un simple alignement en mots, les mots sont traduits indépendamment les uns des autres. En plus, la segmentation d'une phrase en mots est un problème difficile, particulièrement pour les langues avec une morphologie complexe comme l'allemand ou pour les langues avec une frontière de mots ambiguë comme le chinois.

L'alignement en groupe de mots présente donc beaucoup d'avantages et il est utilisé dans le cas où une séquence consécutive de n mots dans la phrase f correspond à une séquence consécutive de m mots dans la phrase e ($n,m \geq 1$). Le terme « groupe de mots » ici n'a pas de sens linguistique. Il y a cependant un nombre important de travaux qui étudient la traduction à base de groupes de mots ayant une motivation linguistique (un groupe de mots correspond à un sous-arbre dans l'arbre syntaxique qui représente la structure syntaxique d'une phrase) comme [Yamada 2001], [Collins 1997], [Imamura 2002], etc. Cependant, nous n'abordons pas cette approche dans nos travaux.

L'approche de traduction probabiliste par groupe de mots est largement utilisée aujourd'hui. Les avantages de la traduction basée sur des groupes de mots par rapport à des mots seuls sont :

- L'ordre local est capturé dans le groupe de mots donc le système n'a plus à gérer les réarrangements locaux tandis que dans un système à base de mots, il fait partie intégrante du modèle d'alignement ;
- l'approche traite mieux les expressions idiomatiques que l'approche basée sur les mots ;
- avec un grand nombre de données disponibles, des phrases entières peuvent quasiment être couvertes.

L'inconvénient de cette approche est qu'elle nécessite un espace important pour stocker tous les groupes de mots.

[Koehn 2003b], [Marcu 2002], [Och 2004] ont présenté l'approche de traduction probabiliste par groupe de mots dans leurs travaux. Cette approche n'aborde pas le concept du mot NULL et la fertilité de chaque mot source. Chaque groupe de mots de la langue source est non vide et il est traduit en exactement un groupe de mots non vide de la langue cible. Le processus de transformation d'une la phrase source de I mots f_1^I vers une phrase cible de J mots e_1^J est décrit de la façon suivante :

1. la phrase source f_1^I est segmentée en Z groupes de mots distincts $\tilde{f}_1^Z = \tilde{f}_1 ... \tilde{f}_Z$. Chaque groupe $\tilde{f}_z = f_{i_1}^{i_2}, i_1 \neq i_2$;
2. chaque groupe de mots $\tilde{f}_z$ est remplacé par un groupe de mots de la langue cible $\tilde{e}_z$. Cette traduction en groupe de mots est modélisée par une probabilité de traduction en groupe de mots $t_g(\tilde{f}_z | \tilde{e}_z)$. Nous avons Z groupes $\tilde{f}_z$ donc nous obtenons Z groupes $\tilde{e}_z$. La longueur de la phrase cible e_1^J est $J = \sum_{z=1}^{Z} |\tilde{e}_z|$;
3. les groupes de mots dans la langue cible sont permutés pour obtenir la phrase cible finale. avec la probabilité de distorsion relative $d_g(a_z - b_{z-1})$, où : a_z indique la position de départ du groupe de mots source qui a été traduit en le z^{eme} groupe de mots cibles ; et b_{z-1}

indique la dernière position du groupe de mots source traduits en le *(z-1)*^{eme} groupe de mots cibles. [Koehn 2003b] a proposé un modèle de distorsion simple $d_g(a_z - b_{z-1}) = \alpha^{|a_z - b_{z-1} - 1|}$ avec un paramètre α approprié.

La Figure 2-6 visualise un exemple du modèle de transformation en groupes de mots qui transforme une phrase anglaise vers une phrase chinoise.

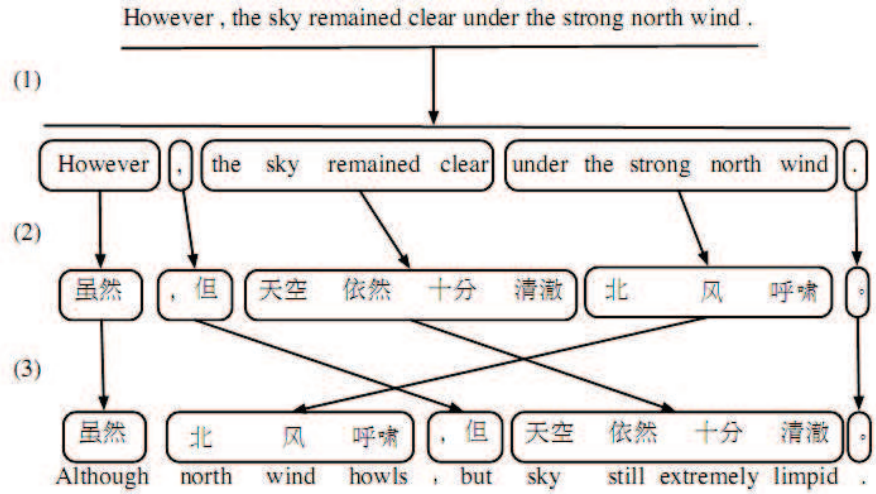


Figure 2-6 : La transformation d'une phrase anglaise vers une phrase chinoise avec le modèle de transformation à base de groupes de mots (src [Lopez 2007])

Un modèle de traduction dans cette approche inclut à la fois un modèle de traduction en groupes de mots et un modèle de distorsion. Le modèle de traduction en groupes de mots est décomposé en :

$$p(f \mid e) = p(\tilde{f}_1^Z \mid \tilde{e}_1^Z) = \prod_{z=1}^{Z} t_g(\tilde{f}_z \mid \tilde{e}_z) d_g(a_z - b_{z-1}) \quad \textbf{(2-15)}$$

[Marcu 2002] a présenté une autre méthode différente qui permet d'établir la traduction en groupes de mots. Il présente un modèle de probabilités jointes pour mettre en évidence les équivalences de mots et de groupes de mots directement à partir d'un corpus bilingue. Ce modèle génère la phrase source et la phrase cible simultanément. Ainsi, les approches classiques forment la probabilité *p(e|f)*, le modèle de probabilité jointe décrit la manière d'utiliser la probabilité jointe *p(e,f)*. Ce modèle donne lieu à deux distributions, $t_g(\tilde{f}_z, \tilde{e}_z)$ qui est la probabilité jointe des groupes de mots $\tilde{f}_z, \tilde{e}_z$, et $d_g(i,j)$ qui est la probabilité de distorsion entre les positions *i* et *j* au sein d'une paire de phrases (voir [Marcu 2002] pour plus de détails).

2.2.2. Vers des modèles discriminants

La combinaison d'un modèle de traduction et d'un modèle de langue avec un produit simple n'est pas forcément optimale. Elle est considérée optimale si les distributions réelles de la probabilité sont utilisées, alors que la méthode d'apprentissage ne peut fournir que des approximations de distributions réelles. Par conséquent, une combinaison différente des deux modèles pourrait donner de meilleurs résultats. En plus, il n'y a aucun moyen d'ajouter d'autres sources de connaissances aux modèles génératifs. Les modèles discriminants sont donc une alternative intéressante et sont aujourd'hui largement utilisés sous la forme pratique des modèles log-linéaires.

Le modèle log-linéaire propose une modélisation directe de la probabilité *p(e,a|f)* comme une combinaison linéaire de *K* fonctions de traits $h_k(e,a,f)$, $k=1..K$ [Och 2002]. Une fonction de traits

peut être n'importe quelle fonction qui attribue une valeur non négative à chaque paire de phrases source – cible.

$$p(e \mid f) = p_{\lambda_1^K}(e \mid f) = \frac{\exp \sum_{k=1}^{K} \lambda_k h_k(e, f)}{\sum_{e'} \exp \sum_{k=1}^{K} \lambda_k h_k(e', f)} \quad \textbf{(2-16)}$$

Ce modèle contient K paramètres $\lambda_1^K = \{\lambda_k \mid k = 1..K\}$, qui déterminent la contribution de chaque fonction de traits $h_k(e,a,f)$ à la valeur finale de $p(e,a|f)$, donc nous les appelons les poids du trait. $\lambda_k > 0$ indique que la fonction de traits $h_k(e,a,f)$ est en corrélation avec $p(e,a|f)$, $\lambda_k < 0$ indique une corrélation inverse, et $\lambda_k = 0$ indique que $h_k(e,a,f)$ est inutile pour calculer $p(e,a|f)$.

L'équation (2-3) : $p(e \mid f) = \frac{p(e) \cdot p(f \mid e)}{p(f)}$ est un cas particulier de l'équation (2-16), où K=2, $\lambda_1 = \lambda_2 = 1$, $h_1(e,a,f) = logp(f|e)$, $h_2(e,a,f) = logp(e)$.

Au décodage, la meilleure traduction $\hat{e}$ est une solution de :

$$\hat{e} = \arg\max_e (\exp \sum_{k=1}^{K} \lambda_k h_k(e,a,f)) = \arg\max_e \sum_{k=1}^{K} \lambda_k h_k(e,a,f) \quad \textbf{(2-17)}$$

La plupart des systèmes de traduction utilisent le modèle log-linéaire qui intègre des modèles génératifs comme les fonctions de traits. Plusieurs fonctions de traits peuvent être utilisées dans ces systèmes, tels que :

- la probabilité du modèle de langue *logp(e)* ;
- le nombre de mots de chaque phrase ;
- le nombre de groupes de mots de chaque phrase ;
- la probabilité de traduction en groupes de mots dans les deux sens *logp(f|e) ; logp(e|f)*
- le modèle de distorsion $d_g()$;
- etc.

N'importe quel modèle ou une combinaison de modèles peut être une fonction de traits pour augmenter la qualité de la traduction.

2.3. Estimation des paramètres

Une fois que nous avons défini le modèle mathématique pour calculer $p(e|f)$, nous avons besoin d'attribuer des valeurs à ses paramètres, les valeurs qui sont utilisées pour chercher la meilleure traduction d'une phrase source. Nous appelons cette étape l'estimation des paramètres ou l'étape d'apprentissage. Le but de cette étape est d'estimer les valeurs des paramètres pour qu'ils soient les plus proches possibles de la valeur réelle.

Dans la traduction, nous utilisons un algorithme d'apprentissage sur un corpus parallèle pour apprendre les valeurs des paramètres. De tels corpus sont obtenus en alignant chaque segment d'un corpus source avec sa traduction dans le corpus cible. L'alignement peut se faire à différents niveaux comme la phrase, le paragraphe ou encore le document. Nous nous concentrons notamment sur ce problème d'alignement dans la partie II.

2.3.1. Estimation des paramètres pour les modèles génératifs

2.3.1.1 Obtenir les alignements en mots et estimer des paramètres par EM

Supposons que nous voulions estimer les paramètres pour les modèles de traduction génératifs à base de mots. Si le corpus parallèle d'apprentissage nous offrait les alignements en mots pour chaque paire de phrases, il serait alors facile d'observer la probabilité de traduction lexicale, la fertilité, la substitution, et la distorsion pour chaque mot en comptant le nombre de fois que ce phénomène a lieu dans le corpus d'apprentissage. Malheureusement, le corpus d'apprentissage est au mieux constitué de phrases source – cible qui sont la traduction l'une de l'autre, sans inclure des informations d'alignement au niveau des mots.

Une solution à ce problème consiste à générer automatiquement les alignements et à les utiliser ensuite pour estimer les paramètres. Une méthode qui est couramment utilisée en TA probabiliste est l'algorithme EM (*Expectation-Maximization*) [Dempster 1977].

Cet algorithme est adapté à l'estimation des paramètres des modèles IBM par exemple. En bref, cela fonctionne comme suit :

1. étant donné une valeur initiale pour des paramètres ;
2. nous calculons la probabilité d'un alignement particulier et comptons le nombre de fois qu'un phénomène a lieu ;
3. nous réestimons alors les paramètres dans l'étape (1) en utilisant les valeurs calculées dans l'étape (2) ;
4. nous répétons les étapes ci-dessus jusqu'à convergence des valeurs.

Les itérations de ce processus mènent aux paramètres qui donnent une probabilité de plus en plus approchée à l'ensemble de paires de phrases que nous observons. Cet algorithme conduit à un maximum local de la probabilité des paires observées en fonction des paramètres du modèle. Donc nous l'appelons l'estimation du maximum de vraisemblance (MLE : *maximum likelihood estimation*).

Dans la Figure 2-7, nous présentons un exemple du processus appliquant l'algorithme EM pour estimer les paramètres $t(f|e)$ du modèle IBM-1 [Callison-Burch 2007]. Le détail de l'algorithme EM pour les autres modèles d'IBM peut être trouvé dans [Brown 1993].

2.3.1.2 Obtenir les alignements en groupes de mots et estimer les probabilités de traduction

Pour estimer les probabilités de traduction en groupe de mots, tout d'abord les alignements en groupe de mots doivent être définis. Plusieurs méthodes heuristiques pour créer l'alignement en groupe de mots sont proposées. En général, elles consistent en deux étapes ; la première étape est de symétriser deux alignements en mots de la paire de phrases dans les deux sens de traduction : $f - e$ et $e - f$; et la deuxième étape est d'extraire les groupes de mots correspondants.

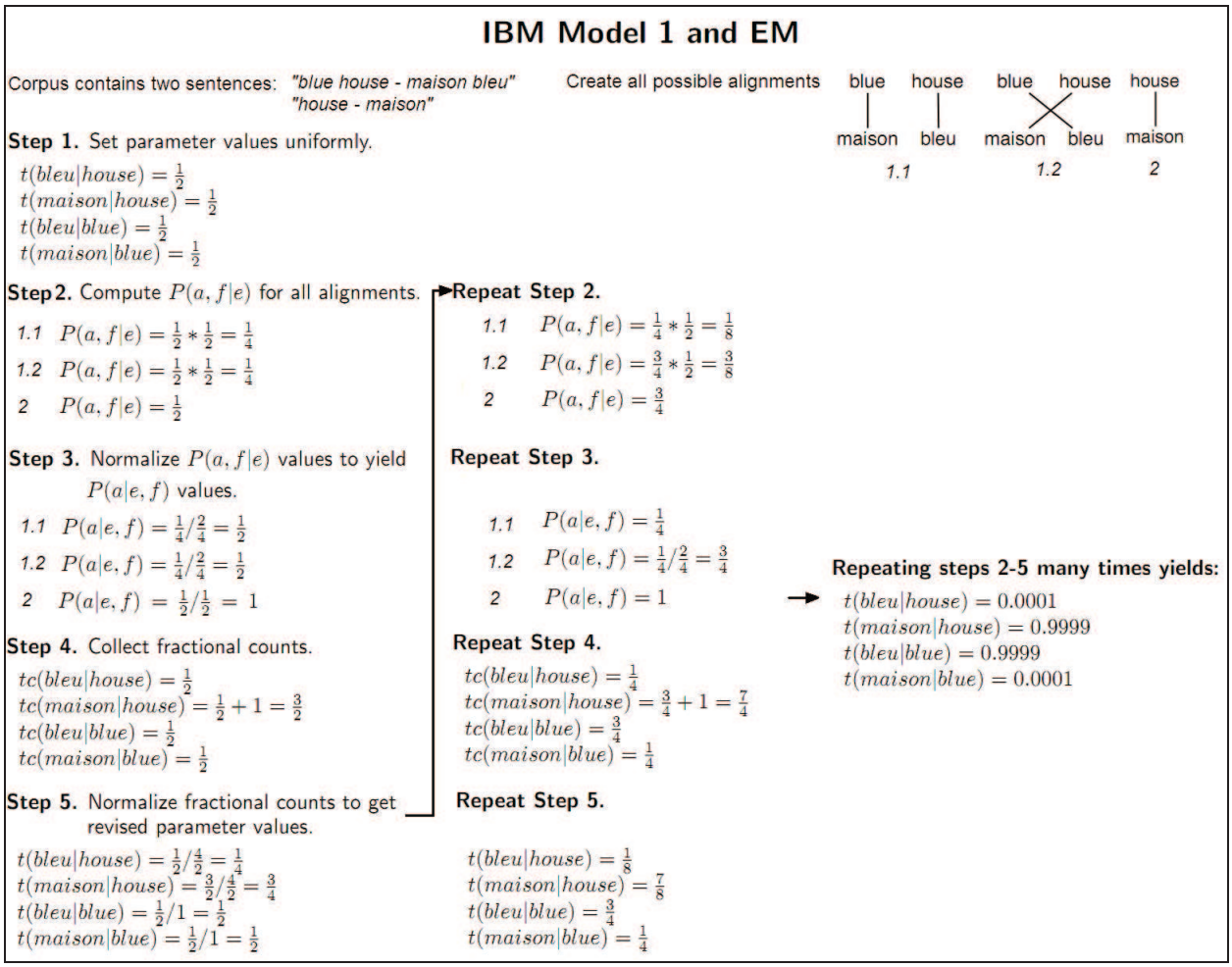


Figure 2-7 : Exemple du processus de l'algorithme EM pour estimer les paramètres t(f|e) du modèle IBM-1 (src [Callison-Burch 2007])

Symétrisation de deux alignements

Dans la première étape, un alignement symétrisé en mots peut être obtenu en combinant deux alignements en mots. Il y a plusieurs manières de combiner ces deux alignements en formant une union, une intersection ou en utilisant des méthodes heuristiques (Figure 2-8).

Appliquons une méthode pour générer automatiquement les alignements en mots pour une paire de phrases *f, e*. Dans le sens de *f* vers *e* (un mot de la phrase *f* peut être aligné avec plusieurs mots de la phrase *e*) nous obtenons l'alignement a_1^J et dans le sens inverse de *e* vers *f* (un mot de la phrase *e* peut être aligné avec plusieurs mots de la phrase *f*) nous obtenons l'alignement b_1^I. Notons l'ensemble des correspondances de a_1^J par $A_1 = \{(a_j,j) \mid a_j>0\}$ et celui de b_1^I par $A_2 = \{(i,b_i) \mid b_i>0\}$. Parce que ces alignements en mots sont asymétriques, les correspondances dans A_1 sont différentes de celles dans A_2. En formant une union ou une intersection entre A_1 et A_2, nous obtenons un alignement symétrisé.

Évidemment, l'intersection entre deux alignements produit un nouvel alignement avec une précision plus grande et un rappel plus faible que pour chaque alignement pris séparément. L'union des alignements au contraire donne un rappel plus grand et une précision plus faible.

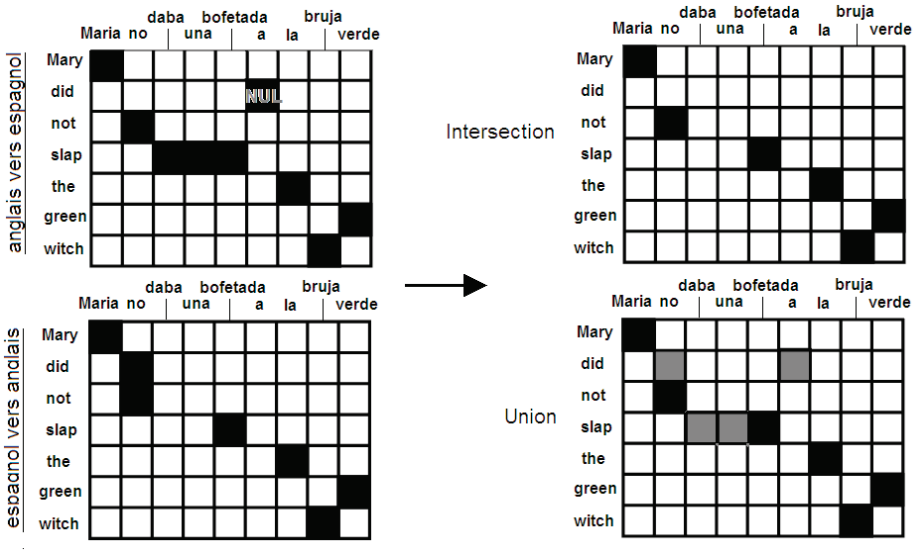


Figure 2-8 : L'intersection et l'union entre deux alignements en mots

[Och 2004] a proposé une méthode heuristique pour créer l'alignement symétrisé. L'intersection A entre A_1 et A_2 est déterminée. Puis, l'alignement A est élargi itérativement en ajoutant la correspondance (i,j) si :

- la correspondance *(i,j)* apparaît seulement dans *A1* ou *A2* et *fi* et *ej* ne possèdent pas de correspondance dans *A* ;
- ou si les deux conditions suivantes sont satisfaites :
  - la correspondance *(i,j)* possède un voisin horizontal *(i-1, j), (i+1, j),* ou un voisin vertical *(i, j-1), (i, j+1)* qui est déjà dans *A* ;
  - l'ensemble $A \cup \{i,j\}$ ne contient pas de correspondance avec les voisins horizontaux et verticaux.

[Koehn 2003b] a présenté d'autres décisions heuristiques pour créer l'alignement symétrisé. Les décisions sont basées aussi sur la proposition de Och et Ney décrite ci-dessus avec quelques modifications. Ils commencent par l'intersection de deux alignements en mots. Ils ajoutent seulement des nouvelles correspondances (i,j) qui existent dans l'union de deux alignements en mots et relient au moins un mot non aligné précédemment. Pour le faire, premièrement, ils étendent seulement aux points de correspondances adjacentes. Ils vérifient les points à partir du coin supérieur droit de la matrice d'alignement, et commencent avec le premier mot cible, puis le deuxième mot, et ainsi de suite. Cela se fait par itération jusqu'à ce qu'il n'y ait plus de point d'alignement qui peut être ajouté. Dans une dernière étape, ils ajoutent des points d'alignements non adjacents, avec les mêmes exigences. Les auteurs ont proposé plusieurs modifications de la base heuristique :

- *grow* (croissance) : uniquement ajouter des points de voisins dans le bloc
- *grow-diag* : sans utiliser la dernière étape
- *srctotgt* : seulement envisager les correspondances en mots dans le sens de la langue source à la langue cible
- *tgttosrc* : seulement envisager les correspondances en mots dans le sens de la langue cible à la langue source

Extraire l'ensemble des correspondances en groupes de mots

Après avoir réalisé l'alignement symétrisé A, l'étape suivante consiste à extraire l'ensemble des correspondances en groupe de mots à partir de A. Ces correspondances doivent être constituées avec A : les mots dans une correspondance sont alignés les uns aux autres, et non pas à des mots à l'extérieur. La correspondance avec le mot NULL n'existe pas dans l'alignement en groupe de mots. La Figure 2-9 présente un exemple de correspondance consistante et non-consistante et la

Figure 2-10 présente les correspondances en groupe de mots extraites à partir d'un alignement symétrisé.

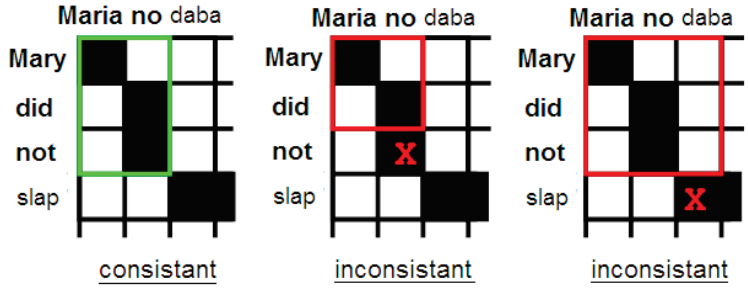


Figure 2-9 : Les correspondances consistantes et non-consistantes

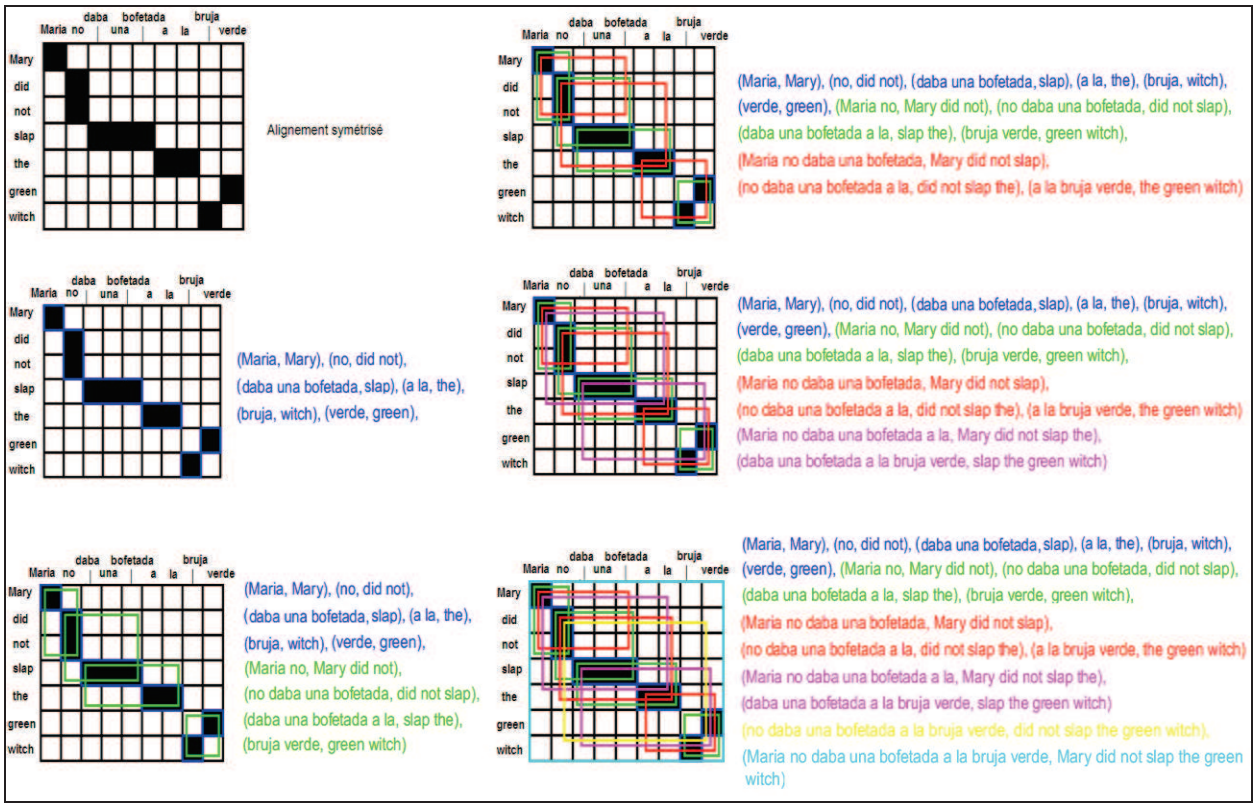


Figure 2-10 : Les correspondances en groupe de mots extraites à partir d'un alignement symétrisé (src [Knight 2003])

Le modèle de traduction en groupe de mots est décomposé en un modèle de traduction en groupe de mots et un modèle de distorsion (équation (2-15)). Le modèle de distorsion peut être défini très simplement comme $d_g(a_z - b_{z-1}) = \alpha^{|a_z - b_{z-1} - 1|}$ [Koehn 2003b].

Les paramètres $t_g(\tilde{f}_z \mid \tilde{e}_z)$ sont estimés en comptant :

$$t_g(\tilde{f}_z \mid \tilde{e}_z) = \frac{count(\tilde{f}_z, \tilde{e}_z)}{\sum_{\tilde{f}_z} count(\tilde{f}_z, \tilde{e}_z)} \qquad \textbf{(2-18)}$$

Les paires de correspondances en groupe de mots $(\tilde{f}_z, \tilde{e}_z)$ et leurs probabilités de traduction $t_g(\tilde{f}_z \mid \tilde{e}_z)$ sont conservées dans une table de traduction (*phrase table*) qui va être utilisée dans l'étape de décodage. En plus, d'autres valeurs sont estimées et conservées aussi :

- les probabilités de traduction : $t_g(\tilde{f}_z \mid \tilde{e}_z)$, $t_g(\tilde{e}_z \mid \tilde{f}_z)$;
- les pondérations lexicales : $p_w(\tilde{f}_z \mid \tilde{e}_z)$, $p_w(\tilde{e}_z \mid \tilde{f}_z)$:

$p_w(\widetilde{f}_z \mid \widetilde{e}_z) = \max_a p_w(\widetilde{f}_z \mid \widetilde{e}_z, a) = \max_a \left(\prod_i \frac{1}{|\{j \mid (i,j) \in a\}|} \sum_{\forall (i,j) \in a} w(f_i \mid e_j) \right)$ où *a* est un alignement en mots entre deux groupes de mots $\widetilde{f}_z$, $\widetilde{e}_z$; i, j sont les positions des mots correspondants dans l'alignement ; w($\widetilde{f}_z \mid \widetilde{e}_z$) est la probabilité de traduction lexicale [Koehn 2003b]

[Och 1999] a proposé des patrons d'alignement (*alignment template*) qui permettent d'aligner une classe de mots sources avec une classe de mots cibles (voir pour plus de détail [Och 1999]). Pour le modèle de probabilité jointe de [Marcu 2002], les paramètres peuvent être estimés en utilisant l'algorithme EM aussi.

2.3.2. Estimation des paramètres pour les modèles discriminants

Dans les modèles discriminants (sous la forme des modèles log-linéaires), les paramètres des fonctions de traits $h_k(e,a,f)$, $k=1..K$ sont estimés comme les paramètres des modèles génératifs. Le problème qui reste ici est l'estimation des poids du trait $\lambda_1^K = \{\lambda_k \mid k = 1..K\}$. Le corpus parallèle pour estimer les paramètres des poids doit être différent de celui pour estimer les paramètres des fonctions de traits, nous l'appelons le corpus de développement.

Dans l'algorithme d'entraînement à entropie maximale de [Och 2002], les auteurs utilisent l'algorithme GIS (*generalized iterative scaling*) de Darroch et Ratcliff avec le critère standard MMI (*maximum mutual information*).

$$\hat{\lambda}_1^K = \arg\max_{\lambda_1^K} \left\{ \sum_{s=1}^{S} \log p_{\lambda_1^K}(e_s \mid f_s) \right\} \quad \textbf{(2-19)}$$

Si nous avons R_s références ($e_{s,1}$, $e_{s,2}$, ... $e_{s,Rs}$) pour chaque f_s, le critère est transformé comme suit :

$$\hat{\lambda}_1^K = \arg\max_{\lambda_1^K} \left\{ \sum_{s=1}^{S} \frac{1}{R_s} \sum_{r=1}^{Rs} \log p_{\lambda_1^K}(e_{s,r} \mid f_s) \right\} \quad \textbf{(2-20)}$$

Une autre algorithme d'entraînement appliqué largement aujourd'hui est l'algorithme de minimisation du taux d'erreur (MERT : *Minimum Error Rate Training*) [Och 2003a] qui suppose que le meilleur modèle est celui qui produit la plus petite erreur globale par rapport à une fonction d'erreur donnée.

Étant donné un corpus parallèle de développement $C=(f_s, r_s)_{i=1..S}$. Une fonction $E(e_s,r_s)$ mesure l'erreur de la traduction e_s par rapport à la traduction de référence r_s (voir les métriques d'évaluation automatique dans la section 2.6). Le critère d'optimisation des paramètres est défini comme suit :

$$\hat{\lambda}_1^K = \arg\min_{\lambda_1^K} \left\{ \sum_{i=1}^{S} E(\hat{e}(f_s, \lambda_1^K), r_s) \right\} \quad \textbf{(2-21)}$$

où $\hat{e}(f_s, \lambda_1^K)$ est la traduction de f_s avec le système de traduction dont les paramètres sont λ_1^K :

$$\hat{e}(f_s, \lambda_1^K) = \arg\max_e \left\{ \sum_{k=1}^{K} \lambda_k h_k(e \mid f_s) \right\} \quad \textbf{(2-22)}$$

L'algorithme génère itérativement des valeurs λ_1^K aléatoires, puis améliore chaque paramètre en gardant les autres paramètres. Les valeurs λ_1^K optimisées qui produisent la réduction d'erreur la

plus grande sont utilisées comme entrée à l'itération suivante. Le pseudo code ci-dessous est présenté dans [Lopez 2007].

Algorithm 1 Minimum Error Rate Training

Input initial estimate $\lambda_{1,0}^K$ ▷ From MLE or prior knowledge
Input training corpus C
$\lambda_1^K = \lambda_{1,0}^K$
$E_{best} = \sum_{(\mathbf{e},\mathbf{f})\in C} E(\text{argmax}_{\hat{\mathbf{e}}}\, P_{\lambda_1^K}(\hat{\mathbf{e}}|\mathbf{f}), \mathbf{e})$
repeat
  Generate M random estimates $\lambda_{1,1}^K, \ldots, \lambda_{1,M}^K$ ▷ To avoid poor local maximum
  for $m = \{0, 1, \ldots, M\}$ **do**
    for $k = \{1, 2, \ldots, K\}$ **do**
      $\lambda'_{k,m} = \text{LINE-MINIMIZE}(k, \lambda_{1,m}^K, C)$
      $E_{k,m} = \sum_{(\mathbf{e},\mathbf{f})\in C} E(\text{argmax}_{\hat{\mathbf{e}}}\, P_{\lambda_{1,m}^{k-1}\lambda'_{k,m}\lambda_{k+1,m}^K}(\hat{\mathbf{e}}|\mathbf{f}), \mathbf{e})$
      if $E_{k,m} < E_{best}$ **then**
        $\lambda_1^K = \lambda_{1,m}^{k-1}\lambda'_{k,m}\lambda_{k+1,m}^K$
        $E_{best} = E_{k,m}$
      end if
    end for
  end for
  $\lambda_{1,0}^K = \lambda_1^K$
until no change in λ_1^K
return $\lambda_{1,0}^K$

Figure 2-11 : Le pseudo code de l'algorithme de minimisation du taux d'erreur (src [Lopez 2007])

La fonction LINE-MINIMIZE(k, $\lambda_{1,m}^K$, C) optimise un seul paramètre λ_k en gardant les autres paramètres constants. Dans le cas où le système de traduction peut sortir N meilleures traductions pour une phrase f_s : $\{e_{s1}, e_{s2}, \ldots e_{sN}\}$, il calcule N probabilités $P^*(e_{sn}|f_s)$

$$P^*(e_{sn} \mid f_s) = \lambda_k h_k(e_{sn}, f_s) + (\sum_{k'=1}^{k-1} + \sum_{k'=k+1}^{K})\lambda_{k'} h_{k'}(e_{sn}, f_s) \qquad \textbf{(2-23)}$$

Parce que les autres paramètres soient gardés constants, le P* est une fonction linéaire de λ_k. Dessinons toutes les fonctions $P^*(e_{sn}|f_s)$ dans le graphique. Choisissons P* maximum dans chaque intervalle et son $\hat{e}_{sn}$ correspondant. Calculons les erreurs $E(\hat{e}_{sn}, r_s)$ pour chaque intervalle. Nous faisons de la même manière pour toutes les paires de phrases (f_s, r_s) dans le corpus d'apprentissage C et calculons l'erreur totale pour $C = \sum_{i=1}^{S} E(\hat{e}_{sn}, r_s)$. Nous choisissons tout simplement l'intervalle qui minimise l'erreur totale de C. La valeur λ_k est le point central de l'intervalle. Pour plus de détails, se référer à [Och 2003a].

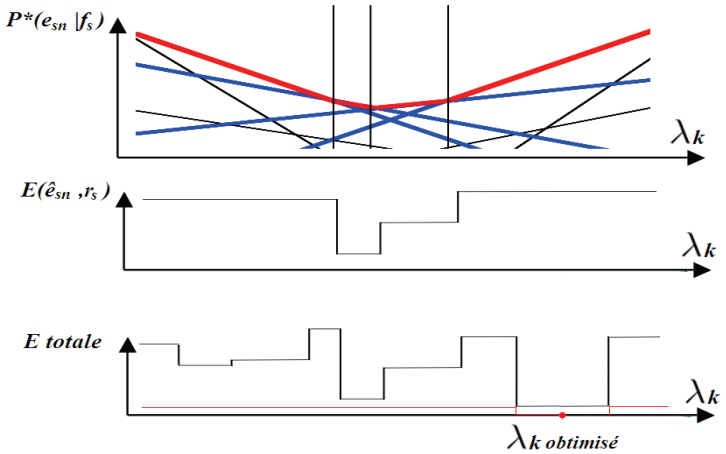


Figure 2-12 : Visualisation de la fonction LINE-MINIMIZE dans l'algorithme de MERT (src [Och 2005])

[Venugopal 2005] a présenté une comparaison entre deux critères le MMI et le critère de minimisation du taux d'erreur, et ce dernier a été plus performant que le critère MMI dans l'algorithme d'entraînement. Le processus d'itération de la méthode basée sur le critère MMI est plus erratique que celui de la méthode basée sur la minimisation du taux d'erreur.

En résumé, le flux de données, les modèles et les procédés couramment impliqués dans le déploiement d'un système de traduction probabiliste sont présentés dans la Figure 2-13 :

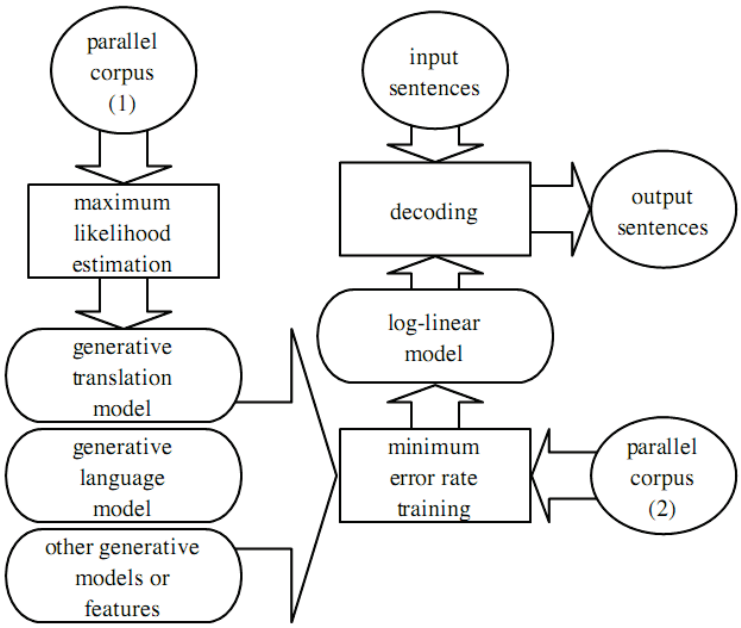


Figure 2-13 : Le flux de données, les modèles et les procédés couramment impliqués dans le déploiement d'un système de traduction probabiliste
(src [Lopez 2007])

2.4. Décodage

Le troisième problème que nous allons présenter est le décodage. Après avoir estimé les modèles pour calculer $p(e|f)$, nous pouvons traduire une phrase source f en cherchant la phrase cible e qui est conforme au modèle de transformation donné et maximise la probabilité $p(e|f)$: $\hat{e}(f) = \arg\max_e \Pr(e \mid f)$.

Le processus de traduction d'une phrase espagnole vers une phrase anglaise peut être illustré dans la Figure 2-14. Commençant avec une hypothèse vide, nous sélectionnons un ou plusieurs mots sources. Les mots cibles qui correspondent à ces mots sources (identifiés par les options de traduction dans la table de traduction) sont ajoutés dans l'hypothèse. Les mots sources sont choisis dans n'importe quel ordre, tandis que les mots cibles sont ajoutés de gauche à droite. Le décodeur va continuer jusqu'à ce que tous les mots sources soient couverts. Nous obtenons un graphe d'hypothèses. Trouver la meilleure traduction de la phrase source consiste à trouver le meilleur chemin dans le graphe avec coût de génération minimal. Le coût de génération est égal à l'opposé du logarithme de la probabilité de génération de cette hypothèse ($= -log[p_{LM}(e).p_{TM}(f|e)]$).

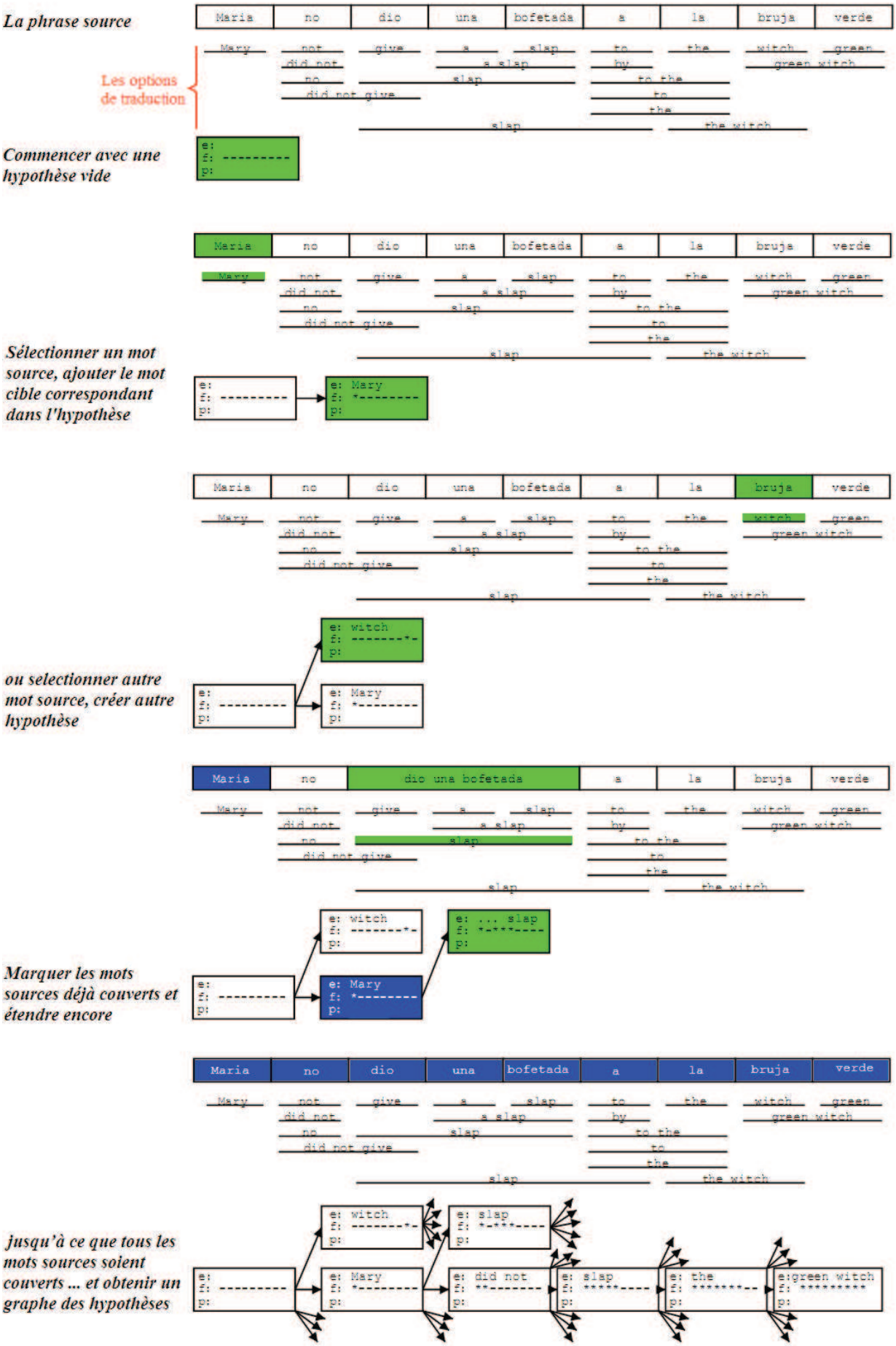


Figure 2-14 : Graphe généré lors de la traduction d'une phrase espagnole vers une phrase anglaise (src [Callison-Burch 2005])

Même si *f* est fixé et si les modèles exigent beaucoup de contraintes, il y a encore un grand nombre d'hypothèses à envisager afin de maximiser la fonction. Beaucoup d'approches de décodage en TA probabiliste suivent l'algorithme de décodage par « piles » (*stack decoder*) utilisé largement en reconnaissance de parole et adapté pour la TA probabiliste.

Par exemple, [Wang 1997] et [Och 2001] ont appliqué l'algorithme de recherche « A* ». Une pile est utilisée pour garder les hypothèses partielles. Chaque hypothèse partielle se voit associer un coût défini par le somme d'un score préfixe *Q* et d'un score heuristique *H*.

Le score préfixe *Q* est le coût de génération de cette hypothèse partielle ($-log[p_{LM}(e_h).p_{TM}(f_h|e_h)]$) avec f_h l'hypothèse partielle, et e_h l'ensemble des mots source couverts.

Le score heuristique *H* estime le coût futur pour finir la traduction de l'hypothèse totale à partir de l'hypothèse partielle. Il prend en compte les mots source non couverts. Plusieurs méthodes pour déterminer ce score *H* peuvent être trouvés dans les travaux de [Wang 1997], [Och 2001], etc. En général, le score *H* est calculé par le produit des coûts de génération pour tous les mots sources non couverts.

La pile est initialisée avec une hypothèse vide. Le décodeur commence par l'hypothèse ayant le coût défini minimal. Donc la pile est triée par coût croissant. L'hypothèse en tête de la pile (qui a le coût minimal) est sortie et étendue en couvrant une position source de plus (pour toutes les positions possibles). Après les extensions, les nouvelles hypothèses partielles générées sont incorporées à la pile avec les coûts accompagnés, et la pile est retriée. Le décodeur continue l'extension jusqu'à ce que tous les mots sources soient couverts. Le décodeur sort l'hypothèse en tête comme la meilleure traduction. Le coût défini de cette hypothèse devient le coût de génération *Q* + 0 (le score heuristique *H* devient zéro car il ne reste aucun mot source non couvert). Donc cette hypothèse satisfait le coût de génération minimal ou la probabilité de génération *p(e).p(f|e)* maximal. L'algorithme est terminé.

L'utilisation de l'algorithme de recherche « A* » permet de trouver une solution exacte de l'équation $\hat{e}(f) = \arg\max_e P(e \mid f)$ [Och 2001], mais il devient très coûteux en temps pour des phrases longues. La recherche en faisceau (*beam search*) est adaptée pour optimiser le processus de recherche en TA en réduisant le besoin de mémoire [Tillmann 2003], [Koehn 2003a]. Il n'étend qu'un nombre limité d'hypothèses les plus prometteuses, autrement dit, il élague des hypothèses non pertinentes.

Plusieurs piles sont utilisées, chacune contient les hypothèses partielles de mêmes mots sources. Les hypothèses dans chaque pile sont triées par coût croissant et l'hypothèse ayant le coût le plus bas sur toutes les piles (ou *N* meilleures hypothèses) est sélectionnée pour étendre le graphe. Les nouvelles hypothèses générées sont placées dans la pile appropriée. A chaque fois, la pile élague les hypothèses qui ne peuvent pas faire partie du chemin de la meilleure traduction. Deux méthodes d'élagage sont couramment utilisées :

- l'élagage de seuil : l'hypothèse dont la probabilité est inférieure à *t* fois la probabilité de la meilleure hypothèse dans la même pile est élaguée ;
- l'élagage d'histogrammes : seuls les *n* meilleures hypothèses sont conservées dans une pile.

Nous notons *t* ou *n* la taille du faisceau. La traduction finale est la meilleure hypothèse dans la dernière pile (la pile couvrant tous les mots sources).

L'algorithme de décodage d'une phrase source qui contient *nf* mots, proposé dans [Koehn 2003a] est décrit dans la Figure 2-15 :

```
initialize hypothesisStack[0 .. nf];
create initial hypothesis hyp_init;
add to stack hypothesisStack[0];
for i=0 to nf-1:
  for each hyp in hypothesisStack[i]:
    for each new_hyp that can be derived from hyp:
      nf[new_hyp] = number of foreign words covered by new_hyp;
      add new_hyp to hypothesisStack[nf[new_hyp]];
      prune hypothesisStack[nf[new_hyp]];
find best hypothesis best_hyp in hypothesisStack[nf];
output best path that leads to best_hyp;
```

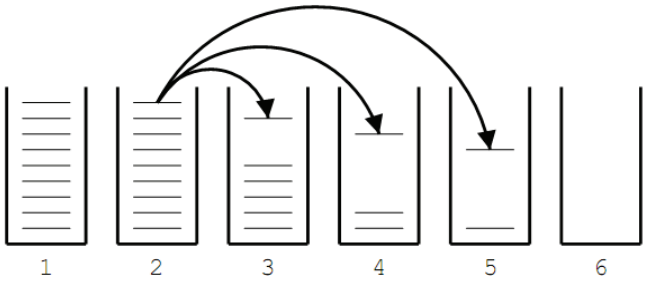


Figure 2-15 : L'algorithme de décodage et les piles des hypothèses pour la recherche en faisceaux (src [Koehn 2003a])

2.5. Approche de traduction probabiliste hiérarchique

L'approche de traduction probabiliste par groupe de mots étend l'unité fondamentale de traduction aux groupes de mots, ce qui permet de capturer des contraintes de réarrangement local dans le groupe de mots ; ceci permet par exemple de mieux traiter les expressions idiomatiques. Cependant, des réarrangements plus *longue distance* des séquences de mots peuvent avoir lieu, notamment pour des paires de langues avec ordre de mots très différent tels que le chinois et l'anglais.

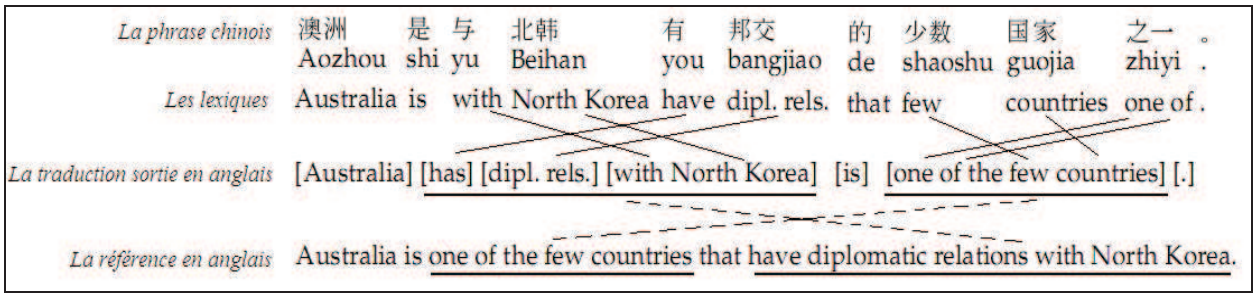


Figure 2-16 : Exemple d'une phrase chinoise et sa traduction en anglais (src [Chiang 2007])

La Figure 2-16 présente une phrase en chinois et sa traduction en anglais issue d'un système de traduction par groupe de mots. Les réarrangements locaux ont été résolus avec succès par le système de traduction (la séquence source de mots correspondant à « *with North Korea have dipl. rels.* » a été réordonnée par « *have diplomatic relations with North Korea* », et la séquence de mots source « *few countries one of* » a été réordonnée par « *one of (the) few countries* »). Cependant, le réarrangement global entre ces deux séquences de mots n'a pas été effectué.

Ce type de réarrangement peut être modélisé dans le système de traduction à base de groupe de mots, mais il faut utiliser des paires de phrases très longues. Ceci est difficilement envisageable lorsqu'il manque des données d'apprentissage (*data sparseness*). Pour résoudre cette limitation, une nouvelle « approche de traduction probabiliste par l'expression hiérarchique de mots » (*hierarchical phrase-based translation*) a été proposée [Chiang 2007]. La motivation pour cette

approche est d'utiliser les expressions hiérarchiques de mots pour modéliser les réarrangements globaux. Les expressions hiérarchiques de mots (les règles) sont formalisées par une grammaire de type CFG synchrone (*synchronous context-free grammar*). Chaque élément dans la grammaire CFG synchrone est une règle de réécriture : $X \rightarrow \langle \gamma, \alpha, \sim \rangle$ ou X est un symbole non terminal, γ et α sont les séquences de mots qui contiennent des symboles non terminaux X, et « ~ » est l'alignement entre les symboles non terminaux dans γ et α. Pour illustrer l'exemple dans la Figure 2-16, nous pouvons utiliser les trois règles de réécriture suivantes :

($r1$) $X \rightarrow \langle yu\, X_1\, you\, X_2,\ have\, X_2\, with\, X_1 \rangle$

($r2$) $X \rightarrow \langle X_1\, de\, X_2,\ the\, X_2\, that\, X_1 \rangle$; ($r3$) $X \rightarrow \langle X_1\, zhiyi,\ one\, of\, X_1 \rangle$

La grammaire utilise aussi les règles conventionnelles qui ne contiennent pas des symboles non terminaux dans le côté droit :

($r4$) $X \rightarrow \langle Aozhou, Australia \rangle$

($r5$) $X \rightarrow \langle Beihan, North\ Korea \rangle$

($r6$) $X \rightarrow \langle shi, is \rangle$

($r7$) $X \rightarrow \langle bangjiao,\ diplomatic\ relations \rangle$

($r8$) $X \rightarrow \langle shaoshu\ guojia,\ few\ countries \rangle$

et deux règles simples pour commencer une dérivation et assembler les séquences, nommées « *glue rules* » :

($r9$) $S \rightarrow \langle S_1 X_2, S_1 X_2 \rangle$ avec S_i, symbole de début de phrase,

($r10$) $S \rightarrow \langle X_1, X_1 \rangle$ pour enchaîner les expressions hiérarchiques ou les règles.

Avec la grammaire définie par les règle *r1…r10*, nous avons la dérivation suivante pour transformer la phrase source chinoise vers la phrase cible anglaise (pour faciliter la lecture, commencer à partir du bas de la Figure 2-17).

Les règles de la grammaire CFG synchrone peuvent être apprises automatiquement à partir d'une base de données d'apprentissage parallèle et sans annotations syntaxiques. Si la grammaire ne contient que les « *glue rules* » et les règles conventionnelles, le système devient le système à base de groupe de mots sans réarrangement global.

2.5.1. Modélisation

Pour une paire de phrases source et cible *(f,e)*, soient D une des dérivations possibles de cette paire avec la grammaire donnée. Selon un modèle log-linéaire, nous pouvons modéliser la probabilité d'une dérivation D par $P(D) \propto \prod_i \phi_i(D)^{\lambda_i}$ ou ϕ_i sont des fonctions de traits et λ_i correspond aux poids associés à ces traits. Une des fonctions est le modèle de langage $P_{LM}(e)$, et d'autres sont les fonctions de règles de cette dérivation : $\phi_i(D) = \prod_{(X \rightarrow <\gamma,\alpha>) \in D} \phi_i(X \rightarrow \langle \gamma, \alpha \rangle)$.

La probabilité d'une dérivation peut être écrite sous la forme :

$$P(D) \propto P_{LM}(e)^{\lambda_{LM}} \times \prod_{i \neq LM} \prod_{(X \rightarrow <\gamma,\alpha>) \in D} \phi_i(X \rightarrow \langle \gamma, \alpha \rangle)^{\lambda_i} \quad \textbf{(2-24)}$$

En définissant la fonction w qui assigne les poids pour les fonctions : $w(D) = \prod_{(X \to \langle \gamma, \alpha \rangle) \in D} w(X \to \langle \gamma, \alpha \rangle)$ et en définissant $w(X \to \langle \gamma, \alpha \rangle) = \prod_{i \neq LM} \phi_i(X \to \langle \gamma, \alpha \rangle)^{\lambda_i}$, la probabilité d'une dérivation devient :

$$P(D) \propto P_{LM}(e)^{\lambda_{LM}} \times w(D) \quad \textbf{(2-25)}$$

L'ensemble de fonctions de règles ϕ_i comprendra alors, par exemple : $P(\gamma|\alpha)$ et $P(\alpha|\gamma)$; les poids du lexique $P_w(\gamma|\alpha)$ et $P_w(\alpha|\gamma)$; le pénalité du mot et les pénalités pour chaque groupe de règles.

$$\begin{aligned}
&\langle S_1, S_1 \rangle \\
&\xrightarrow{(r9)} \langle S_2 X_3, S_2 X_3 \rangle \\
&\xrightarrow{(r9)} \langle S_4 X_5 X_3, S_4 X_5 X_3 \rangle \\
&\xrightarrow{(r10)} \langle X_6 X_5 X_3, X_6 X_5 X_3 \rangle \\
&\xrightarrow{(r4)} \langle Aozhou\ X_5 X_3, Australia\ X_5 X_3 \rangle \\
&\xrightarrow{(r6)} \langle Aozhou\ shi\ X_3, Australia\ is\ X_3 \rangle \\
&\xrightarrow{(r3)} \langle Aozhou\ shi\ X_7\ zhiyi, Australia\ is\ one\ of\ X_7 \rangle \\
&\xrightarrow{(r2)} \langle Aozhou\ shi\ X_8\ de\ X_9\ zhiyi, Australia\ is\ one\ of\ the\ X_9\ that\ X_8 \rangle \\
&\xrightarrow{(r1)} \left\langle \begin{array}{l} Aozhou\ shi\ yu\ X_1\ you\ X_2\ de\ X_9\ zhiyi, \\ Australia\ is\ one\ of\ the\ X_9\ that\ have\ X_2\ with\ X_1 \end{array} \right\rangle \\
&\xrightarrow{(r5)} \left\langle \begin{array}{l} Aozhou\ shi\ yu\ Beihan\ you\ X_2\ de\ X_9\ zhiyi, \\ Australia\ is\ one\ of\ the\ X_9\ that\ have\ X_2\ with\ North\ Korea \end{array} \right\rangle \\
&\xrightarrow{(r7)} \left\langle \begin{array}{l} Aozhou\ shi\ yu\ Beihan\ you\ bangjiao\ de\ X_9\ zhiyi, \\ Australia\ is\ one\ of\ the\ X_9\ that\ have\ diplomatic\ relations\ with\ North\ Korea \end{array} \right\rangle \\
&\xrightarrow{(r8)} \left\langle \begin{array}{l} Aozhou\ shi\ yu\ Beihan\ you\ bangjiao\ de\ shaoshu\ guojia\ zhiyi, \\ Australia\ is\ one\ of\ the\ few\ countries\ that\ have\ diplomatic\ relations\ with\ North\ Korea \end{array} \right\rangle
\end{aligned}$$

Figure 2-17 : Une dérivation de la grammaire définie pour transformer la phrase source chinoise vers la phrase cible anglaise (src [Chiang 2007])

2.5.2. Estimation des paramètres

Tout d'abord, les règles sont extraites à partir d'un corpus d'apprentissage parallèle aligné en mots. Le processus d'alignement en mots est réalisé dans les deux sens, et une union entre les deux ensembles d'alignements en mots est formée (revoir la section 2.3.1).

Puis, pour chaque paire de phrases déjà alignée en mot (*f,e*), les correspondances consistantes $\langle f_i^j, e_{i'}^{j'} \rangle$ pour chaque groupe de mots sont identifiées (voir la Figure 2-9 pour la notion de correspondances consistantes) .

Ensuite, on recherche les correspondances consistantes qui contiennent une autre correspondance consistante et nous remplaçons alors les sous-correspondances par des symboles non terminaux. Enfin, l'ensemble des règles pour chaque paire de phrases (*f,e*) est l'ensemble minimal qui satisfait :

- Si $\langle f_i^j, e_{i'}^{j'} \rangle$ est une correspondance consistante, $X \to \langle f_i^j, e_{i'}^{j'} \rangle$ est une règle de (*f,e*).

- Si $X \rightarrow \langle \gamma, \alpha \rangle$ est une règle de (*f*,*e*), et si $\langle f_i^j, e_{i'}^{j'} \rangle$ est une correspondance consistante qui satisfait $\gamma = \gamma_1 f_i^j \gamma_2$ et $\alpha = \alpha_1 e_{i'}^{j'} \alpha_2$, alors $X \rightarrow \langle \gamma_1 X_k \gamma_2, \alpha_1 X_k \alpha_2 \rangle$ est une règle de (*f*,*e*).

Nous présentons un exemple d'extraction d'une règle dans la Figure 2-18.

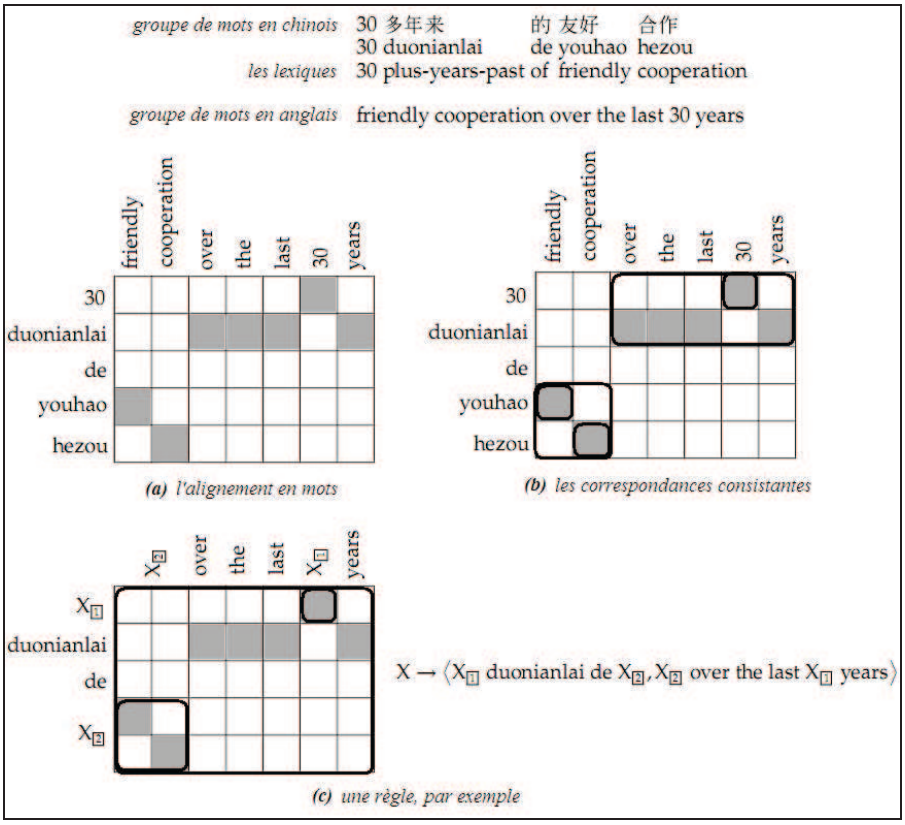


Figure 2-18 : Exemple d'extraction d'une règle de réécriture à partir d'alignement en mots (src [Chiang 2007])

Pour optimiser le processus d'extraction, certaines contraintes sont appliquées : les règles contiennent deux paires de symboles non terminaux au maximum (X_1, X_2) ; la longueur de correspondance consistante est limitée (par exemple ≤ 10) ; deux symboles non terminaux ne peuvent pas être adjacents côté source ; etc.

Après avoir extrait toutes les règles dans le corpus d'apprentissage, les paramètres des fonctions de traits du modèle sont estimés en comptant les événements existants et les poids du trait sont estimés en utilisant l'algorithme de minimisation du taux d'erreur MERT.

2.5.3. Décodage

Etant donné une phrase source *f*, le décodage cherche la meilleure dérivation (ou *N*- meilleures dérivations) qui génère $\langle f, e \rangle$.

$$\hat{e} = e\left(\underset{D s.t. f(D)=f}{\arg\max} P(D) \right) \quad \textbf{(2-26)}$$

Le processus de décodage comprend trois étapes. Etant donné un ensemble de règles, le décodage utilise l'analyseur CKY+ (*Cocke-Kasami-Younger*) pour faire l'analyse grammaticale des phrases entrées. Le graphe sorti représente toutes les dérivations possibles à partir de cette grammaire. Après, la recherche en faisceau est appliquée pour chercher la meilleure dérivation. Enfin, la phrase en langue source correspondant à la meilleure dérivation est l'hypothèse de

traduction de la phrase cible. Plus de détails sur le processus de décodage sont données dans [Chiang 2007] et [Huang 2007].

Le système à base d'expression hiérarchique de mots peut être hybridé avec un autre système comme le système à base de mots (dans le travail de [Hayashi 2010]), et le système à base de groupe de mots (dans le travail de [Dymetman 2010]). Nos travaux dans le cadre de cette thèse se focalisent sur l'extraction de données parallèles pour construire un système de traduction probabiliste. Nous utiliserons l'approche de traduction probabiliste à base de groupe de mots pour construire nos systèmes de traduction de référence.

2.6. Évaluation

Une fois qu'une traduction est réalisée, il faut en évaluer la qualité, mais ceci n'est pas un problème trivial en traduction automatique.

Le problème est, entre autres, que la langue naturelle est ambigüe et qu'il peut exister plusieurs traductions possibles d'une phrase source donnée. Ces traductions peuvent varier selon le choix des mots ou selon l'ordre des mots, même quand elles utilisent les mêmes mots. Qu'est-ce qu'une bonne traduction ? Comment pouvons-nous clairement distinguer une bonne traduction d'une mauvaise ? Il existe plusieurs approches pour évaluer la qualité de la traduction.

La qualité de traduction d'un système de TA est évaluée sur un *corpus de test*. Ce corpus est un ensemble de textes parallèles qui contiennent des phrases source *f* et leurs traductions d'experts (*les références*) e_r. Ce corpus doit être différent du corpus d'apprentissage. Lorsque le corpus d'apprentissage et le corpus de test appartiennent à un même domaine, il est considéré comme *dans le domaine*. En revanche, lorsque les deux corpus se réfèrent à des domaines différents, le corpus de test est considéré comme *hors domaine*. Le score de l'évaluation va déterminer le degré de ressemblance entre la traduction émise par le système (*l'hypothèse*) e_h et la/les références e_r.

Traditionnellement, l'examen de la qualité de la traduction est fait par un être humain. Nous l'appelons *l'évaluation humaine* ou *l'évaluation subjective*. La fidélité de la signification de la référence e_r, et les critères de correction, dans la langue cible, de l'hypothèse e_h sont considérés.

La fidélité de la signification indique quel pourcentage du sens exprimé dans la référence est également exprimé dans l'hypothèse. Nous pouvons définir plusieurs seuils tels que « tout », « la plupart », « beaucoup », « peu », « rien », etc.

Les critères de correction (grammaticale) indiquent la fluidité, en langue cible, de la traduction. Les valeurs qualitatives qui peuvent être utilisées sont « parfait », « bien », « non native », « mauvaise », « incompréhensible », etc.

L'évaluation humaine est coûteuse cependant. De plus, elle a le défaut d'être non reproductible puisque il existe une variabilité entre les annotateurs évaluateurs. C'est pourquoi plusieurs méthodes de mesure automatique, si possible corrélées avec l'évaluation humaine, ont été développées. L'évaluation utilisant les mesures automatiques est appelée *l'évaluation automatique* ou *l'évaluation objective*. L'évaluation automatique ne nécessite quasiment aucune intervention humaine a posteriori (mais il faut un corpus de test bilingue a priori).

2.6.1. Mesures reposant sur des taux de mots erronés

2.6.1.1 Le score WER

Le score WER (en anglais : *Word Error Rate*) est initialement utilisé en reconnaissance automatique de la parole. Il compare une hypothèse e_h à la référence e_r en se fondant sur la distance de Levenshtein au niveau de mots. Il compte le nombre minimum d'opérations à effectuer sur e_h pour la transformer en e_r. Moins il y a d'opérations à effectuer, meilleur est le score. Nous pouvons calculer le score WER avec la formule suivante :

$$WER(e_h) = \frac{n_{ins} + n_{\sup} + n_{sub}}{|e_r|} \quad \textbf{(2-27)}$$

où n_{ins}, n_{sup}, n_{sub} sont respectivement les nombres minimums d'insertions, de suppressions et de substitutions pour transformer e_h à e_r.

Lorsque plusieurs références sont utilisées, la formule peut être modifiée avec alors le numérateur qui est le nombre d'opérations minimal pour toutes les références (la référence la plus proche est considérée) et le dénominateur qui devient la moyenne des longueurs des références.

Malheureusement, cette mesure simple est moins appropriée pour la traduction parce qu'elle pénalise des hypothèses correctes dont l'ordre des mots ne correspond pas à la référence. Un mot qui est traduit correctement mais qui est mis à la mauvaise position sera pénalisé par une suppression et une insertion par exemple. On peut ainsi assigner des scores WER différents pour deux hypothèses équivalentes « *je vois un chat et un chien* » et « *je vois un chien et un chat* » avec la même référence « *je vois un chat et un chien* ».

2.6.1.2 Le score PER

Le score PER (en anglais : *Position-independent Word Error Rate*) est semblable au score WER, mais il ne prend pas en compte l'ordre des mots. Il considère l'hypothèse e_h et la référence e_r comme des ensembles de mots non ordonnés plutôt que des phrases totalement ordonnées. Le score PER a été proposé par Tillmann en 1997 [Tillmann 1997]. Il ne compte que le nombre de fois que des mots identiques sont produits dans les deux phrases. Les mots qui ne correspondent pas sont comptés comme des substitutions. Selon que la phrase traduite est plus longue ou plus courte que la référence, le reste des mots est compté comme insertion ou suppression. Ainsi, le score PER assigne le même score pour les deux hypothèses « *je vois un chat* » et « *un chat vois je* » lorsque la référence est « *je vois un chat* ».

2.6.1.3 Le score TER

Le score TER (en anglais : *Translation Edit Rate* ou *Translation Error Rate*) compte aussi le nombre minimum d'opérations à effectuer sur e_h pour la transformer en e_r. Comme le score WER, les opérations considérées sont l'insertion, la suppression, la substitution, mais aussi le déplacement d'une suite de mots [Snover 2006]. Un déplacement permet de déplacer un groupe de mots contigus vers la gauche ou la droite. Chaque déplacement est compté comme une seule opération quels que soient le nombre de mots déplacés et l'amplitude du déplacement. Le score TER est donc formulé comme suit :

$$TER(e_h) = \frac{n_{ins} + n_{\sup} + n_{sub} + n_{dep}}{|e_r|} \quad \textbf{(2-28)}$$

où n_{dep} est le nombre minimum de déplacements.

HTER (en anglais : *Human-targeted Translation Edit Rate*) est une version modifiée du score TER avec l'intervention des traducteurs humains [Snover 2006]. Apès avoir lu les références, les traducteurs humain éditent l'hypothèse du système pour générer une nouvelle phrase qui a la même signification que les références originales. Cette phrase est considérée comme une nouvelle référence humaine de l'hypothèse. Puis, le score HTER est le score TER minimum calculé entre l'hypothèse et les références originales plus la nouvelle référence humaine.

Le score HTER est moins subjectif que les jugements humains, mais il est encore coûteux, en ce que le traducteur perd environ de 3 à 7 minutes pour éditer chaque phrase. Et le score HTER n'est pas adapté pour être utilisé dans le cycle de développement d'un système de TA.

La métrique d'évaluation TER-Plus (TERp) est une autre extension du score TER avec des paramètres ajustables et l'incorporation avec la morphologie, la synonymie et des paraphrases [Snover 2009]. TERp aligne un mot de l'hypothèse avec un mot de la référence non seulement quand deux mots sont les correspondances exactes, mais aussi quand ils possèdent la même racine ou ils sont les synonymes. Plus, TERp utilise aussi la substitution de groupe de mots (des paraphrases) pour aligner deux phrases. Il utilise donc toutes les opérations de TER : l'insertion, la suppression, la substitution de mots, le déplacement d'une suite de mots ; et trois nouvelles opérations : la correspondance en racine, la correspondance en synonyme et la substitution de paraphrases. Le coût de toutes les opérations est optimisé afin de maximiser la corrélation avec les jugements humains.

Le score TERp permet de mieux aligner l'hypothèse et les références, mais le calcul dépend du dictionnaire de synonymes, de la liste de mots possédant la même racine, de la liste des paraphrases, qui ne sont pas toujours disponible pour toutes les langues.

2.6.2. Mesures reposant sur des ressemblances avec des références

2.6.2.1 Le score BLEU

Depuis ces dernières années, la métrique la plus souvent utilisée est le score BLEU (en anglais : *BiLingual Evaluation Understudy*). Le score BLEU est proposé par [Papineni 2002]. Il ne considère pas seulement la ressemblance au niveau des mots mais aussi la ressemblance au niveau des *n-grammes* entre l'hypothèse et les références.

La tâche principale est de comparer les n-grammes de l'hypothèse avec les n-grammes de la référence et de compter le nombre d'équivalences. Les correspondances sont indépendantes de la position. Plus il y a de correspondances, meilleure est l'hypothèse. Tout d'abord, les précisions modifiées de n-gramme (p_n) avec l'ordre de *1* à *N* ($n=1..N$) sont calculées pour chaque paire d'hypothèses et sa référence (ou ses références lorsque plusieurs références sont utilisées).

$$p_{n\ \text{chaque paire}} = \frac{\sum_{n-gram \in e_h} Compte_{clip}(n-gram)}{\sum_{n-gram \in e_h} Compte_{e_h}(n-gram)}$$

Pour un n-gramme *n-gram* donné, soient $Compte_{eh}(n\text{-}gram)$ le nombre de fois que ce *n-gram* apparaît dans e_h. Si nous notons c le nombre de mots de l'hypothèse e_h, e_h contient $c-n+1$ n-grammes. Le dénominateur devient $c-n+1$.

$Compte_{clip}(n\text{-}gram)$ est le nombre d'appariements de ce *n-gram* entre e_h et e_r, donc il est calculé par : $min(Compte_{eh}(n\text{-}gram), max_{\{er\}}(Compte_{er}(n\text{-}gram)))$ où $max_{\{er\}}(Compte_{er}(n\text{-}gram))$ est le nombre maximal de fois que ce *n-gram* apparaît dans une référence, parmi toutes les références disponibles.

Pour calculer la précision n-gramme modifiée sur le corpus de test entier, nous accumulons simplement les comptes pour chaque paire d'hypothèses et sa référence.

$$p_{n\ corpus} = \frac{\sum_{e_h \in corpus} \sum_{n-gram \in e_h} Compte_{clip}(n-gram)}{\sum_{e_h \in corpus} \sum_{n-gram \in e_h} Compte_{e_h}(n-gram)}$$

Pour combiner les *N* précisions n-grammes modifiées, le score BLEU utilise le logarithme moyen pondéré, ce qui est équivalent à une moyenne géométrique, et pour pénaliser les hypothèses plus courtes que leurs références, une pénalité de brièveté *BP* est introduite. Le score BLEU est finalement calculé comme suit :

$$BLEU = BP \cdot \exp(\sum_{n=1}^{N} w_n \log p_n) \quad \textbf{(2-29)}$$

w_n sont les poids positifs tels que $\sum_{n=1}^{N} w_n = 1$, souvent nous utilisons des poids uniformes $w_n=1/N$ et N=4.

$$BP = \begin{cases} 1 & c > r_p \\ e^{(1-r_p/c)} & c \le r_p \end{cases}$$

pour une paire de phrases, c est la longueur de l'hypothèse e_h, et r_p est la longueur de la référence la plus proche de e_h parmi les références. Pour le corpus entier, la somme totale de c et la somme totale de r_p de toutes les hypothèses du corpus sont calculées.

Dans le domaine des logarithmes,

$$\log BLEU = \min(1 - \frac{r_p}{c}, 0) + \sum_{n=1}^{N} w_n \log p_n \quad \textbf{(2-30)}$$

BLEU est un score de précision, sa valeur varie de 0 à 1. Plus le score est élevé, meilleure est la traduction. Une hypothèse se voit attribuer un score BLEU de « *1* » lorsqu'elle est identique à une des références ; au contraire, elle aura un score BLEU de « *0* » si aucun de ses n-grammes n'est présent dans une référence.

2.6.2.2 Le score NIST

Le score NIST (dont le nom vient de *National Institute of Standards and Technology*) a été proposé dans [Doddington 2002] et reprend le principe du score BLEU mais avec quelques adaptations légères. Il repose aussi sur la précision n-gramme, mais il utilise la moyenne arithmétique des n-grammes au lieu de la moyenne géométrique. L'expression de la pénalité de brièveté est différente de celle utilisée pour le score BLEU. Dans le score NIST, les n-grammes sont aussi pondérés selon leur fréquence d'apparition : les n-grammes rares contribuent plus au score final que les n-grammes fréquents. Par exemple, le bi-gramme anglais « *interesting calculations* » contribue plus au score que le bi-gramme « *of the* » qui apparaît souvent en anglais.

Les poids *Info()* d'un n-gramme *n-gram*= $m_1..m_n$ sur un ensemble de références sont calculés par

$$Info(n-gram) = Info(m_1...m_n) = \log_2\left(\frac{\text{compte}(\text{m}_1...\text{m}_{\text{n-1}})}{\text{compte}(\text{m}_1...\text{m}_{\text{n}})}\right)$$

où *compte*($m_1..m_n$) est le nombre de fois que le n-gramme $m_1..m_n$ apparaît dans l'ensemble.

Le score NIST est alors calculé comme suit :

$$NIST = \sum_{n=1}^{N} \left\{ \frac{\sum_{n-gram \in e_h \cap n-gram \in e_r} Info(n-gram)}{\sum_{n-gram \in e_h} compte(n-gram)} \right\} \cdot \exp\left\{ \beta \log^2 \left[\min\left(\frac{c}{\bar{r}}, 1 \right) \right] \right\} \quad \textbf{(2-31)}$$

où c est le nombre de mots de l'hypothèse e_h, $\bar{r}$ est le nombre moyen de mots de toutes les références et β est un facteur pour ajuster la pénalité de brièveté.

2.6.2.3 Le score METEOR

Les scores BLEU et NIST sont des scores de précision. [Banerjee 2005] a proposé le score METEOR (en anglais : *Metric for Evaluation of Translation with Explicit ORdering*) qui équilibre entre la précision et le rappel. Ce score est calculé sur la base d'un alignement entre les uni-grammes d'une hypothèse et ceux d'une référence.

Un alignement est un ensemble d'appariements d'uni-grammes. Un uni-gramme d'une phrase est mis en correspondance avec zéro ou un seul uni-gramme d'une autre phrase. Les appariements sont établis d'abord sur les formes orthographiques, puis les mots de même racine (par exemple : « *joli* » et « *jolie* ») et enfin sur les synonymes (par exemple : « *joli* » et « *beau* »). Il permet de valider la fidélité de la signification de l'hypothèse avec plusieurs choix de lexiques, différents de la référence. Le meilleur alignement est celui qui contient le plus grand nombre d'appariements d'uni-grammes avec le plus petit nombre de réarrangements. Le score METEOR est déterminé à partir de ce meilleur alignement.

$$METEOR = F_{moyenne} \cdot (1 - \text{Pénalité}) \quad \textbf{(2-32)}$$

où $F_{moyenne} = \dfrac{P \cdot R}{\alpha \cdot P + (1-\alpha) \cdot R}$ et $\text{Pénalité} = \gamma \cdot \left(\dfrac{\text{Compter(segment)}}{\text{Compter(unigram.align.)}} \right)^{\beta}$

La précision P est le nombre d'uni-grammes appariés de l'hypothèse divisé par la taille de cette hypothèse, et le rappel R est ce nombre d'uni-grammes appariés divisé par la taille de la référence. La *Pénalité* favorise l'hypothèse qui contient des segments d'uni-grammes consécutifs appariés plus longs. *Compter(unigram.align.)* est le nombre d'appariements d'une hypothèse. *Compter(segment)* est le nombre de segments (le plus long) de l'hypothèse mis en correspondance avec les segments de la référence. Les poids originaux sont α=0,9 ; β=3 ; γ=0,5.

Ci-dessous nous présentons un exemple de calcul des scores pour une paire référence et hypothèse. La référence pour cet exemple est la phrase anglaise « *it is a guide to action which ensures that the military always obeys the commands of the party* ».

La première hypothèse est identique à la référence, donc les scores pour cette hypothèse sont les meilleurs scores dans tous les cas. Dans la deuxième hypothèse, le mot « *that* » est proposé au lieu du mot « *which* ». Ce changement influence tous les scores sauf le METEOR parce que celui-ci considère des synonymes. Pour la troisième hypothèse, l'ordre de mot est changé donc le score WER est influencé plus que le score PER. L'hypothèse 4 possède plusieurs synonymes avec la référence, donc son score METEOR est encore grand. La dernière hypothèse est la plus mauvaise, comme ses scores l'indiquent. Il semble que le score METEOR est le score le plus efficace, mais le calcul dépend du dictionnaire de synonymes qui n'est pas disponible pour toutes les langues.

Tableau 2-1 : Exemple de calcul des scores pour une paire référence / hypothèse.

La reference : « *it is a guide to action which ensures that the military always obeys the commands of the party* » [1]

	Les hypothèses	**# mots**	**WER**	**PER**	**TER**	**BLEU**	**NIST**	**METEOR**
1.	it is a guide to action which ensures that the military always obeys the commands of the party	18	0	0	0	100	4.18	99.99
2.	it is a guide to action that ensures that the military always obeys the commands of the party	18	5.56	5.56	5.55	83.94	3.95	99.99
3.	it is a guide to action that ensures that the military obeys the party commands always	16	33.33	16.67	27.77	47.28	3.69	88.52
4.	it is a guide to action that ensures that the military will forever heed party commands	16	44.44	33.33	38.88	42.29	2.95	82.71
5.	this guide to action ensures the army following the commands of the party	13	44.44	44.44	44.44	27.84	1.98	67.30
6.	this is good guide to the activities for sure which united states the commander obeys and the company	18	77.78	55.56	72.22	0	1.58	46.03

La métrique la plus souvent utilisée au sein de la communauté de TA est le score BLEU. Une bonne corrélation entre le score BLEU et l'évaluation humaine a été constatée [Papineni 2002], [Doddington 2002], etc. Mais le score présente encore des restrictions comme citées dans le travail de [Callison-Burch 2006] :

- Il n'impose pas de contraintes explicites sur l'ordre des n-grammes. Donc il peut exister plusieurs variations d'une hypothèse (créées par la permutation et la substitution des n-grammes) pour lesquelles le score BLEU est le même
- Il traite tous les mots (les mots outils, les mots lexicaux, les mots d'arrêt, etc.) également
- Il ne considère pas les synonymes
- L'évaluation au niveau d'une phrase n'est pas bonne à cause de la précision 4-gramme qui est souvent égale à 0 (comme la sixième hypothèse du Tableau 2-1), donc le score BLEU est souvent évalué au niveau du corpus.

Cependant, BLEU est considéré comme la mesure automatique efficace pour les utilisations appropriées telles que l'observation des changements progressifs d'un système unique, la comparaison des systèmes qui emploient des approches de traduction similaires. Ainsi, dans notre travail, nous continuons à utiliser le score BLEU qui répond à ces deux objectifs.

2.7. Systèmes de référence existants aujourd'hui

Les systèmes de traduction probabiliste connaissent un véritable succès depuis plusieurs années. La mise en place d'un système de traduction probabiliste est rendue très accessible notamment grâce à la disponibilité de plusieurs systèmes de traduction en code source libre.

[1] Pour calculer les scores, nous utilisons des outils suivants :
- WER et PER : apertium-eval-translator. https://apertium.svn.sourceforge.net/svnroot/apertium/trunk/apertium-eval-translator/
- TER : tercom-0.7.25. http://www.cs.umd.edu/~snover/tercom/
- BLEU (4-grammes) et NIST (5-grammes): mteval-v11b.pl. http://www.itl.nist.gov/iad/mig/tests/mt/2008/scoring.html
- METEOR est calculé manuellement

Le programme *GIZA* a été élaboré en 1999, dans la boîte à outil nommé *Egypt*[1], pour apprendre les alignements en mots et estimer les paramètres des modèles de traduction IBM 1-3 à partir d'un corpus parallèle bilingue aligné au niveau des phrases [Al-Onaizan 1999]. Après, [Och 2000] a développé le programme *GIZA++*[2], une extension du programme *GIZA*, avec des nombreuses fonctionnalités supplémentaires pour traiter le modèle IBM-4 et le modèle HMM. Daniel Marcu et Ulrich Germann [Germann 2001] a développé l'outil *ISI ReWrite Decoder*[3] pour décoder avec le modèle IBM-4. Pour implémenter le décodage de la traduction en groupe de mots, le décodeur *Pharaoh*[4] est devenu disponible [Koehn 2004] qui implémente la recherche en faisceau.

Plus récemment, une autre boîte à outils en code source libre a été construite et publiée : *MOSES*[5] [Koehn 2007a]. Cette boîte à outils contient un ensemble complet de scripts pour construire un système complet de traduction probabiliste par groupe de mots, tels que le script pour aligner des mots par *GIZA++*, le script pour l'extraction de groupes de mots, le script pour l'entraînement et l'estimation des paramètres du système et un décodeur.

Le système utilise le modèle log-linéaire combinant un modèle de langue, deux modèles de traduction en groupe de mots ($t_g(\widetilde{f}_z|\widetilde{e}_z)$, $t_g(\widetilde{e}_z|\widetilde{f}_z)$), deux modèles de traduction de lexique ($p_w(\widetilde{f}_z|\widetilde{e}_z)$, $p_w(\widetilde{e}_z|\widetilde{f}_z)$), une pénalité de groupe de mots, une pénalité de mots et un modèle de distorsion.

Le système de TA *MOSES* continue à être développé avec de nombreuses fonctionnalités supplémentaires telles que :

- des modèles de traduction factoriels [Koehn 2007b] qui permettent d'intégrer d'autres informations (forme de surface, lemme, partie du discours, etc.) au niveau des mots
- le décodage avec des réseaux de confusion et de treillis de mots permet d'intégrer facilement le système de traduction avec des systèmes de reconnaissance automatique de la parole ou des analyseurs morphologiques [Bertoldi 2008a]
- depuis peu, *MOSES* permet de construire des modèles hiérarchiques similaires à ceux décrits dans [Chiang 2007].

Le décodage de *MOSES* se fait par l'algorithme de recherche en faisceau. *MOSES* est le successeur du décodeur *Pharaoh*, mais il permet d'utiliser plus de types de représentation en entrée du décodeur (forme de surface, partie du discours, réseaux de confusion, treillis de mots) que *Pharaoh* (forme de surface seule).

Il y a d'autres systèmes tels que *Cdec*[6] (un ensemble d'outils pour aligner, entraîner et décoder) [Dyer 2010], etc., mais le système *MOSES* est utilisé le plus largement et il est considéré aujourd'hui comme un système de référence dans la communauté de TA probabiliste.

En plus, certains groupes de recherches ont développé des boîtes à outils de traduction hiérarchiques telles que le décodeur *Joshua*[7] de l'université de John Hopkins [Li 2009], la boîte à outil *Jane*[8] de l'université de RWTH Aachen [Vilar 2010].

[1] http://www.clsp.jhu.edu/ws99/projects/mt/
[2] http://www-i6.informatik.rwth-aachen.de/Colleagues/och/software/GIZA++.html
[3] http://www.isi.edu/licensed-sw/rewrite-decoder/
[4] http://www.isi.edu/licensed-sw/pharaoh/
[5] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial
[6] http://cdec-decoder.org
[7] http://joshua.sourceforge.net/Joshua/Welcome.html
[8] http://www-i6.informatik.rwth-aachen.de/jane/

Dans le cadre de cette thèse, nous utilisons la boîte à outil *MOSES* (avec *GIZA++*) pour construire le système de traduction (aligner, entraîner et décoder), et l'outil *SRILM* [Stolcke 2002] pour construire le modèle de langue de type n-gramme.

Chapitre 3 : Langue vietnamienne, une des langues peu dotées

Les chapitres suivants de cette thèse présentent l'application de nos travaux sur l'extraction des corpus parallèles pour la langue vietnamienne, qui peut être considérée du point de vue de la traduction et du TALN comme une langue peu dotée. Tout d'abord, pour traiter les données, pour réaliser un système d'extraction de corpus parallèle et pour construire un système de traduction pour la langue vietnamienne, une connaissance minimale des caractéristiques linguistiques de la langue vietnamienne est nécessaire. Ce chapitre 3 présente en bref la langue vietnamienne et l'état de l'art sur la traduction pour la langue vietnamienne.

Le vietnamien (en vietnamien : « *tiếng Việt* ») est parlé par environ 86 millions de personnes au Vietnam et environ 4 millions de personnes à l'étranger (en 2009)[1]. Le vietnamien est la langue nationale et officielle du Vietnam. Il est la langue maternelle du groupe ethnique « *Kinh* » (86 % de la population) et la langue secondaire des 53 ethnies minoritaires du Vietnam. À l'étranger, le vietnamien est parlé principalement aux Etats-Unis, au Canada, en Australie, et aussi au Cambodge, au Laos, en Chine, en France, etc. (selon Ethnologue[2]).

Selon les linguistes, le vietnamien appartient au groupe Viet-Muong, branche Môn-khmer de la famille Austro Asiatique (numéro 14 dans la Figure 3-1) qui comprend également le khmer, parlé au Cambodge, ainsi que diverses langues parlées par des groupes minoritaires, telles que les langues Munda parlées dans l'est de l'Inde, et d'autres langues dans le sud de la Chine. Le vietnamien du Vietnam et le khmer du Cambodge rassemblent 92 % des locuteurs de cette famille qui est parlée par environ 1,1 % de la population mondiale, en comparaison de la famille Indo-européenne (numéro 1 dans la Figure 3-1) parlée par 48 % de la population mondiale[3].

[1] Bureau de statistique générale du Vietnam http://www.gso.gov.vn
[2] Un ouvrage de référence encyclopédique catalogue de toutes les 6 909 langues vivantes connu du monde http://www.ethnologue.com
[3] Site Aménagement linguistique dans le monde. Site hébergé par le Trésor de la langue française au Québec (TLFQ), Université Laval, Québec. http://www.tlfq.ulaval.ca/axl/index.html

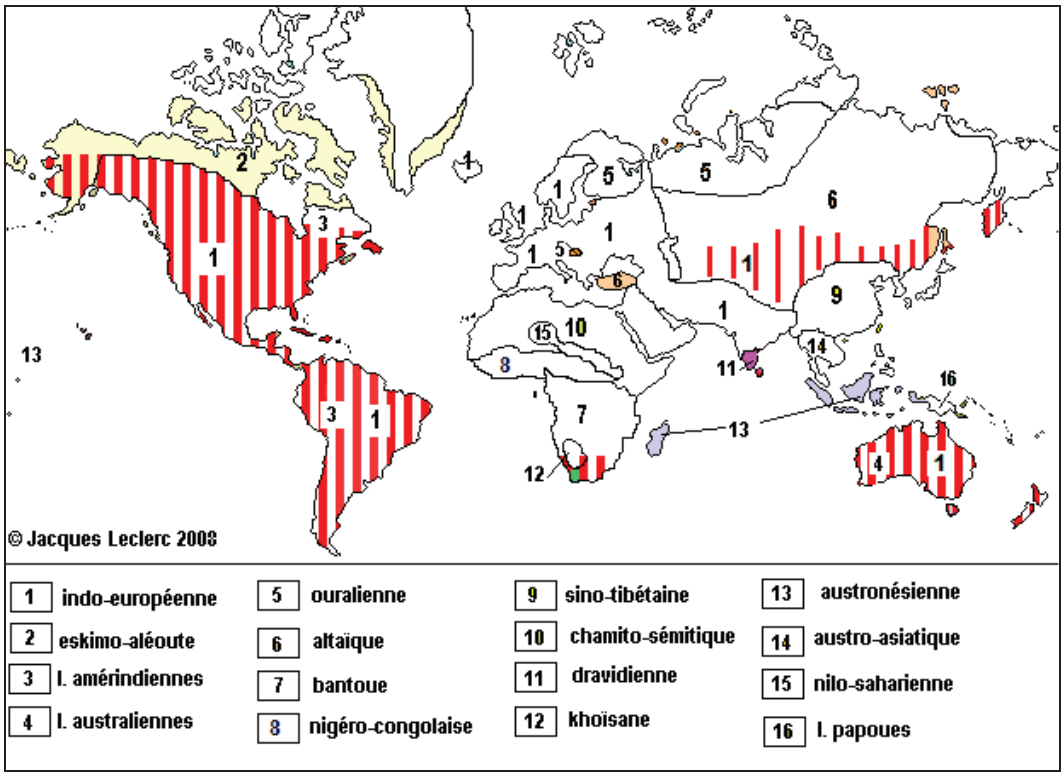


Figure 3-1 : Les familles linguistiques dans le monde
(src: Site Aménagement linguistique dans le monde[1])

3.1. Évolution historique de la langue vietnamienne

L'histoire de la langue vietnamienne est très ancienne. Au fil des années, le Vietnamien a beaucoup changé pour atteindre sa forme actuelle. Nous pouvons compter environ 6 périodes dans le développement du vietnamien [Nguyen T.C. 1978], [Nguyen H.Q. 2008].

1. *La période du « pré- Viet-Muong »* : des milliers d'années avant Jésus-Christ jusqu'au premier siècle après Jésus-Christ. C'est la période pendant laquelle le vietnamien et toutes les langues du groupe Viet-Muong étaient alors une seule langue commune dans la branche Môn-khmer. À ce moment là, le vietnamien utilisait entièrement le vocabulaire de la branche Môn-khmer qui comprenait des mots mono-syllabiques et des mots multi-syllabiques. Par ailleurs, le vietnamien dans cette période n'avait pas de ton comme toutes les autres langues de la branche Môn-Khmer.

2. *La période du « Viet-Muong ancien» :* du premier siècle ap. J. C. au huitième ou neuvième siècle. Il s'agit d'une période où le vietnamien s'est séparé de la branche Môn-khmer pour devenir la langue Viet-Muong mono-syllabique. Historiquement, à l'époque, le Vietnam était dominé par la Chine, donc « le *Hán* », la langue du chinois classique, dominait tout le pays (les caractères sont issus du système chinois, mais la prononciation est vietnamienne). Le *Hán* devint la langue administrative et a été utilisée dans toutes les œuvres « savantes ». Pourtant, le vietnamien qui n'avait pas encore de système d'écriture, se développait en parallèle dans le peuple. Le vietnamien de cette période avait été influencé par la tendance du monosyllabisme. De plus, trois tons apparaissent dans la langue pendant cette période.

[1] http://www.tlfq.ulaval.ca/axl/monde/familles.htm

3. *La période du « Viet-Muong général» :* du 10ème siècle au 14ème siècle. Le 10ème siècle a marqué le début d'une période d'indépendance du Vietnam par rapport à la domination de la Chine. Socialement, le vietnamien devenait la langue parlée de la population mais l'écriture *Hán* jouait encore un rôle dans l'éducation, l'administration, et la littérature. Cependant, au 12ème siècle, les Vietnamiens ont inventé leur propre système d'écriture : « le *Nôm* » pour transcrire les mots de leur langue vernaculaire. Les caractères du *Nôm* sont basés sur l'écriture idéographique au moyen de caractères chinois simples ou combinés entre eux. Un mot du *Nôm* est créé de la manière suivante :
    - garder le caractère et le sens du Hán, mais changer la prononciation ;
    - garder le caractère et la prononciation du Hán, mais changer le sens ;
    - garder le caractère du Hán, changer le sens et la prononciation ;
    - assembler deux caractères du Hán pour former un autre mot ; cette manière était extrêmement populaire et elle permettait de former une classe de mots appelés des mots sino-vietnamiens (en vietnamien : « từ Hán-Việt »).

- Le grand inconvénient de cette écriture était qu'elle impliquait aussi la connaissance du chinois, mais l'avantage était de maintenir la tradition historique et culturelle avec le chinois. Les deux systèmes (le *Hán* et le *Nôm*) ont co-existé jusqu'au 20ème siècle. Dans cette période, le *Nôm* était utilisé principalement par le peuple.

- La langue dans cette période a montré les caractéristiques suivantes :
    - le phénomène d'emprunt des mots chinois pour former une classe de mots importante plus tard dans la langue vietnamienne : le « sino-vietnamien » ; mais, l'emprunt n'a pas été uniforme selon les régions du pays, c'est la cause de la différenciation des lexiques selon les régions plus tard ;
    - le vietnamien a également amplifié sa tendance au mosyllabisme ; dès lors, il n'y a eu que des mots monosyllabiques ;
    - enfin, c'est à cette période que le vietnamien a adopté six tons.

4. *La période du « Viet ancien » :* du 14ème siècle à la fin du 15ème siècle. Dans cette période, le *Viet-Muong* a été séparé complètement en deux classes individuelles : le *Viet* et le *Muong*. Le *Nôm* a atteint son zénith, notamment dans la littérature. La classe de mots sino-vietnamiens qui n'existe pas dans le *Muong* a été perfectionnée et rendue stable.

5. *La période du « Viet médiéval » :* de la fin du 15ème siècle jusqu'au début du 19ème siècle. Le fait marquant de cette période a été l'apparition d'un nouveau système d'écriture, le « *Quốc Ngữ* ». Ce n'est qu'au 17ème siècle que, des jésuites franco-portugais, emmenés par Alexandre de Rhodes (1591-1660), introduisirent une écriture vietnamienne latinisée qui est une transcription de la langue parlée vietnamienne en signes alphabétiques empruntés aux langues d'origine latine. Ils ont utilisé 6 accents différents pour transcrire les 6 tons vietnamiens. Le premier dictionnaire vietnamien utilisant cette écriture est le « *Dictionarium Annamiticum Lusitanum et Latinum* » (un dictionnaire Vietnamien – Portugais – Latin) d'Alexandre de Rhodes, qui fut imprimé à Rome en 1651. Toutefois, le nouvel alphabet romanisé s'est implanté difficilement pour toutes sortes de raisons, dont des raisons politiques et idéologiques.

6. *La période du « vietnamien moderne » :* depuis le milieu du 19ème siècle. Lorsque la France a envahi le Vietnam à la fin du 19ème siècle, l'utilisation du *Hán* a été abolie. Le français a remplacé le chinois dans l'administration et l'éducation. Sous la colonisation française, le vietnamien a également évolué par emprunt des mots et des constructions grammaticales françaises. Le vietnamien a adopté de nombreux termes français aussi bien du domaine scientifique et technique que de la vie courante, tels que « *đầm* » (madame), « *ga* » (la gare), « *sơ mi* » (la chemise), « *búp bê* » (la poupée), « *vít* » (la vis), « *bu lông* » (le boulon), etc.

Toutefois, le nouveau mode d'écriture « *Quốc Ngữ* » n'est parvenu à prédominer qu'au début du 20ème siècle, lorsque l'éducation est devenue généralisée et lorsqu'un système d'écriture plus simple a semblé plus approprié pour l'enseignement et la communication avec la population en général. Après l'indépendance en 1945, le gouvernement vietnamien a décidé que l'écriture « *Quốc Ngữ* » devait devenir la langue nationale et officielle du Vietnam. Aujourd'hui, le « *Quốc Ngữ* » est utilisé dans la vie courante et pour toutes les activités politiques et sociales.

3.2. Introduction générale du vietnamien

Le vietnamien aujourd'hui possède une écriture romanisée. Le système d'écriture du vietnamien est latinisé depuis le 17ème siècle. L'alphabet vietnamien utilise des caractères latins auxquels ont été ajoutés quatre signes diacritiques pour créer des nouvelles lettres. Les lettres « *f* », « *j* », « *w* », « *z* » n'existent pas dans l'alphabet vietnamien. Il y a 29 lettres dans l'alphabet du vietnamien, neuf bi-grammes et un tri-gramme (voir la Figure 3-2).

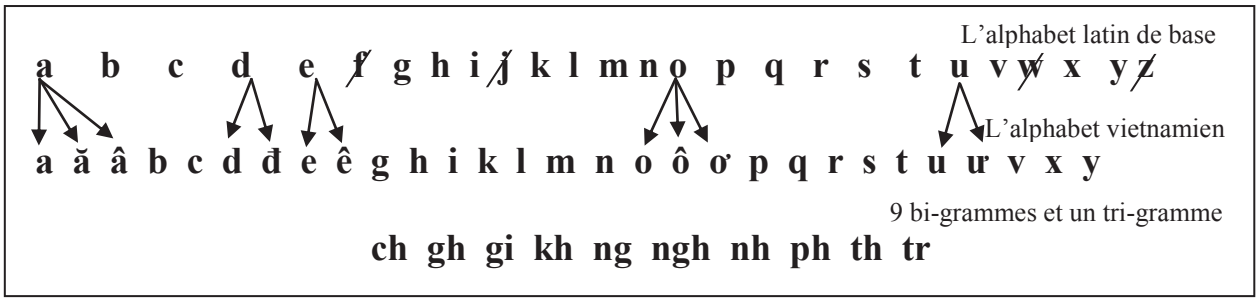


Figure 3-2 : L'alphabet vietnamien

Le vietnamien est devenu une langue tonale avec 6 tons après un long processus de développement [Doan T.T. 1999]. Cinq signes diacritiques sont utilisés pour représenter les six tons : « aucun » (ton plat), « ` » (ton descendant), « ' » (ton interrogatif), « ~ » (ton brisé), « ´ » (ton montant), « . » (ton grave). Il n'y a qu'un ton unique dans chaque syllabe. Ils sont relativement difficiles à prononcer pour les étrangers. Tous les diacritiques sont placés sur les voyelles de la syllabe, sauf le signe diacritique du ton grave qui est placé sous la voyelle.

Tableau 3-1 : Les tons vietnamiens et leur exemple

Ton vietnamien	Registre – Inflexion	Voyelles avec diacritiques	Exemples	
			Mot vietnamien	Signification
Ton montant	haut – oblique bref	*á, ắ, ấ, é, ế, í, ó, ố, ớ, ú, ứ, ý*	*có*	avoir
Ton brisé	haut – oblique long	*ã, ẵ, ẫ, ẽ, ễ, ĩ, õ, ỗ, ỡ, ũ, ữ, ỹ*	*mỹ*	l'américain
Ton plat	haut – égal	*a, ă, â, e, ê, i, o, ô, ơ, u, ư, y*	*gai*	l'épine
Ton interrogatif	bas – oblique long	*ả, ẳ, ẩ, ẻ, ể, ỉ, ỏ, ổ, ở, ủ, ử, ỷ*	*hưởng*	apprécier
Ton descendant	bas – égal	*à, ằ, ầ, è, ề, ì, ò, ồ, ờ, ù, ừ, ỳ*	*trà*	le thé
Ton grave	bas – oblique bref	*ạ, ặ, ậ, ẹ, ệ, ị, ọ, ộ, ợ, ụ, ự, ỵ*	*lại*	venir

Le ton est très important pour trouver la signification d'un mot vietnamien. Si deux mots mono-syllabiques phonétiquement équivalents ont des tons différents, ils ont des significations différentes.

Tableau 3-2 : La signification d'un mot vietnamien dépend de son ton (src : [Tran D.D, 2007])

Mot vietnamien	*bá*	*bã*	*ba*	*bả*	*bà*	*bạ*
Signification	le roi	le marc (de café)	trois	la pâture	grand-mère	n'importe

Le Vietnam est divisé en 3 régions dialectales principales : les dialectes du Nord (*Hà Nội*), du Centre (*Huế*) et du Sud (la ville *Hồ Chí Minh*). Ces dialectes ont des tonalités et des lexiques différents ce qui peut, en pratique, rendre difficile les échanges d'une région à l'autre [Hoang T.C. 2004]. La forme parlée considérée comme le standard du vietnamien est le vietnamien de la capitale *Hà Nội*.

3.3. Lexique vietnamien

Le vietnamien possède une unité qui s'appelle « *tiếng* », elle peut être considérée comme « la syllabe vietnamienne », une unité linguistique indispensable dans tous travaux en traitement de la langue vietnamienne. Dans le langage parlé, chaque « *tiếng* » est prononcé comme une seule syllabe, et dans le langage écrit, les « *tiếng* » sont séparés par le blanc typographique (espace).

Cette unité peut donc constituer tour à tour « une syllabe du point de vue phonologique », « un morphème du point de vue syntaxique » ou « un mot simple du point de vue des constituants de la phrase » [Nguyen T.M.H. 2006]. Par exemple, la phrase « *tôi đi đến trường* » (*je/aller/à/l'école* : je vais à l'école) est une phrase qui contient quatre « *tiếng* » correspondants à quatre syllabes, quatre morphèmes, et quatre mots.

Cependant, les mots vietnamiens sont formés d'une seule syllabe (« *tiếng* ») ou de plusieurs syllabes. Par exemple le mot « *xã hội* » (le social) contient 2 syllabes, le mot « *xã hội hóa* » (socialiser) contient 3 syllabes. Les divers types de combinaison donnent naissance à plusieurs types de mots (Figure 3-3). Donc les linguistes sont parvenus à un accord et considèrent qu'un mot en vietnamien est « la plus petite unité ayant un sens spécifié, une structure *stable*, et utilisée pour composer des constituants de phrase » [Nguyen T.M.H. 2006], [Mai N.C. 1997].

En se basant sur le nombre de syllabes, les mots vietnamiens sont :

- soit des mots simples : les mots ne contiennent qu'une syllabe, tels que « *nhà* » (la maison), « *cửa* » (la porte), « *bàn* » (la table), « *ghế* » (la chaise), « *cười* » (rire), « *nói* » (parler), « *hát* » (chanter), « *đi* » (aller), etc. ;
- soit des mots complexes : les mots multi syllabiques, tels que « *ví dụ* » (l'exemple), « *hài lòng* » (content), « *công nghiệp hóa* » (industrialiser), etc.

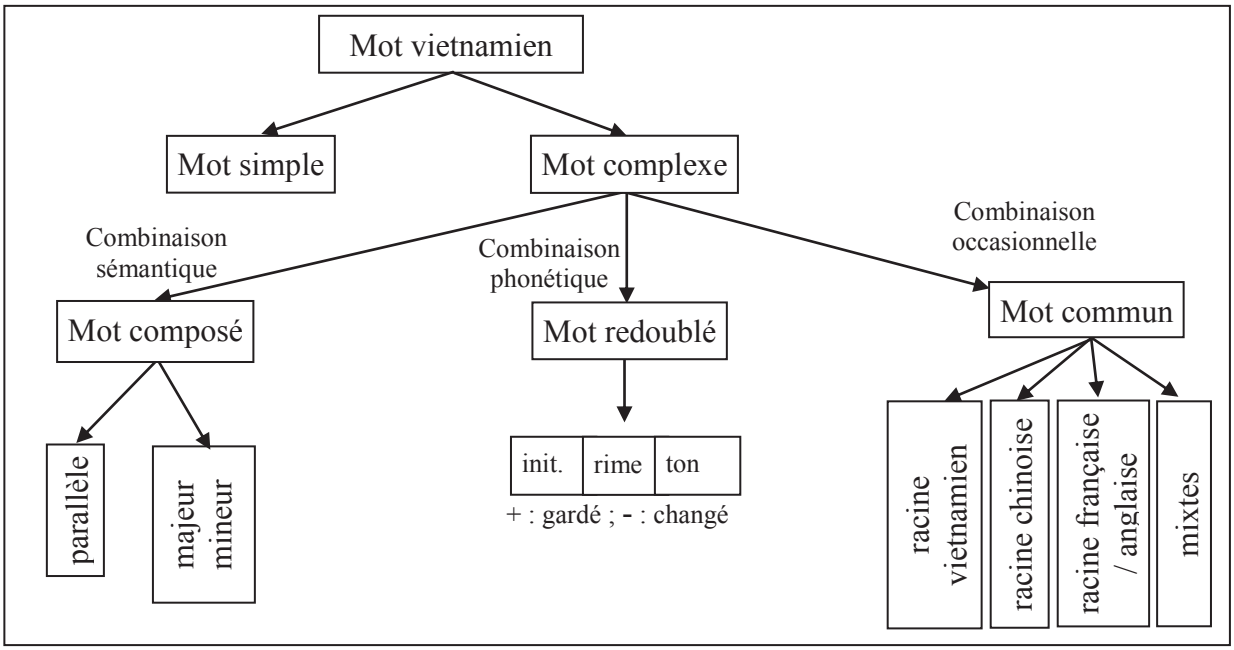


Figure 3-3 : La classification d'un mot vietnamien

Dans le « Dictionnaire Vietnamien » de Hoang Phe [Hoang P. 2002], un des plus grands dictionnaires vietnamiens, il y a environ 37.000 mots distincts (et 40.000 mots en ajoutant les mots homographes). On y trouve un grand pourcentage, environ 80 %, de mots complexes [Dinh D. 2003c]. A cause de la grande fréquence des mots bi-syllabiques et multi-syllabiques, la segmentation de texte vietnamien en mot est compliquée et il y a beaucoup d'ambiguïtés de segmentation.

Toutefois, dans une expérimentation de [Le V.B. 2006], l'auteur présente une statistique sur le nombre de mots vietnamiens selon le nombre de syllabes par mot dans un corpus de textes vietnamien récupéré sur les sites Web (environ de 5 millions de phrases). Bien que les mots multi-syllabiques possèdent un grand pourcentage dans le dictionnaire, le nombre de mots multi-syllabiques n'est que de 21,2 % dans ce corpus de textes. Les mots monosyllabiques semblent ainsi les plus utilisés dans la langue courante.

3.3.1. Des mots vietnamiens complexes

Les mots complexes ou mots multi syllabiques consistent à associer deux ou plusieurs syllabes, de façon à créer une nouvelle signification. Les mots complexes sont classés en fonction du mode de formation du mot ou de la façon dont les syllabes sont composées [Nguyen L.T. 2010].

3.3.1.1 La combinaison sémantique

La première façon de combiner les syllabes pour créer *un mot composé* est la combinaison sémantique. Lorsque les syllabes d'un mot sont liées l'une à l'autre par le sens, nous obtenons des mots composés. Nous distinguons deux classes de mots composés, en fonction de la composition sémantique des syllabes: des *mots composés parallèles* et des *mots composés majeur/mineur*.

Mots composés parallèles. Ce sont des mots dont chaque syllabe possède son propre sens mais joue un rôle égal dans la signification du mot composé. Cependant le mot composé ainsi créé présente un nouveau sens qui est souvent plus large/général et plus abstrait que les sens des syllabes composantes. Par exemple :

- « *quần* » (le pantalon), « *áo* » (la chemise) => « *quần áo* » ou « *áo quần* » (le vêtement) ;
- « *giang* » (la rivière), « *sơn* » (le montagne) => « *giang sơn* » (le pays natal) ;
- « *phố* » (la rue), « *phường* » (l'arrondissement) => « *phố phường* » (les rues, les avenues, les quartiers d'une ville).

L'ordre des syllabes constituantes est normalement stable, mais dans certains cas cet ordre peut être modifié sans changer le sens :

- « *mạnh khỏe* » = « *khỏe mạnh* » (en bonne santé) ;
- « *mong chờ* » = « *chờ mong* » (attendre avec impatience) ;
- « *quần áo* » ou « *áo quần* » (le vêtement) ;

Mots composés majeur/mineur. La deuxième classe constitue des mots composés majeur/mineur. Ce mode de composition consiste à greffer sur un mot servant de base un autre élément dont le rôle est de délimiter le sens souvent trop général de l'unité de base.
Par exemple : « *xe* » (le véhicule) : le mot de base ;

- « *đạp* » (pédaler) => « *xe đạp* » (le vélo) ;
- « *lửa* » (le feu) => « *xe lửa* » (le train).

3.3.1.2 La combinaison phonétique

La combinaison phonétique constitue une particularité de la langue vietnamienne : les mots redoublés. Elle consiste à mettre côte à côte des syllabes, généralement deux syllabes, ayant des particularités phonétiques telles qu'elles puissent créer ensemble une certaine harmonie euphonique du mot dissyllabique nouvellement formé [Nguyen L.T. 2010].

En particulier, la syllabe vietnamienne se compose de trois parties fondamentales : la partie initiale, la rime et le ton (par exemple, dans le mot « *người* » (personne) : l'initiale est « *ng* », la rime est « *ươi* », et le ton est descendant « ` »). Un mot est considéré comme un mot redoublé lorsque ses syllabes possèdent une répétition complète ou partielle de la partie fondamentale. Nous pouvons compter sept sous-classes des mots redoublés :

Tableau 3-3 : Sept sous-classes des mots redoublés vietnamiens (« + » : gardé ; « - » : changé)

	Partie initiale	**Rime**	**Ton**	**Exemples**
1.	+	+	+	« *khăng khăng* » (persister) *« hiu hiu »* (souffler légèrement (en parlant du vent)) *« lâng lâng »* (dispos)
2.	–	+	+	« *lí nhí* » (balbutier) ; « *lật đật* » (pressé) ; « *càu nhàu* » (grogner)
3.	+	–	+	« *xanh xao* » (pâle) ; « *gày gò* » (maigre) ; « *chắc chắn* » (solide)
4.	+	+	–	« *đu đủ* » (le papayer) ; « *bong bóng* » (la vessie) ; « *lừng lững* » (trop grand)
5.	–	–	+	« *tình cờ* » (par hasard) ; « *ướt át* » (humide) ; « *thịnh vượng* » (prospère)
6.	+	–	–	*« khang khác »* (un peu différent) ; *« chênh chếch »* (légèrement oblique) ; « *thật thà* » (franc)
7.	–	+	–	« *bình tĩnh* » (calme) ; « *bình minh* » (l'aube) ; « *chơi bời* » (s'amuser)

Il existe aussi des mots redoublés contenant trois ou quatre syllabes mais ils sont rares. Ces redoublements multiples renforcent le sens du mot initial. Par exemple :

- « *khít khìn khịt* » = *« khít khịt »* (bien ajusté) *;*
- *« sát sàn sạt »* = *« sát sạt »* (très près ; très exactement) ;
- *« dửng dừng dưng »* = *« dửng dưng »* (détaché) *;*
- *« lếch tha lếch thếch »* = *« lếch thếch »* (en désordre) ;
- *« vội vội vàng vàng »* = *« vội vàng »* (se dépêcher).

Dans la combinaison sémantique, deux syllabes du mot composé possèdent un sens différent respectivement, alors nous pouvons dire que chaque syllabe est un morphème et le mot composé est un autre morphème. Par exemple :

- « *nhà* » (la maison), « *khách* » (le visiteur) => « *nhà khách* » (la maison de réception)
- « *nhanh* » (rapide), « *ý* » (idée, esprit) => « *nhanh ý* » (avoir de la présence d'esprit)

Au contraire, dans la combinaison phonétique, nous trouvons des mots redoublés dont :

- une seule syllabe possède un sens relatif au sens global, l'autre n'en a pas : (« *nhỏ* » (petit), « *nhắn* » (faire avertir) => « *nhỏ nhắn* » (mignon / minime) ;
- ou aucune des syllabes n'est dotée d'un sens relatif au sens global du mot : « *trục* » (l'axe), « *trặc* » (ϕ) => « *trục trặc* » (détraqué, en panne) ; « *càu* » (ϕ), « *nhàu* » (ϕ) => « *càu nhàu* » (grogner).

Les syllabes de ces mots ne sont pas des morphèmes, sauf que le mot entier est un morphème. En général, l'ordre des syllabes dans le mot redoublé est relativement fixe. Cependant, certains cas sont possibles : « *thiết tha* » = « *tha thiết* » (s'attacher à qqch), « *vẩn vơ* » = « *vơ vẩn* » (futilement), etc.

3.3.1.3 La combinaison occasionnelle

Les mots dont les syllabes sont constituées occasionnellement, sans relation du sens ou de la phonétique, forment la dernière classe, la classe de mots communs qui contient :

1. Des mots possédant une racine vietnamienne : « *bồ câu* » (le pigeon), « *mồ hôi* » (la sueur), « *mặc cả* » (négocier), etc.
2. Des mots empruntés possédant une racine chinoise. Une grande partie du vocabulaire vietnamien est composée de mots empruntés au chinois, avec une prononciation vietnamisée. Par exemple : « *mâu thuẫn* » (le désaccord), « *hi sinh* » (sacrifier), « *kinh tế* » (l'économie), « *câu lạc bộ* » (le club), « *tài xế* » (le chauffeur), « *lục tàu xá* » (un type de soupe sucrée), etc.
3. Des mots empruntés possédant une racine française ou anglaise : « *mít tinh* » (en anglais *meeting*), « *phông* » (en anglais *font*), « *sơ mi* » (en français *la chemise*), « *mùi xoa* » (en français *le mouchoir*), « *xà phòng* » (en français *le savon*), etc. Le vietnamien adopte aussi des termes scientifiques et techniques français arrivés par la voie de la transcription phonétique. Par exemple : « *a-xít* » (l'acide), « *pê-ni-xi-lin* » (la pénicilline), « *pê-đan* » (la pédale), « *phanh* » (le frein). Parallèlement, il y a aussi les mots ayant des formes lexicales « étrangères » qui ne sont pas transcrit du vietnamien mais sont souvent utilisés dans la vie quotidienne, comme les mots « *internet* », « *e-mail* », « *mobile* », « *fax* », etc.
4. Des mots mixtes : ce sont des mots utilisant un mélange de trois types ci-dessus. Par exemple :
   - « *vôi hoá* » (la calcification) = « *vôi* » (le calcin) est un mot originaire de vietnamien + « *hoá* » (la transformation) est un mot sino-vietnamien
   - « *ôm kế* » (l'ohmmètre) = « *ôm* » (ohm) est un mot anglais + « *kế* » (l'instrument pour mesurer) est un mot sino-vietnamien
   - « *nhà băng* » (la banque) = « *nhà* » (la maison) est un mot originaire de vietnamien + « *băng* » (la banque) est un mot français.

3.3.2. La segmentation en mots d'un texte vietnamien

Dans l'écriture, les syllabes vietnamiennes d'une phrase sont séparées par un espace. C'est pourquoi, le processus de segmentation syllabique est trivial pour le vietnamien. Par contre, pour la plupart des langues, la segmentation d'une phrase en syllabes est plus difficile, car un mot est souvent écrit en chaînes de syllabes (français, anglais, etc.) ou une phrase est une chaîne de caractères sans espace entre les mots et les syllabes (khmer, thaï, etc.).

Cependant, un mot vietnamien correspond à une ou plusieurs syllabes mais les espaces ne sont pas utilisés pour identifier les limites de mots. La procédure de segmentation d'une phrase vietnamienne en mots peut s'avérer très difficile à cause de lexiques gros et ambigus.

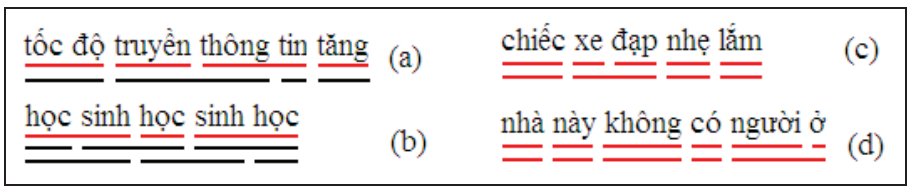


Figure 3-4 : L'ambiguïté dans la segmentation en mot de la phrase vietnamienne

Par exemple, dans la Figure 3-4, les phrases peuvent être segmentées de plusieurs façons. Dans les cas (a) et (b), la première segmentation est correcte, alors que la/les segmentation(s) suivante(s) est/sont incorrecte(s) :

1. Les segmentations correctes :
   - « *tốc độ / truyền / thông tin / tăng*» (*la vitesse/transmettre/l'information/accélérer* : la vitesse de transmission de l'information est accélérée)

- « *học sinh / học / sinh học* » (*l'étudiant/apprendre/la biologie* : les étudiants apprennent la biologie)

2. Les segmentations incorrectes qui ne constituent pas une phrase acceptable :
   - « *tốc độ / truyền thông / tin / tăng* » (*la vitesse / la communication / l'informatique / accélérer*)
   - « *học sinh / học sinh / học* » (*l'étudiant/l'étudiant/la biologie*)
   - « *học / sinh học / sinh học* » *(apprendre/la biologie/la biologie*)

Avec ce type de segmentation, l'ambiguïté peut être détectée avec l'analyse grammaticale de la phrase.

Par contre, toutes les segmentations des cas (c) et (d) sont correctes, avec des sens qui peuvent être bien différents selon la segmentation choisie:

- « *chiếc / xe / đạp / nhẹ / lắm* » (*[classificateur,* voir la section 3.4.1*]/le véhicule/pédaler/léger/très* : le véhicule (normalement le vélo) pédale très légèrement) ;
- « *chiếc / xe đạp / nhẹ / lắm* » (*[classificateur]/le vélo/léger/très* : le vélo est léger) ;
- « *nhà / này / không / có / người / ở* » (*la maison/ce/ne…pas/avoir/personne/vivre* : personne ne vit dans cette maison) ;
- « *nhà / này / không / có / người ở* » (*la maison/ce/ne…pas/avoir/la servante* : il n'y pas de servante dans cette maison).

Le traitement d'un corpus de textes vietnamien nécessite donc de résoudre le problème de la segmentation en mots. Les méthodes de segmentation en mots à base d'un vocabulaire sont utilisées largement pour des langues asiatiques non segmentées [Berment 2004]. Cette méthode utilise un algorithme de programmation dynamique qui analyse la phrase de gauche à droite, recherche dans un vocabulaire les mots du texte et segmente cette phrase en optimisant un critère quelconque. Ce critère est choisi en fonction de la stratégie de segmentation « plus longue chaîne d'abord » (*Longest Matching*) ou « plus petit nombre de mots » (*Maximal Matching*). La stratégie de segmentation « plus petit nombre de mots » est implémentée dans [Nguyen T.B. 2004]. Dans le cas d'une ambiguïté, une décision humaine est nécessaire. Cette stratégie est aussi combinée avec la technique d'automates d'états finis et l'analyse d'expression régulière dans [Le H.P. 2008] pour créer l'outil *vnTokenizer*[1].

[Nguyen C.T. 2007] utilisent des techniques d'apprentissage statistique fondées sur les CRF (en anglais : *Conditional Random Fields*) et les SVM (en anglais : *Support Vector Machines*) pour identifier les limites de mots (l'outil JVnSegmenter[2]). Une autre méthode appliquée au vietnamien consiste à utiliser un algorithme d'apprentissage à base d'un réseau de neurones combiné avec un modèle à base d'automates d'états finis [Dinh D. 2001]. Cependant, ces méthodes statistiques nécessitent de disposer d'un grand corpus de textes segmenté au préalable.

3.4. Grammaire

Le vietnamien est une langue isolante où tous les mots restent invariables quelle que soit leur fonction syntaxique. Il n'y a ni conjugaison des verbes, ni accord des noms, adjectifs, etc. [Nguyen T.M.H. 2006]. Nous pouvons citer d'autres langues isolantes comme le chinois, le lao, le thaï, etc.

Ces langues sont souvent opposées aux langues flexionnelles, agglutinantes ou polysynthétiques. Pour les langues flexionnelles (comme le latin), les mots changent de forme selon leur rapport

[1] http://www.loria.fr/~lehong/tools/vnTokenizer.php
[2] http://jvnsegmenter.sourceforge.net/

grammatical (genre, nombre, fonction syntaxique, classe lexicale, temps, mode, etc.), ce qui crée pour chaque mot un ensemble de lemmes. Les langues agglutinantes et les langues polysynthétiques, d'autre part, appliquent des affixes et des suffixes sur le radical pour changer la fonction d'un mot.

Dans une langue isolante, les fonctions syntaxiques et les sens grammaticaux sont manifestés par des mots outils et l'ordre des mots dans la phrase. Les nuances sont généralement rendues par le contexte et l'intonation. Par exemple en vietnamien : « *tôi đánh nó* » (*je/battre/il* : je le bats), « *nó đánh tôi* » (*il/battre/je* : il me bat). Ou, pour dire « je vais le battre », nous disons « *tôi sẽ đánh nó* » (*je/[marqueur de temps au futur,* voir la section 3.4.1*]/battre/il*), le mot « *sẽ* » étant un mot outil représentant le futur.

3.4.1. Classification des mots

Des mots dans une langue sont classifiés en se basant sur des critères syntaxiques et sémantiques. Les mots d'une classe possèdent les mêmes caractéristiques grammaticales. D'une façon générale, nous pouvons distinguer neuf classes de mots en vietnamien : les noms (ou substantifs), les verbes, les adjectifs, les jonctions, les classificateurs, les pronoms, les marqueurs, les quantificateurs, les *modus* [Nguyen L.T. 2010]. Dans cette section, nous présentons seulement les particularités spécifiques à la langue vietnamienne.

1. Les classificateurs constituent une particularité de la langue vietnamienne.

En général, tous les substantifs doivent être précédés par un classificateur propre (de même que les articles précédent les noms en français). Les deux classificateurs principaux en vietnamien sont « *cái* » et « *con* ». Le mot « *cái* » sert à classifier les objets ou choses inanimés par contre le mot « *con* » sert à classifier les êtres animés. Nous disons :

- « *cái nhà* » (la maison), « *cái bàn* » (la table), « *cái quần* » (le pantalon) ;
- mais : « *con trâu* » (le buffle), « *con bò* » (le bœuf), « *con người* » (l'homme).

Dans certains cas, pour insister sur le caractère d'activités ou de déplacement, dans certains objets non animés, nous remplaçons « *cái* » par « *con* ». Par exemple : « *con tàu* » : le train (le train qui va et vient), « *con sông* » : le fleuve (le fleuve qui coule).

Il y a aussi d'autres classificateurs comme « *chiếc* », « *bức* » [« *bức tranh* » (la peinture), « *bức vẽ* » (le dessin)], « *quyển* » [« *quyển sách* » (le livre), « *quyển báo* » (le journal)], etc. parmi lesquels « *chiếc* » est le plus important et quelques fois, il peut remplacer le mot « *cái* » : « *chiếc bàn* » = « *cái bàn* » ; « *chiếc quần* » = « *cái quần* » ; mais nous ne disons pas « *chiếc nhà* » .

Quelquefois, « *cái* » peut apparaître immédiatement avant « *con* », mais le contraire est impossible : « *cái con chó này* » (*[classificateur]/[classificateur]/chien/ce* : ce chien (ou ce chien là), par contre « *con cái chó này* » n'est pas grammaticalement correct.

2. Les marqueurs constituent une autre particularité du système lexical en vietnamien. C'est un élément grammatical participant à la formation des syntagmes (surtout le syntagme verbal) sans jamais en être le noyau. Ils forment la classe des mots grammaticaux (s'oppose aux mots lexicaux, dont le sens est aussi important que le rôle syntaxique : les noms, les adjectifs, les verbes et les adverbes)). Les marqueurs en vietnamien remplacent les adverbes de temps, de lieu, de degré, de comparaison, de négation, certains verbes dits "auxiliaires", les adjectifs démonstratifs, etc. [Nguyen L.T. 2010]. Ci-dessous les principaux marqueurs en vietnamien :

Marqueurs de

- Démonstration : « *này* » (ce), « *ấy* » (ce), « *nọ* » (celui là), « *kia* » (celui là), etc. Exp. « *cái áo sơ-mi này đẹp* » (*[classificateurs]/habit/chemise/ce/beau* : cette chemise est belle)

- État : « *rồi* » (fini), « *xong* » (fini), « *vẫn* » (encore, toujours), etc. Exp. « *trời vẫn mưa* » (*ciel/toujours/pleuvoir* : il pleut toujours)
- Négation : « *không* » (non, ne … pas), « *chẳng* » (sans), « *chưa* » (pas encore), etc. Exp. « *chẳng có tiền* » (*sans/avoir/argent* : sans argent)
- Affirmation : « *có* » (oui). Exp. « *tôi có đi* » (*je/oui/aller* : oui, je vais)
- Ordre, souhait, impératif : « *hãy* » (à faite d'abord), « *đừng* » (ne pas), « *chớ* » (ne pas), « *phải* » (devoir), « *cần* » (nécessiter), « *đi* » (ϕ), etc. Exp. « *ra đi !* » (*sortir /*ϕ : sors !) ; « *đừng chạy !* » (*ne pas/marcher* : ne pas marcher !)
- Espace : « *đấy* » (là), « *đó* » (là), « *kia* » (là-bas), « *trên* » (dessus), « *dưới* » (dessous), « *ngoài* » (extérieur), « *sau* » (derrière), etc. Exp. « *cô ấy không ở đấy* » (*elle/ce/ne…pas/être/là* : elle n'est pas là)
- Temps : ce sont des mots qui expriment le sens grammatical de temps. « *đã* » (temps passé), « *sẽ* » (temps futur), « *đang* » (être en train de), « *vừa* » (venir juste de), « *sắp* » (qqch/qqun va venir bientôt), etc. Exp. « *thầy giáo đang đến* » (*le professeur/ être en train de /venir* : le professeur est en train de venir)
- Degré : « *rất* » (vraiment), « *khá* » (assez), « *hơi* » (un peu), « *lắm* » (beaucoup), « *quá* » (très), etc. Exp. « *con chó này rất ác* » (*[*classificateur*] / chien / ce / vraiment / méchant* : ce chien est très méchant)
- Résultat : « *nổi* » (être capable de), « được » (avoir la possibilité de), etc. Exp. « *tôi không về được* » (*je/ne…pas/rentrer/avoir la possibilité* : je ne peux pas rentrer)
- Réciprocité : « nhau », « lẫn nhau » (l'un l'autre), etc. Exp. « *họ đánh nhau* » (*ils/battrer/l'un l'autre* : ils se battent)
- Comparaison : « cũng » (aussi), « đều » (également), etc. Exp. « *họ cũng làm việc* » (*ils/aussi/travailler* : ils travaillent aussi)

3. Les *modus* constituent la troisième particularité de la langue vietnamienne. Ils regroupent toutes les interjections, les mots expressifs d'emploi très fréquent en vietnamien.

Les éléments servant à indiquer l'attitude du locuteur : une surprise, un doute, le respect, l'ironie, etc. ou encore une affirmation particulière se placent à la fin de la phrase :

- « *ư* » : phrase interrogative rhétorique, exprime la surprise ou l'ironie.
- « *ạ* » : phrase affirmative, exprime le respect
- « *phỏng* » : phrase interrogative rhétorique, exprime la menace.
- « *nhỉ* » : phrase interrogative, demande de partager une opinion.
- « *nhé* » : phrase interrogative, demande un acquiescement.
- « *sao* » : phrase interrogative souvent indirecte, exprime le doute.
- « *mà* » : phrase affirmative, exprime la conséquence
- « *ơi* » : appellation, exprime la gentillesse

Par contre, quelques *modus* se placent en tête de la phrase. Ils servent en général à exprimer les cris de joie, de douleur, de plainte, de déception, d'injure, etc.

- « *ái chà* » (oh là là) ; « *trời ơi* » (mon dieu) ; « *chao ôi* », « ôi » (hélas) ; « *ô* » (oh), « *a* » (ah)
- « *vâng* », « *dạ* » (oui) ; « *không* », « *ứ ừ* » (non) ;
- « *này* » (tiens, tenez) ; « *nào* » : par exemple « *đi nào* » (*aller* : allons, allez)

4. Les pronoms personnels. Les pronoms personnels en vietnamien sont utilisés pour exprimer non seulement la personne et le nombre, mais aussi la relation sociale ou familiale entre le locuteur et la personne désignée, ainsi que le sentiment du locuteur (Tableau 3-5). Par exemple, tous ces mots « *tôi, tao, cháu, con, em, anh/chị, cô/chú, bác, etc.* » peuvent être utilisés pour dire « *je* ». En outre, plusieurs de ces mots comme « *bạn, ấy, cậu, đằng ấy, mày, etc.* » peuvent être également utilisés pour dire « tu » et « *ông/bà* », pour dire « vous ».

5. Quantificateur. Les quantificateurs sont des mots qui se placent avant le nom noyau (avec ou sans classificateur) d'un syntagme nominal. Les quantificateurs peuvent être les nombres cardinaux ou les mots indiquant des quantités. Ci-dessous quelques exemples de quantificateurs vietnamiens :

Tableau 3-4 : Quelques exemples de quantificateurs vietnamiens

Quantificateur	Explication	Quantificateur	Explication
một, hai, etc.	nombres cardinaux : un, deux, etc.	*mấy*	quelques, combien, beaucoup
vài, dăm, vài ba, dăm bảy	quelques, environ trois, environ cinq	*bao nhiêu*	plusieurs
mọi	tout	*những*	pluriel indéfini
mỗi/từng	chaque	*các*	pluriel défini
nhiều	beaucoup	*ít*	peu

Par exemple :

- « *ba đêm* » (*trois/nuit* : trois nuits) ; « *dăm bữa* » (*quelque/jour* : quelques jours) ; « *nhiều người* » (*beaucoup/personne* : beaucoup de personnes).

Tableau 3-5 : Quelques pronoms vietnamiens

Quelques pronoms	Nous parlons avec	relation exprimée
Je :		
- *tôi*	personnes du même rang	formel
- *tao*	amis, personnes plus jeunes	familial
- *cháu*	personnes plus âgées, grands-parents, tantes/oncles	formel/familial
- *con*	parents, professeurs	familial
- *em*	personnes plus jeunes	formel/familial
- *anh/chị*	personnes plus âgées, mais de peu	formel/familial
- *cô/chú/bác*	petits enfants, les jeunes	formel/familial
Tu :		
- *bạn/ấy/đằng ấy/cậu*	personnes du même rang, amis	familial
- *mày*	amis, personnes plus jeunes	familial
- *cháu*	petits enfants, les jeunes	formel/familial
- *con*	enfants	familial
- *em*	personnes moins âgées, mais de peu	formel/familial
- *anh/chị*	personnes plus âgées, mais de peu	formel/familial
- *cô/chú/bác*	personnes plus âgées, tantes/oncles	formel/familial
Vous :		
- *ông/bà*	personnes plus âgées, grands-parents	formel/familial

6. Syntagmes nominaux

Les syntagmes nominaux contiennent un nom noyau et des modificateurs qui comprennent les classificateurs, les démonstratifs, les quantificateurs, les modificateurs attributifs, etc. La structure d'un syntagme nominal vietnamien est la suivante :

totalité + article ou quantificateur + classificateur + **nom noyau** + modificateur attributif + démonstratif + modificateur possessif

Par exemple :

cả	*hai*	*cuốn*	*từ điển*	*Việt Anh*	*này*	*của*	*nó*
tous	deux	[classificateur]	dictionnaire	vietnamien anglais	[marqueur démonstratif]	de, à	[pronom de la 3ème personne]
totalité	quantificateur	classificateur	nom noyau	modificateur attributif	démonstratif	modificateur possessif	
« ses deux dictionnaires vietnamien-anglais (à lui) »							

3.4.2. Syntaxe

Comme d'autres langues de la région, la syntaxe vietnamienne est conforme à l'ordre des mots *Sujet – Verbe – Objet*. Les prépositions, les classificateurs et les chiffres précèdent le nom ; les syntagmes nominaux du possesseur et les adjectifs suivent le nom.

Par exemple :

- « Mai là sinh viên » (*Mai/être/étudiant* : Mai est étudiante) ;
- « Giáp có ba quyển sách » (*Giap/avoir/trois/livre* : Giap a trois livres) ;
- « tôi thích ngựa đen » (*je/aimer/cheval/noir* : j'aime le cheval noir).

Dans les langues isolantes comme le vietnamien, le « mot » conserve toute son autonomie linguistique. La flexion est impossible. Chaque mot possède une forme unique qui n'est jamais modifiée. Chaque mot en vietnamien possède un certain sens selon son ordre dans la phrase. Modifier l'ordre des mots implique souvent que le sens de la phrase est modifié. Cela signifie que la grammaire repose sur l'ordre des mots et la structure des phrases plutôt que sur la morphologie.

Par exemple :

- « một vài *khó khăn* » (*quelque/difficulté* : quelques difficultés) : « *khó khăn* » est un substantif ;
- « rất *khó khăn* » (*très/difficile* : très difficile) : « *khó khăn* » est un adjectif ;
- « nó *suy nghĩ* nhiều » (*il/réfléchir/beaucoup* : il réfléchit beaucoup) : « *suy nghĩ* » est un verbe ;
- « những *suy nghĩ* của nó » (*les/réflexion/de/il* : ses réflexions) : « *suy nghĩ* » est un substantif ;
- « đó là một gia đình *hạnh phúc* » (*ce/être/un/famille/heureux* : c'est une famille heureuse) : « *hạnh phúc* » est un adjectif ;
- « *hạnh phúc* của tôi » (*bonheur/de/je* : mon bonheur) : « *hạnh phúc* » est un substantif ;
- « anh *cho* em cuốn sách này » (*je/donner/tu/livre/ce* : je te donne ce livre) : « *cho* » est un verbe ;
- « anh gửi cuốn sách này *cho* em » (*je/envoyer/livre/ce/à/tu* : je t'envoie ce livre) : « *cho* » est une conjonction « à » ;

Nous présentons un exemple très intéressant de l'influence de l'ordre de mots sur la signification d'une phrase. À partir de cinq mots : « *sao* » (pourquoi), « *nó* » (il), « *bảo* » (dire), « *không* » (ne pas), « *đến* » (venir), nous pouvons construire des phrases différentes :

- « *sao nó bảo không đến?* » : pourquoi dit-il qu'il ne vient pas ?
- « *sao bảo nó không đến?* » : pourquoi quelqu'un dit qu'il ne vient pas ?
- « *sao không đến bảo nó?* » : pourquoi ne l'appelles-tu pas ?
- « *sao nó không bảo đến?* » : pourquoi ne dit-il pas qu'il vient ?
- « *sao? đến bảo nó không?* » : comment ? on vient de lui dire ?
- « *sao? bảo nó đến không?* » : comment ? on va l'appeler ?

Une autre remarque que l'on peut faire en vietnamien, est le fait que l'omission des pronoms est permise dans certaines circonstances, et la présence de mots interrogatifs ne modifie pas l'ordre habituel des mots de la phrase.

- « *khi nào tới ?* » (*quand/venir* : l'omission des pronoms) ou « *khi nào cậu tới ?* » (*quand/tu/venir* : l'ordre de mots n'est pas modifié) : quand viens-tu ?

Quelquefois, il n'y a pas de sujet impersonnel en vietnamien. Par exemple, pour dire « il fait très froid ici », nous disons tout simplement « *ở đây rất lạnh* » (*ici/très/froid*).

3.5. Etat de l'art de la traduction automatique pour la langue vietnamienne

Le vietnamien peut être considéré du point de vue du TALN et de la traduction comme une langue peu dotée à cause de sa présence limitée sur le Web[1] et du manque de ressources pour le TALN. Bien que l'histoire du TAL commence dans les années 50 avec beaucoup de recherches et d'applications sur le traitement des langues telles que l'anglais, le chinois, l'arabe, le français, etc., le traitement automatique de la langue vietnamienne fait juste ses premiers pas. En ce moment, au Vietnam, on compte peu de groupes de recherche sur le traitement automatique de la langue vietnamienne.

En 2005, il y avait deux groupes de recherche principaux qui obtiennent des résultats encourageant sur la TAL vietnamien : l'Institut de la Technologie de l'Information[2] et le Centre de recherche international MICA[3] [Ho T.B. 2005]. L'Institut de la Technologie de l'Information concentre ses recherches sur la synthèse de la parole, la reconnaissance vocale, la reconnaissance optique de caractères, la classification de documents, le résumé automatique de texte, la recherche d'information et la fouille de textes. Le centre MICA se concentre sur le traitement des signaux complexes (parole et image), la synthèse de la parole, la reconnaissance vocale, l'extraction d'informations extralinguistiques, essentiellement transportées par la prosodie. Récemment deux groupes de recherche commencent les travaux sur la traduction automatique de la langue vietnamienne. Il y a d'autres groupes de recherche qui relèvent des universités, des instituts de recherche, les entreprises, etc.

Cependant, les activités sur le TAL vietnamien présentent les caractéristiques suivantes [Ho T.B. 2005]:

- Il y a peu de recherches fondamentales
- Il manque des outils et des ressources de données indispensables pour le TAL tels que les analyseurs morphologiques, les étiqueteurs, les dictionnaires électroniques, corpus de données, etc.
- Il y a peu de coopérations, de partage entre les groupes de recherche.

Jusqu'à 2010, il n'y a eu que deux projets nationaux concernant la Recherche et le Développement en Reconnaissance, Synthèse et Traitement de la Langue Vietnamienne : le projet *KC.01.03* de 2001 à 2005 [Khang B.H. 2004] et le projet *KC.01.01* de 2006 à 2010[4]. Les résultats de ces projets qui concernent le traitement de texte comprennent les outils d'analyse de la phrase vietnamienne (outil de segmentation en mots, étiqueteurs grammaticaux) ; le dictionnaire électronique, un corpus parallèle bilingue anglais – vietnamien de 100 000 paires de phrases dans le domaine de l'économie, du social et des technologies de l'information[5]; et le premier système de traduction à transfert pour la paire de langue anglais – vietnamien (EVTRAN, voir au dessous).

Retournons sur l'état de l'art de la traduction automatique pour la langue vietnamienne. Le premier système de TA de la langue vietnamienne est le système de « Logos Corporation » des

[1] En 2008, le nombre d'hôtes de l'Internet au vietnam est 84 151, en comparaison avec 14 256 000 hôtes en France, 22 606 000 hôtes en Allemand, et 316 000 000 hôtes aux Etats-Unis. (src : CIA The World Factbook http://www.cia.gov/cia/publications/factbook/index.html)
[2] http://www.ioit.ac.vn/
[3] Centre de recherche international MICA, L'Institut de Polytechnique de Hanoi - CNRS/UMI2954 Grenoble INP. http://www.mica.edu.vn/
[4] Le site du bureaux du gouvernement : http://kc01.vpct.gov.vn/Default.aspx?tabid=83&News=257&CategoryID=170
[5] http://vlsp.vietlp.org:8080/demo/?page=resources

années 1970. Ce système a été développé pour traduire des manuels militaires de l'anglais vers le vietnamien lors de la guerre au Vietnam [Hutchins 2001]. Le système pouvait traduire quelques millions de mots techniques. Cependant, le développement de ce système de traduction de l'anglais vers le vietnamien s'est arrêté en 1973, et malheuresement ce système a été perdu.

Au Vietnam, avec le développement des technologies de l'information, la recherche sur la TA de la langue étrangère (l'anglais) vers le vietnamien a démarré à la fin des années 1980. Cependant, jusqu'à présent, on compte peu de groupes de recherche travaillant sur la TA vietnamien – anglais/français et les résultats obtenus par les systèmes sont modestes. Dans le rapport de l'atelier ADD sur le TALN des langues asiatiques[1], [Le H.P. 2006] indique qu'au Vietnam il n'y a que quatre groupes de recherches principaux sur la TA pour la langue vietnamienne en 2006.

Le premier groupe vient du centre national pour le progrès technologique *Nacentech*. Le groupe a présenté le premier système commercial de traduction de la langue anglaise vers la langue vietnamienne en 1997 avec le nom EVTRAN. Le groupe a participé au projet *KC.01.03* avec la tâche de traduction automatique anglais – vietnamien mais après il est devenu une compagnie privée (*Softex*). EVTRAN est basé sur l'approche de traduction à transfert avec des règles d'analyse, de transfert et de génération [Le K.H. 2003a, b][2]. Cependant le groupe ne publie plus les détails de ses recherches et des ressources utilisées[3].

Le deuxième groupe qui vient du Département de Technologie de l'Information à l'Université des Sciences Naturelles d'Ho Chi Minh. Le groupe suit l'approche de traduction à transfert aussi. Au début, le groupe construit un corpus parallèle bilingue anglais – vietnamien à partir de diverses sources bilingues (textes dans le domaine scientifique). Le corpus contient 400 000 bi-phrases (5 millions de mots) [Dinh D. 2002] qui sont alignés en mots/groupe de mots et étiquetés par fonction grammaticale [Dinh D. 2003a]. Les systèmes de traduction anglais – vietnamien sont construits. Les règles de transfert de l'anglais vers le vietnamien sont apprises à partir du corpus parallèle ci-dessus par les techniques d'apprentissage automatique [Dinh D. 2003b, 2004]. Les règles extraites sont utilisées dans le module de transfert d'un système de traduction à transfert, ou elles sont utilisées pour prétraiter les phrases d'entrée d'un système de traduction probabiliste (l'ordre des mots et la structure de la phrase entrée en langue source sont convertis vers ceux en langue cible, avant que la phrase d'entrée soit traduite par le système de traduction probabiliste) [Nguyen T.H.N. 2008], [Vu H. 2008]. Cependant, le groupe de recherche reconnaît qu'il utilise un corpus d'apprentissage limité.

Le troisième groupe vient de l'université Polytechnique de la ville d'Ho Chi Minh. Le système de traduction utilise un modèle de transfert de mot à groupe de mots (word-to-phrase transfer model) pour résoudre le problème qu'un mot en vietnamien peut être traduit vers un groupe de mots en langue source cible (anglais). Le groupe a proposé plusieurs méthodes pour résoudre le problème et augmenter la qualité de traduction [Hai L.M. 1997, 2009, 2010].

Le quatrième groupe vient en fait de l'Institut des sciences et technologies *JAIST* du Japon. Il se concentre sur les problèmes d'ordre des mots et l'analyse morphosyntaxique pour la traduction anglais – vietnamien. La phrase à traduire est analysée par un module de traitement linguistique avant qu'elle n'entre dans un système de traduction automatique probabiliste par groupe de mots. Un corpus d'apprentissage de 24 000 paires de phrases est utilisé (extraites à partir des livres de grammaires anglais, les livres informatiques) [Nguyen T.P. 2007, 2008]. Dans un autre travail du groupe, le module de prétraitement est utilisé pour normaliser le corpus, segmenter des mots et

[1] Workshop on Asian applied natural language processing for linguistics Diversity and language resource Development. http://www.tcllab.org/modules.php?name=events&file=ADD
[2] Actuellement, ce système est déployé au site http://vdict.com/#translation
[3] Une autre compagnie qui développe le produit commercial de traduction anglais – vietnamien est *Lac Viet*, mais il ne publie pas aussi ses recherches et ses données. http://tratu.vietgle.vn/hoc-tieng-anh/dich-van-ban.html

faire un étiquetage morphosyntaxique. Ensuite, le module de prétraitement est combiné avec un système de traduction automatique probabiliste par groupe de mots [Ho T.B. 2008].

La recherche sur la TA vietnamien – français est encore plus rare. [Doan N.H. 2001] a proposé un module de traduction pour le vietnamien dans ITS3, un système de TA multilingue, développé au Laboratoire d'Analyse et de Technologie du Langage (LATL) à Genève, fondé sur l'approche à transfert. Cependant ce travail n'est pas terminé.

Nous trouvons aussi le travail de Nguyen M.C. sur la traduction japonais – vietnamien, qui se concentre sur la différence entre la structure adnominale japonaise (qui fonctionne comme un nom) et l'expression correspondante en vietnamien [Nguyen M.C. 2005, 2006].

En conclusion, nous voyons que la recherche sur la TA de la langue vietnamienne est assez rare et isolée. Des techniques proposées sont évaluées avec des petits corpus de données et dans des domaines limités (textes dans le domaine scientifique, les romans bilingues, etc.). C'est pourquoi dans la suite, nous présentons notre contribution sur la construction de corpus parallèles pour des langues peu dotées, en appliquant notre méthodologie au vietnamien.

Partie II. Contributions

Chapitre 4 : Extraction de corpus parallèles – Motivation

L'approche de traduction automatique probabiliste (chapitre 2) et la disponibilité des outils prêts à l'emploi permettent de construire rapidement un système de traduction automatique avec des données d'apprentissage parallèles suffisantes.

Pour la traduction depuis ou vers une langue peu dotée, ce type de corpus d'apprentissage parallèle n'existe pas toujours, ou bien il n'existe qu'avec une quantité faible de données, qui n'est pas suffisante pour apprendre des modèles probabilistes robustes. Ainsi, une des difficultés de la construction d'un système de traduction probabiliste pour une langue peu dotée réside dans la constitution de ces corpus parallèles, étape indispensable à l'apprentissage des modèles probabilistes.

Pour collecter les données de texte, le World Wide Web est une source possible. Il contient des textes de plusieurs langues accessibles librement en grande quantité. Les chercheurs en traitement automatique du langage utilisent de plus en plus le Web comme source de données linguistiques [Kilgarriff 2003]. Plusieurs travaux de recherche se sont concentrés sur la construction de corpus de textes monolingues en grande quantité en collectant des pages Web. Nous pouvons citer, par exemple, les travaux de [Fletcher 2004] ou [Baroni 2004], et pour les langues peu dotées [De Schryver 2002] qui a montré que le Web pouvait être une source de texte potentielle pour les langues africaines. [Ghani 2001] propose de former des requêtes sur le Web pour collecter des documents en une langue peu dotée (le slovène, le tchèque). Enfin, [Scannell 2007] a présenté un projet appelé Crúbadán[1] pour créer des corpus de textes pour un grand nombre de langues peu dotées (dans la version 2, il y a 487 langues), etc.

4.1. Les sites Web multilingues

Le Web n'est pas seulement une source de données monolingues, mais aussi une source de textes multilingues. Par exemple, les sites dans le domaine « *.de* » peuvent contenir des données bilingues allemandes – anglaises, le domaine « *.ca* » peut comprendre des données bilingues

[1] Projet Crúbadán http://borel.slu.edu/crubadan/

françaises – anglaises [Ma 1999]. Il existe aussi des pages Web qui possèdent des liens vers d'autres pages en d'autres langues. Par exemple, si une page Web en anglais a un lien avec le texte « *Espagnol* » ou « *en Espagnol* », cette page Web et la page liée par ce lien peuvent contenir des données bilingues espagnoles – anglaises [Resnik 1998]. Pour localiser les pages Web bilingues, [Chen 2000] ont proposé d'envoyer des requêtes particulières aux moteurs de recherche. Par exemple, deux requêtes « *anchor : « english version » [« in english » , ...]* » et « *anchor : « chinese version » [« in chinese» , ...]* » sont envoyées à un moteur de recherche (*anchor* est l'élément HTML qui représente un lien hypertexte). Deux ensembles de documents, l'un en anglais et l'autre en chinois, sont retournés. L'insertion ou l'union de ces deux ensembles est considérée comme des sites Web candidats, c'est-à-dire contenant potentiellement des données parallèles.

Plus simplement, des données bilingues peuvent être trouvées dans les sites Web des organisations internationales (par exemple : les sites Web de l'ONU[1], l'OMS[2], l'UNESCO[3], etc. qui contiennent des documents en 6 langues, le site Web de la Commission européenne[4] contient des documents en 23 langues), et l'encyclopédie libre Wikipédia[5], propose une quantité raisonnable d'articles pour une trentaine de langues. Mais les données multilingues les plus intéressantes restent les articles de sites Web multilingues de nouvelles journalistiques grâce à leur quantité énorme, leur actualisation permanente et leur grand nombre de domaines et de paires de langues. Par exemple : la BBC[6] propose des contenus en 32 langues, l'AFP[7] affiche 6 langues, la VOANews[8] environ 40 langues, etc. Alors que le Web continue de se diversifier et de croître, même les sites Web de langues peu dotées commencent à apparaître en grand nombre.

Une fois que les sites bilingues ou multilingues sont trouvés, les données textuelles de ces sites Web sont téléchargées et stockées dans une base de données afin de constituer un corpus. Les documents et les phrases parallèles doivent alors être extraits à partir de ce corpus. Dans cette thèse, nous ne nous concentrons pas sur la recherche des sites Web multilingues à partir du Web, mais nous nous focalisons sur l'extraction de documents parallèles et de phrases parallèles à partir d'un site Web multilingue de nouvelles journalistiques donné.

4.2. Corpus parallèle et corpus comparable

Les textes récupérés à partir du Web ne contiennent pas toujours des données parallèles, c'est le cas si les données sont récupérées à partir d'un site Web multilingue de nouvelles journalistiques.

Un corpus de données bilingues « parallèle » est un corpus qui contient des paires de documents bilingues ou des paires de phrases bilingues qui sont la traduction directe l'un de l'autre. Les deux phrases (ou documents) parallèles sont souvent de longueur similaire. L'ordre des phrases parallèles est maintenu dans deux documents parallèles [Fung 2004a]. Une phrase dans la langue *L1* est souvent traduite par une phrase dans la langue *L2* (nous appelons cela une correspondance *1:1*). La plupart des phrases de deux documents parallèles est alignée *1:1*. Il y a peu de

[1] Organisation des Nations unies (ONU) http://www.un.org
[2] Organisation mondiale de la santé (OMS) http://www.who.int
[3] Organisation des Nations unies pour l'éducation la science et la culture (UNESCO) http://www.unesco.org
[4] Commission européenne (EC) http://ec.europa.eu/
[5] http://fr.wikipedia.org
[6] British Broadcasting Corporation (BBC) http://www.bbc.co.uk/
[7] Agence France-Presse (AFP) http://www.afp.com/afpcom/en/
[8] Voice of America (VOA) http://www.voanews.com/english/news/

correspondances *1:n*, *n:n* et peu de suppressions *1:0* ou *n:0* dans un corpus bilingue parallèle [Wu 1994].

La source de données extraites à partir de sites Web multilingues de nouvelles journalistiques ne peut être considérée comme un véritable corpus parallèle. On parlera plutôt de corpus parallèle bruité ou même de corpus comparable. En fait, dans la littérature, la notion de « *corpus comparable* » est assez vague. D'une manière générale, la communauté de chercheurs considère qu'un corpus comparable contient des documents qui ne sont pas des traductions l'un de l'autres, mais « étroitement liés par les mêmes contenus » aux « niveaux de parallélisme différents, tels que des mots, des chaînes de mots, des phrases, etc. » [Zhao 2002], [Fung 2004a, b], [Kumano 2007].

Plus concrètement, [Fung 2004a, b] ont défini un corpus *comparable* (qu'ils appellent d'autre fois corpus *« parallèle bruité »*) comme un corpus qui contient des phrases non alignées mais dont la plupart sont des traductions bilingues d'un même morceau de document. Les paires de documents dans ce corpus sont des traductions approximatives l'un de l'autre, avec des insertions et des suppressions dans le contenu, et portent sur les mêmes sujets. L'ordre des phrases dans les deux documents est presque similaire. Des exemples de ce type de corpus sont le corpus de « *Hong Kong News* » ou celui de « *Xinhua News* » [Fung 2004a, b]. Un autre type de corpus défini par [Fung 2004a, b] est le corpus *quasi-comparable* ou *très non parallèle (very-non-parallel corpus).* Ce type de corpus contient des documents dans deux domaines différents, avec l'existence d'une quantité considérable de documents hors sujets. Très peu de documents sont comparables ou contiennent des phrases parallèles.

[Munteanu 2006a] définit quant à lui divers types de corpus comparables avec plusieurs niveaux de parallélisme possibles. Le premier type de corpus comparable est constitué de corpus dont les documents dans une langue sont, soit entièrement traduits dans l'autre langue, soit ne possèdent pas de correspondance. Un exemple peut être illustré par les articles d'information de la revue « *Le Monde Diplomatique* », où certains articles sont traduits en plusieurs langues, tandis que d'autres sont spécifiques à chaque région et n'existent que dans une seule langue. Le deuxième type de corpus comparable défini contient des documents qui peuvent être soit traduits, soit partiellement traduits mais partageant des phrases parallèles, ou encore non traduits. Deux exemples sont le corpus de « *Xinhua News* », le corpus de « *Agence France Presse* ». Dans ces corpus, la plupart des données parallèles peuvent être trouvées au niveau de la phrase. L'ordre des phrases n'est pas toujours respecté. Enfin, le dernier type de corpus comparable défini est constitué de corpus qui présentent peu de parallélisme au niveau du document ou de la phrase, mais, par contre, du parallélisme au niveau des chaînes de mots (des fragments). L'auteur donne un exemple avec des articles de nouvelles produites par la « *BBC »*. Une paire d'articles rapporte le même événement d'un même instant mais il y a peu ou pas de paires de phrases complètement parallèles, il y a seulement certains fragments parallèles.

Pour être plus clair, nous définissons dans cette thèse divers niveaux de parallélisme selon la granularité de texte considérée (voir les Figure 4-1, Figure 4-2, Figure 4-3).

Premièrement, au niveau de la phrase, nous définissons :

- *les phrases parallèles* : deux phrases sont la traduction l'une de l'autre ;
- *les phrases comparables* : deux phrases ne sont pas exactement la traduction l'une de l'autre, mais elles contiennent des fragments parallèles ;
- *les phrases non parallèles* : deux phrases sont des phrases sans rapport l'une avec l'autre.

Au niveau du document, nous définissons :

- *les documents parallèles* : toutes les phrases dans ces deux documents sont des phrases parallèles, et l'ordre des phrases est respecté

- *les documents parallèles bruités* : la plupart des phrases dans ces deux documents sont des phrases parallèles, l'ordre des phrases peut être respecté ou non ; il peut y avoir des insertions ou des suppressions de phrases dans un document par rapport à un autre
- *les documents comparables* : les deux documents contiennent peu de phrases parallèles, mais quelques phrases comparables ; l'ordre des phrases peut être similaire ou différent ; ces documents peuvent contenir aussi certaines phrases non parallèles
- *les documents non parallèles* sont des documents qui ne contiennent pas de données parallèles.

Au niveau du corpus, nous définissons :

- *le corpus parallèle* : contient des documents parallèles. Un sous type de *corpus parallèle bruité* est le corpus parallèle mais il est bruité « un peu » par d'autres types de paires de documents.
- *le corpus comparable* : contient tous les types de documents mais possède une majorité de documents parallèles, parallèles bruités et comparables.

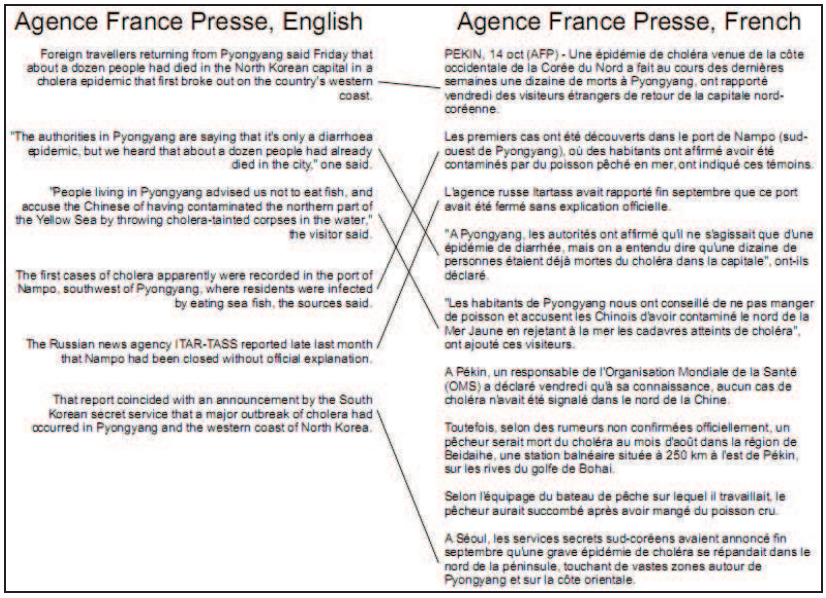


Figure 4-1 : Exemple d'une paire de documents parallèles bruités ; les lignes présentent les phrases parallèles (image originale dans [Munteanu 2006a]).

Figure 4-2 : Exemple d'une paire de documents comparables ; les lignes et les blocs présentent les fragments parallèles (image originale dans [Munteanu 2006a]).

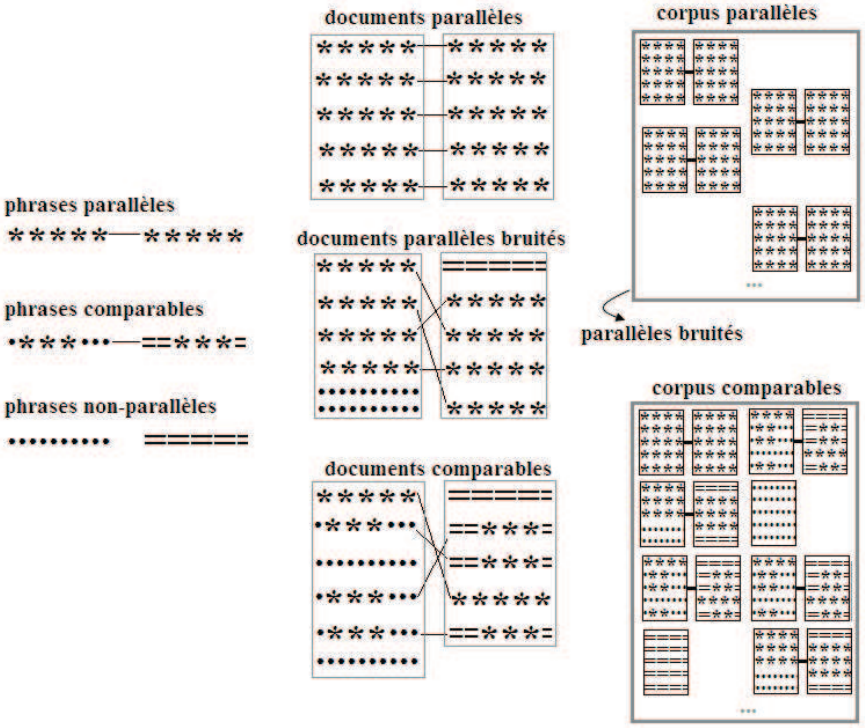


Figure 4-3 : Résumé de nos divers niveaux de parallélisme selon la granularité du texte considéré

Deux articles de nouvelles (deux documents) dans deux langues différentes qui décrivent le même événement peuvent être des documents parallèles. Mais ils sont produits fréquemment de façon indépendante, donc ils peuvent ne pas être des traductions directes l'une de l'autre car ils peuvent contenir des phrases non traduites dans l'autre langue ou des phrases placées dans un ordre différent. Cependant, ces deux articles sont susceptibles de contenir des données parallèles (des phrases, des fragments, des mots) qui expriment le chevauchement du contenu. Alors ils peuvent constituer des documents comparables.

La tâche d'extraction des documents comparables/parallèles et des phrases comparables/parallèles à partir de corpus comparables est plus difficile qu'à partir de corpus parallèles ou parallèles bruités. En plus, dans les documents comparables, les phrases de deux documents ne sont pas souvent alignées selon les correspondances *1:1* ; elles peuvent être alignées selon les correspondances *1:n* ou *n:n*. Il y a aussi des suppressions ou des insertions dans une paire de textes comparables [Ma 2006].

4.3. Les méthodes proposées pour extraire les données parallèles à partir d'un corpus comparable

La meilleure façon d'extraire les données parallèles à partir d'un corpus comparable est encore une question ouverte qui a reçu beaucoup d'attention de la part de la communauté de chercheurs dans les années récentes. Plusieurs méthodes sont proposées pour extraire (ou chercher) à partir de ce type de corpus les documents comparables [Resnik 1999], [Resnik 2003], etc. ; les phrases parallèles [Zhao 2002], [Fung 2004a], [Munteanu 2005], etc. ; et des fragments ou des dictionnaires bilingues [Munteanu 2006b], [Quirk 2007], [Cettolo 2010], etc. Les méthodes de recherche peuvent être regroupées de la façon suivante :

- les approches de recherche qui utilisent des caractéristiques générales des documents (telles que le titre du document, le lien du document, l'information dans le lien, la structure du document, etc.) ainsi que des informations lexicales du document ;
- les approches qui utilisent des techniques de recherche d'information translingue (*cross-language information retrieval*) nécessitant un module de traduction.

Dans nos travaux, nous nous concentrons sur l'extraction de documents parallèles et de phrases parallèles pour construire un grand corpus d'apprentissage afin de développer un système de traduction probabiliste dans le cas d'une langue peu dotée. Les deux approches ci-dessus sont considérées. Au point de départ, nous n'avons qu'un seul site web multilingue de cette langue peu dotée, nous n'avons pas de données parallèles disponibles ou de données supplémentaires. Les trois chapitres suivants présentent nos contributions et méthodes proposées, qui sont au nombre de trois.

La première méthode concerne la première approche pour extraire à la fois les documents comparables et les phrases parallèles. Cette méthode requiert des données supplémentaires sur la paire de langues (telles qu'un dictionnaire bilingue, une liste de mots d'arrêt, etc.) Lorsque ces données supplémentaires sont disponibles, nous pouvons appliquer cette méthode pour toute paire de langues.

La deuxième méthode suit l'approche de recherche d'information translingue mais nous proposons une méthode non supervisée qui peut s'appliquer dans le cas de langues peu dotées (où l'application directe de la technique de recherche d'information translingue est impossible à cause des données requises en entrée qui ne sont pas disponibles). Le processus d'extraction ne requiert aucune donnée supplémentaire comme dans la première méthode. Cette méthode peut être appliquée pour toute paire de langues, pour un corpus comparable en entrée seulement, et aucune donnée supplémentaire n'est requise sur les deux langues considérées.

La dernière méthode aborde l'utilisation d'une troisième langue dans l'extraction du corpus comparable bilingue. C'est une application de l'approche de triangulation, introduite pour la traduction automatique, au processus d'extraction de données parallèles. L'approche de triangulation a été utilisée récemment pour augmenter la qualité d'un système de traduction. Nous proposons d'utiliser une troisième langue dans le processus d'extraction des données parallèles à partir d'un corpus comparable.

Les détails de ces approches sont présentés dans les chapitres suivants avec des expériences appliquées sur la langue peu dotée vietnamienne et les langues française et anglaise.

Chapitre 5 : Recueillir rapidement des ressources et construire un premier système de traduction

Comme mentionné dans l'introduction, le premier objectif de notre travail était de construire un système de TA probabiliste pour les langues peu dotées. Une des difficultés réside dans la constitution des corpus parallèles indispensables à l'apprentissage des modèles qui ne sont pas toujours disponibles, en particulier pour les langues peu dotées.

Nos premiers travaux concernent la construction d'un système de traduction probabiliste pour le couple de langues vietnamien – français dans le domaine des nouvelles journalistiques. Nous nous concentrons sur l'extraction d'un corpus bilingue de nouvelles journalistiques vietnamien – français collecté à partir du Web.

Généralement, le processus d'extraction d'un corpus de textes bilingues pour la traduction automatique se décompose en cinq étapes [Koehn 2005] : la collection de données brutes, l'alignement des documents, la segmentation en phrases, la normalisation des écritures, l'alignement des phrases. Les paires de documents parallèles sont d'abord déterminées et ensuite les paires de phrases parallèles sont identifiées.

Ce chapitre présente les méthodologies générales d'extraction d'un corpus bilingue de texte pour obtenir un corpus de phrases parallèles. Nous présentons une vue d'ensemble des méthodes de recherche qui utilisent des caractéristiques générales avec des informations lexicales des documents. Parce que nous étudions une langue peu dotée aux ressources limitées et parce que nos sources de données sont bruitées, nous suggérons une nouvelle méthode combinant des caractéristiques générales et des informations d'alignement de lexiques pour identifier simultanément des paires de documents et des paires de phrases parallèles extraites à partir d'un corpus comparable.

Nous exposons alors les résultats d'expériences sur l'exploitation automatique d'un site Web multilingue de nouvelles journalistiques vietnamien – français et présentons également la construction d'un premier système de traduction automatique probabiliste vietnamien – français et français – vietnamien à partir de ce corpus. Nous discutons finalement l'opportunité d'utiliser des unités lexicales ou sous lexicales pour le vietnamien (syllabes, mots, ou leurs combinaisons).

5.1. Techniques d'alignement à base de caractéristiques générales et d'informations lexicales

Cette section présente une vue d'ensemble des méthodes de recherche basées sur les caractéristiques générales et les informations lexicales, pour aligner des documents et des phrases. Nous les partageons en des méthodes d'alignement de documents et des méthodes d'alignement de phrases.

5.1.1. Alignement de documents

Pour aligner deux documents à partir d'un corpus de textes, plusieurs approches sont proposées. Tout d'abord, nous pouvons mentionner les approches basées sur des caractéristiques générales du document telles que l'adresse URL, le titre, la structure, le nombre de paragraphes du document, les longueurs des deux documents, etc.

[Resnik 1998] a présenté un système d'extraction de données STRAND (*Structural Translation Recognition for Acquiring Natural Data*) avec une technique très simple pour extraire les documents parallèles à partir du Web. Cette technique est basée sur l'hypothèse que deux documents parallèles possèdent les adresses URLs semblables ou une mise en page équivalente. Le fait est que les adresses URLs de plusieurs pages Web parallèles sont parfois nommées de façon similaire pour faciliter la maintenance du site, par exemple, les adresses dans un même site Web multilingue anglais – français peuvent s'écrire : « *http://..../index.en* » et « *http://..../index.fr* », ou bien « *http://..../en/program.html* » et « *http://..../fr/program.html* ». De plus, le(la) concepteur(-trice) de sites Web peut créer les pages Web bilingues d'un même contenu avec une même apparence. L'auteur a proposé que si les adresses URLs du document *D1* et du document *D2* présentent d'étroites similitudes, il est possible que les documents *D1* et *D2* soient des traductions mutuelles. Ensuite, la structure HTML du document est considérée. Un score de similarité est calculé basé sur les étiquettes HTML non correspondantes et les longueurs des segments entre ces étiquettes, et un seuil est utilisé pour identifier les paires de documents parallèles. Le système PTMiner (*Parallel Text Miner*) de [Chen 2000] a utilisé aussi les adresses URLs de documents, avec d'autres critères tels que le rapport de la longueur des documents, la proportion de paires de phrases qui contiennent des éléments d'un dictionnaire bilingue, etc.

Le titre d'article est considéré comme un résumé condensé du document. [Yang 2002] a donc suggéré la comparaison des titres des documents pour trouver des paires de documents. Une autre méthode de comparaison de la structure de documents a été proposée par [Shi 2006] qui utilisent le modèle DOM (*Document Object Model*) pour représenter les deux pages Web comme une paire d'arbres de DOM. Ensuite, un modèle d'alignement d'arbres stochastique est utilisé pour aligner les textes équivalents.

Toutefois, les informations basées sur l'adresse URL, le titre, ou la structure de document ne sont pas des informations stables. Les approches basées sur ces caractéristiques générales ne sont pas toujours performantes, surtout avec la grande diversité de styles que présentent les pages Web. C'est pour cela que d'autres techniques, basées quant à elles sur des informations lexicales, ont été proposées.

L'algorithme BITS (*Bilingual Internet Text Search*) a été développé par [Ma 1999]. Les similarités entre un document *D1* dans la langue *L1* et un document dans la langue *L2* sont calculées en se basant sur le rapport du nombre de paires de mots ayant des traductions mutuelles et le nombre de mots de *D1* (les traductions mutuelles sont déterminées par un dictionnaire bilingue). Le document *D2* qui est le plus semblable au document *D1* est sélectionné à partir d'un ensemble de documents dans la langue *L2*. Lorsque la similarité entre *D1* et *D2* est

supérieure à un seuil donné, deux documents sont déclarés comme appartenant à une paire de traductions. Pour améliorer l'efficacité de l'algorithme, avant de calculer la similarité entre des paires de documents, quelques filtrages simples sont appliqués pour filtrer les paires candidates, tels que des filtrages basés sur la taille du document, le nombre de paragraphes de chaque document, le nombre de mots « identité » (les mots qui ne sont pas changés après la traduction).

La présence de mots « identité » ou des mots invariants d'une langue à l'autre (comme des chiffres, des marques de ponctuation, et des noms d'entités nommées) est utilisée aussi dans [Patry 2005] comme un indice intéressant. Les auteurs ont représenté chaque document par un vecteur contenant des séquences de chiffres, des marques de ponctuation, et des noms d'entités nommées de ce document. Le degré de parallélisme entre deux documents est présenté par la mesure de cosinus et la distance d'édition normalisée entre deux vecteurs. Le système de classification est entraîné sur un ensemble d'entraînement pour identifier une paire de documents comme parallèles ou non.

Dans les travaux de [Zhao 2002], la probabilité que deux documents soient alignés est calculée selon un modèle de combinaison entre la longueur des documents (en mots ou en caractères) et un modèle de traduction IBM-1. Une paire de documents est considérée comme parallèle si sa probabilité calculée par ce modèle est supérieure à un seuil.

Il existe aussi des techniques de recherche d'information translingue qui reposent sur de la traduction, mais celles-ci seront présentées dans le chapitre suivant.

5.1.2. Alignement de phrases

L'alignement au niveau de la phrase entre deux documents est un ensemble de correspondances entre les phrases des deux documents. Plusieurs méthodes pour aligner automatiquement deux documents parallèles au niveau de la phrase ont été proposées. Ces méthodes peuvent utiliser des informations non lexicales, des informations lexicales ou combinent les deux.

La première technique utilise uniquement la longueur de la phrase (l'information non lexicale) et elle ne nécessite presque aucune connaissance préalable. Cette technique est proposée dans [Brown 1991] et [Gale 1993] et elle est basée sur l'idée principale que les phrases plus longues (respectivement plus courtes) dans une langue ont tendance à être traduites en phrases longues (respectivement plus courtes) dans l'autre langue. Un modèle statistique simple des longueurs de phrases en mots (comme dans les travaux de [Brown 1991]) ou en caractères (dans les travaux de [Gale 1993]) est utilisé. Un score probabiliste est attribué à chaque paire de phrases candidates, basé sur la différence des longueurs des deux phrases et la variance de cette différence. Avec ce score probabiliste, la technique de programmation dynamique est appliquée pour trouver l'alignement qui maximise la vraisemblance entre les phrases. Il permet de trouver la phrase longue qui est traduite en deux ou plusieurs phrases courtes. Parce que les techniques n'utilisent pas des détails lexicaux de la phrase, le calcul est rapide et donc pratique pour une application sur une très grande collection de textes. Cependant, ces techniques peuvent échouer dans les régions qui contiennent des petites suppressions ou des phrases d'une longueur similaire. En plus, ces techniques ne sont appropriées que pour aligner les phrases entre deux documents qui présentent un important degré de parallélisme au départ.

[Chen 1993] a utilisé des informations lexicales en construisant un modèle statistique simple de traduction mot à mot avec les paramètres initiaux appris à partir d'un corpus aligné. La probabilité d'un alignement est le produit des probabilités de toutes les correspondances en phrase entre deux documents. A son tour, la probabilité de chaque correspondance est calculée en fonction des paires de mots qui sont des traductions mutuelles. La technique de programmation dynamique est appliquée et l'alignement qui maximise la probabilité totale est

trouvé. Dans les travaux de [Tillmann 2009], le score de similarité entre deux phrases est calculé selon des probabilités de traduction dans les deux sens de traduction du modèle IBM-1 qui est appris à partir d'un corpus parallèle disponible.

L'hybridation entre le modèle de longueur de la phrase et le modèle lexical est proposée dans plusieurs travaux. [Wu 1994] a proposé d'utiliser la longueur de la phrase avec l'occurrence des « indices lexicaux », des paires de mots bilingues sélectionnées manuellement selon chaque corpus, dans deux documents. Dans [Moore 2002], l'auteur propose un processus d'extraction en trois étapes. Premièrement, le modèle de longueur de phrase issu de [Brown 1991] est utilisé pour trouver l'alignement initial. Puis, les paires de phrases les plus probables selon cet alignement sont choisies pour entraîner un modèle d'IBM-1. Finalement, le corpus est réaligné en modifiant l'alignement initial par le modèle à base de mots. Dans les travaux de [Zhao 2002], une fois que les documents parallèles sont trouvés, les auteurs alignent les phrases avec un critère de maximum de vraisemblance qui combine des modèles de longueur de phrases et un modèle de lexique extrait d'un corpus parallèle aligné existant. La technique de programmation dynamique est appliquée pour trouver le meilleur alignement. Un processus itératif est proposé pour réapprendre le modèle de lexique en utilisant les données extraites.

5.2. Notre première méthode d'extraction – Système de référence qui utilise quelques ressources disponibles

La première méthode que nous proposons utilise des informations lexicales pour extraire à la fois les documents et les phrases parallèles. Cette méthode requiert des données supplémentaires sur la paire de langues visée (le dictionnaire bilingue, la liste de mots d'arrêt[1] de deux langues, la liste de formes de surface de mots[2] de deux langues, etc.). Lorsque ces données supplémentaires sont disponibles, nous pouvons appliquer cette méthode pour toutes les paires de langues.

5.2.1. Nos hypothèses

Nous nous concentrons sur l'extraction de phrases parallèles à partir d'un corpus comparable. Nous supposons que les données textuelles sont récupérées à partir d'un site Web multilingue qui peut contenir des textes qui sont des traductions mutuelles les uns des autres. Cet ensemble de données de texte peut contenir des documents parallèles, non parallèles et aussi des documents parallèles bruités ou des documents comparables. L'objectif de notre travail étant de construire un système de TA probabiliste pour les langues peu dotées, nous nous concentrons sur l'extraction de toutes les phrases parallèles possibles à partir de toutes les paires de documents pertinents : parallèles, parallèles bruités ou comparables. Nous chercherons donc les trois types de paires de documents (à partir de maintenant nous les nommerons *paires de documents pertinents*), et nous extrayons les paires de phrases parallèles correspondantes.

Nous supposons que :

- il n'y a aucun lien existant entre deux documents pertinents d'une paire, même dans leur adresse URL ;
- les titres des deux documents d'une paire ne sont pas la traduction mutuelle l'un de l'autre;

[1] Les mots d'arrêt (*stop words*) sont des mots tellement communs qu'il devient inutile de les indexer ou de les utiliser dans une recherche. En français, des mots d'arrêt sont « *le* », « *la* », « *de* », « *du* », « *ce* », « *ça* », etc.
[2] Par exemple, les mots anglais « *run* », « *runs* », « *runing* », « *ran* » ont la même forme de surface « *run* »

- la paire de documents parallèles bruités ou comparables contient des suppressions ou insertions de phrases ou de paragraphes : les longueurs des deux documents d'une paire sont donc différentes ;
- il est possible que les phrases parallèles n'apparaissent pas dans le même ordre entre documents source et cible ;

Généralement, l'étape d'alignement de phrases est exécutée sur les paires de documents qui sont considérées comme parallèles à l'issue de l'étape d'alignement de documents. Cela signifie que deux étapes d'alignement de documents et de phrases sont consécutives. Cependant, les méthodes d'alignement de documents fondées sur des caractéristiques générales ne sont pas utilisables ici en raison notamment des hypothèses listées ci-dessus. Donc, pour aligner les paires de documents pertinents, nous préférons utiliser des informations lexicales. De plus, avec les hypothèses ci-dessus, le seul indice que nous pouvons utiliser pour aligner les phrases est celui lié aux informations lexicales. Ainsi, nous proposons d'intégrer l'étape d'alignement de phrases dans l'étape d'alignement de documents. En particulier, nous utiliserons les informations issues d'un alignement de phrases pour remettre en cause l'étape d'alignement de documents. La section suivante présente notre méthode d'extraction.

5.2.2. Notre méthode d'extraction utilisant des informations lexicales

L'entrée de notre processus d'extraction est constituée de deux corpus monolingues *S1* et *S2* de documents dans deux langues différentes *L1*, *L2*. Nous cherchons le document *D2*, dans l'ensemble *S2*, qui peut être apparié au document *D1*, dans l'ensemble *S1*. Notre méthode d'extraction se compose de trois modules (voir Figure 5-1).

Le premier module cherche les paires de documents candidats pour réduire l'espace de recherche. D'abord, la date de publication est utilisée pour limiter le nombre de documents possibles *D2*. Ensuite, nous utilisons un filtrage basé sur des mots spéciaux contenus dans le document pour déterminer les candidats *D2*. Les paires de documents candidats sont notées *{D1-D2}*.

Le deuxième module est le module d'alignement de phrases pour les paires de documents candidats. Comme la plupart des modules d'alignement de phrases, ce module utilise des informations lexicales de document. Nous proposons d'utiliser un outil d'alignement des phrases existant. Normalement, l'entrée du outil d'alignement est constituée de paires de documents parallèles (ou parallèles bruités). Mais ici nous alignons des documents candidats. Donc le résultat d'alignement peut nous donner des indices importants pour classifier des paires de documents.

La sortie du module d'alignement en phrases est constituée des ensembles *{Alignement-Phrases$_{D1-D2}$}*, chaque ensemble contient toutes les paires de phrases alignées (les correspondances) « *phrase1–phrase2* » pour chaque paire de documents candidats *{D1-D2}*. Nous appelons « *phrase1–phrase2* » une correspondance du type *m : n* quand *phrase1* contient *m* phrases consécutives et *phrase2* contient *n* phrases consécutives. Le troisième module utilise les résultats de l'alignement en phrases pour re-estimer les paires de documents pertinents et pour filtrer les paires des phrases alignées afin de récupérer des paires de phrases parallèles. Le détail de chaque module est donné dans les sous-sections suivantes.

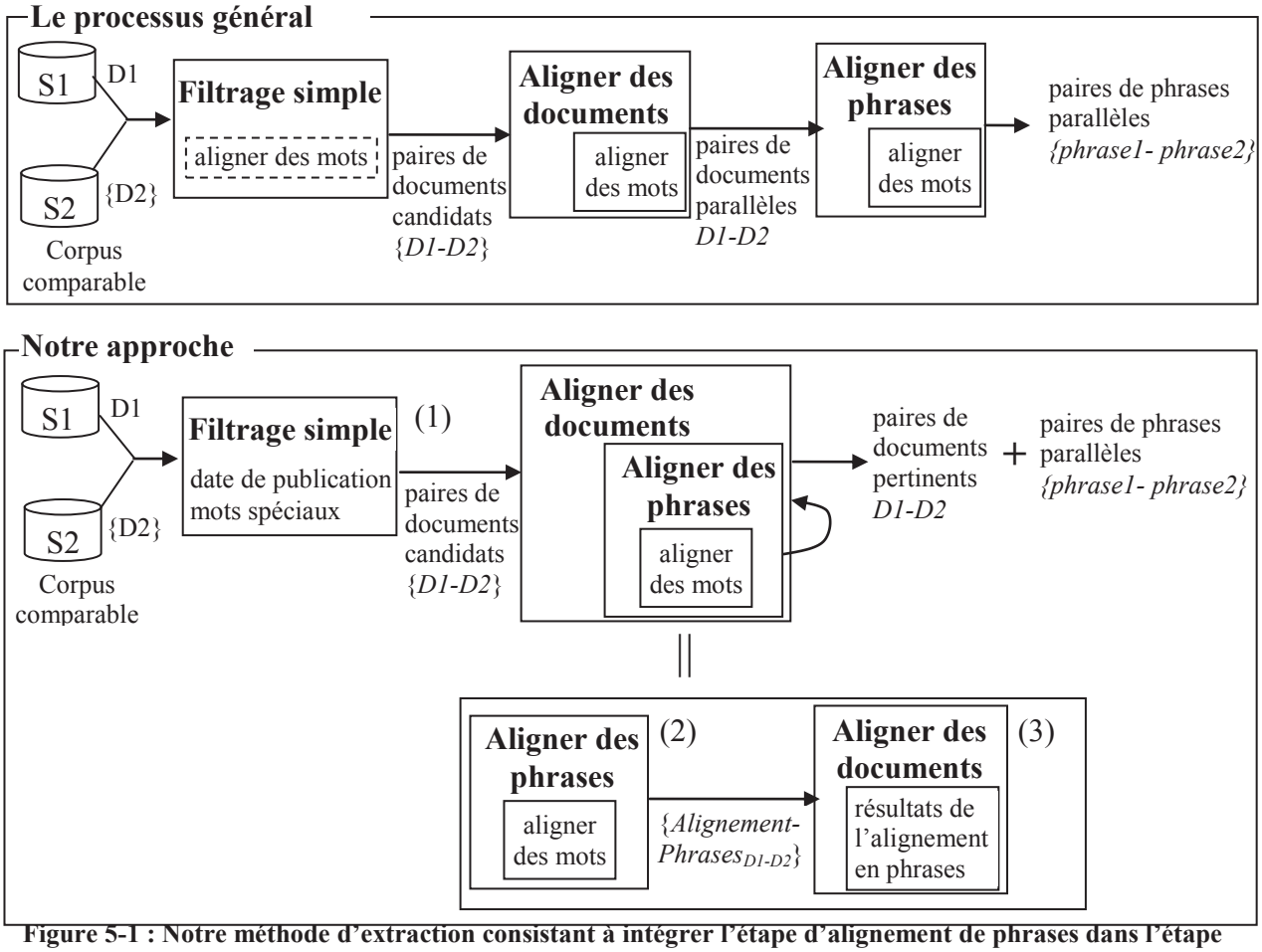


Figure 5-1 : Notre méthode d'extraction consistant à intégrer l'étape d'alignement de phrases dans l'étape d'alignement de documents.

5.2.2.1 Premier module : deux filtrages simples utilisant la date de publication et les mots spéciaux

Pour réduire le nombre de paires de documents candidats, nous appliquons deux filtrages simples sur le corpus comparable. Nous supposons que le document est traduit et publié au plus *d* jours après la date de publication du document original. Dans la mesure où nous ignorons si *D1* ou *D2* est le document original, nous supposons que le document *D2* est publié *d* jours avant ou après le document *D1*. Après ce filtrage, nous obtenons un ensemble *S2'* contenant un jeu de documents possibles de *D2* (la Figure 5-2).

Nous définissons les mots spéciaux comme étant les mots qui sont invariants d'une langue à l'autre, tels que des nombres, des symboles et des noms propres. Non seulement les nombres, mais aussi les symboles joints ('*$*', '*%*', '*‰*', '*.*', '*,*', etc.) sont extraits à partir des documents, par exemple : « *12.000 $* », « *13,45* », « *50 %* », etc. Les nombres et les symboles joints sont normalisés pour être comparés entre deux langues. Par exemple, le nombre « *13,45* » écrit en français est normalisé par « *13.45* » pour être comparé avec le nombre écrit en anglais. Mais il n'est pas normalisé pour être comparé avec le nombre écrit en vietnamien où la virgule indique aussi les décimales.

En plusieurs langues, les noms propres sont précisés par une chaîne de mots dans laquelle tout mot commence par une lettre majuscule, par exemple, « *Paris* », « *Nations Unies* » en français[1]. Le premier mot d'une phrase est commencé par une lettre majuscule aussi, mais il n'est pas toujours le nom propre. L'utilisation d'un dictionnaire monolingue de mots communs nous permet de supprimer ces mots. Par exemple, en français, le mot « selon » qui existe dans le dictionnaire de mots communs sera supprimé dans le groupe de mots « Selon François Girault », et nous obtenons le nom propre « François Girault ». Bien que les noms propres dans la langue *L1* puissent être traduits en une autre chaîne de caractères dans la langue *L2*, il existe des cas où les noms propres dans la langue *L1* (tels que les noms de personnes ou les noms d'organisations) ne changent pas dans la langue *L2*. Pour prendre un exemple très simple, « *Nations Unies* » en français est traduit vers « *United Nations* » en anglais, mais « *Paris* », *« Microsoft »* ne sont pas changés en anglais. Dans le travail de [Patry 2005], chaque document est représenté par la fréquence et l'ordre des mots ou symboles invariants dans ce document. L'ordre des mots spéciaux dans les documents est considéré comme un critère important. Cependant, dans le cas d'un corpus comparable, la fréquence et l'ordre des mots dans deux documents pertinents ne sont pas toujours respectés à cause de la suppression ou de l'insertion. C'est pourquoi notre méthode ne prend pas en compte la fréquence et l'ordre des mots spéciaux dans les documents mais juste la présence d'un tel mot. Si un même mot spécial apparaît plusieurs fois dans un document, ceci n'affecte pas le résultat d'alignement de documents. Nous n'utilisons pas de marques de ponctuation de phrase (tels que la virgule, le deux-points, le point, les parenthèses, les guillemets, etc.) ; nous utilisons par contre les symboles joints avec les nombres.

À partir du document *D1*, tous les mots spéciaux sont identifiés et nous extrayons une liste de mots spéciaux $w_1, w_2,...,w_n$. Pour chaque mot w_i, nous recherchons dans l'ensemble *S2'* les documents *D2* qui contiennent ce mot, nous obtenons une liste de documents *D2*. Le document *D2* qui apparaît le plus dans toutes les listes est choisi. C'est le document contenant le plus grand nombre de mots spéciaux du document D1, en comparaison avec d'autres documents dans le corpus S2. Nous pouvons trouver zéro, un ou plusieurs documents qui satisfont cette condition. Nous appelons cet ensemble de documents *S2''*. La Figure 5-2 présente notre premier module avec deux filtrages simples utilisant la date de publication et les mots spéciaux.

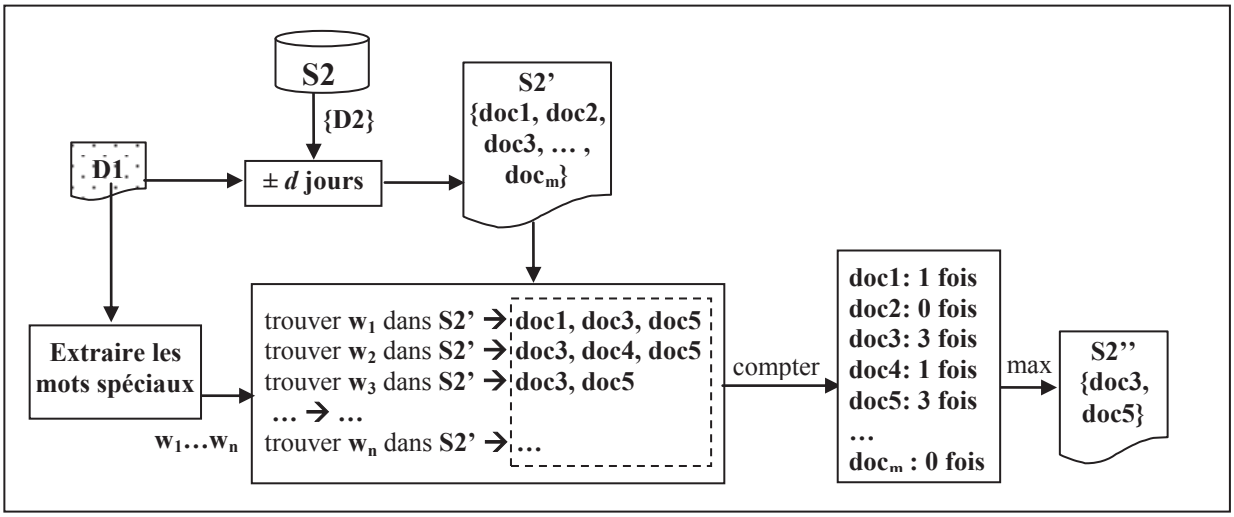


Figure 5-2 : Le premier module : utiliser la date de publication et les mots spéciaux

[1] Un nom propre peut être détecté lorsqu'une lettre majuscule est rencontrée derrière un espace blanc. Nous étendons ce nom propre jusqu'à ce que nous rencontrions une ponctuation ou une lettre minuscule devant un espace blanc. Nous envisageons de remplacer cette technique de détection très rudimentaire par une technique véritable dans le futur. Cependant, cette technique s'avère assez performante pour notre besoin.

5.2.2.2 Deuxième module : alignement en phrases

A cette étape, pour chaque document *D1*, nous avons déjà extrait un ensemble *S2''*, qui contient zéro, un ou plusieurs documents candidats *D2*. Un outil d'alignement des phrases existant est utilisé pour aligner chaque paire de documents *D1* et *D2*. La sortie est l'ensemble des paires de phrases alignées $\{Alignement\text{-}Phrases_{D1\text{-}D2}\} : D2 \in S2''$.

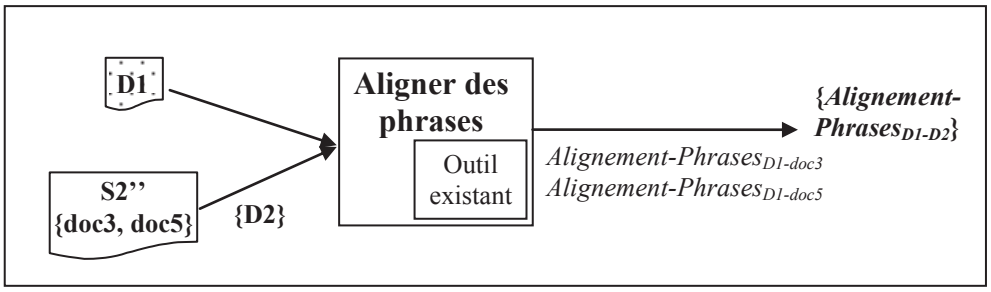


Figure 5-3 : Le deuxième module : alignement en phrases

Dans notre travail, nous utilisons l'outil d'alignement des phrases appelé *Champollion*[1], qui un est un aligneur à base de lexiques et il semble performant pour traiter les données parallèles bruitées [Ma 2006]. Un autre outil qui satisfait aux hypothèses de données bruitées pourrait cependant être utilisé.

Champollion diffère d'autres aligneurs de deux façons. Premièrement, il permet de travailler sur des documents parallèles bruités, avec un grand nombre de correspondances qui ne sont pas des correspondances *1:1*, et pour lesquels le nombre de suppressions et d'insertions est significatif. Pour le faire, il prend la décision sur la correspondance entre deux segments, chaque segment étant constitué de zéro ou plusieurs phrases (de 0 à 4 phrases) au lieu d'une seule. Donc il permet les correspondances de type *m:n* (*m, n = 0, 1, 2, 3, 4*). Les suppressions (insertions) sont présentées par les correspondances du type *0:n* (*m:0*).

La similarité entre deux segments est calculée en utilisant une mesure bien connue en recherche d'information, le poids TF-IDF (*term frequency - inverse document frequency*). Pour chaque paire de segments, les paires de mots et leur traduction mutuelle sont déterminées par un dictionnaire bilingue[2]. La fréquence d'un terme TF dans chaque segment et la fréquence inverse de document IDF sont calculées pour chaque paire de mots étant une traduction mutuelle l'un de l'autre, dans ces segments. En utilisant l'IDF, Champollion attribue des poids différents pour les paires de mots qui sont des traductions mutuelles (il attribue le poids inférieur à la paire de mots qui apparaît plus fréquemment, telles que les paires de mots français – anglais « *le – the* », « *un/une – a/an* », etc.). Dans la plupart des algorithmes d'alignement de phrases, les paires de mots sont traitées d'une manière égale. C'est la deuxième différence de l'outil Champollion.

La mesure TF-IDF est combinée avec une pénalité d'alignement et une pénalité de longueur pour estimer le score de similarité entre deux segments. La valeur de la pénalité d'alignement est égale à *1* avec des correspondances de type *1:1* et varie de *0* à *1* avec d'autres correspondances. La pénalité de longueur est fonction de la longueur du segment source et de la longueur du segment cible. Champollion utilise la technique de programmation dynamique pour trouver le meilleur alignement (l'ensemble des correspondances) qui maximise la similarité entre deux textes entre tous les alignements possibles.

[1] http://champollion.sourceforge.net. Voir plus en Annexe 1.
[2] Les mots sont normalisés avant par leur forme de surface, par exemple le mot français « *utilisées* » est normalisé par « *utiliser* »

Pour utiliser l'outil Champollion, plusieurs informations lexicales sont nécessaires telles que les listes de formes de surface de mots (pour la normalisation de texte) en deux langues, les listes de mots d'arrêt en deux langues, un dictionnaire bilingue, etc.

5.2.2.3 Troisième module : filtrer les paires de documents et de phrases en utilisant les résultats de l'alignement en phrases

La sortie du deuxième module est l'ensemble des paires de phrases alignées *{Alignement-Phrases$_{D1\text{-}D2}$} : D2∈S2''*. Le troisième module est utilisé pour filtrer les paires de documents ou de phrases de faible qualité. Nous ajoutons alors deux seuils α et β pour filtrer les documents *D2*.

Premièrement, pour chaque paire de documents *D1-D2*, le nombre de correspondances dans l'ensemble *Alignement-Phrases$_{D1\text{-}D2}$* est nommé *card(Alignement-Phrases$_{D1\text{-}D2}$)*. Le nombre de phrases qui ne peuvent pas trouver leur partenaire (quand « *phrase1-phrase2* » est une correspondance du type *0:m* ou *n:0*) est noté *nbr_abandonné (Alignement-Phrases$_{D1\text{-}D2}$)*.

Lorsque le rapport *[nbr_abandonné (Alignement-Phrases$_{D1\text{-}D2}$) / card(Alignement-Phrases$_{D1\text{-}D2}$)]* est supérieur à un seuil α , la paire de documents *D1-D2* est éliminée. Ce rapport représente relativement la couverture de contenu entre deux documents. Si le rapport est faible, la possibilité que deux documents soient alignés est grande. L'utilisation de ce seuil α nous permet de contrôler aussi les longueurs de deux documents lorsque les suppressions ou les insertions apparaissent.

$$\text{Le premier seuil}: \quad \frac{\text{nbr\_abandonné}(\text{Alignement - Phrases}_{\text{D1-D2}})}{\text{card}(\text{Alignement - Phrases}_{\text{D1-D2}})} > \alpha \Rightarrow \text{D1 - D2 est éliminé}$$

Deuxièmement, pour chaque paire de phrases « *phrase1-phrase2* » dans *Alignement-Phrases$_{D1\text{-}D2}$*, nous ajoutons deux scores x_{P1} et x_{P2} pour les phrases *phrase1* et *phrase2* :

$$x_{P1} = \frac{\text{Le nombre de mots traduits dans la phrase 1}}{\text{Le nombre de mots dans la phrase 1}} \quad ; \quad x_{P2} = \frac{\text{Le nombre de mots traduits dans la phrase 2}}{\text{Le nombre de mots dans la phrase 2}}$$

Les mots traduits sont les mots invariants ou les mots qui ont des équivalents dans l'autre phrase. L'identification de ces mots est simple grâce au fait qu'ils sont marqués dans la sortie du deuxième module. Les mots d'arrêt ne sont pas comptés ici, même dans le dénominateur.

Lorsque toutes les paires de phrases dans *Alignement-Phrases$_{D1\text{-}D2}$* ont les deux scores x_{P1} et x_{P2} qui sont plus petits que β, la paire de documents *D1-D2* est éliminée. Ce seuil élimine les paires de documents non pertinents qui génèrent des paires de phrases de faible qualité. Parce que nous traitons un corpus comparable, nous supposons que deux documents sont comparables lorsque ils contiennent au moins un tiers de paires de phrases alignées dans *Alignement-Phrases$_{D1\text{-}D2}$* qui ont les deux scores x_{P1} et x_{P2} plus grands que β. Donc le deuxième seuil pour filtrer les paires de documents est présenté comme suit :

$$\frac{\#\text{ de paires de phrases satisfaisant } [(x_{P1} \geq \beta) \text{ et } (x_{P2} \geq \beta)]}{\#\text{ de paires de phrases alignées}} < \frac{1}{3} \Rightarrow D1 - D2 \text{ est éliminé}$$

où $\#\text{ de paires de phrases alignées} = \text{card}(\text{Alignement - Phrases}_{\text{D1-D2}}) - \text{nbr\_abandonné}(\text{Alignement - Phrases}_{\text{D1-D2}})$

Après avoir utilisé ces trois modules, nous obtenons un corpus de paires de documents alignés, et aussi un corpus de paires de phrases alignées (les paires de phrases alignées dans *Alignement-Phrases$_{D1\text{-}D2}$* ont leurs deux scores x_{P1} et x_{P2} qui sont plus grands que β).

La seule référence correspondant à notre approche est [Munteanu 2006a] qui utilise des alignements en phrases pour considérer les paires de documents pertinentes, mais celle-ci utilise

des critères différents de nos critères, tels que le rapport de la longueur des documents, le pourcentage de phrases alignées (>30 %), le pourcentage de correspondances monotones (la correspondance qui n'est pas croisée par les autres) (>90 %). En plus, le processus d'extraction a le défaut de répéter plusieurs fois les comparaisons lexicales entre les documents (3 fois dans deux étapes d'alignement de documents et de phrases). Ces comparaisons semblent redondantes contrairement à notre approche.

5.3. Recueillir rapidement des ressources vietnamiennes – françaises

5.3.1. Collection et prétraitement des données

La langue peu dotée sur laquelle nous concentrons nos travaux est la langue vietnamienne. Comme présenté dans la section 3.3 et 3.4, l'unité de base de la langue vietnamienne est la syllabe. A l'écrit, les syllabes sont séparées par un espace. Un mot correspond à une ou plusieurs syllabes. Les fonctions syntaxiques sont également déterminées par l'ordre des mots dans la phrase.

Tableau 5-1 : Un exemple d'une phrase vietnamienne segmentée en syllabes et en mots

La phrase vietnamienne : Thành phố hy vọng sẽ đón nhận khoảng 3 triệu khách du lịch nước ngoài trong năm nay ***Segmentée en syllabes*** : Thành | phố | hy | vọng | sẽ | đón | nhận | khoảng | 3 | triệu | khách | du | lịch | nước | ngoài | trong | năm | nay ***Segmentée en mots*** : Thành_phố | hy_vọng | sẽ | đón_nhận | khoảng | 3 | triệu | khách_du_lịch | nước_ngoài | trong | năm | nay
La phrase française correspondant : La ville a prévu de recevoir 3 millions de touristes étrangers cette année

La plupart des mots vietnamiens présente des traductions correspondantes en d'autres langues. En revanche, les noms propres de langues telles que l'anglais, le français, etc., sont conservés en vietnamien. Heureusement, les noms en vietnamien sont souvent traduits dans d'autres langues en supprimant juste les diacritiques et les accents. Par exemple, le nom en vietnamien « *Nông Đức Mạnh* » est traduit en français « *Nong Duc Manh* » ; le nom « *Điện Biên* » est traduit « *Dien Bien* ».

Tableau 5-2 : Exemples de noms propres vietnamiens traduits en français

Le nom propre vietnamien	La traduction en français	Manière de traduire
Liên Hợp Quốc	*Nations Unies*	changé complètement
Pháp	*France*	
Michel Platini	*Michel Platini*	conservé
Paris	*Paris*	
Nông Đức Mạnh	*Nong Duc Manh*	suppression des diacritiques
Điện Biên	*Dien Bien*	

En vue de la construction d'un corpus de textes parallèles vietnamien – français, nous avons appliqué notre méthodologie d'alignement pour exploiter un corpus de textes à partir d'un site Web multilingue de nouvelles journalistiques, le Vietnam News Agency[1] (VNA). Ce site contient des articles de presse écrits en quatre langues (vietnamien, anglais, français et espagnol)

[1] http://www.vnagency.com.vn/. En 2009, il a été changé sur le domaine http://www.vietnamplus.vn/

et divisés en 9 catégories : Politique – Diplomatie, Société – Education, Economie – Finances, Culture – Sports, Sciences, Santé publique, Environnement, Région, International.

Chaque article est obtenu via un lien URL depuis le site Web de VNA. L'adresse d'un article a la forme http://www.vnagency.com.vn/Vnanetvn/FR/tabid/145/itemid/210000/Default.aspx. L'interface du site Web de VNA est présentée dans la Figure 5-4. Il n'y a pas de lien ou d'information dans les URLs pour découvrir la relation entre une paire d'articles. La partie d'URL « /FR/» n'identifie que la langue du contenu de trois rubriques : « INFO », « GRANDS TITRES », « AUTRES INFOS » dans l'interface du site Web, pas la langue du contenu d'article. Ces trois régions contiennent des informations telles que menus, références, annonces, publicités, etc., et elles sont répétées dans toutes les pages. La deuxième partie « 210000/Default.aspx » est l'identification (ID) de l'article affiché dans n'importe quelle langue. Pour obtenir tous les articles, nous avons changé l'ID de la deuxième partie de 0 à l'ID du dernier article et enlevé les documents vides.

Nous avons utilisé *GNU wget* pour récupérer à distance des pages à partir de ce site Web. *GNU wget* est un outil pour récupérer des documents binaires à partir du Web par l'utilisation du protocole HTTP et FTP, et pour les enregistrer sur le disque. Il peut récupérer des pages Web autant de fois que nécessaire, ou jusqu'à une limite spécifiée par l'utilisateur. Entre le 12 avril 2006 et le 14 août 2008, nous avons obtenu environ 121 000 documents dans les quatre langues. Pour chaque page obtenue, nous avons filtré les informations redondantes et nous ne gardons que le contenu. Chaque document contient en moyenne 10 phrases, avec environ 30-35 mots par phrase.

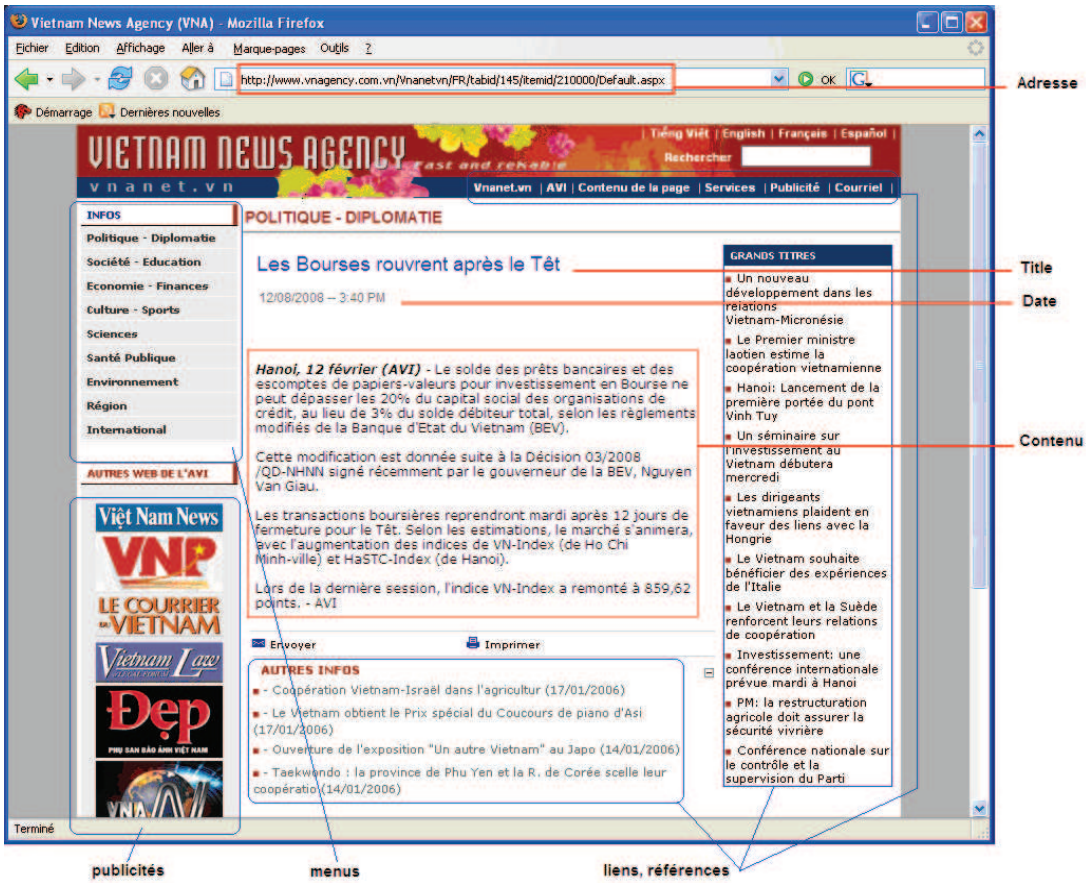


Figure 5-4 : L'interface d'une page dans le site Web « Vietnam News Agency »

Cependant, tous les articles vietnamiens ne sont pas forcément disponibles dans les trois autres langues. La répartition de la quantité de données dans les quatre langues est différente (40 % en vietnamien, 27 % en anglais, 20 % en français et 13 % en espagnol). En plus, nous pouvons

trouver manuellement des paires de documents parallèles, des paires de documents comparables, et aussi des documents qui ne possèdent pas de traduction. Ce type de corpus est vraiment un corpus comparable, car il a tendance à contenir des phrases parallèles ou des traductions approximatives des phrases sur les mêmes sujets. La Figure 5-5 et la Figure 5-6 présentent deux exemples d'une paire de documents parallèles bruités et d'une paire de documents comparables dans ce corpus de textes.

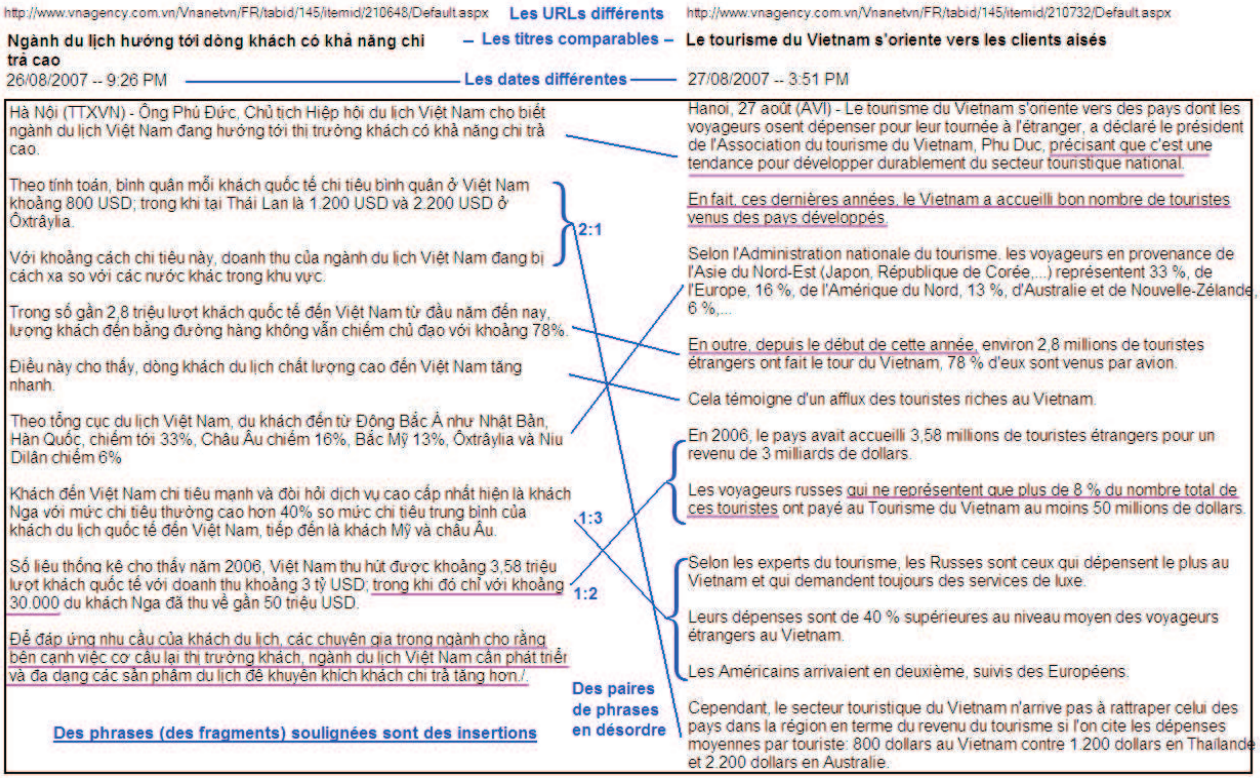


Figure 5-5 : Exemple d'une paire de documents parallèles bruités dans le corpus de textes de VNA

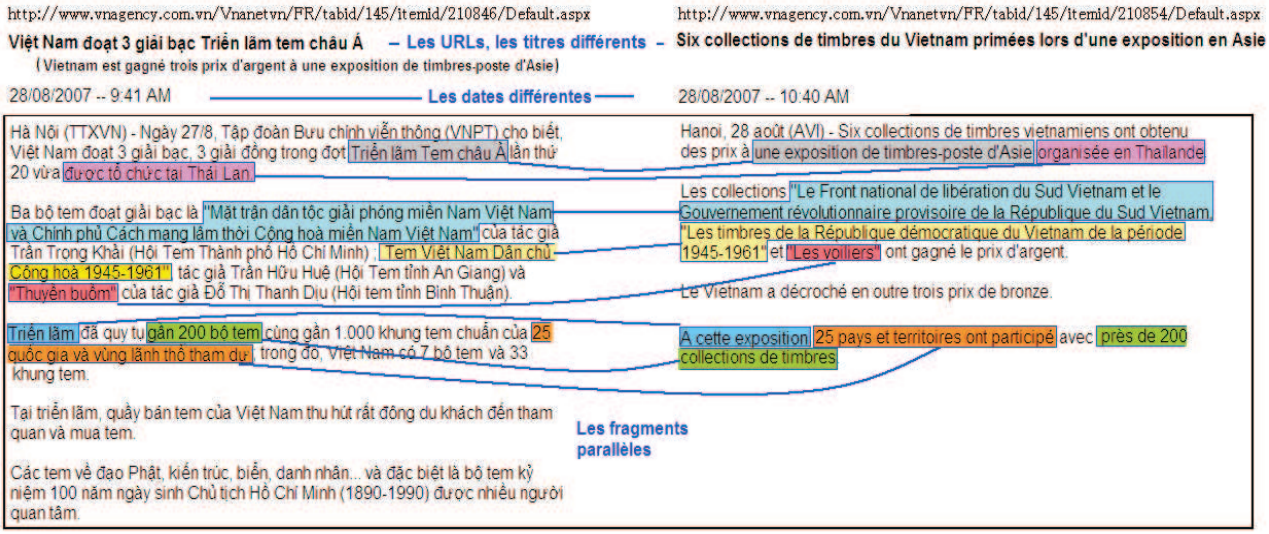


Figure 5-6 : Exemple d'une paire de documents comparables dans le corpus de textes de VNA

Nous avons séparé notre corpus de 121 218 documents dans les quatre langues en deux ensembles. Le premier ensemble contient 1 000 documents, nommé E_{1k}, dont l'appariement de documents correct français vietnamien est connu, suite à une annotation manuelle. Il a été utilisé pour régler les paramètres du système de filtrage décrit précédemment. Le reste des données est appelé E_{all}. Pour ce large corpus, les paramètres du système de filtrage, réglés sur E_{1k}, ont été

appliqués pour construire le corpus de phrases entier. Nous avons appliqué le processus de prétraitement ci-dessous pour les deux corpus de textes E_{1k}, E_{all} :

(1) pour chaque page obtenue, nous avons filtré les informations redondantes pour ne garder que le contenu ; nous avons éliminé aussi les petits signes au début de chaque document (par exemple les fragments « *Hà Nội (TTXVN)* - », « *Hanoi, ... (AVI)* - » dans la Figure 5-5 et la Figure 5-6) ;

(2) classer automatiquement les documents par langue, en utilisant l'outil TextCat[1], un outil d'identification des langues fondé sur les n-grammes de mots ;

(3) traiter et nettoyer les documents vietnamiens, français et anglais en utilisant l'outil CLIPS-Text-Tk, une boîte à outils générique « multilingue » de type « open source » de traitement d'un corpus de textes [Le V.B. 2003] ; afin de rendre les données recueillies sur le Web exploitables, l'outil CLIPS-Text-Tk nous permet de réaliser un certain nombre de traitements nécessaires :

1. transformation html vers texte ;
2. normalisation des tags et restructuration des documents ;
3. conversion des encodages ;
4. segmentation en phrases ;
5. segmentation en mots ;
6. transcription des caractères spéciaux, par exemple remplacement des éléments « ^{2} » par « carrés », « ^{o}C » par « degrés celcius », « ^{o} » par « degrés », « % » par « pour cent » ;
7. transcription des nombres : conversion des nombres par leur équivalent textuel (comme par exemple « 2 » qui devient « deux ») ;
8. enlèvement de la casse du caractère ;
9. suppression des marques de ponctuation ;
10. filtrage en fonction d'un vocabulaire donné.

Actuellement, l'outil supporte cinq langues : français, anglais, vietnamien, khmer et chinois. Nous avons utilisé les cinq premiers modules de l'outil pour convertir les documents html en documents textes, convertir le code des caractères, segmenter en phrases et en mots les documents français, vietnamiens et anglais. Des caractères spéciaux, des nombres, des ponctuations et la casse du caractère ont été gardés pour être utilisés dans notre méthode d'extraction.

Les corpus obtenus sont nommés S_{FR} pour le français, S_{VN} pour le vietnamien et S_{EN} pour l'anglais. Notre première méthode a été testée sur la paire de langues français – vietnamien.

[1] http://www.let.rug.nl/~vannoord/TextCat/

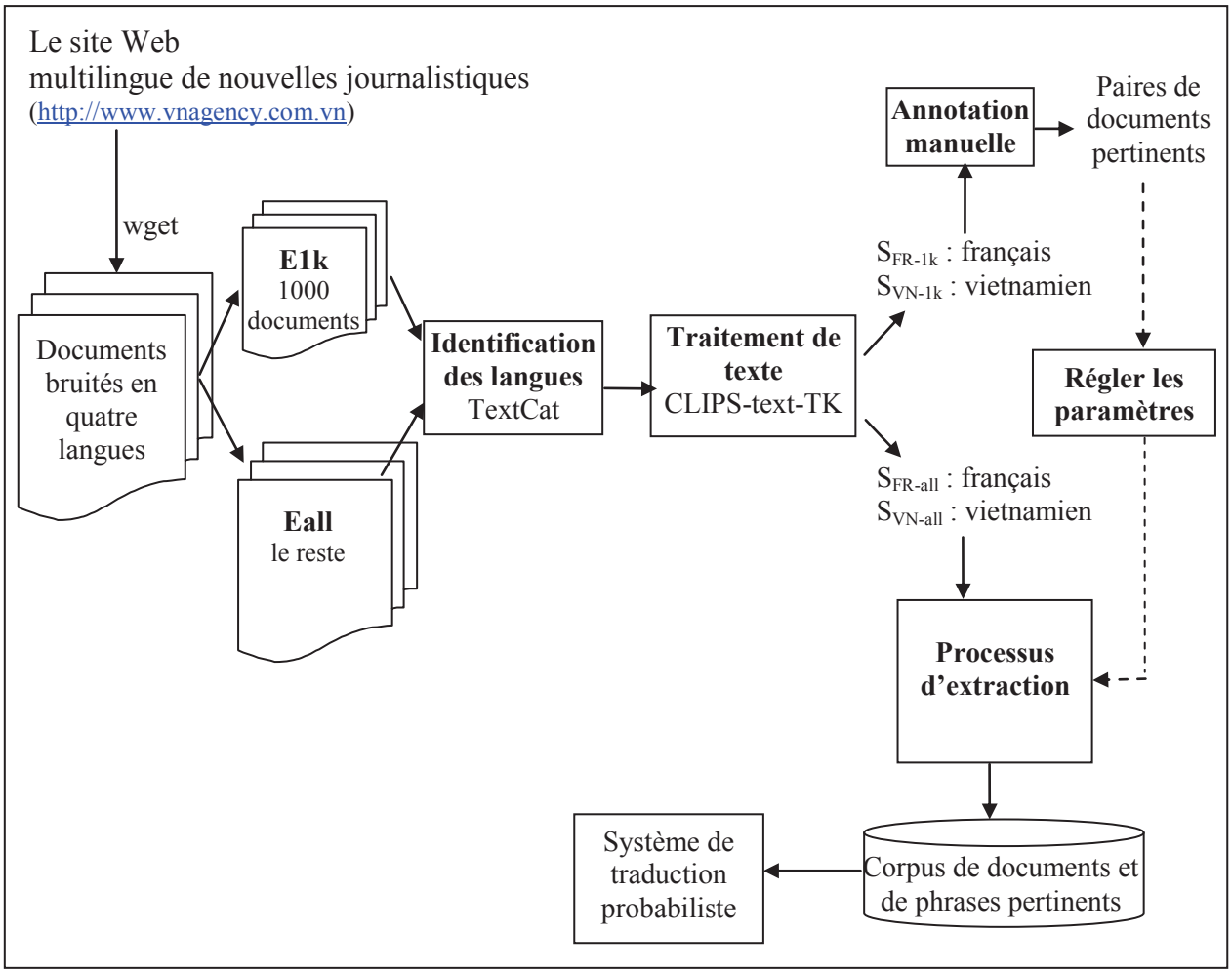


Figure 5-7 : Nos expériences pour la première méthode d'extraction

5.3.2. Réglage des paramètres de filtrage sur E_{1k}

Les valeurs absolues de α et β sont dépendantes du corpus comparable utilisé : nous devons donc estimer ces paramètres pour chaque corpus utilisé.

Notre méthode d'extraction a été appliquée sur les corpus $S_{FR\text{-}1k}$ et $S_{VN\text{-}1k}$ qui sont extraits à partir de l'ensemble E_{1k}. A partir de ces 1 000 documents, nous avons obtenu 173 documents français et 348 documents vietnamiens. Une annotation manuelle a été réalisée et nous avons trouvé 129 paires de documents pouvant être considérés comme pertinents, incluant 29 paires de documents parallèles, 31 paires de documents parallèles bruités et 69 paires de documents comparables.

Dans le premier module, selon notre corpus, nous avons supposé que $d = 2$ (nous utilisons une fenêtre de 4 jours autour la date de publication du document). Le deuxième filtrage a été réalisé sur l'ensemble $S_{FR\text{-}1k}$ et l'ensemble $S_{VN\text{-}1k}$ pour identifier les correspondances des noms propres. Après l'utilisation des deux filtrages du premier module, nous avons obtenu 379 paires de documents, toutes les paires de documents pertinents ont été trouvées. Le Tableau 5-3 résume le résultat d'extraction après application du premier module.

Tableau 5-3 : Le résultat d'extraction après application du premier module

E_{1k}	Nbr. de doc. : 1 000 Nbr. de doc. en français : 173 ($S1_{E1k}$) Nbr. de doc. en vietnamien : 348 ($S2_{E1k}$) Nbr. de paires de documents pertinents vrais : 129
Après le premier module	Nbr. de paires de documents obtenus : 379 ($S2''_{E1k}$) Nbr. de paires de documents corrects : 129 Précision = 129/379= 34,04 % Rappel = 100 % F-mesure (F1 score) = 47,18 %

Le processus d'alignement en phrases du deuxième module a été réalisé en utilisant les ensembles $S1_{E1k}$, $S2''_{E1k}$ et l'outil Champollion. Nous avons adapté Champollion pour traiter la paire de langues français – vietnamien. Nous avons aussi modifié le code source de l'outil pour marquer les correspondances entre des mots de deux phrases (les mots qui ont des équivalents dans l'autre phrase). Les paramètres de Champollion pour cette paire de langues sont estimés sur l'ensemble E_{1k}. Suite à une vérification manuelle, la proportion de mots français par rapport aux mots vietnamiens dans une paire de phrases parallèles est assignée à *1,2*. La valeur de la pénalité d'alignement est fixée à *1* avec des correspondances de type *1:1* ; 0,8 avec des correspondances de type *0:1*, *1:0*, et 0,75 avec des correspondances de type *2:1, 1:2, 2:2*. Les autres types de correspondance n'ont pas été utilisés.

Le troisième module a ensuite été appliqué en faisant varier le paramètre α (0,4 ; 0,5 ; 0,6 ; 0,7 ; 0,8) et le paramètre β (0,1 ; 0,15 ; 0,2 ; 0,25 ; 0,3). La précision, le rappel et le F-mesure de la recherche des paires de documents pertinents ont été estimés. Trois catégories de paires de documents pertinents sont validées : le groupe de paires de documents parallèles (*Gp*) ; le groupe de paires de documents parallèles et parallèles bruités (*Gpb*) ; le groupe de paires de documents parallèles, parallèles bruités et comparables (*Gpbc*). Les scores de chaque groupe sont présentés dans la Figure 5-8.

Parce que la méthode d'extraction se concentre sur les hypothèses d'un corpus comparable, la précision du groupe *Gpbc* est la plus grande et la précision du groupe *Gp* est la moins grande (précision de *Gp* < précision de *Gpb* < précision de *Gpbc*). Inversement, le rappel du groupe *Gp* est le plus grand tandis que le rappel du groupe *Gpbc* est le moins grand (rappel de *Gp* > rappel de *Gpb* > rappel de *Gpbc*). Dans un même groupe, plus les seuils sont restreints (α plus grand et β plus petit), moins grandes sont les précisions obtenues et plus grands sont les rappels obtenus. De plus, quand nous observons la F-mesure de chaque groupe et pour chaque seuil α, le seuil $\beta = 0,15$ est approprié pour le groupe *Gpbc* alors que le seuil $\beta = 0,25$ convient avec le groupe *Gp* et *Gpb*, car ils donnent la meilleure F-mesure pour chaque groupe.

Pour obtenir la meilleure F-mesure du groupe *Gpbc*, nous choisissons la valeur $\alpha = 0,7$ et $\beta = 0,15$. Les valeurs de α et β sont dépendantes du degré de parallélisme entre deux documents pertinents de chaque corpus comparable, et la valeur de β est dépendante aussi du dictionnaire utilisé. Ces deux valeurs peuvent être petites dans notre cas mais elles peuvent devenir plus grandes dans le cas où le degré de parallélisme de corpus en entrée serait plus grand et où la qualité du dictionnaire serait meilleure.

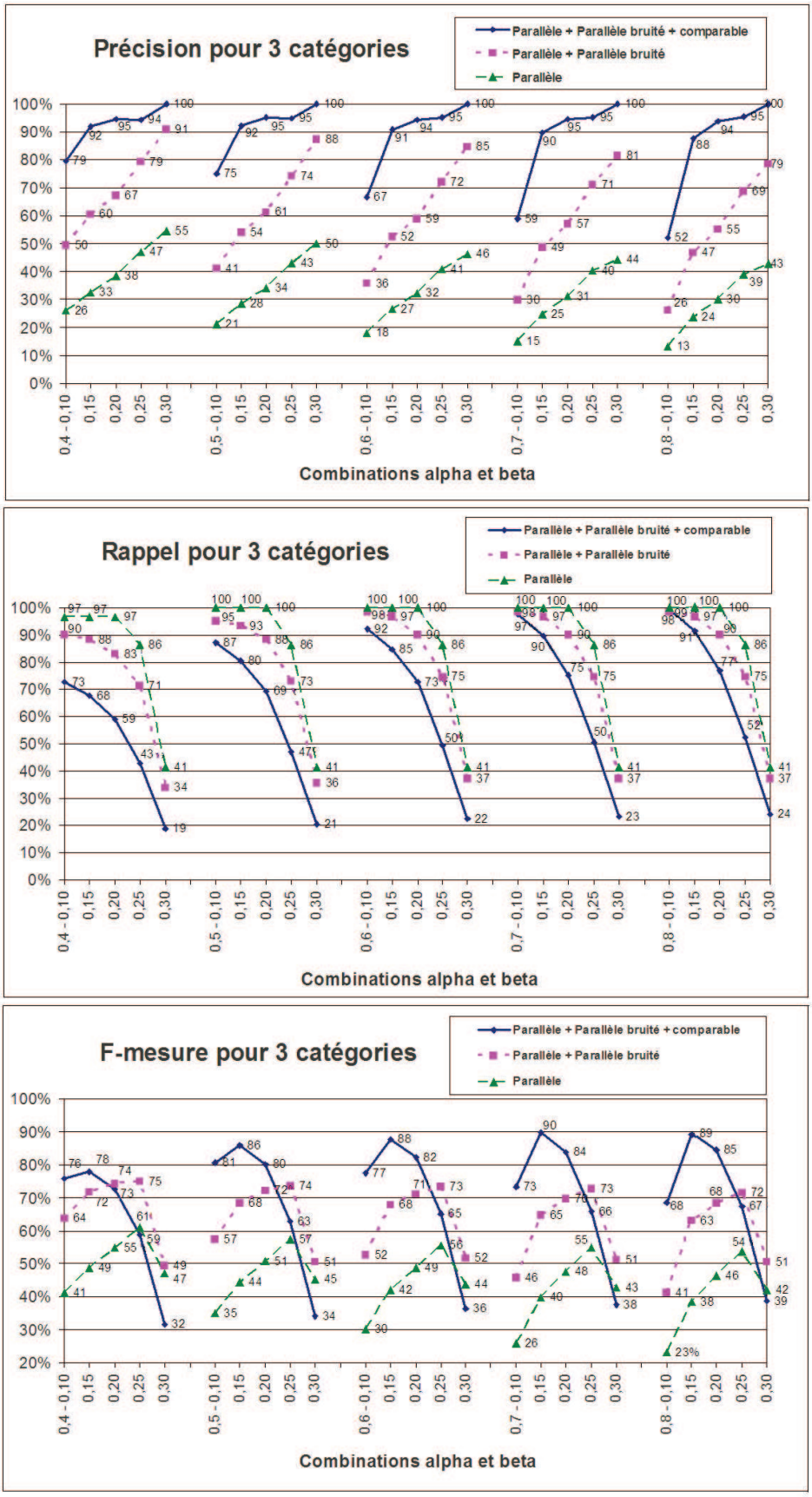


Figure 5-8 : Les scores pour 3 catégories de paires de documents pertinents : le groupe de paires de documents parallèles (Gp) ; le groupe de paires de documents parallèles et parallèles bruités (Gpb) ; le groupe de paires de documents parallèles, parallèles bruités et comparables (Gpbc)

5.3.3. Application sur le corpus entier E_{all}

Nous avons appliqué la méthode d'extraction avec les paramètres estimés dans la section 5.3.2 sur l'ensemble E_{all}. Au vu des résultats obtenus sur E_{1k}, nous choisissons les paramètres $\alpha = 0{,}7$ et $\beta = 0{,}15$ (qui optimise la F-mesure pour les documents de catégorie Gpbc). Les caractéristiques du corpus entier traité sont présentées dans le Tableau 5-4. Nous obtenons un total de 50 000 paires de phrases considérées par le système comme parallèles qui constituent ainsi notre futur corpus d'apprentissage pour un modèle de traduction français – vietnamien.

Tableau 5-4 : Caractéristiques du corpus obtenu

E_{all}	Nombre de documents : 120 218 Nombre de documents en français : 20 884 Nombre de documents en vietnamien : 54 406
Après le premier module	Nombre de paires de documents possibles : 21 446
Corpus parallèle obtenu	Nombre de paires de documents récupérées : 12 108 Nombre de paires de phrases récupérées : 50 322 (environ de 30 MBytes de textes bruts)

La qualité d'un sous-ensemble des paires de phrases extraites du corpus E_{all} est évaluée manuellement. 100 paires de phrases sont choisies au hasard, et sont classées dans 3 groupes : parallèles, comparables, et non pertinentes (les phrases non parallèles ou celles pour lesquelles moins de 30 % de parties de la phrase sont comparables). Nous avons trouvé 42 % de paires de phrases parallèles, 38 % de paires de phrases comparables, et 20 % de paires de phrases non pertinentes. Le Tableau 5-5 présente les exemples de paires de phrases des trois groupes dans notre corpus obtenu. Les textes en **gras** d'une phrase possèdent une traduction dans l'autre phrase, et les textes soulignés sont des parties non traduites de l'autre côté.

5.4. Application : premier système de traduction probabiliste pour le couple de langue vietnamien – français

Avec 50 mille paires de phrases pertinentes, nous avons désormais une quantité de bi-texte utilisable pour l'apprentissage d'un premier système de TA probabiliste. Nous avons construit des systèmes de TA probabiliste français–vietnamien et vietnamien–français à partir du corpus obtenu en utilisant les outils libres GIZA++ [Och 2000] et MOSES [Koehn 2007a] (se référer à la section 2.7). Les scripts fournis avec MOSES nous permettent de construire un modèle de traduction fondé sur des séquences de mots (*phrase table*). Il contient également des outils pour régler les paramètres des modèles et pour estimer le score BLEU.

5.4.1. Préparation des données

A partir du corpus entier, nous avons choisi 50 paires de documents pertinents pour le réglage des paramètres du système de traduction (MERT) (l'ensemble de développement contient au final 351 paires de phrases pertinentes), 50 paires de documents pertinents pour le test (l'ensemble de test contient au final 384 paires de phrases pertinentes), et le reste est réservé pour l'apprentissage (l'ensemble d'apprentissage contient au final 49 587 paires de phrases pertinentes). En ce qui concerne les ensembles de développement et de test, ceux-ci ont été vérifiés manuellement et nous avons éliminé les paires de phrases comparables et gardé seulement les paires de phrases parallèles. Au final, nous obtenons 198 paires de phrases parallèles pour le développement et 210 paires de phrases parallèles pour le test. Les données qui

ont servi pour créer les modèles de traduction et de langage ont été extraites automatiquement depuis les 49 587 paires de phrases pertinentes de l'ensemble d'apprentissage, non vérifiées manuellement.

Tableau 5-5 : Exemples de paires de phrases récupérées dans notre corpus: parallèles, comparables, et non pertinentes (les phrases sont normalisées : sans casse et tokénisées)

Paires de phrases parallèles	**- nous vous soutenons pour continuer votre politique de développement en harmonie entre les intérêts nationaux et la réalité de la globalisation mondiale laquelle apportera des intérêts pour le peuple vietnamien** ***- chúng tôi ủng hộ các bạn tiếp tục đường lối phát triển hài hòa giữa lợi ích quốc gia với thực tế của một thế giới toàn cầu hóa mang lại nhiều lợi ích cho nhân dân việt nam***
	- ce programme de coopération inclut encore le partage des expériences des échanges de technologies et la formation des cadres ***- chương trình hợp tác còn bao gồm cả việc trao đổi kinh nghiệm công nghệ và đào tạo cán bộ***
Paires de phrases comparables	**- c' est la première fois** qu' un tel **prix honore des entreprises ayant eu d' importantes contributions à la prospérité commune** de l' asean **en matière de développement économique** , **de création d' emploi , de rénovation d' entreprises et de responsabilité sociale** ***- đây là lần đầu tiên giải thưởng*** được trao tặng , nhằm ***tôn vinh những doanh nghiệp có thành tích nổi bật đối với sự thịnh vượng chung*** trong khu vực . giải thưởng được xét tặng ***theo tiêu chí là tăng trưởng , tạo việc làm , đổi mới doanh nghiệp và trách nhiệm xã hội***
	- il a **informé** son interlocuteur **des résultats de ses discussions** avec **l' amiral adjoint nguyen van hien** , **commandant de la marine populaire du vietnam** , **concernant certaines questions d' intérêt commun** , pour la paix , la stabilité et la prospérité dans la région et le monde - đô đốc nam haeil bày tỏ vui mừng được đến thăm việt nam ***và thông báo kết quả trao đổi với tư lệnh hải quân nhân dân việt nam - phó đô đốc nguyễn văn hiến về một số vấn đề cùng quan tâm , hợp tác***
Paires de phrases non pertinentes	**- les données obtenues serviront de base pour les études approfondies** , elle se concentrera dans les foyers ruraux , les infrastructures rurales , le processus de l' industrialisation et la modernisation rurale et agricole , l' économie fermière , les entreprises et coopératives opérant dans les domaines agricole et sylvicole , les compétences dans la production aquatique , les investissements en 2005 , les fonds dormants des foyers ruraux , les plantes et animaux utiles ***- các dữ liệu*** về nông thôn , nông nghiệp và thủy sản ***thu được*** từ cuộc tổng điều tra còn ***phục vụ công tác nghiên cứu chuyên sâu*** và làm dàn chọn mẫu cho các cuộc điều tra định kỳ của các năm tiếp theo
	- le président de l ' assemblée nationale du vietnam , nguyên phu trong , qui a reçu lundi à hanoi **le président de la commission européenne , josé manuel barroso** , a plaidé en faveur du renforcement des relations avec le parlement européen et ses pays membres ***- chủ tịch uỷ ban châu âu jose manuel barroso*** tới hà nội bắt đầu chuyến thăm chính thức việt nam đã tham dự đ ê m nhạc kịch châu âu , được tổ chức tối 25 / 11 , tại nhà hát lớn thành phố hà nội

Tableau 5-6 : Préparation des données

Ensemble	**Paires de documents**	**Paires de phrases**	**Après verif. manuelle**
Développement (DEV)	50	351	198
Test (TST)	50	384	210
Apprentissage (TRN)	12 108	49 587	Non vérifié

5.4.2. Systèmes de référence

Nous avons construit des systèmes de TA dans les deux sens : français vers vietnamien (F→V) et vietnamien vers français (V→F). Les données en vietnamien ont été segmentées en syllabes et en mots. La segmentation de la phrase vietnamienne en mots est réalisée en utilisant *Clips-text-TK* dont l'algorithme est fondé sur l'algorithme de « longest matching» à partir d'un dictionnaire monolingue de mots vietnamiens. Nous avons au total quatre systèmes de TA :

- le système de traduction du texte français vers le texte vietnamien segmenté en syllabe S1FV
- le système de traduction du texte vietnamien segmenté en syllabe vers le texte français S1VF
- le système de traduction du texte français vers le texte vietnamien segmenté en mot S2FV
- le système de traduction du texte vietnamien segmenté en mot vers le texte français S2VF

Nous obtenons les performances présentées dans les Tableau 5-7 et Tableau 5-8 pour tous ces systèmes. Le modèle de langage est entraîné par le corpus monolingue de VNA. Dans l'étape d'apprentissage, nous avons supprimé les phrases dans l'ensemble TRN qui sont plus longues que 100 unités (mots ou syllabes).

Tableau 5-7 : Les quatre premiers systèmes de TA développés

Segmentation du vietnamien	**Nombre de paires de phrases**	**Langue**	**Taille du vocabulaire (K)**	**Nombre de mots (syllabes) (K)**
Syllabe Système S1FV (F→V) Système S1VF (V→F)	TRN[1] : 47 081	Fr	38,6	1783,6
		Vn	21,9	2190,2
	DEV : 198	Fr	1,8	6,3
		Vn	1,2	6,9
	TST : 210	Fr	1,9	6,4
		Vn	1,3	7,1
Mot Système S2FV (F→V) Système S2VF (V→F)	TRN : 48 864	Fr	39,7	1893,0
		Vn	33,4	1629,0
	DEV : 198	Fr	1,8	6,3
		Vn	1,5	4,8
	TST : 210	Fr	1,9	6,3
		Vn	1,6	4,9

Tableau 5-8 : Evaluation des systèmes de TA sur l'ensemble de test (après tuning sur le DEV)

	Systèmes de TA	**BLEU (%)**
français vers vietnamien	S1FV	40,09
	S2FV	40,59
vietnamien vers français	S1VF	31,73
	S2VF	30,58

Dans le cas de systèmes où le texte vietnamien a été segmenté en mots, les phrases traduites en vietnamien sont re-segmentées en syllabes avant de calculer le score BLEU, ceci afin que tous les scores BLEU évalués soient comparables. Les scores BLEU pour le sens de traduction français vers vietnamien sont environ 40 % et pour le sens vietnamien vers français environ 31 %, ce qui est encourageant pour un premier résultat. Le sens de traduction français vers

[1] L'ensemble TRN est supprimé les phrases qui sont plus longues que 100 syllabes, donc le nombre de paires de chaque ensemble est inférieur à 49 587 paires.

vietnamien peut être « plus facile » que le sens de traduction inverse, parce que la forme d'un mot vietnamien est unique pendant que les mots français changent de forme selon leur fonction grammaticale. Donc un mot vietnamien correspond à plusieurs formes de mot français. Par exemple, toutes les formes « *étudie* », « *étudies* », « *étudions* », « *étudié* », etc. du verbe français « *étudier* » sont traduites en un seul et même mot vietnamien « *học* ». Mais la traduction du mot vietnamien « *học* » en français est « plus difficile » à cause de plusieurs formes possibles en français. Le contexte du groupe de mots doit être utilisé pour trouver la traduction correspondante. De plus, une seule référence est utilisée pour estimer les scores BLEU dans nos expériences. Si nous avions plusieurs références, ces scores BLEU seraient évidemment plus hauts.

Il est également intéressant de noter que la segmentation des phrases vietnamiennes en syllabes ou en mots a peu d'influence sur la performance pour les deux sens de traduction. C'est un peu étonnant, mais une raison est que nous utilisons l'approche de traduction probabiliste en groupe de mots. Par exemple, le mot « *F* » en français est la traduction d'un mot vietnamien de deux syllabes « V_1 V_2 ». Lorsque l'approche de traduction probabiliste en mots uniquement est utilisée, la segmentation en syllabes nous donne deux probabilités $p(V_1|F)$, $p(V_2|F)$ et la segmentation en mots nous donne une probabilité $p(V_1\_V_2|F)$. Donc la traduction du mot « *F* » est soit « V_1 », soit « V_2 », soit « $V_1\_V_2$ ». Ainsi, dans ce cas, la segmentation influence bien le choix du mot final et la performance du système de traduction peut dépendre du type de segmentation. Par contre, dans l'approche de traduction probabiliste en groupe de mots que nous utilisons, l'alignement entre groupes de mots est appris. « V_1 » et « V_2 » sont deux syllabes d'un mot donc la probabilité qu'ils apparaissent ensemble est grande. Par conséquent, le système de traduction obtient l'alignement entre « *F* » et « V_1 V_2 » même dans le cas d'une segmentation en syllabe. Ainsi, la probabilité que « V_1 V_2 » et « *F* » apparaissent dans un alignement peut être plus grande que la probabilité que « V_1 » – « *F* » et « V_2 » – « *F* » apparaissent. On note en tout cas que l'influence de la segmentation en syllabes ou en mots sur la performance de traduction n'est pas vraiment significative (dans le sens de traduction français vers vietnamien, le score BLEU est changé de 40,09 à 40,59, et dans le sens de traduction vietnamien vers français, le score BLEU est changé de 31,73 à 30,58). Un exemple de la sortie de nos systèmes de traduction est présenté dans le Tableau 5-9.

Tableau 5-9 : Exemples de sorties de nos systèmes de traduction

Une paire de phrases parallèles dans l'ensemble de test : **FR** : selon le département de gestion des travailleurs à l'étranger le qatar est un marché prometteur et nécessite une grande quantité de travailleurs étrangers **VN syl** : theo cục quản lý lao động ngoài nước cata là thị trường đầy tiềm năng và có nhu cầu lớn lao động nước ngoài **VN mot** : theo cục quản_lý lao_động ngoài nước cata là thị_trường đầy tiềm_năng và có nhu_cầu lớn lao_động nước_ngoài	
S1VF	**Entrée** : VNsyl **Référence** : FR **Sortie** : selon le département de gestion des travailleurs étrangers cata était un marché plein de potentialités et aux besoins importants travailleurs étrangers
S2VF	**Entrée** : VNword **Référence** : FR **Sortie** : selon le département de gestion des travailleurs étrangers cata marché plein de potentialités et la grande travailleurs étrangers
S1FV	**Entrée** : FR **Référence** : VNsyl **Sortie** : theo cục quản lý lao động ở nước ngoài phía cata là một thị trường đầy tiềm năng và cần một lượng lớn lao động nước ngoài
S2FV	**Entrée** : FR **Référence** : VNword **Sortie** : theo thống_kê của cục quản_lý lao_động ngoài nước cata là một thị_trường đầy tiềm_năng và cần có sự lớn lượng lao_động nước_ngoài

5.4.3. Expériences complémentaires

5.4.3.1 Réduire le nombre de paires de phrases comparables pour augmenter la qualité du corpus d'apprentissage

Le corpus d'apprentissage pour construire les systèmes de référence comporte des paires de phrases parallèles, comparables, et aussi des paires non pertinentes en raison d'erreurs dans le processus d'extraction. Dans la section 5.3.3, nous avons choisi les paramètres $\alpha = 0,7$ et $\beta = 0,15$, et nous avons obtenu 50 322 paires de phrases pertinentes. Après avoir choisi les corpus de DEV et TST, nous avons 49 587 paires de phrases pour entraîner le modèle de traduction. La question que nous posons dans cette section est : que se passe-t-il si nous filtrons encore les paires de phrases pertinentes pour entraîner le modèle de traduction ? Sur la base de 49 587 paires de phrases pertinentes, nous avons extrait trois autres corpus des paires de phrases qui possèdent les deux scores x_{P1} et x_{P2} plus grands que $\beta = 0,2 ; 0,25 ; 0,3$. Le relèvement de β augmente le pourcentage de paires de phrases parallèles mais il réduit le nombre de paires de phrases extraites au total.

Trois systèmes de traduction du vietnamien (en syllabe) vers le français ont été construits avec le modèle de traduction entraîné sur ces trois corpus séparément. Les scores BLEU des trois systèmes de traduction ont été estimés avec réglage sur le même corpus de DEV et de TST. Les résultats sont présentés dans le Tableau 5-10.

Tableau 5-10 : Le relèvement de β et le score BLEU de système de traduction associé

β	Nombre de paires de phrases pour entraîner le modèle de traduction	BLEU (%) vietnamien (en syllabe) vers français
0,15	49 587	31,73 (baseline voir le Tableau 5-8)
0,2	38 283	29,14
0,25	24 928	28,35
0,3	13 734	26,80

Bien que le relèvement de β élimine les paires de phrases dont le degré de parallélisme est faible, la diminution du nombre de paires de phrases pour entraîner le modèle de traduction réduit le score BLEU du système de traduction. Les raisons possibles sont que, dans notre cas le seuil $\beta = 0,15$ permet déjà d'extraire des paires de phrases pertinentes pour l'entraînement du modèle de traduction et que les modèles probabilistes sont relativement robustes à des données non parallèles présentes dans le corpus.

5.4.3.2 Combinaison des systèmes fondés sur mot et syllabe en vietnamien

Dans la section 5.4.2, nous avons vu que la segmentation des phrases vietnamiennes en syllabes ou en mots ne modifie pas sensiblement la performance pour les deux sens de traduction. Dans cette section, nous avons effectué un autre test sur la combinaison des unités lexicales (syllabes et mots) sur le vietnamien. Nous avons réalisé le test dans le sens de traduction vietnamien vers français. Deux manières de réaliser les combinaisons ont été testées.

La première méthode est la combinaison des tables de traduction. En fait, le décodeur MOSES permet la combinaison de deux ou plusieurs tables de traduction : nous pouvons utiliser les options de traduction (section 2.4) *dans les deux tables de traduction* (le mode « *both* ») ou *dans n'importe quelle table de traduction* (le mode « *either* »). Dans le mode « both », une option de traduction est sélectionnée lorsqu'elle existe dans les deux tables de traduction. Et un score composite est créé pour cette option. En revanche, dans le mode « either », une option de traduction est sélectionnée depuis n'importe quelle table de traduction et elle prend le score de cette table. Ce dernier mode permet de recueillir toutes les options de traduction pour les deux

tables. Dans tous les cas, chaque table de traduction possède son propre ensemble de poids. Dans cette expérience, nous avons utilisé le mode « either ».

Les tables de traduction du système S1VF (T_{syl}) et du système S2VF (T_{mot}) ont été utilisées. Une autre table (T_{mot*}) a été créée à partir de la table T_{mot}, dans laquelle tous les mots ont été re-transformés en syllabe (dans ce dernier cas, la segmentation en mot a été utilisée durant le processus d'alignement et de construction de la table de traduction, mais la partie en vietnamien de la table finale est re-segmentée en syllabes). Les combinaisons de ces trois tables de traduction ont également été créées. Pour le test, les phrases de traduction en vietnamien étaient segmentées soit en mot soit en syllabe. Comme précédemment, l'ensemble de développement a été utilisé pour régler les paramètres et l'ensemble de test a été utilisé pour estimer le score BLEU. Les résultats obtenus sont présentés dans le Tableau 5-11. Des cellules sont marquées par *X* car certaines combinaisons n'ont pas de sens (par exemple la combinaison entre l'entrée en mots et la table de traduction en syllabes). Ces résultats montrent que la performance peut être améliorée en combinant les informations de mots et de syllabes du côté vietnamien. Le score BLEU est amélioré de 35,30 à 38,02 sur l'ensemble DEV (de 7 %) et de 31,73 à 32,08 (1 %) sur l'ensemble TST. L'amélioration sur le test n'est cependant pas significative.

Tableau 5-11 : Scores BLEU (%) obtenus avec combinaisons entre tables de traduction (calculés sur l'ensemble de développement et l'ensemble de test)

	Tables de traductions utilisées	Entrée en syllabe		Entrée en mots	
		DEV	TST	DEV	TST
VN vers FR	Tsyl	**35,30**	**31,73**	X	X
	Tmot	X	X	35,70	**30,58**
	Tmot*	37,31	31,76	X	X
	Tsyl + Tmot	35,30	31,43	36,80	30,68
	Tsyl + Tmot*	**38,02**	**32,08**	X	X
	Tmot + Tmot*	37,42	30,23	36,67	30,21

Nous avons analysé les sorties de ces systèmes de traduction. Dans la phrase en sortie, chaque groupe de mot est marqué selon la table de traduction utilisée. Les traductions de 210 phrases dans l'ensemble de test ont été marquées. Nous avons aussi calculé le nombre de groupes de mots utilisés de chaque table (Tableau 5-12).

Tableau 5-12 : Le nombre de groupes de mots choisis à partir de chaque table

Tables de traductions utilisées (la première table + la deuxième table)	Nombre de groupes de mots de 210 phrases sorties	Nombre de groupes de mots collectés à partir de la première table	Nombre de groupes de mots collectés à partir de la deuxième table
Tsyl	2 654	2 654 (100 %)	
Tmot*	2 485	2 485 (100 %)	
Tsyl + Tmot	2 623	2 045 (78 %)	578 (22 %)
Tsyl + Tmot*	2 423	1 076 (44 %)	1 347 (56 %)
Tmot + Tmot*	2 485	14 (0,5 %)	2 471 (99,5 %)

L'utilisation conjointe mot et syllabe peut améliorer légèrement le score BLEU, mais pas vraiment de façon significative sur un corpus de test inconnu. La segmentation en mot durant le processus d'alignement est efficace mais la plupart des groupes de mots est extrait à partir de la table en syllabe.

La deuxième méthode de combinaison entre l'approche « mot » et l'approche « syllabe » est l'utilisation d'un réseau de confusion. MOSES nous permet de représenter l'entrée du processus de décodage par un réseau de confusion. Nous avons reformé les phrases d'entrée de l'ensemble de DEV par des réseaux de confusion simples (avec les probabilités égales pour chaque chemin) qui combinent les syllabes et les mots de la phrase (Figure 5-9). Le résultat de traduction est

estimé sur l'ensemble et présenté dans le Tableau 5-13. Les premiers résultats obtenus montrent que la représentation en réseau de confusion simple ne semble pas efficace dans notre cas.

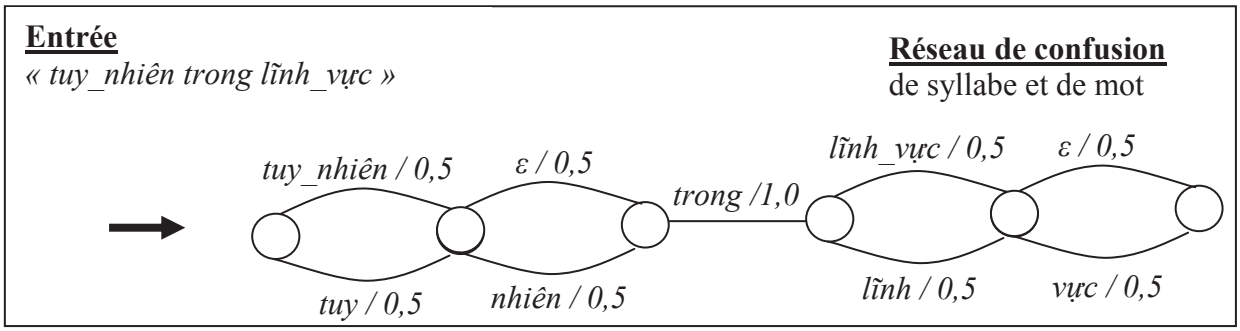


Figure 5-9 : La représentation d'un réseau de confusion simple combinant les syllabes et les mots pour un fragment de mots vietnamiens

Tableau 5-13 : Utilisation de l'entrée en réseau de confusion simple

Tables de traductions utilisées	Entrée en syllabe	Entrée en mots	Entrée en réseau
Tsyl	**35,30**	X	33,54
Tmot	X	35,70	34,87
Tmot*	37,31	X	36,13
Tsyl + Tmot	35,30	36,80	36,57
Tsyl + Tmot*	**38,02**	X	36,56
Tmot + Tmot*	37,42	36,67	36,40

5.4.3.3 Comparaison avec le système de TA de Google

Le système de TA *Google Translate*[1] a ajouté la langue vietnamienne à sa liste de langues traitées depuis septembre 2008. Dans la plupart des cas, il utilise l'anglais comme une langue intermédiaire. Pour une première évaluation comparative, un test simple a été réalisé. Deux ensembles de données ont été utilisés : un dans le domaine des nouvelles journalistiques (l'ensemble TST de la section 5.4.2), et un hors du domaine des nouvelles journalistiques. Ce dernier a été obtenu à partir du site Web bilingue vietnamien français de l'Ambassade de France au Vietnam[2]. Après avoir traité préalablement et aligné manuellement, nous avons obtenu 100 paires de phrases parallèles pour l'ensemble hors du domaine de données. Les phrases vietnamiennes ont été segmentées en syllabes. Les deux ensembles de données ont été traduits par nos systèmes de TA (S1FV, S1VF) et le système de TA de *Google* (le 4 février 2009). Les résultats des systèmes ont été post-traités (passage en minuscule) et les scores BLEU ont été estimés. Le Tableau 5-14 présente les résultats de ce test. Bien que notre système soit logiquement meilleur sur l'ensemble de données dans le domaine, il est également légèrement meilleur que le système de TA de *Google* sur l'ensemble de données hors domaine (pour le couple de langues vietnamien français).

Tableau 5-14 : Comparaison avec le système de TA de Google (le 4 février 2009)

	Sens de traduction	Score BLEU (%)	
		Notre système	Google
Dans le domaine news (210 paires de phrases)	F→V	40,09	24,82
	V→F	32,08	15,63
Hors du domaine (100 paires de phrases)	F→V	25,00	24,38
	V→F	20,22	15,82

[1] http://translate.google.com
[2] http://www.ambafrance-vn.org

Tableau 5-15 : Un exemple de la traduction d'une phrase vietnamienne vers français par notre système et le système de TA de Google (le 4 février 2009)

Référence en français	le chef d'état vietnamien a proposé que le vietnam et l'australie renforcent les dialogues politiques de haut rang , ainsi que les contacts
La traduction de notre système	le président vietnamien a demandé le vietnam et l'australie renforcer les dialogues politiques de haut rang , ainsi que des rencontres
La traduction de Google 2009	président du vietnam et l'australie pour renforcer le dialogue politique , ainsi que des niveaux élevés d'exposition

Le Tableau 5-15 montre un exemple de la traduction d'une phrase vietnamienne vers français par notre système et le système de TA de Google. La phrase vietnamienne est prise au hasard dans le corpus de test. La traduction de notre système est assez acceptable, lorsque la traduction du système de TA de Google ne veut rien dire.

5.5. Conclusion

Dans ce chapitre, nous avons présenté notre première méthode sur l'extraction d'un corpus bilingue comparable pour construire des systèmes de traduction probabilistes pour une paire de langues incluant une langue peu dotée. Nous avons décrit la méthode d'alignement de documents, qui est basée sur des informations lexicales et extrait à la fois les documents et les phrases pertinents.

Tout d'abord, les paires de documents possibles sont filtrées en utilisant la date de publication et les mots spéciaux (les numéros, les symboles joints, les noms propres). Deuxièmement, des phrases dans une paire de documents possibles sont alignées en utilisant un outil existant, *Champollion*, qui utilise des informations lexicales (la liste des formes de surface de mots, la liste de mots d'arrêt, le dictionnaire bilingue, etc.) et peut traiter des données bruitées. Enfin, des paires de documents et de phrases pertinentes sont extraites en utilisant des informations d'alignement de phrases. Cette méthode requiert des données supplémentaires sur la paire de langues pour aligner des paires de phrases. Lorsque ces données supplémentaires sont disponibles, nous pouvons appliquer simplement cette méthode pour toutes les paires de langues.

La méthode proposée est appliquée aux données vietnamiennes et françaises récupérées depuis un site Web multilingue de nouvelles journalistiques, le Vietnam News Agency (contenant 20 884 documents français et 54 406 documents vietnamiens), ce qui nous fournit un corpus véritablement comparable. Nous avons obtenu près de 12 100 paires de documents considérés comme pertinents et 50 300 paires de phrases considérées comme pertinentes. Nous avons construit des systèmes de TA utilisant l'outil MOSES. Les scores BLEU obtenus sont de 40,09 % pour le système de traduction français vers vietnamien et de 32,08 % pour le système vietnamien vers français. De plus, des expériences complémentaires autour de la segmentation optimale du vietnamien (mots ou syllabes) ont été réalisées.

Bien que le parallélisme du corpus d'apprentissage puisse être amélioré par élimination de paires de phrases moins pertinentes qui contiennent peu de lexique parallèle, la diminution du nombre de paires de phrases pour entraîner le modèle de traduction réduit la qualité du système de traduction. Donc, pour les langues peu dotées, en plus de l'extraction de textes parallèles, l'extraction de textes comparables est aussi avantageuse. La combinaison des informations entre les mots et les syllabes vietnamiennes peut aussi être utile pour améliorer les performances des systèmes de traduction mais les améliorations observées sur notre corpus de test ne sont pas significatives. Dans la suite de cette thèse, la segmentation en syllabe sera utilisée pour le texte vietnamien. Concernant la comparaison avec le système de TA de *Google* (à la date de février 2009), bien que notre système soit logiquement meilleur sur l'ensemble de données dans le

domaine, il est également légèrement meilleur que le système de TA de *Google* sur l'ensemble de données hors domaine (pour le couple de langues vietnamien – français).

Chapitre 6 : Apprentissage non supervisé pour la traduction automatique : application à l'extraction d'un corpus comparable

Comme présenté dans le chapitre 4, les méthodes proposées pour extraire des données parallèles à partir d'un corpus comparable suivent les deux approches suivantes : approche de recherche à base de caractéristiques générales et approche « recherche d'information translingue » (RIT) nécessitant un module de traduction.

La première méthode, que nous venons de présenter dans le chapitre 5, concerne l'utilisation d'informations d'alignement pour extraire à la fois des documents pertinents et des phrases parallèles. Cette méthode requiert des données supplémentaires pour la paire de langues considérée (telles que le dictionnaire bilingue, la liste de mots d'arrêt, la liste des formes de base d'un mot, etc.). Lorsque ces données supplémentaires sont disponibles, cette méthode est applicable, en théorie, pour n'importe quelle autre paire de langues.

Cependant, dans certains cas, mêmes ces données minimales peuvent être indisponibles, notamment pour des langues peu dotées. L'approche RIT utilisant un module de traduction peut s'avérer alors utile. En effet, quelques travaux de recherche autour de cette approche ont été présentés mais ils requièrent aussi des données supplémentaires telles que des grands dictionnaires bilingues ou un corpus parallèle initial pour construire le module de traduction qui sera utilisé dans la technique RIT (voir plus de détails dans la section suivante). Ceci peut paraître paradoxal puisque pour des langues peu dotées, un tel modèle sera indisponible dans la plupart des cas. Dans ce travail, nous supposons que dans le cas d'un couple de langues peu dotées, même un petit corpus parallèle n'est pas forcément disponible pour développer le système de TA initial. La question que nous nous posons alors est : *est-ce qu'un processus totalement non supervisé, initialisé à partir d'un corpus comparable particulièrement bruité, permet d'apporter des solutions au problème du manque de données parallèles ?*

Ce chapitre présente notre méthode d'extraction non supervisée appliquée dans le cas de langues peu dotées. Le processus d'extraction ne requiert aucune donnée supplémentaire, même en petite quantité. Cette méthode réellement non supervisée peut être appliquée pour toutes les paires de

langues, avec un corpus comparable seulement, et aucune donnée supplémentaire n'est requise pour ces deux langues.

6.1. Présentation des approches utilisant la technique RIT à base d'un système de traduction

La tâche standard en recherche d'information (sur du texte) est de récupérer des textes (des documents) qui répondent à la requête de l'utilisateur à partir d'une grande collection de données. Normalement, la langue de la requête et la langue de la collection de documents sont les mêmes. Plus généralement, la recherche d'information inclut deux aspects : l'indexation des corpus, et l'interrogation du fonds documentaire constitué. L'indexation des documents consiste à passer d'un document textuel à une représentation exploitable par le modèle de recherche d'information. Cette représentation peut être un ensemble de descripteurs : l'ensemble des termes (des mots) qui apparaissent dans un document (convertis éventuellement en forme de surface et avec suppression des mots outils tels que déterminants, prépositions et conjonctions par exemple). Après, il est possible d'indexer le document par un vecteur dans l'espace des termes. Une fois les documents indexés, nous pouvons rechercher ceux qui répondent le mieux à une requête d'un utilisateur en comparant leurs vecteurs grâce à un score. Les moteurs de recherche sur le Web sont les applications les plus populaires de la recherche d'information.

La technique RIT (recherche d'information translingue) est un sous domaine de la recherche d'information qui récupère des informations écrites dans une langue différente de la langue de la requête de l'utilisateur. Par exemple, l'utilisateur peut poser sa requête en anglais, et le système lui répond par des documents pertinents rédigés en français. Ainsi, la technique RIT doit résoudre en plus le problème de traduction de la requête vers la langue de la collection de documents.

[Collier 1997] a présenté une application de la technique RIT pour aligner des documents parallèles à partir d'un corpus de textes de nouvelles journalistiques en langues anglaise et japonaise (Figure 6-1). Dans leur cas, un document japonais est le résumé d'environ quatre ou cinq phrases d'un document anglais correspondant. Les paires de phrases japonaises correspondent donc à des phrases clés dans le document anglais. Ainsi les documents japonais peuvent être considérés comme les requêtes. Les auteurs ont proposé d'utiliser un dictionnaire bilingue pour traduire les termes du document japonais en anglais. Chaque document japonais, ou chaque requête, a été ensuite représenté par un vecteur avec une cinquantaine de termes anglais. De leur côté, les documents anglais sont indexés par leurs termes. Pour estimer la similarité entre un vecteur qui représente la requête et un document anglais, cinq modèles de score ont été proposés basés sur la fréquence d'un terme TF, la fréquence inverse de document IDF, le nombre de termes, etc. Un seuil a été utilisé pour décider les paires de documents correspondants.

[Utiyama 2003] a utilisé aussi la technique RIT pour aligner des documents parallèles et ensuite la technique de programmation dynamique pour trouver le meilleur alignement en phrases d'une paire de documents parallèles. Les termes de documents (convertis en forme de surface) du côté source (en japonais) ont été traduits en langue cible (anglais) par un dictionnaire. La mesure de la similarité entre deux documents est basée sur le score BM25 [Robertson 1994] qui est assez efficace en recherche d'information. Ce score de similarité est calculé à l'aide de la fréquence d'un terme apparaissant dans chaque document, le nombre de documents dans l'espace de recherche, le nombre de documents contenant ce terme, la longueur du document, etc. Pour augmenter la fiabilité, les auteurs ont proposé encore un autre score basé sur des alignements en phrases de la paire de documents.

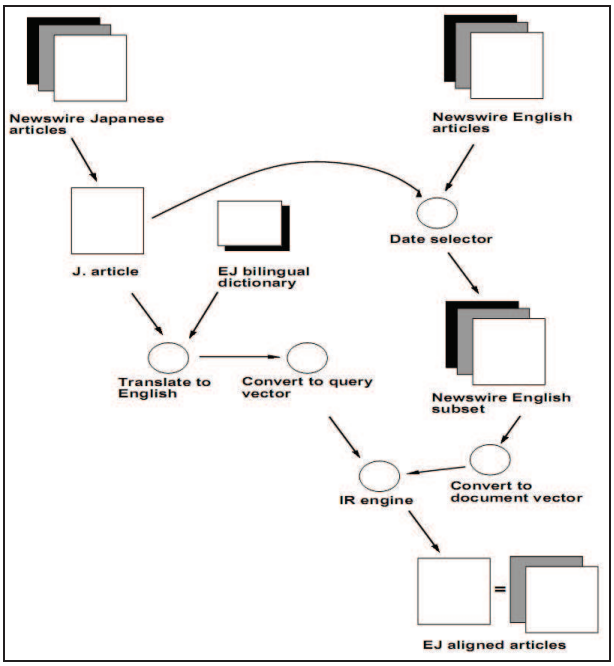


Figure 6-1 : Le processus d'extraction des documents parallèles de Collier et al. (src [Collier 1997])

Des travaux de Fung et Cheung [Fung 2004 a,b] concernent aussi l'utilisation de la technique RIT pour aligner des documents et des phrases parallèles à partir d'un corpus comparable initial. Les documents chinois sont segmentés en mots et leurs mots sont traduits en anglais par un dictionnaire disponible (nous le nommons le dictionnaire initial). Le document chinois traduit et le document anglais candidat sont représentés par des vecteurs de mots avec les poids du mot. La similarité est calculée pour toutes les paires de documents possibles, et la paire de documents dont la similarité est plus grande qu'un certain seuil est sélectionnée comme une paire comparable. La similarité est calculée à base de la TF, IDF et le score TF-IDF aussi (le produit de TF et IDF). De la même façon, pour aligner des phrases parallèles, chaque phrase est présentée en un vecteur de mots. Pour chaque paire de documents extraite, la similarité est calculée pour toutes les paires de phrases possibles dans cette paire de documents. Un autre seuil est utilisé pour extraire les paires de phrases parallèles. Les auteurs utilisent les données extraites pour mettre à jour le dictionnaire initial et le corpus comparable initial, et le processus itératif est effectué pour réaligner les paires de documents et les phrases parallèles.

Une autre application de la technique RIT pour aligner des documents et des phrases parallèles peut être trouvée dans les travaux de [Munteanu 2006b]. L'architecture générale de leur système d'extraction est présentée dans la Figure 6-2. Commençant avec deux corpus monolingues en documents, les auteurs effectuent l'extraction des paires de documents en utilisant l'outil de recherche d'information Lemur [Ogilvie 2001]. Le document chinois est traduit en anglais par un dictionnaire appris à partir d'un corpus parallèle initial chinois – anglais pour créer la requête en anglais. En appliquant l'outil Lemur avec le score de TF-IDF et dans une fenêtre de *n*- jours autour la date de publication du document chinois, les K-meilleurs documents anglais sont extraits depuis le corpus de documents anglais. Pour chaque document chinois et K-meilleurs documents anglais, toutes les paires de phrases possibles sont prises et passées vers un filtrage qui vérifie le rapport des longueurs de deux phrases, et le pourcentage de mots communs (à base d'un dictionnaire) entre deux phrases avec des seuils predéfinis (le rapport des longueurs < 2 et le pourcentage de mots communs > 0.5). Les paires de phrases résultantes de ce filtrage sont vérifiées par un classificateur à entropie maximale avec plusieurs caractéristiques telles que les longueurs des deux phrases, le pourcentage de mots communs, non communs, la longueur de la séquence de mots communs la plus longue, etc.

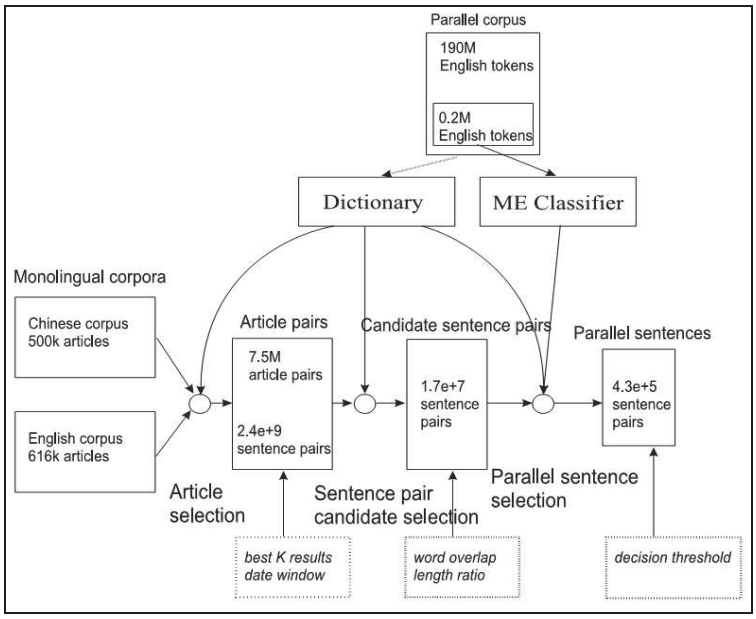


Figure 6-2 : Le processus d'extraction des documents et des phrases parallèles de Munteanu et al. (src [Munteanu 2006b])

Les méthodes présentées ci-dessus utilisent un modèle simple de traduction lexicale à base d'un dictionnaire bilingue pour traduire la requête, et utilisent une mesure de similarité de type TF-IDF (la fréquence d'un terme et la fréquence inverse de document).

[Abdul-Rauf 2009] présente une technique RIT similaire, mais un système de TA statistique est utilisé au lieu d'un dictionnaire bilingue, et le classificateur d'entropie maximale est remplacé par une métrique d'évaluation (TER) plus simple. Un système de TA statistique construit à partir d'un corpus de phrases parallèles initial est utilisé pour traduire le côté source du corpus de phrases. Chaque phrase après traduction est considérée comme la requête pour le processus de recherche d'information. L'outil de recherche d'information Lemur est utilisé aussi pour exécuter le processus de recherche d'information. Pour chaque requête, les 5- meilleures phrases du côté cible sont collectées. Les paires de phrases source et 5- meilleures phrases cibles sont filtrées encore par le rapport de la longueur, et une métrique d'évaluation est utilisée pour vérifier le degré de parallélisme entre les phrases. La métrique d'évaluation TER d'une paire de phrases est calculée et des seuils sont utilisés pour décider les paires de phrases parallèles. Les paires de phrases après ce filtrage sont ajoutées au corpus de phrases parallèles initial pour construire le système de TA statistique.

Dans le travail de [Sarikaya 2009], le processus d'extraction des paires de documents utilise aussi la technique RIT avec un système de TA statistique construit à partir d'un corpus de phrases parallèles initial et la mesure de la similarité est fondée sur le score de TF-IDF. Dans une paire de documents extraite, la métrique d'évaluation BLEU d'une paire de phrases source et cible est utilisée pour évaluer le degré de parallélisme entre deux phrases. Les auteurs ont proposé aussi d'itérer le processus et à chaque itération, les nouvelles données extraites sont ajoutées au corpus initial parallèle pour construire un nouveau système de traduction qui sera utilisé lors de l'itération suivante.

Toutes ces méthodes sont présentées comme des méthodes efficaces pour extraire des documents ou des phrases parallèles à partir d'un corpus comparable.

D'une manière générale, le processus d'extraction présenté dans les travaux ci-dessus peut être résumé par la Figure 6-3. Les méthodes diffèrent sur le module de traduction ou le score de similarité utilisé dans le module de recherche d'information. Le côté source (les documents ou les phrases) du corpus comparable est traduit par le module de traduction vers la langue cible. Le

module de traduction peut être un système de traduction automatique existant (limité par sa disponibilité) ; un modèle de traduction lexicale à base d'un dictionnaire bilingue ; ou un modèle de traduction probabiliste appris à partir de textes parallèles. La sortie du module de traduction est considérée comme la requête et est ensuite comparée avec le côté cible du corpus comparable par le module de recherche d'information. Le module de recherche d'information peut être un outil existant (comme *Lemur*) ou un système qui implémente des mesures classiques en RI telles que la TF-IDF. Les paires de documents obtenues sont alors alignées au niveau des phrases avec des techniques telles que la programmation dynamique. Les paires de phrases obtenues au final sont elles validées en utilisant, par exemple des métriques d'évaluation issues de la TA (BLEU, TER, etc.) ou des classificateurs à entropie maximale, etc.

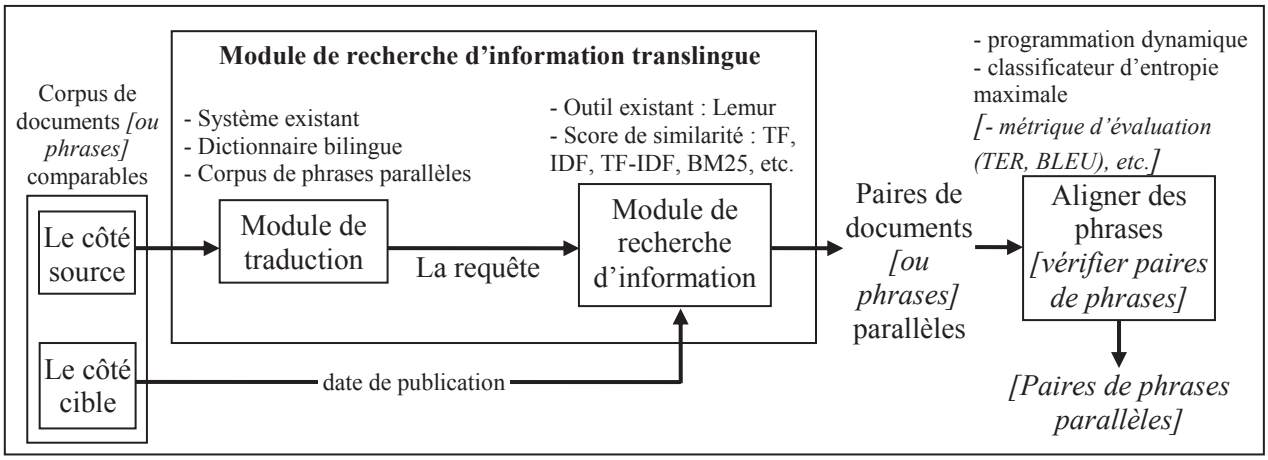


Figure 6-3 : La technique RIT appliquée pour l'extraction de données parallèles

6.2. Nos hypothèses

Au point de départ, nous n'avons qu'un seul site web multilingue de cette langue peu dotée, nous n'avons pas de données parallèles disponibles ou de données supplémentaires.

Premièrement, nous n'utilisons pas de module RIT pour aligner les paires de documents. Nous utilisons deux filtrages simples utilisant la date de publication et les mots spéciaux (qui sont présentés dans la section 5.2.2.1) pour réduire l'espace de recherche.

Deuxièmement, nous utilisons le module RIT pour aligner les paires de phrases. Les phrases du document source sont traduites en langue cible. Parce que nous nous concentrons sur l'extraction de phrases parallèles, nous comparons la phrase traduite automatiquement en langue cible (la requête) avec les phrases du document cible. Cette comparaison peut se faire avec des métriques mesurant la qualité de la traduction (par exemple les mesures automatiques présentées dans la section 2.6). Pour évaluer l'efficacité de différentes métriques, une expérience préliminaire a été effectuée et nous la présentons dans la section 6.2.1.

Troisièmement, les méthodes présentées dans la section 6.1 peuvent être considérées comme des méthodes semi supervisées, qui ont besoin d'un corpus de phrases parallèles initial (ou au moins un dictionnaire bilingue) pour construire le module de traduction. Nous supposons que dans le cas des langues peu dotées, ce corpus parallèle, même de petite taille, n'est pas disponible. Le point de départ est uniquement un corpus comparable (le corpus contient des paires de phrases parallèles, comparables et non parallèles). Dans la section 6.2.2, nous comparerons les méthodes semi-supervisées (corpus parallèle disponible au départ) et non supervisées (corpus comparable disponible au départ). Enfin, nous proposons un processus itératif, afin d'améliorer la qualité du système de traduction, puis d'augmenter le nombre de paires de phrases extraites correctement (expérience présentée dans la section 6.2.3).

Après ces expériences préliminaires, nous proposons un processus totalement non supervisé pour extraire les paires de phrases pertinentes (les paires de phrases parallèles et comparables) à partir d'un corpus comparable sans aucune information supplémentaire (section 6.3). Toutes les données pour construire le système sont extraites directement à partir de ce même corpus comparable initial.

6.2.1. Comparaison de plusieurs mesures automatiques dans le module de recherche d'information

Nous avons réalisé un test simple d'utilisation des mesures automatiques dans le module de recherche d'information d'un système semi supervisé (Figure 6-4).

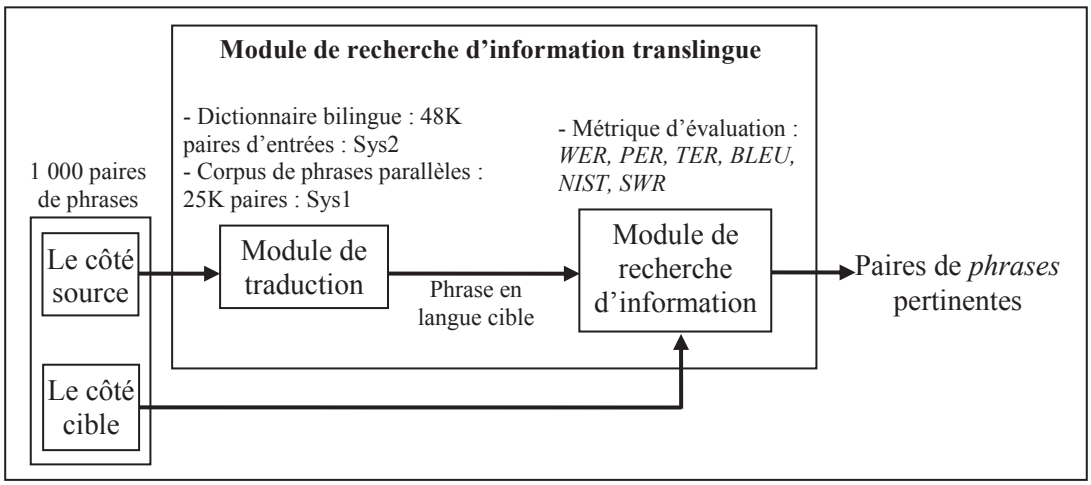


Figure 6-4 : Comparaison de plusieurs mesures automatiques dans le module de recherche d'information

1 000 paires de phrases bilingues françaises – vietnamiennes contenant des phrases parallèles, comparables et non pertinentes, étiquetées manuellement, ont été prélevées à partir du corpus VNA pour l'entrée du test. Nous avons trois groupes comme indiqués dans le Tableau 6-1 (voir les exemples de paires de phrases des trois groupes dans le Tableau 5-5).

Tableau 6-1 : Quatre groupes de phrases dans le test 6.2.1

Type	Nombre de paires
Paires de phrases parallèles	235
Paires de phrases comparables	348
Paires de phrases non pertinentes	417
Total	1 000

Le côté source du corpus a été traduit par un module de traduction construit :

- soit à partir d'un corpus de 25K paires de phrases « parallèles » français – vietnamien avec l'outil MOSES (paires de phrases extraites avec β=0,25 dans le Tableau 5-10 du chapitre précédent et différentes des phrases d'entrée) ; nous le nommons Sys1 ;
- soit à partir du dictionnaire bilingue français – vietnamiens de Ho Ngoc Duc[1], le dictionnaire de référence largement utilisé dans le TAL vietnamien, avec 48K paires d'entrées (chaque élément français correspond à *n* éléments en vietnamien) ; le dictionnaire a été transformé en une table de traduction en modifiant le script de MOSES ; nous le nommons Sys2.

[1] Ho Ngoc Duc, 1998. Vietnamese French Online Dictionary. http://www.informatik.uni-leipzig.de/~duc/Dict/index.html

La traduction de chaque phrase source a été appariée avec la phrase cible, et six métriques d'évaluation ont été calculées pour chaque paire de phrases. Nous avons utilisé six métriques d'évaluation : les mesures reposant sur des taux de mots erronés telles que WER, PER, TER ; et les mesures reposant sur des ressemblances avec des références telles que BLEU, NIST et notre métrique d'évaluation SWR. SWR (*Similar Word Rate*) est une mesure se fondant sur la similitude entre les hypothèses et la référence qui ne pénalise pas le réarrangement des mots. Ainsi la formule de notre SWR est la suivante :

$$SWR = \frac{2 * nombre\ de\ mots\ identiques\ entre\ hypothèse\ et\ référence}{longueur\ de\ l'\ hypothèse + longueur\ de\ la\ référence}$$

Ensuite, les distributions des scores d'évaluation pour trois groupes de phrases (235 paires pour chaque groupe) sont calculées et présentées dans la Figure 6-5. Le but est de vérifier si l'utilisation des mesures automatiques dans le module de recherche d'information permet de bien classifier les phrases.

Si nous comparons les deux systèmes Sys1 et Sys2, nous constatons que les distributions des scores ont tendance à avoir la même forme. L'utilisation des mesures reposant sur des taux de mots erronés tels que WER, PER, TER pour classifier des groupes de phrases semble cependant peu efficace. Par contre, les mesures reposant sur des ressemblances telles que BLEU, NIST et SWR peuvent classifier des phrases dans une certaine mesure avec un seuil. Parmi les métriques d'évaluation, NIST et SWR semblent donner les meilleurs scores.

6.2.2. Comparaison selon l'état initial : corpus parallèle ou parallèle bruité ?

L'un des objectifs de ce test est de savoir si nous pouvons construire un premier module de traduction acceptable pour la tâche RIT à partir d'un corpus parallèle bruité, versus un corpus vraiment parallèle. Afin de bien contrôler l'expérimentation, nous avons choisi le couple de langues français – anglais. Les paires de phrases parallèles ont été choisies dans le corpus Europarl, version 3 [Koehn 2005].

Deux modules de traduction ont été construits, l'un fondé sur un corpus vraiment parallèle C1 (Sys3), un autre basé sur un corpus parallèle bruité C2 (Sys4) (Figure 6-6). Le corpus « artificiel » C2 a été construit par l'introduction d'un grand nombre de paires de phrases non-parallèles dans les données parallèles (50 %) (il peut ainsi être considéré comme un corpus parallèle bruité). Pour être cohérents avec le cas réel des langues peu dotées, la taille des données expérimentales a été choisie relativement petite. Ainsi, le corpus C1 ne contient que 50 000 paires de phrases parallèles correctes. Le corpus C2 contient 25 000 paires de phrases parallèles correctes (retirées à partir de C1) et 25 000 paires de phrases non-parallèles (créées manuellement). Le corpus D, données d'entrée pour le processus d'extraction, a été construit, quant à lui, avec 10 000 paires de phrases parallèles correctes et 10 000 paires de phrases non-parallèles, différentes des paires de phrases de C1 et C2. Afin de contrôler la précision et le rappel du processus d'extraction, ces paires de phrases sont marquées comme étant parallèles ou non parallèles.

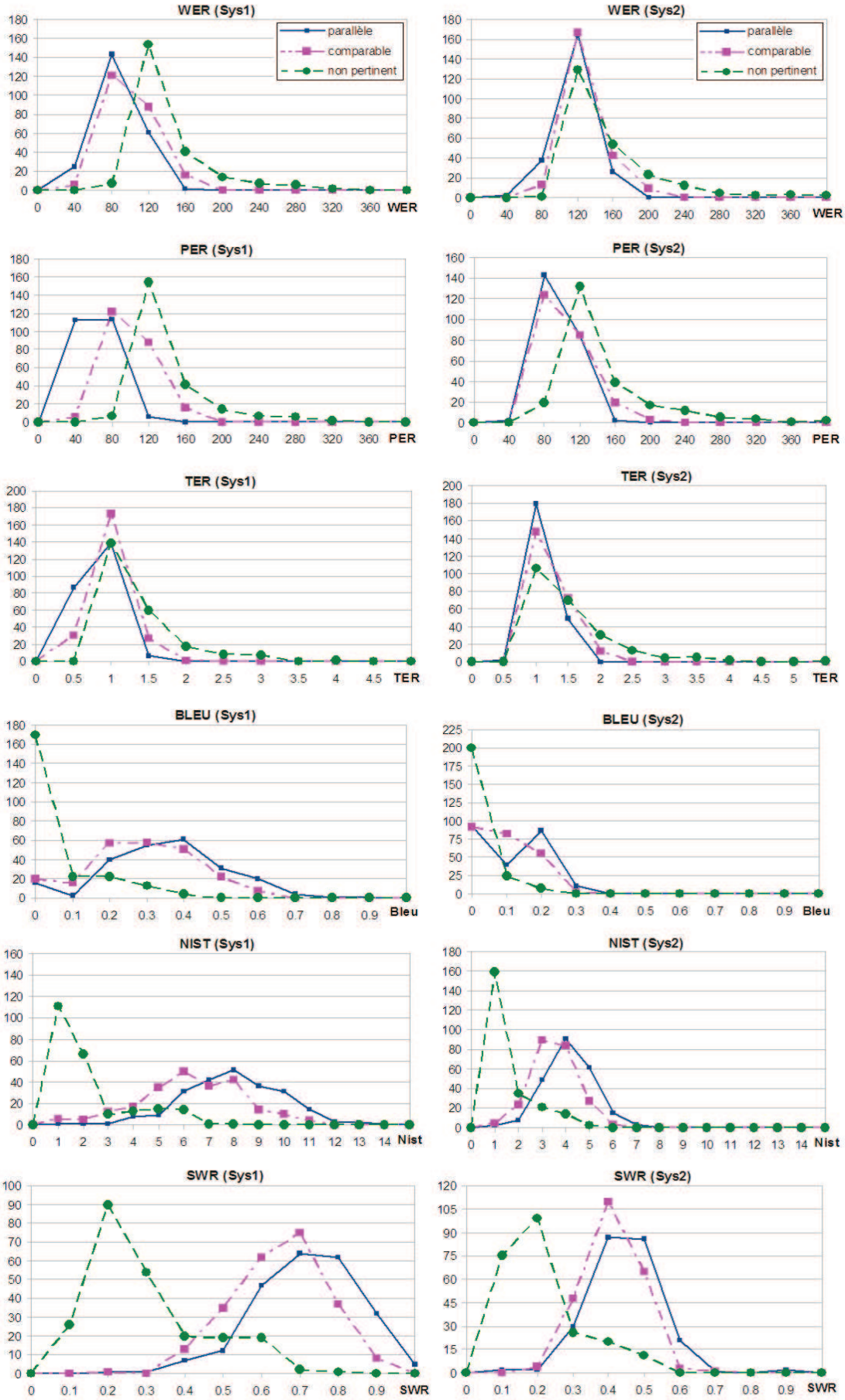


Figure 6-5 : Les distributions des scores d'évaluation pour quatre groupes de phrases dans le test 6.2.1

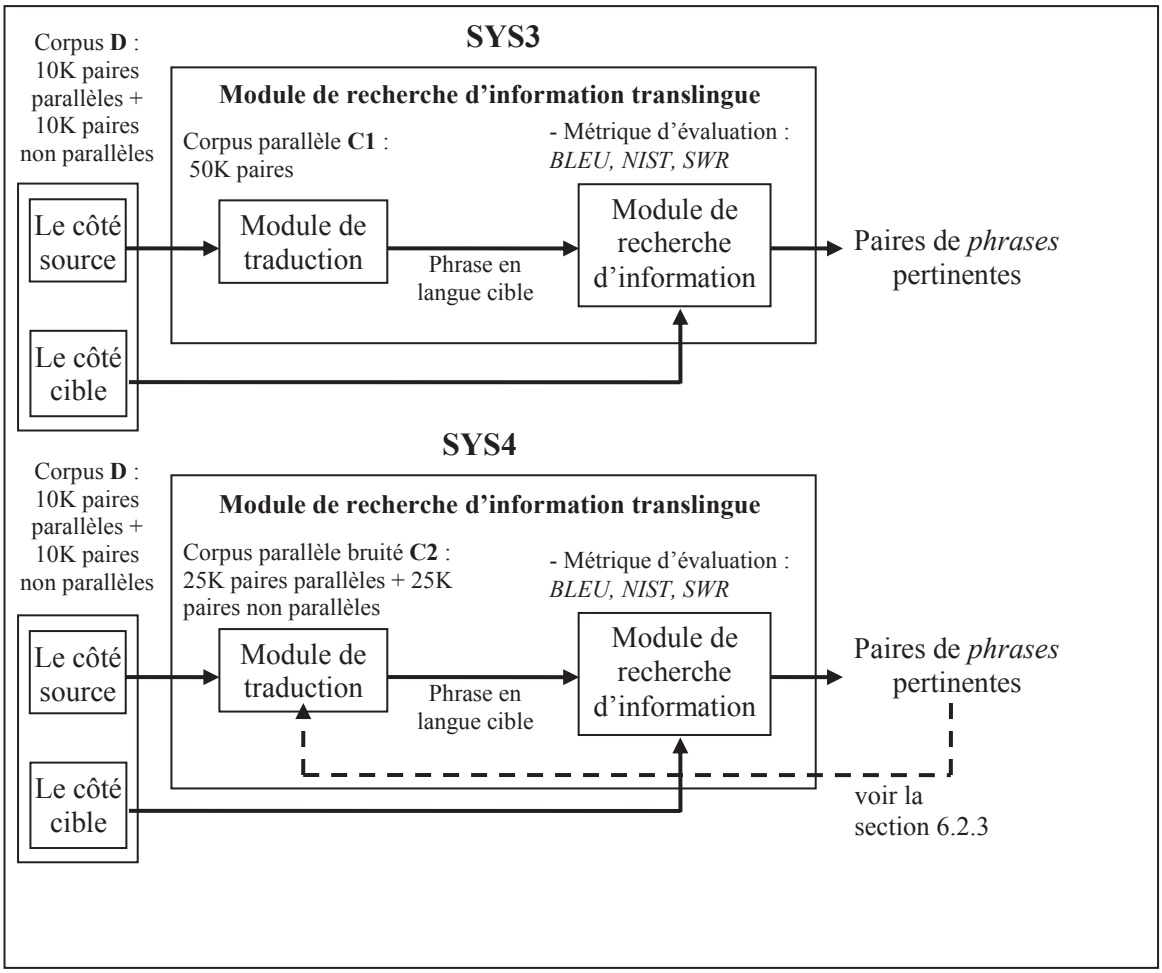


Figure 6-6 : Comparaison selon l'état initial : corpus parallèle ou parallèle bruité pour le module de recherche d'information

La première question à laquelle nous voulons répondre tout d'abord est de savoir si le module de traduction basé sur un corpus parallèle bruité peut être utilisé pour filtrer les données d'entrée aussi efficacement que le module de traduction basé sur un corpus parallèle propre. Pour répondre à cette question, le côté français du corpus D a été traduit par les systèmes de TA Sys3 et Sys4. Ensuite, les traductions ont été comparées avec le côté anglais du corpus D. Trois métriques d'évaluation ont été utilisées pour cette comparaison : BLEU, NIST et SWR (les métriques d'évaluation WER, PER, TER ont, quant à elles, été écartées). Ensuite, les distributions des scores d'évaluation pour les paires de phrases parallèles correctes et les paires de phrases non-parallèles sont calculées et présentées dans la Figure 6-7.

A partir de la Figure 6-7 nous pouvons faire des observations intéressantes : les distributions des scores ont la même forme entre les deux systèmes Sys3 et Sys4. En particulier, les distributions des scores pour les paires non-parallèles sont presque identiques pour les deux systèmes. Ainsi, un corpus parallèle bruité (issu d'un corpus comparable) peut remplacer un corpus parallèle dans la construction du module de traduction initial. Par conséquent, cette méthode peut être réellement appliquée dans le cas du manque de données parallèles initial.

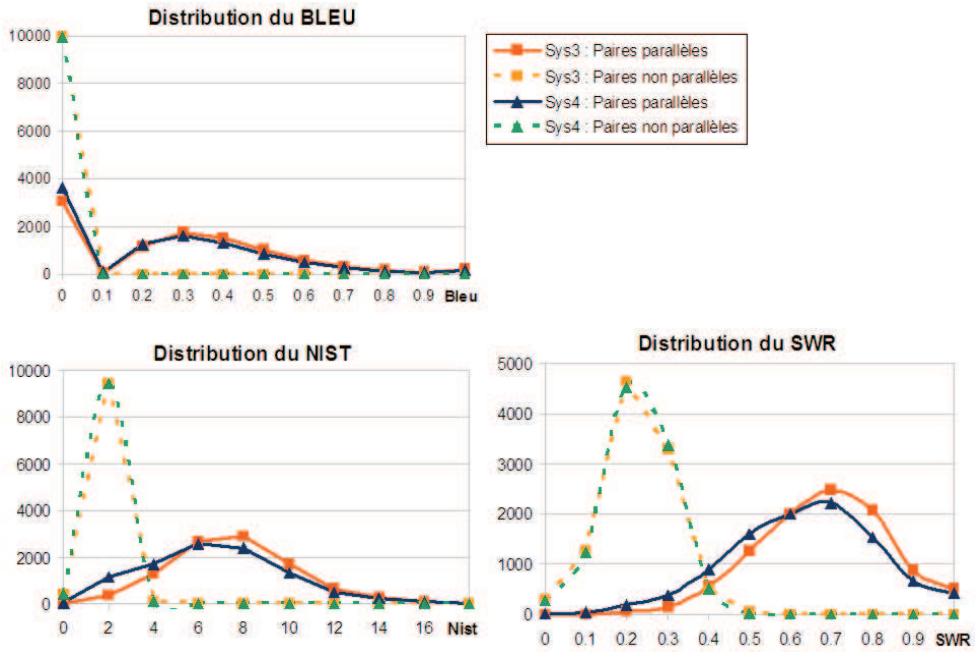


Figure 6-7 : Les distributions des scores d'évaluation pour les paires de phrases parallèles et non parallèles des deux systèmes Sys3 et Sys4

Pour filtrer les paires de phrases, un seuil a été utilisé. Une paire est considérée comme une paire parallèle si sa métrique d'évaluation (BLEU, NIST, SWR) est plus grande qu'un certain seuil. Un autre résultat important que nous pouvons voir est que le SWR, un score simple et facile à calculer, peut être considéré comme le meilleur score pour filtrer les paires de phrases parallèles correctes. La F-mesure du processus d'extraction atteint 97,10 % (pour Sys3) et 94,24 % (pour Sys4) lorsque le seuil de SWR est 0,35. Le Tableau 6-2 présente la précision et le rappel du filtrage des paires de phrases parallèles des deux systèmes Sys3 et Sys4.

Tableau 6-2 : Précision et rappel du filtrage des paires de phrases parallèles (avec 10K paires des phrases parallèles correctes)

Filtré par	Sys3 – corpus vraiment parallèle					Sys4 – corpus parallèle bruité				
	# de paires trouvées	# de paires correctes	Précision (%)	Rappel (%)	F1-mesure (%)	# de paires trouvées	# de paires correctes	Précision (%)	Rappel (%)	F1-mesure (%)
BLEU=0,1	6 908	6 892	99,76	68,92	81,52	6 233	6 218	99,75	62,18	76,61
NIST=4	8 350	8 347	99,96	83,47	90,97	7 110	7 108	99,97	71,08	83,08
SWR=0,3	10 342	9 785	94,61	97,85	96,20	10 110	9 468	93,65	94,68	94,16
SWR=0,35	9 764	9 595	98,27	95,95	97,10	9 236	9 064	98,14	90,64	94,24
SWR=0,4	9 390	9 333	99,39	93,33	96,27	8 682	8 629	99,38	86,29	92,37
SWR=0,5	8 191	8 187	99,95	81,87	90,00	7 154	7 150	99,94	71,50	83,36

6.2.3. Processus itératif d'extraction

Malgré les résultats encourageants du paragraphe précédent, on remarque toutefois que le résultat du filtrage du système Sys4 est légèrement plus faible que celui du système Sys3 (le nombre de paires de phrases correctes extraites est réduit). C'est pourquoi nous proposons un processus itératif, afin d'améliorer la qualité du module de traduction, puis d'augmenter le nombre de paires de phrases extraites correctement.

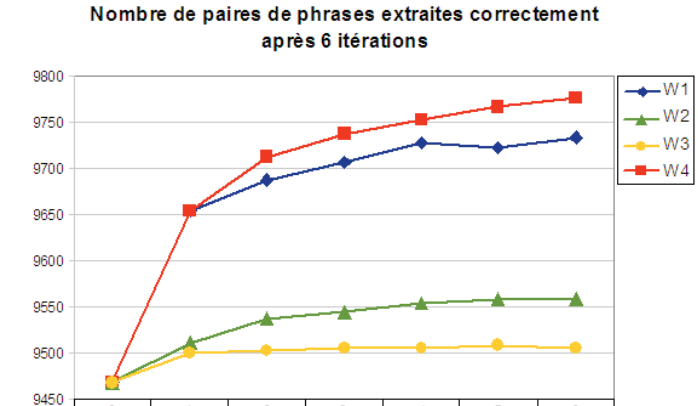
A l'issue de la première extraction, les paires de phrases extraites sont combinées avec les données ayant servi à développer le module de traduction de référence S_0, selon plusieurs manières pour créer un nouveau module de traduction. Nous espérons ici que les données

extraites vont permettre d'adapter le module de traduction vers le corpus d'extraction D, qui sera ainsi mieux traduit avec pour conséquence souhaitée une augmentation du nombre de paires de phrases correctes extraites.

Lors d'une nouvelle itération, on re-traduit le côté source par le nouveau module de traduction, re-calcule les métriques d'évaluation et re-filtre les paires de phrases parallèles. Pour utiliser les données extraites, quatre combinaisons différentes sont proposées :

- W1 : le module de traduction à la i^{eme} étape est entraîné par un corpus consistant en C2 et E_{i-1} (les données extraites à la dernière itération) ; E_0 étant les données extraites lorsque le module de traduction est entraîné par C2 seulement (S_0) ;
- W2 : à la i^{eme} itération, une nouvelle table de traduction est construite basée sur des données extraites E_{i-1} ; le module de traduction décode en utilisant deux tables de traduction combinées dans un modèle log-linéaire : S_0 et cette nouvelle table ; les poids associés à chacune des tables sont identiques ;
- W3 : la même combinaison que W2, mais la table de traduction de S_0 et la nouvelle table sont combinées en donnant plus d'importance aux données extraites E_{i-1} (par exemple 1:2) ;
- W4 : le module de traduction à la i^{eme} étape est entraîné par un corpus comprenant C2 et $E_0 \cup E_1 \cup ... \cup E_{i-1}$ (les données sont extraites aux itérations précédentes).

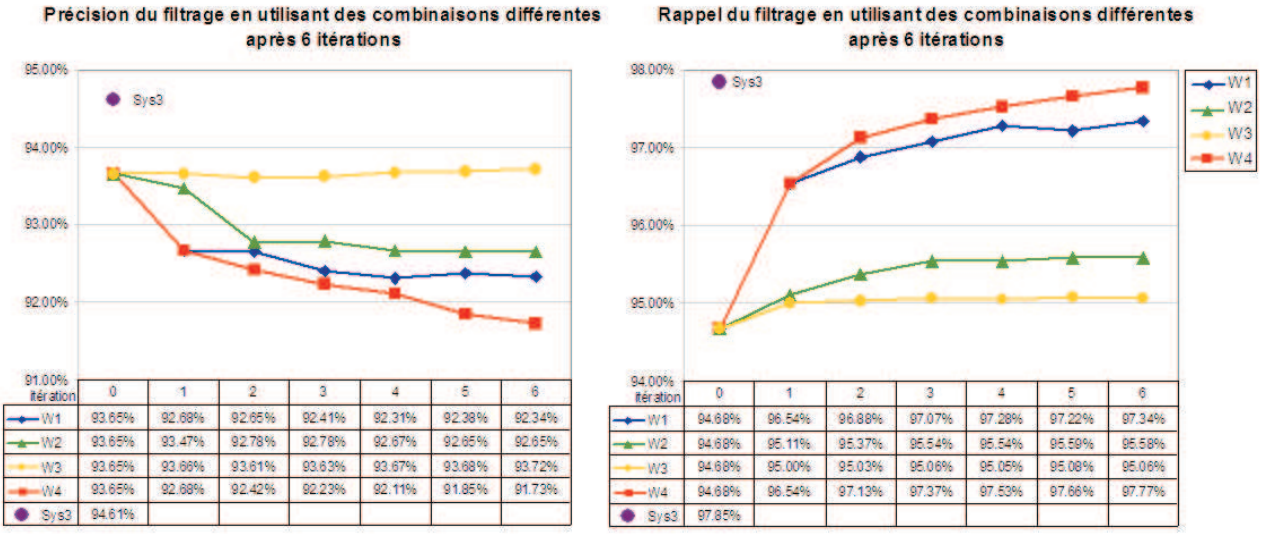
<u>Augmenter le nombre de paires de phrases correctes extraites</u> :
Les paires de phrases extraites sont combinées avec le système de référence S_0 des quatre manières différentes citées ci-dessus. L'expérience avec les itérations a été effectuée pour le système Sys4 (corpus initial bruité). Afin d'obtenir le nombre maximal de paires de phrases correctes extraites, pour toutes les itérations, nous avons choisi le score d'évaluation SWR avec un seuil égal à 0,3, ce qui a donné un rappel maximum de 94,68 % pour le système de référence. La Figure 6-8 présente le nombre de paires de phrases extraites correctement après 6 itérations pour les quatre combinaisons différentes W1, W2, W3 et W4. Le nombre de paires correctes extraites est augmenté dans tous les cas, mais la combinaison W4 introduit le plus grand nombre de paires de phrases correctes. Les combinaisons W2 et W3 ne sont pas vraiment efficaces parce que la deuxième table de traduction est construite à partir d'une faible quantité de paires de phrases. La combinaison W4 semble plus efficace que la combinaison W1 car les données d'apprentissage de W4 sont meilleures que celles de W1.



itération	0	1	2	3	4	5	6
W1	9468	9654	9688	9707	9728	9722	9734
W2	9468	9511	9537	9544	9554	9559	9558
W3	9468	9500	9503	9506	9505	9508	9506
W4	9468	9654	9713	9737	9753	9766	9777

Figure 6-8 : Nombre de paires de phrases extraites correctement après 6 itérations pour quatre combinaisons différentes

Augmenter la précision et le rappel du processus de filtrage :
La précision et le rappel de ces quatre combinaisons sont présentés dans la Figure 6-9. Parce que le processus de filtrage se concentre sur l'extraction du plus grand nombre de paires de phrases correctes extraites, la précision diminue. Toutefois, en utilisant la combinaison W4, le rappel après 6 itérations (97,77 %) atteint presque le rappel du système Sys3 (97,85 %).



Précision	itération 0	1	2	3	4	5	6
W1	93.65%	92.68%	92.65%	92.41%	92.31%	92.38%	92.34%
W2	93.65%	93.47%	92.78%	92.78%	92.67%	92.65%	92.65%
W3	93.65%	93.66%	93.61%	93.63%	93.67%	93.68%	93.72%
W4	93.65%	92.68%	92.42%	92.23%	92.11%	91.85%	91.73%
Sys3	94.61%						

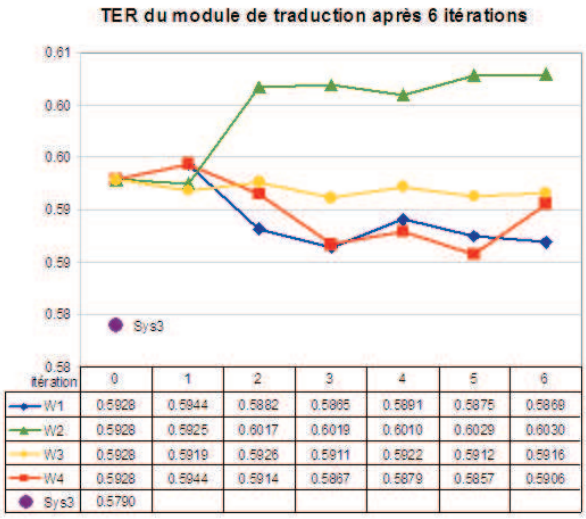
Rappel	itération 0	1	2	3	4	5	6
W1	94.68%	96.54%	96.88%	97.07%	97.28%	97.22%	97.34%
W2	94.68%	95.11%	95.37%	95.54%	95.54%	95.59%	95.58%
W3	94.68%	95.00%	95.03%	95.06%	95.05%	95.08%	95.06%
W4	94.68%	96.54%	97.13%	97.37%	97.53%	97.66%	97.77%
Sys3	97.85%						

Figure 6-9 : Précision et rappel du filtrage en utilisant des combinaisons différentes

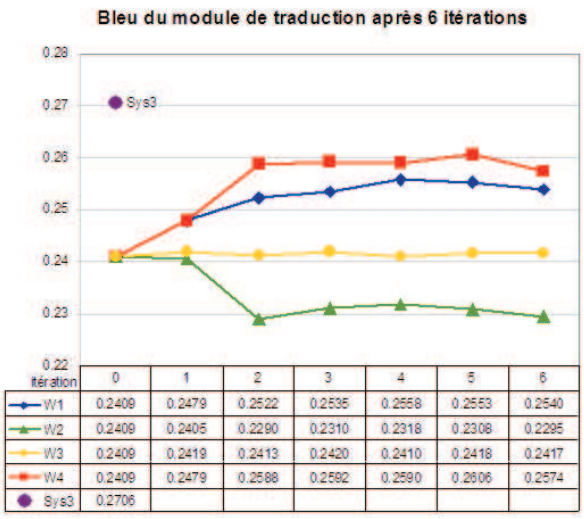
Évaluation du module de traduction :
La qualité du module de traduction est évaluée également. Un ensemble de test contenant 400 paires de phrases parallèles français – anglais qui ont été extraites du corpus Europarl, est utilisé. Chaque phrase française n'a qu'une seule référence en anglais. La qualité est calculée selon BLEU, NIST et TER. La Figure 6-10 donne les scores d'évaluation pour les systèmes après chaque itération.

L'évaluation du module de traduction révèle un résultat important : la qualité du module de traduction augmente rapidement au cours des premières itérations, mais diminue après. Nous pouvons expliquer que, dans les premières itérations, un grand nombre de paires de phrases parallèles nouvelles sont extraites et sont incluses dans le modèle de traduction. Toutefois, dans les itérations suivantes, lorsque la précision du processus d'extraction diminue, des paires de phrases non pertinentes sont ajoutées au système ; le modèle de traduction est alors dégradé et la qualité du module de traduction est réduite. Après environ 3 ou 5 itérations, le score BLEU peut augmenter d'environ 2 points lorsque les données ajoutées sont faibles. Nous noterons qu'il n'y a ici aucun réglage des paramètres du modèle log-linéaire à chaque itération (pas de données de développement utilisées, etc.).

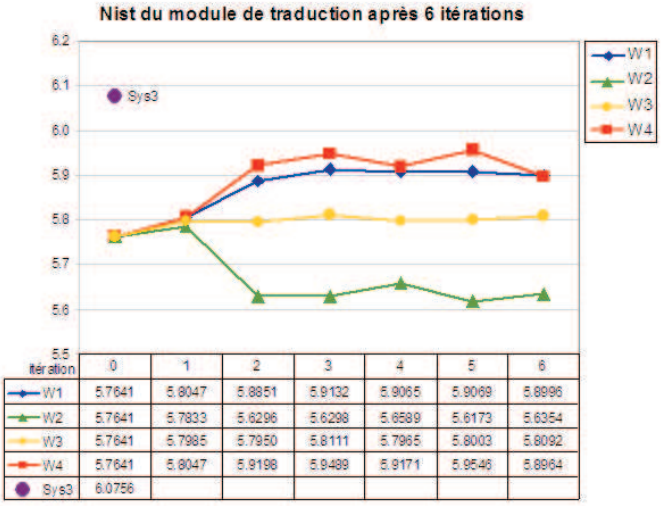
[Sarikaya 2009] présente une méthode semi-supervisée avec des itérations mais le système de TA initial est fondé sur un corpus parallèle. Ils utilisent la métrique d'évaluation BLEU pour le filtrage, et une combinaison semblable à notre combinaison W4. Cependant, à la différence de nos travaux systématiques, les auteurs ne fournissent pas une explication complète sur la façon dont ils choisissent la métrique d'évaluation, ou la méthode de combinaison (une seule méthode de combinaison est proposée) ; de plus, la fluctuation de la qualité du module de traduction après plusieurs itérations n'est pas mentionnée dans cette étude.



itération	0	1	2	3	4	5	6
W1	0.5928	0.5944	0.5882	0.5865	0.5891	0.5875	0.5869
W2	0.5928	0.5925	0.6017	0.6019	0.6010	0.6029	0.6030
W3	0.5928	0.5919	0.5926	0.5911	0.5922	0.5912	0.5916
W4	0.5928	0.5944	0.5914	0.5867	0.5879	0.5857	0.5906
Sys3	0.5790						



itération	0	1	2	3	4	5	6
W1	0.2409	0.2479	0.2522	0.2535	0.2558	0.2553	0.2540
W2	0.2409	0.2405	0.2290	0.2310	0.2318	0.2308	0.2295
W3	0.2409	0.2419	0.2413	0.2420	0.2410	0.2418	0.2417
W4	0.2409	0.2479	0.2588	0.2592	0.2590	0.2606	0.2574
Sys3	0.2706						



itération	0	1	2	3	4	5	6
W1	5.7641	5.8047	5.8851	5.9132	5.9065	5.9069	5.8996
W2	5.7641	5.7833	5.6296	5.6298	5.6589	5.6173	5.6354
W3	5.7641	5.7985	5.7950	5.8111	5.7965	5.8003	5.8092
W4	5.7641	5.8047	5.9198	5.9489	5.9171	5.9546	5.8964
Sys3	6.0756						

Figure 6-10 : Évaluation du module de traduction après itérations

6.3. Notre méthode d'extraction non supervisée

Les expériences présentées dans la section 6.2 nous encouragent à développer une méthode d'extraction vraiment non supervisée, nous la nommons *Méthode 2,* pour extraire les paires de phrases pertinentes à partir d'un corpus comparable donné sans aucune information supplémentaire. Les données pour construire le module de traduction initial seront extraites directement à partir du corpus comparable disponible.

Supposons que nous devons extraire les paires de phrases pertinentes à partir d'un corpus de documents comparables (un seul site web multilingue) et qu'il n'y a aucune information supplémentaire. Le processus d'extraction, résumé dans la Figure 6-11 ci-dessous, contient alors deux étapes : l'étape d'initialisation et l'étape d'extraction.

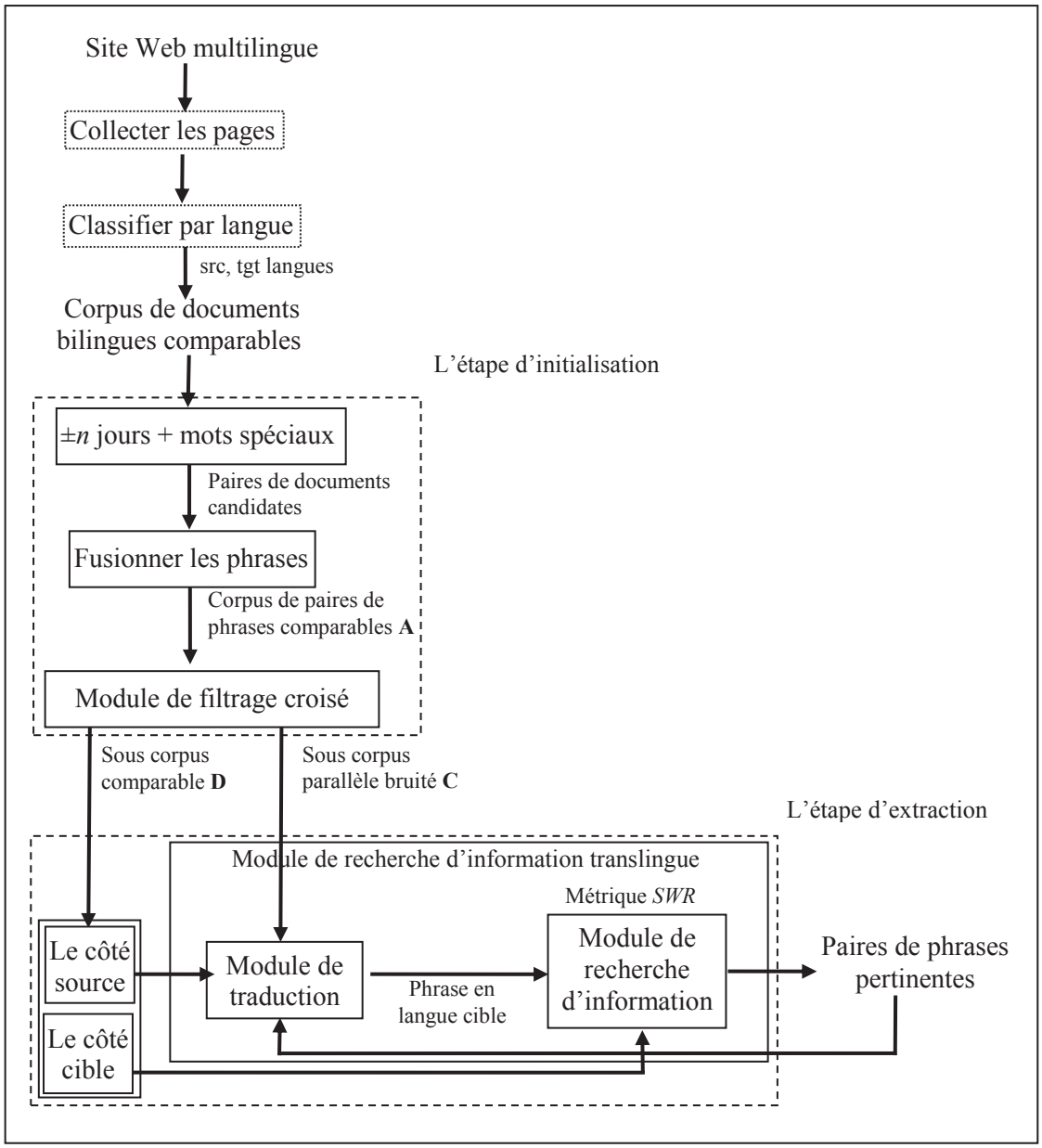


Figure 6-11 : Le processus d'extraction non supervisée pour extraire les paires de phrases pertinentes à partir d'un corpus comparable sans aucune information supplémentaire - Méthode 2

6.3.1. Étape d'initialisation – Module de filtrage croisé

Le but de cette étape est d'extraire un corpus assez fiable de phrases pertinentes à partir de corpus de documents comparables pour construire le module de traduction dans le module RIT. Deux filtrages simples utilisant la date de publication et les mots spéciaux sont utilisés pour l'extraction de paires de documents candidates. Ensuite, chaque phrase dans un document source est fusionnée avec toutes les phrases dans le document cible correspondant. Ainsi une paire de documents source (contenant *m* phrases*)* et cible (contenant *n* phrases*)* produit *m x n* paires de phrases. Nous appelons ce corpus le corpus de paires de phrases comparables bruitées *A*.

Nous proposons de diviser le corpus de paires de phrases comparables bruitées en deux ensembles : un corpus d'apprentissage initial *C* et un corpus à « fouiller » *D* (*A, C* et *D* sont indiquées dans la Figure 6-11). Pour garantir une qualité minimale de *C* (et par conséquent pour le module de traduction initial), nous proposons ci-dessous un processus de filtrage croisé (Figure 6-12) pour extraire le corpus *C* :

- diviser le corpus A en un nombre pair de sous-corpus contenant des paires de phrases différentes (par exemple 4 sous-corpus A1, A2, A3, A4) ;
- construire un système de traduction différent pour chaque sous-corpus (A1 → SMTA1, A2 → SMTA2, A3 → SMTA3, A4 → SMTA4) ;
- pour chaque deux sous corpus, la même manière de traduction et de filtrage par le score SWR est appliquée. Le côté source d'un sous-corpus (par exemple A1) est traduit par le système de traduction d'un autre corpus (par exemple SMTA2) ; ensuite, les traductions sorties sont comparées avec le côté cible (du corpus A1) et les paires de phrases dont la métrique d'évaluation SWR est plus grande que 0,45 sont filtrées (un seuil élevé de SWR égal à 0,45 est choisi pour assurer la fiabilité des paires de phrases extraites (assure une bonne précision selon la Figure 6-7) et un nombre acceptable de paires pour construire le système de traduction) ; appliquer la même manière pour chaque paire (A1, SMTA2), (A2, SMTA1), (A3, SMTA4), (A4, SMTA3), etc. ;
- les paires de phrases extraites C1, C2, C3, C4, etc., et leur union est considérée comme suffisamment fiable pour servir comme corpus comparable initial C ; le reste est traité comme le corpus D.

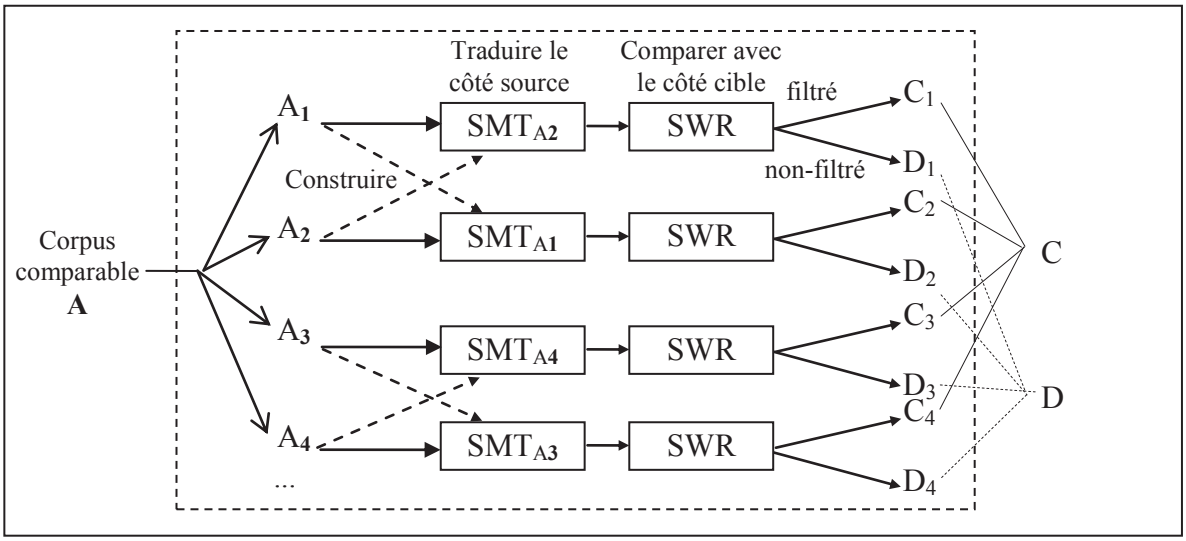


Figure 6-12: Le module de filtrage croisé dans l'étape d'initialisation

6.3.2. Étape d'extraction – Module de recherche d'information translingue

Cette étape utilise le module RIT pour aligner les paires de phrases. Le module de traduction est entraîné par le corpus C. Ensuite, toutes les phrases source sont traduites en langue cible. La traduction sortie est comparée avec la phrase cible appariée par le métrique d'évaluation SWR. Une paire est considérée comme une paire pertinente si son score SWR est plus grand qu'un certain seuil. Un processus itératif est utilisé afin d'améliorer la qualité du module de traduction, puis d'augmenter le nombre de paires de phrases extraites. Les paires de phrases pertinentes extraites après chaque itération sont ajoutées dans le corpus d'apprentissage C. Le module de traduction est re-entraîné par $C \cup E_0 \cup E_1 \cup ... \cup E_{i-1}$ (E_i est les données extraites de chaque itération *i*). Le processus (re-traduire le côté source + re-calculer le score + re-filtrer les paires de phrases) est répété jusqu'à $E_{n-1}=E_n$ (pas de nouvelles paires de phrases extraites). E_n est alors considéré comme l'ensemble des paires de phrases pertinentes extraites.

6.4. Application de la méthode d'apprentissage non supervisée à des couples de langue peu dotés

Notre méthode d'extraction non supervisée (*Méthode 2*) a été appliquée sur les données bilingues vietnamien – français et vietnamien – anglais du corpus VNA. Une comparaison avec notre première méthode d'extraction utilisant des informations lexicales (*Méthode 1*, présentée dans le chapitre 5) est également réalisée pour le couple vietnamien – français.

6.4.1. Application pour le couple de langues vietnamien – français

La *Méthode 2* a été appliquée sur les données bilingues vietnamien – français E_{all} (pour être comparable avec la *Méthode 1*). Les deux filtrages simples utilisant la date de publication et les mots spéciaux nous donnent 21 446 paires de documents possibles (voir la 2ème ligne du Tableau 5-4). Après filtrage selon la date de publication (±2 jours), chaque document français est apparié avec environ 500 documents vietnamiens. Et après le deuxième filtrage (mots spéciaux), il y a environ 2 documents candidats appariés avec chaque document français. La probabilité de trouver des paires de documents pertinentes après ces deux filtrages est très grande (dans le test reporté Tableau 5-3, le rappel est égal à 100 %).

Pour cette expérience, l'ensemble de test contient 400 paires de phrases parallèles vérifiées manuellement (190 paires de phrases de l'ensemble de DEV et 210 paires de phrases de l'ensemble TST dans l'expérience de la *Méthode 1* ; voir le Tableau 5-6). Nous n'avons pas utilisé ici le processus de réglage des poids des modèles log-linéaires dans le module de traduction ; nous évaluons donc la qualité de la table de traduction obtenue à poids constants. Les 100 paires de documents qui constituent le corpus de test ont bien sûr été écartées des 21 446 paires de documents possibles. Nous avons ainsi un ensemble de 21 346 paires de documents possibles vietnamien – français.

Chaque phrase dans un document français a été fusionnée avec toutes les phrases dans le document vietnamien candidates. Nous avons obtenu un corpus comparable de 1 442 448 paires de phrases. Nous avons gardé seulement les paires avec le rapport de longueur (compté en mots) de la phrase française et de la phrase vietnamienne entre 0,8 et 1,3 (ce filtrage nous aide à réduire le nombre de phrases traitées, sans influence sur le résultat parce que les paires de phrases ayant une longueur très différente seront de toute façon éliminées après par le score SWR). Nous avons au final obtenu un corpus comparable de 345 575 paires de phrases (nommé $A_{vn\text{-}fr}$).

Création du système de traduction initial

Le processus de filtrage croisé a été appliqué sur le corpus $A_{vn\text{-}fr}$. Le corpus $A_{vn\text{-}fr}$ a été divisé en 4 sous-corpus contenant des paires de phrases différentes : A_1 (85 011 paires de phrases), A_2 (85 008 paires de phrases), A_3 (86 529 paires de phrases) et A_4 (89 027 paires de phrases).

4 systèmes de TA différents sont ainsi construits : $A_1 \rightarrow SMT_{A1}$, $A_2 \rightarrow SMT_{A2}$, $A_3 \rightarrow SMT_{A3}$ et $A_4 \rightarrow SMT_{A4}$. Nous appliquons la méthode proposée pour chaque paire (A_1, SMT_{A2}), (A_2, SMT_{A1}), (A_3, SMT_{A4}) et (A_4, SMT_{A3}), nous obtenons ainsi les paires de phrases extraites C_1, C_2, C_3, C_4 dans le Tableau 6-3.

Tableau 6-3 : Données extraites pour C et D

Sous-corpus	Traduit par	Nombre de paires $C_{vn\text{-}fr}$	Nombre de paires $D_{vn\text{-}fr}$
A_1	SMT_{A2}	C_1 : 2 916	D_1 : 82 095
A_2	SMT_{A1}	C_2 : 3 495	D_2 : 81 513
A_3	SMT_{A4}	C_3 : 3 820	D_3 : 82 709
A_4	SMT_{A3}	C_4 : 3 892	D_4 : 85 135

Après cette étape, nous avons obtenu un corpus $C_{vn\text{-}fr}$ contenant 14 123 paires de phrases, et un corpus $D_{vn\text{-}fr}$ contenant les 331 452 paires de phrases restantes. Le module de recherche d'information translingue est ensuite appliqué sur $C_{vn\text{-}fr}$ et $D_{vn\text{-}fr}$ pour extraire plus de paires de phrases pertinentes.

Application du module de recherche d'information translingue

Le premier module de traduction vietnamien – français S_0 a été construit à partir du corpus d'apprentissage $C_{vn\text{-}fr}$ de 14 123 paires de phrases. Le corpus $D_{vn\text{-}fr}$ contient 331 452 paires de phrases. Les phrases en vietnamien sont segmentées en syllabes (pas de segmentation en mots). Le seuil utilisé pour le score SWR est 0,3 (valeur ayant donné le meilleur rappel sur le couple français – anglais).

Après 5 itérations, nous extrayons 25 635 paires à partir du corpus $D_{vn\text{-}fr}$. Le nombre total de paires de phrases extraites à partir du corpus $A_{vn\text{-}fr} = C_{vn\text{-}fr} + 25\,635 = 39\,758$ paires. La qualité du système de TA est évaluée également sur un ensemble de test de 400 paires de phrases parallèles. Chaque phrase vietnamienne n'a qu'une seule référence en français. Le nombre de paires de phrases extraites et les scores d'évaluation à chaque itération sont reportés dans la Figure 6-13.

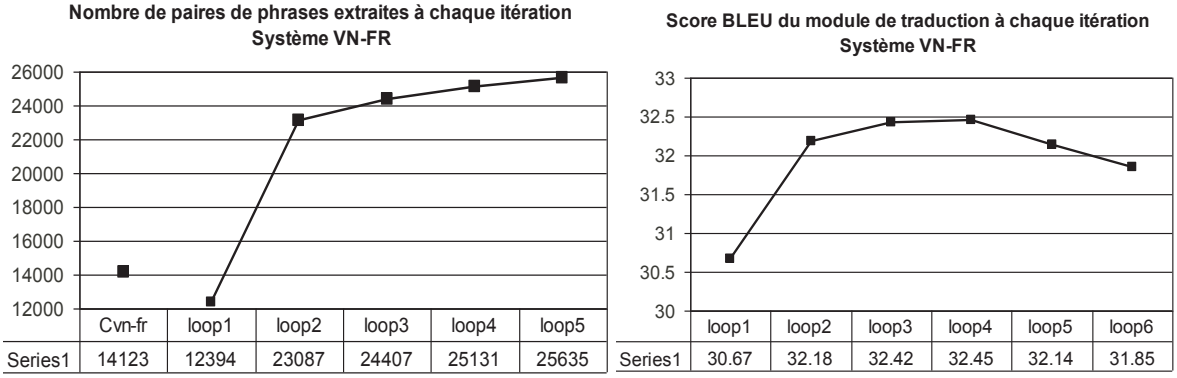


Figure 6-13 : Le nombre de paires de phrases extraites et les scores BLEU du module de traduction à chaque itération dans le système VN-FR (Méthode2-Exp1)

Les résultats dans ce cas sont similaires à ceux obtenus lors des expériences préliminaires pour le couple de langues français – anglais (voir les sections 6.2.2 et 6.2.3) : le nombre de paires de phrases extraites augmente après quelques itérations, la qualité du système de TA augmente également lors des premières itérations et diminue par la suite. Nous appelons cette expérience *Méthode 2 - Exp 1*.

Nous notons que la métrique SWR se fonde sur la similitude entre les hypothèses et la référence, et elle mesure tous les mots identiques, même les mots d'arrêt (*stop words*). Nous proposons alors de faire l'expérience d'enlever les mots d'arrêt et les ponctuations pendant le calcul du score SWR. Le même processus est appliqué sur le corpus $A_{vn\text{-}fr}$. Après le filtrage croisé, nous n'avons obtenu que 4 067 paires de phrases pour construire le module de traduction : $C_{vn\text{-}fr} = 4\,076$ paires et $D_{vn\text{-}fr} = 341\,449$ paires. Nous avons réalisé le processus d'extraction, et les résultats sont présentés dans la Figure 6-14. Nous appelons cette expérience le *Méthode 2 – Exp 2*.

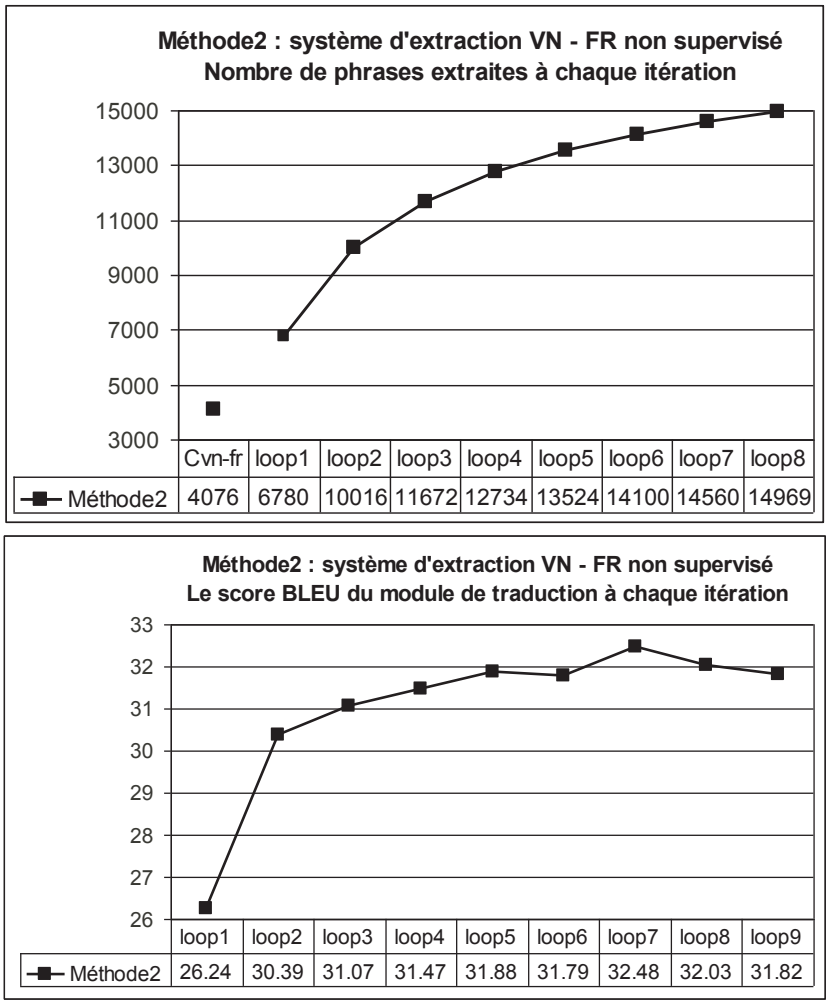
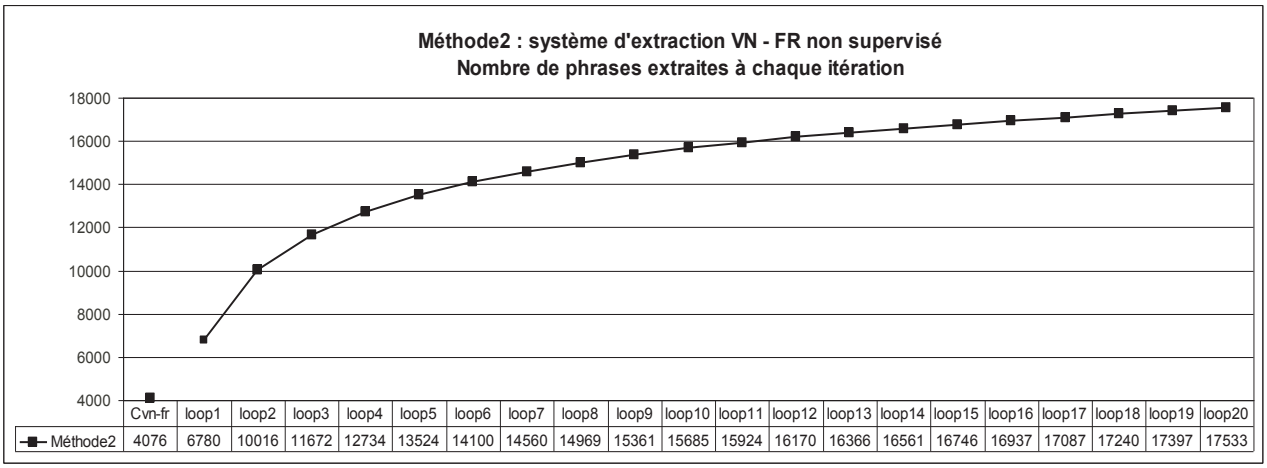


Figure 6-14 : Le nombre de paires de phrases extraites et les scores BLEU du module de traduction à chaque itération dans le système VN-FR (Méthode 2 – Exp 2)

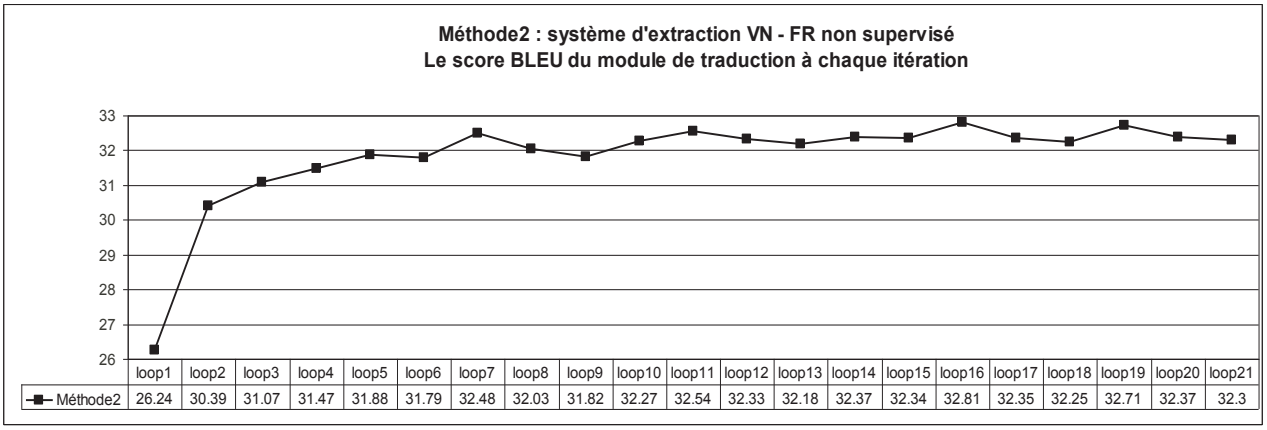
Le nombre de paires de phrases extraites dans la *Méthode 2 – Exp 2* est moins grand que dans le cas précédent de la *Méthode 2 – Exp 1*, à cause de la métrique SWR qui devient plus sévère. Cependant, le résultat que la *Méthode 2 – Exp 2* nous apporte est intéressant. Après 5 itérations, nous avons obtenu plus de 39 000 paires de phrases dans la *Méthode 2 – Exp 1* et le score BLEU maximal du module de traduction est 32,45. Par ailleurs, après 6 itérations dans la *Méthode 2 – Exp 2*, plus de 18 000 paires de phrases sont obtenues (= 4 076 + 14 100) et le score BLEU maximal du module de traduction est atteint déjà à 32,48 (loop7).

Nous pouvons dire que la qualité de données extraites dans la *Méthode 2 – Exp 2* est meilleure que celle de la *Méthode 2 – Exp 1*. La métrique SWR plus sévère nous apporte un meilleur résultat. Dans les expériences suivantes, nous avons donc utilisé la métrique SWR sans compter les mots d'arrêt.

Nous avons continué la *Méthode 2 – Exp 2* jusqu'à la convergence (pas de nouvelles paires extraites). Le nombre de paires extraites à chaque itération est réduit, et le processus d'extraction s'arrête à l'itération 20. Le score BLEU du module de traduction augmente rapidement au cours des 6 – 7 premières itérations, et augmente plus légèrement pour les itérations suivantes. Le score BLEU maximal atteint est 32,81 à l'itération 16.



	Cvn-fr	loop1	loop2	loop3	loop4	loop5	loop6	loop7	loop8	loop9	loop10	loop11	loop12	loop13	loop14	loop15	loop16	loop17	loop18	loop19	loop20
Méthode2	4076	6780	10016	11672	12734	13524	14100	14560	14969	15361	15685	15924	16170	16366	16561	16746	16937	17087	17240	17397	17533



	loop1	loop2	loop3	loop4	loop5	loop6	loop7	loop8	loop9	loop10	loop11	loop12	loop13	loop14	loop15	loop16	loop17	loop18	loop19	loop20	loop21
Méthode2	26.24	30.39	31.07	31.47	31.88	31.79	32.48	32.03	31.82	32.27	32.54	32.33	32.18	32.37	32.34	32.81	32.35	32.25	32.71	32.37	32.3

Figure 6-15 : Le nombre de paires de phrases extraites et les scores BLEU du module de traduction après toutes les itérations dans le système VN-FR (Méthode2-Exp2)

Pour une analyse plus poussée, les précisions du processus d'extraction à chaque itération sont estimées (ici, nous vérifions les 8 premières itérations). Le Tableau 6-4 présente cette évaluation. A chaque itération, la qualité des paires de phrases extraites est évaluée manuellement. 100 paires de phrases sont choisies au hasard parmi celles extraites à chaque itération, et sont classées dans 3 groupes : parallèles, comparables, et non pertinentes. Les colonnes (3), (4), (5), (6) dans le Tableau 6-4 présentent l'évaluation manuelle sur ces 100 paires de phrases. Les colonnes suivantes (7), (8), (9), (10) donnent une estimation pour le corpus entier de paires de phrases extraites. Les colonnes (7) et (9) estiment le nombre de paires pertinentes/non pertinentes du corpus entier à chaque itération. Leur valeur est égale au nombre de paires extraites multiplié par la probabilité de paires de phrases pertinentes/non pertinentes. Les colonnes (8) et (10) indiquent le nombre de nouvelles paires de phrases pertinentes/non pertinentes détectées à chaque itération. Leur valeur $\Delta 1$ ($\Delta 2$) est la soustraction entre le nombre de paires pertinentes/non pertinentes à l'itération *i* et le nombre de paires pertinentes/non pertinentes à l'itération précédente *i-1*.

Tableau 6-4 : L'évaluation manuelle de la qualité de paires de phrases extraites VN – FR : Méthode2-Exp2

(1) après	(2) # paires extraites	Evaluation manuelle sur 100 paires de phrases				Estimation provisoire pour le corpus entier			
		(3) % parallèle	(4) % comparable	(5) % non pertinent	(6) Précision = (3)+(4)	(7) # paires pertinents = (2)x(6)	(8) Δ1 = (7) à itér i – (7) à itér (i-1)	(9) # paires non pertinents = (2) x (5)	(10) Δ2 = (9) à itér i – (9) à itér (i-1)
$C_{vn\text{-}fr}$	4 076	66	32	2	98	3 995	–	81	–
itér1	6 780	60	28	12	88	5 966	5 966	814	814
itér2	10 016	56	32	12	88	8 814	2 848	1 202	388
itér3	11 672	53	34	13	87	10 155	1 341	1 517	315
itér4	12 734	52	34	14	86	10 951	796	1 783	266
itér5	13 524	51	34	15	85	11 495	544	2 029	246
itér6	14 100	51	35	14	86	12 126	631	1 974	-55
itér7	14 560	50	35	15	85	12 376	250	2 184	210
itér8	14 969	49	35	16	84	12 574	198	2 395	211

Cette évaluation peut nous aider à expliquer le problème que la qualité du module de traduction augmente au cours des premières itérations, mais plafonne après. Premièrement, le nombre de paires de phrases pertinentes (estimées) augmente à chaque itération (colonne 7), mais le nombre de paires de phrases non pertinentes augmente aussi (colonne 9). Deuxièmement, dans les premières itérations (1 – 6), le nombre de nouvelles paires pertinentes Δ1 est très grand en comparaison du nombre de nouvelles paires non pertinentes Δ2 (les colonnes 8 et 10). Lorsque les nouvelles paires sont incluses dans le modèle de traduction, la qualité du module de traduction augmente. Ensuite, dans les itérations 7 et 8, le nombre de nouvelles paires pertinentes Δ1 diminue. Même le nombre de nouvelles paires non pertinentes Δ2 est comparable au (ou plus grand que le) nombre de nouvelles paires pertinentes Δ1. Le modèle de traduction est alors dégradé et la qualité du module de traduction est réduite.

Dans les itérations suivantes (9 – 20), le nombre de nouvelles paires extraites est petit, la qualité du module de traduction fluctue. Donc dans les expériences suivantes, nous ne continuons pas la procédure d'extraction jusqu'à la convergence, mais nous continuons jusqu'à la présence de cette fluctuation.

6.4.2. Comparaison entre notre deuxième méthode non supervisée et notre première méthode d'extraction utilisant des informations lexicales

Dans l'application pour le couple de langues vietnamien – français, notre première méthode d'extraction utilisant des informations lexicales *Méthode 1* et notre deuxième méthode non supervisée *Méthode 2 – Exp2* ont été appliquées sur un même corpus comparable de VNA.

La *Méthode 1* nous donne 50 322 paires de phrases, alors que la *Méthode 2* nous donne 18 176 paires de phrases (après l'itération 6). Les nombres de paires extraites des deux méthodes ne peuvent être véritablement comparés car la *Méthode 1* extrait aussi des paires de phrases du type *2:1, 1:2, 2:2* (revoir la section 5.3.2), alors que la *Méthode 2* n'extrait que des paires de phrases du type *1:1*.

Cependant, le pourcentage de paires de phrases pertinentes de la *Méthode 2* est plus grand que celui de la *Méthode 1.* La *Méthode 1* nous donne 42 % paires de phrases parallèles, 38 % paires de phrases comparables et 20 % paires de phrases non pertinentes (section 5.3.3). 18 176 paires

de phrases de la *Méthode 2* correspondant à 4 076 paires de S_0 et 14 100 paires extraites (Tableau 6-4, la première et la 7ème ligne). En moyenne, 55 % paires de phrases parallèles, 33 % paires de phrases comparables et 12 % paires de phrases non pertinentes (Tableau 6-5).

Ensuite, pour comparer ces deux méthodes, les systèmes de traduction du vietnamien (segmenté en syllabe) vers le français sont construits avec les données extraites des deux méthodes, et les performances du système de traduction sont estimées sur un même corpus de test. Les résultats sont donnés aussi dans le Tableau 6-5. Nous notons que le processus de réglage des poids des modèles log-linéaires n'est pas utilisé et l'ensemble de test contient 400 paires de phrases parallèles vérifiées manuellement (ce qui explique que le score du système de traduction pour la *Méthode 1* est légèrement différent de celui montré dans le Tableau 5-8 où le processus de réglage des poids des modèles log-linéaires était utilisé).

Tableau 6-5 : Comparaison entre les méthodes d'extraction Méthode 1 et Méthode 2

Méthode	**Ressources requises**	**# de paires de phrases extraites**	**% parallèles**	**% comparable**	**% non pertinent**	**BLEU**
Méthode1	données parallèles minimales	50 322 (m : n)	42	38	20	32,74
Méthode2 (itération 6)	aucune donnée supplémentaire	18 176 (1 : 1) =4 076+ 14 100	55	33	12	32,48

Bien que le nombre de paires de phrases extraites dans la *Méthode 2* soit beaucoup plus faible que celui obtenu dans la *Méthode 1*, la qualité du système de TA est comparable. La *Méthode 1* dépend cependant de données / informations supplémentaires telles que la qualité du dictionnaire bilingue ou des règles heuristiques. A partir de ces résultats, nous pouvons dire que la méthode non supervisée *Méthode 2* a été appliquée avec succès pour le couple de langues vietnamien – français.

Tableau 6-6 : Un exemple de la traduction d'une phrase vietnamienne vers français par la Méthode 1 et la Méthode 2 à quelques itérations

Référence en français	le chef d'état vietnamien a proposé que le vietnam et l'australie renforcent les dialogues politiques de haut rang , ainsi que les contacts
Méthode 1	le président vietnamien a demandé ***X*** le vietnam et l'australie **renforcer les dialogues politiques** de haut rang , ainsi que des rencontres
Méthode 2-itér1	le président **du** vietnam ***X*** **sur** le vietnam et l'australie d'intensifier les **dialogue politique** de haut rang , ainsi que des rencontres
Méthode 2-itér2	le président vietnamien a proposé que le vietnam et l'australie **de renforcer** davantage leur dialogue politique de haut rang , ainsi que des rencontres
Méthode 2-itér7	**le président de l'état vietnamien** a proposé ***X*** le vietnam et l'australie **,** de renforcer davantage leur dialogue politique de haut rang ainsi que des rencontres
Méthode 2-itér8	**le président de l'état vietnamien** a demandé **au** vietnam et l'australie d'intensifier **les dialogues politiques** de haut rang , ainsi que des rencontres

Le Tableau 6-6 montre un exemple de la traduction d'une phrase vietnamienne vers français par le module de traduction construit dans la *Méthode 1* et la *Méthode 2*. Les textes en **gras** d'une phrase présentent les groupes de mots qui sont bien traduits, en comparaison des textes d'autres phrases. Les textes soulignés et en **gras** sont des parties mal traduites et le symbole ***X*** présente le manque de mots. La traduction de la *Méthode 1* est acceptable mais il reste encore deux erreurs. La traduction à la première itération de la *Méthode 2* contient plusieures erreurs, à cause de la qualité faible du module de traduction initial. Mais à partir de la deuxième itération, les traductions deviennent meilleures. Et à l'itération 8, deux groupes de mots bien traduits sont trouvés.

6.4.3. Application pour le couple de langues vietnamien – anglais

Le corpus de VNA contient des articles en anglais aussi (32 795 documents). Nous avons appliqué la *Méthode 2* pour fouiller les paires de phrases pertinentes vietnamienne – anglaise à partir de deux corpus monolingues vietnamien et anglais.

Après avoir appliqué les deux filtrages de date de publication et de mots spéciaux, et avoir fusionné les phrases entre deux documents correspondants, nous avons obtenu un corpus de paires de phrases comparables bruitées $A_{vn\text{-}en}$ qui contient 479 865 paires de phrases. Le filtrage croisé est réalisé avec un seuil de SWR égal à 0,35 (pour assurer la fiabilité des paires de phrases extraites et un nombre acceptable de paires pour construire le système de traduction, qui dépend du corpus à l'entrée). Après ce filtrage croisé, nous obtenons 4 754 paires de phrases vietnamienne – anglaise pour construire le module de traduction : $C_{vn\text{-}en} = 4\ 754$ paires, et $D_{vn\text{-}en}$= *475 111* paires. Le seuil utilisé pour le score SWR dans les itérations est 0,3.

Le nombre de paires de phrases extraites et le score BLEU du module de traduction sont présentés dans la Figure 6-16 (nous n'avons pas continué jusqu'à la convergence à cause de la fluctuation aux dernières itérations). Les précisions du processus d'extraction à chaque itération sont estimées sur un ensemble de 100 paires de phrases parallèles vietnamiennes – anglaises qui sont choisies au hasard parmi celles extraites à chaque itération et présentées dans le Tableau 6-7.

Les résultats pour le couple de langues vietnamien – anglais sont similaires à ceux pour le couple de langues vietnamien – français : le nombre de paires de phrases extraites augmente après quelques itérations ; la qualité du système de TA augmente également lors des premières itérations et diminue par la suite. Après 5 itérations, nous recevons 25 466 = 4 754 + 20 712 paires de phrases, qui donne le meilleur score BLEU (29,76), avec une moyenne de 44 % paires de phrases parallèles, 34 % paires de phrases comparables, 22 % paires de phrases non pertinentes.

Tableau 6-7 : Evaluation manuelle de la qualité de paires de phrases extraites VN – EN : Méthode2

	# de paires extraites	**% parallèle**	**% comparable**	**% non pertinent**	**Précision = % parallèle + % comparable**
$C_{vn\text{-}en}$	4 754	52	38	10	90
itération 1	11 722	52	30	18	82
itération 2	16 549	52	30	18	82
itération 3	18 696	48	33	19	81
itération 4	19 912	48	32	20	80
itération 5	20 712	44	34	22	78
itération 6	21 286	44	34	22	78
itération 7	21 787	42	36	22	78
itération 8	22 149	40	36	24	76

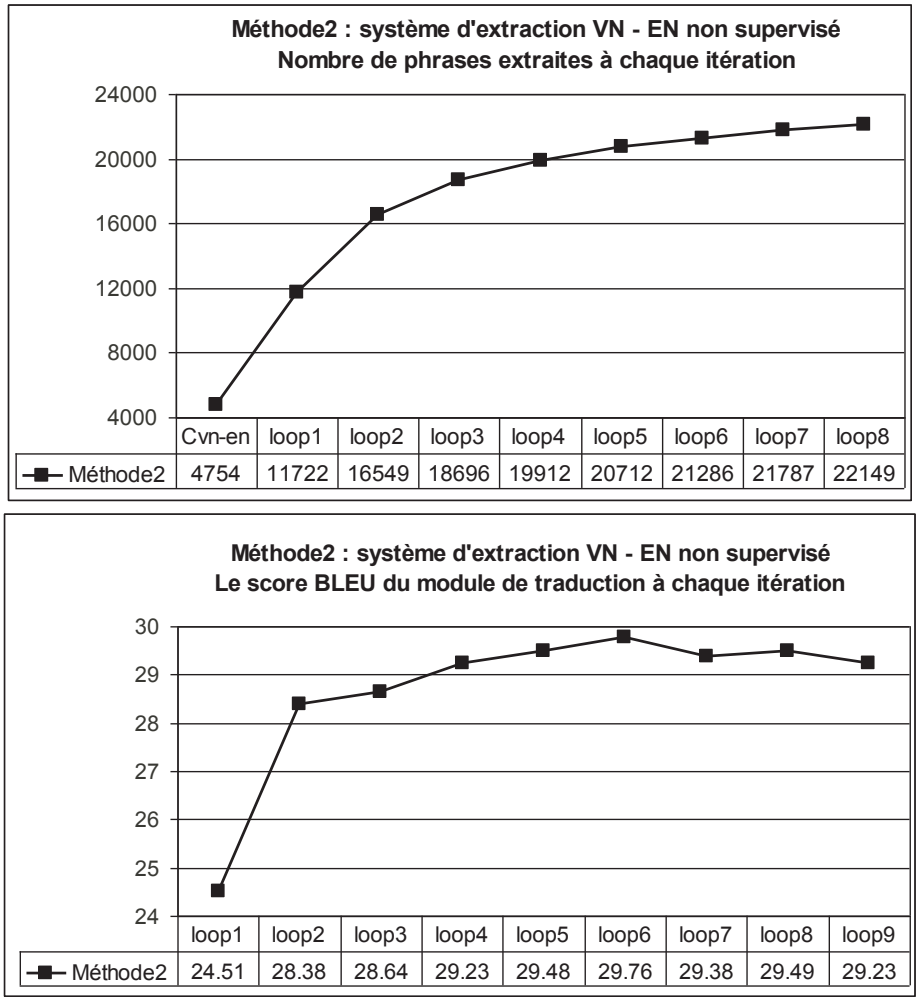


Figure 6-16 : Le nombre de paires de phrases extraites et les scores BLEU du module de traduction à chaque itération dans le système VN-EN (Méthode2)

6.5. Conclusion

Les expériences réalisées dans ce chapitre montrent que notre méthode d'extraction entièrement non supervisée peut être réellement appliquée dans le cas du manque de données parallèles. Aucune information supplémentaire n'est utilisée. L'entrée du processus d'extraction n'a qu'un corpus comparable à fouiller.

Le processus d'extraction contient deux étapes : l'étape d'initialisation et l'étape d'extraction. Dans l'étape d'initialisation, les données pour construire le module de traduction initial sont extraites directement à partir du corpus comparable à l'entrée, par un processus de filtrage croisé original. Le reste du corpus comparable est considéré comme les données à fouiller dans l'étape d'extraction. Le module de traduction initial est utilisé pour traduire le côté source du corpus à fouiller vers la requête. Le module de recherche d'information est basé sur notre métrique d'évaluation SWR et les paires de phrases pertinentes sont extraites. Un processus itératif a été utilisé pour augmenter le nombre de paires de phrases extraites et améliorer la qualité du module de traduction. Un résultat intéressant est que la qualité du module de TA peut être améliorée au cours des premières itérations, mais elle est dégradée plus tard en raison de l'ajout de données bruitées dans le module de traduction.

La méthodologie a été validée sur les corpus comparables vietnamien – français et vietnamien – anglais, et les résultats obtenus sont encourageants. Cette méthode proposée ne nécessite pas de

données supplémentaires, mais la qualité du module de TA construit est comparable à celle d'une autre méthode qui nécessite des données de meilleure qualité comme un dictionnaire bilingue, des heuristiques, etc.

Dans le futur, nous avons l'intention d'appliquer cette méthode à une plus grande échelle pour exploiter un plus grand flux de données comparables extraites du Web.

En plus, avec la disponibilité de corpus de VNA en trois langues, nous continuons notre travail sur l'utilisation d'une troisième langue dans l'extraction du corpus comparable bilingue. Le chapitre suivant présente notre travail sur l'extension de cette méthode d'extraction non supervisée en utilisant la troisième langue.

Chapitre 7 : Extension de la méthode d'extraction non supervisée – utilisation d'une troisième langue

7.1. La triangulation via une troisième langue

Pour améliorer la performance des systèmes de traduction automatique, l'utilisation de ressources auxiliaires d'une troisième langue peut être proposée. Cette troisième langue peut être une langue naturelle ou un langage artificiel. Dans notre cas, nous abordons l'utilisation d'une langue naturelle. Nous appelons *triangulation* le processus d'intégration des ressources auxiliaires de la troisième langue dans un système bilingue, et nous appelons cette troisième langue *la langue pivot*. L'idée de la triangulation est de remplacer la transformation d'informations entre deux langues source (*S*) et cible (*C*) par la transformation d'informations entre deux paires de langues source *S* et pivot (*P*) plus pivot *P* et cible *C*, lorsque la transformation directe entre *S* et *C* est impossible ou très difficile.

7.1.1. La triangulation pour des problèmes connexes

La triangulation a été proposée et utilisée dans plusieurs problèmes tels que la construction automatique de dictionnaires bilingues [Tanaka 1994], [Ahn 2006], [Istvan 2009] ; l'amélioration du processus de recherche d'information translingue [Gollins 2001], [Lehtokangas 2002] ; et l'amélioration des alignements en mots [Borin 2000], [Wang 2006], [Kumar 2007].

[Tanaka 1994] et [Ahn 2006] ont employé une troisième langue (l'anglais) pour construire automatiquement des dictionnaires bilingues japonais–français et espagnol–allemand. Lorsque la traduction d'un mot ne peut pas être trouvée dans le dictionnaire bilingue, elle peut être obtenue en utilisant les dictionnaires intermédiaires mettant en jeu une troisième langue. Par exemple, pour obtenir la traduction française d'un mot japonais, les auteurs cherchent les traductions anglaises de ce mot japonais dans le dictionnaire japonais–anglais, et les traductions françaises des mots anglais intermédiaires dans le dictionnaire anglais–français. Pour choisir les meilleures traductions françaises, les auteurs ont proposé d'utiliser le processus dit de *consultation inverse*.

Pour un mot donné m_S dans la langue source *S*, leur méthode consiste d'abord à traduire m_S dans la langue intermédiaire *P*, puis dans la langue cible *C*, et retourner vers la langue intermédiaire *P* et enfin dans la langue source *S* : $m_S \rightarrow m_P \rightarrow m_C \rightarrow m_P \rightarrow m_S$. Si nous arrivons au même point de départ en langue source m_S, le mot source m_S et le mot cible m_C sont acceptés comme une traduction appropriée. Le résultat a été présenté comme encourageant. En plus de l'appariement fondé sur un dictionnaire, [Istvan 2009] ont employé les informations sémantiques extraites à partir du réseau de mots *WordNet* de la langue pivot pour filtrer les appariements et améliorer le rappel en conservant une bonne précision.

En recherche d'information translingue, lorsque la traduction directe entre la requête et la langue des documents n'existe pas, la triangulation peut être appliquée. La requête est traduite vers la langue pivot, et ensuite vers la langue de documents [Lehtokangas 2002]. Une ou plusieurs langues pivot peuvent être utilisées et les traductions de la requête en langues cibles sont combinées [Gollins 2001]. La triangulation s'avère efficace en comparaison avec la traduction directe dans les expériences de [Lehtokangas 2002] et [Gollins 2001].

La triangulation est utilisée aussi pour améliorer des alignements en mots pour un système de traduction probabiliste. [Borin 2000] a utilisé une ou plusieurs langues pivot pour améliorer l'alignement au niveau de mots entre des textes de deux langues. A partir d'un corpus parallèle multilingue (corpus de plus de deux langues, les phrases du corpus sont alignées en plusieurs langues incluant les langues source, pivot et cible), l'alignement en mots est réalisé directement entre la paire de langues source–cible. Ensuite, les alignements entre les paires de langues source–pivot et pivot–cible sont effectués. En assemblant ces deux derniers alignements, le nouvel alignement indirect source–pivot–cible est produit et combiné avec l'alignement direct source–cible pour former l'alignement final. Le rappel de l'alignement en mots est augmenté, sans réduire la précision selon les expériences présentées. L'auteur a montré que des langues pivot différentes donnent des alignements indirects différents, et donc que l'utilisation de plusieurs langues pivot est plus efficace que l'utilisation d'une seule langue pivot.

Le travail de [Wang 2006] propose, quant à lui, de construire l'alignement en mots entre deux langues source *S* et cible *C* via une langue pivot *P* même si le corpus bilingue de la paire de langue *S* et *C* n'existe pas. Deux grands corpus bilingues des paires de langues *S – P* et *P – C* sont utilisés pour entraîner les modèles intermédiaires du modèle IBM-4 : les modèles de traduction, les modèles de distorsion et les fertilités de la paire de langue *S – P et P – C*. Ensuite, trois nouveaux modèles de traduction, distorsion et fertilité de la paire de langue *S – C* sont induits à partir des modèles intermédiaires. Le modèle IBM-4 induit de la paire de langue *S – P* est la multiplication des modèles intermédiaires induits. Par exemple, la probabilité de traduction en mot $t_{S\text{-}C}$ est calculée par les probabilités de traduction $t_{S\text{-}P}$ et $t_{P\text{-}C}$ (w_i^S est le mot w_i dans la langue *S*).

$$t_{S-C}(w_j^C \mid w_i^S) = \sum_{w_k^P} t_{P-C}(w_j^C \mid w_k^P).t_{S-P}(w_k^P \mid w_i^S) \qquad \textbf{(7-1)}$$

Les expériences ont été réalisées pour la paire de langues chinoise–japonaise via l'anglais. Le modèle d'alignement induit présente une meilleure performance qu'un modèle initial qui est appris à partir d'un petit corpus bilingue de *S* et *C* (le taux d'erreur relatif d'alignement est réduit de 10,41 %). En plus, une interpolation entre le modèle induit et le modèle initial ($P_{inter} = \lambda P_{initial} + (1\text{-}\lambda)P_{induit}$) peut encore réduire le taux d'erreur relatif d'alignement de 21,30 % par rapport au modèle initial.

[Kumar 2007] a présenté une autre méthode pour construire le modèle d'alignement indirect en mots en utilisant des corpus parallèles multilingues avec une ou plusieurs langues pivot (les documents officiels de l'organisation des Nations Unies[1]). Dans leur méthode, le modèle d'alignement induit est la multiplication directe des deux modèles d'alignement intermédiaires. Pour une triade de phrases f_1^J, e_1^I, g_1^K en trois langues *F, E, G* et les alignements correspondants a^{FG}, a^{GE}, les auteurs supposent que l'alignement a^{FE} peut être obtenu par le modèle d'alignement induit :

$$P(a_j^{FE} = i \mid e, f) = P(a_j^{FE} = i \mid G, e, f) = \sum_k P(a_j^{FG} = k \mid g, f) P(a_k^{GE} = i \mid e, g) \quad \textbf{(7-2)}$$

Plusieurs langues pivot peuvent être combinées dans le but de corriger des erreurs d'alignement. Le modèle d'alignement final est calculé par une interpolation linéaire des modèles d'alignements induits par chaque langue pivot : $P(a_j^{FE} = i \mid e, f) = \sum_l \alpha_l \, P(a_j^{FE} = i \mid G_l, e, f)$ où $\sum_l \alpha_l = 1$. Les expériences ont montré que la méthode améliore non seulement la qualité des alignements, mais aussi la performance du système de traduction.

7.1.2. La triangulation pour la traduction automatique probabiliste par groupes de mots

Dans la construction d'un système de traduction automatique probabiliste par groupe de mots, un corpus parallèle bilingue est nécessaire au préalable. Cependant, ce type de corpus n'est pas toujours disponible pour toutes les paires de langues, ou alors il n'existe qu'en faible quantité. Récemment, plusieurs travaux de recherches ont été réalisés qui appliquent le processus de triangulation pour construire ou améliorer la qualité des systèmes de traduction automatique probabiliste. L'application de la triangulation est appropriée dans le cas où le corpus d'apprentissage parallèle pour la paire de langue source–cible n'est pas disponible, mais les systèmes de traduction ou les corpus parallèles des paires de langues source–pivot et pivot–cible existent (et en grande quantité).

En général, trois approches principales ont été proposées pour intégrer le processus de triangulation dans la construction d'un système de traduction automatique probabiliste par groupe de mots : l'approche de traduction *consécutive*, l'approche de traduction *synthétisée* et l'approche de *combinaison des tables de traduction*.

Pour réaliser l'approche de traduction consécutive, deux systèmes de traduction automatique sont utilisés. L'un est le système de traduction de la langue source vers la langue pivot $SMT_{S\text{-}P}$, et l'autre est le système de traduction de la langue pivot vers la langue cible $SMT_{P\text{-}C}$. Pour traduire une phrase p_s en langue source vers la langue cible, tout d'abord nous traduisons p_s vers N phrases en langue pivot par le système $SMT_{S\text{-}P}$ (N-meilleurs hypothèses), puis nous traduisons séparément ces phrases en langue pivot vers M phrases en langue cible par le système $SMT_{P\text{-}C}$. Un score est défini et la phrase ayant le meilleur score est sélectionnée depuis $N \times M$ phrases en langue cible. Nous l'appelons l'approche de traduction consécutive du type *NxM*. Lorsque $N=M=1$, nous avons une concaténation de deux systèmes de traduction $SMT_{S\text{-}P}$ et $SMT_{P\text{-}C}$. Cette approche est simple et facile à mettre en œuvre, mais les erreurs sont accumulées car les erreurs d'un système sont propagées à l'entrée du système suivant.

[1] http://ods.un.org/

La deuxième approche est l'approche de traduction synthétisée. Lorsque nous avons un corpus parallèle de la paire de langue source–pivot $C_{S\text{-}P}$ et un système de traduction de la langue pivot vers la langue cible $SMT_{P\text{-}C}$, nous pouvons construire un corpus parallèle synthétisé pour la paire de langue source–cible. Simplement, le côté en langue pivot du corpus $C_{S\text{-}P}$ est traduit en langue cible par le système de traduction $SMT_{P\text{-}C}$. Les phrases de traduction sorties sont appariées avec le côté source de corpus $C_{S\text{-}P}$ pour former le corpus parallèle synthétisé pour la paire de langue source–cible. Le nouveau système de traduction automatique probabiliste $SMT_{S\text{-}C}$ est entraîné à partir de ce corpus parallèle synthétisé. Cette deuxième approche est simple et facile à réaliser aussi, mais elle est coûteuse en temps et ressources pour traduire entièrement le côté en langue pivot du corpus $C_{S\text{-}P}$.

La troisième approche consiste à combiner deux tables de traduction des deux systèmes de traduction automatique probabiliste par groupe de mots. A partir de deux tables de traduction des systèmes $SMT_{S\text{-}P}$ et $SMT_{P\text{-}C}$, une nouvelle table de traduction pour la paire de langue source–cible est construite. Les équivalences entre les entrées des deux tables source–pivot et pivot–cible sont trouvées. Deux entrées de deux tables sont équivalentes lorsqu'elles possèdent le même groupe de mots du côté de la langue pivot. Les entrées de la nouvelle table de traduction de la paire de langue source–cible sont identifiées par l'union ou l'intersection des équivalences. Les scores correspondants sont calculés en combinant les scores des équivalences. La table de traduction induite est alors utilisée dans le système de traduction automatique probabiliste par groupe de mots.

Ci-dessous, nous présentons certains travaux typiques sur l'utilisation de la triangulation pour un système de traduction automatique.

[De Gispert 2006] présente une expérience sur la traduction automatique probabiliste entre la paire de langues anglais et catalan sans utiliser de corpus parallèle catalan–anglais. La langue pivot espagnole est utilisée via deux corpus parallèles indépendants : anglais–espagnol (les actes du Parlement Européen) et espagnol–catalan (les nouvelles journalistiques). Les auteurs ont mis en œuvre deux stratégies de triangulation : l'approche de traduction consécutive et l'approche de traduction synthétisée. Pour traduire la phrase anglaise vers la langue catalane, dans la première méthode, deux systèmes de traduction probabilistes sont construits à partir de deux corpus parallèles anglais–espagnol et espagnol–catalan et ils sont enchaînés simplement (l'approche de traduction consécutive du type *1x1*). La seconde méthode consiste à traduire tout le côté de la langue espagnole du corpus parallèle anglais–espagnol vers la langue catalane en utilisant le système de traduction probabiliste espagnol–catalan. Ensuite, un système de traduction probabiliste anglais–catalan est entraîné directement par ce corpus synthétisé anglais–catalan. Dans leurs expériences, les performances (en termes des scores BLEU, WER, PER) des deux méthodes sont assez équivalentes et les scores de la tâche de traduction catalan–anglais (au début il n'y a aucune donnée parallèle catalan–anglais) sont assez semblables à ceux de la tâche de traduction espagnol–anglais (dont les données parallèles espagnol–anglais existent). Les auteurs en déduisent qu'il n'y a pas de perte significative à cause de la triangulation.

[Eisele 2006] propose une approche généralisée qui permet d'améliorer la qualité de la traduction par plusieurs langues pivot, au lieu d'une seule langue pivot. Pour cela, un corpus parallèle multilingue est nécessaire. L'auteur suggère que la couverture de lexiques peut être agrandie et les contraintes des autres langues peuvent aider à faire le choix de la traduction finale entre des expressions qui sont trop larges ou trop spécifiques. Le risque d'obtenir des traductions mauvaises est donc réduit. Les systèmes de traduction existants qui partagent une ou plusieurs langues communes peuvent être couplés pour construire le système de traduction pour la paire de langues pour lesquelles aucun corpus parallèle n'existe. L'auteur présente un schéma de type traduction consécutive pour traduire entre les langues principales considérées (telles que l'arabe, le chinois, le russe) et les autres langues européennes (telles que le polonais, l'allemand,

l'ukrainien) en utilisant trois langues pivot : l'anglais, l'espagnol et le français. Cependant, ce schéma n'est pas expérimenté dans l'article.

La troisième approche, l'approche de combinaison des tables de traduction, est mise en œuvre dans le travail de [Cohn 2007]. Les auteurs utilisent un corpus parallèle multilingue pour augmenter la couverture lexicale entre les phrases en langue source et en langue cible. Les tables de traduction pour les paires de langues source–pivot et pivot–cible sont construites à partir du corpus parallèle multilingue. Une nouvelle table de traduction pour la paire de langue source–cible est construite ensuite en combinant directement ces deux tables. Les scores (les probabilités) de la table de traduction sont calculés par la formule suivante :

$$p(s \mid t) = \sum_i p(s,i \mid t)p(i \mid t) \approx \sum_i p(s \mid i)p(i \mid t) \quad \textbf{(7-3)}$$

Pour améliorer encore la performance, la table induite est combinée avec une table de traduction standard apprise à partir d'un petit corpus parallèle source – cible par une interpolation linéaire. Les auteurs annoncent que la couverture des *n*-grammes est élargie et que le système de traduction basé sur la table induite est meilleur que le système standard en termes de score BLEU. Ils observent aussi que la performance de traduction est augmentée lorsque la langue pivot est dans la même famille que la langue source ou la langue cible.

[Wu 2007] présente la même approche de combinaison des tables de traduction sur un corpus parallèle multilingue (le corpus *Europarl*). Les auteurs développent une méthode d'alignement par triangulation (présentée ci-dessus [Wang 2006]). La combinaison des tables de traduction est semblable à la méthode de [Cohn 2007], mais les alignements utilisés sont les alignements induits à partir de la méthode de [Wang 2006]. L'interpolation linéaire avec une table standard est utilisée aussi. Le score BLEU du système de traduction français–espagnol avec la langue pivot anglais augmente de 16 % par rapport à celui du système de traduction normale appris avec 5 000 paires de phrases françaises–espagnoles, et il est comparable à celui du système de traduction normale appris avec 30 000 paires de phrases françaises–espagnoles. Les résultats indiquent également que l'utilisation de deux langues pivot anglais et allemand conduit à une augmentation de 22 % du score BLEU.

[Utiyama 2007] présente une comparaison entre les méthodes de triangulation pour la traduction automatique probabiliste par groupe de mots. Dans ce travail, les auteurs ont mis en œuvre l'approche de traduction consécutive et l'approche de combinaison des tables de traduction. Les expériences ont été réalisées sur un corpus parallèle multilingue (à nouveau le corpus *Europarl*). L'auteur a réalisé l'approche de traduction consécutive entre le français et l'allemand via l'anglais. Chaque phrase anglaise ou allemande possède 8 scores calculés par 8 fonctions de traits $h_k()$ du système de traduction français – anglais ou anglais – allemand (la probabilité du modèle de langue, les probabilités de traduction par groupe de mots dans les deux sens, les probabilités de traduction lexicale dans les deux sens, la pénalité de lexique, la pénalité de phrase et la pénalité de réarrangement). Un score global pour chaque paire de phrases français – allemand est calculé par une interpolation des 16 scores. La phrase allemande possédant le plus grand score est choisie comme la traduction de la phrase source. Les systèmes de traduction consécutive du type *1x1* et *15x15* sont construits. L'inconvénient de cette approche est une vitesse de traduction lente.
La deuxième approche est la combinaison de deux tables de traduction français–anglais et anglais–allemand pour construire une nouvelle table de traduction français–allemand. Quatre probabilités sont estimées pour la nouvelle table : deux probabilités de traduction par groupe de mots $\phi(f|g)$, $\phi(g|f)$ et deux probabilités de traduction lexicale $p_w(f|g)$, $p_w(g|f)$. La formule du calcul est semblable aux autres travaux :

$$\phi(f \mid g) = \sum_{e \in T_{FE} \cap T_{EG}} \phi(f \mid e)\phi(e \mid g) \ , \qquad \phi(g \mid f) = \sum_{e \in T_{FE} \cap T_{EG}} \phi(g \mid e)\phi(e \mid f)$$
$$p_w(f \mid g) = \sum_{e \in T_{FE} \cap T_{EG}} p_w(f \mid e)\, p_w(e \mid g) \ , \ p_w(g \mid f) = \sum_{e \in T_{FE} \cap T_{EG}} p_w(g \mid e)\, p_w(e \mid f) \qquad \textbf{(7-4)}$$

où *e* est le groupe de mots compris dans les deux tables de traduction T_{FE} et T_{EG}.

Le système de traduction directe et le système de traduction par l'approche de combinaison sont mis en œuvre et comparés avec l'approche de traduction consécutive. Le même corpus parallèle multilingue est utilisé pour toutes les expériences. Le système de traduction directe fonctionne mieux que les autres systèmes en termes de score BLEU. Le système de traduction consécutive du type *15x15* n'est pas plus performant que le système de traduction consécutive du type *1x1*. Les auteurs expliquent que *N=15* n'est pas assez grand pour couvrir les bons candidats de traduction. Cependant, l'exécution avec une grande valeur de *N* est peu pratique à cause de la vitesse de traduction lente. L'approche de combinaison de tables de traduction donne une meilleure performance que l'approche de traduction consécutive mais demeure moins performante que l'approche de traduction directe.

Un autre travail qui présente la comparaison entre les approches est le travail de [Bertoldi 2008b]. Les trois approches principales ont été évaluées sur deux types de corpus, l'un est le corpus parallèle multilingue de trois langues source–pivot–cible, et l'autre est deux corpus bilingues indépendants de deux paires de langues source–pivot et pivot–cible. L'approche de traduction consécutive (du type *1x1* et *NxN*) est réalisée. Un score global est calculé par une interpolation de 16 scores des fonctions de traits et la paire de phrases source et cible ayant le meilleur score est sélectionnée.

Dans l'approche de combinaison des tables de traduction, le score de chaque nouvelle entrée de la table de traduction induite est calculé par une intégration ou un maximum :

$$\phi(src \mid tgt) = \begin{cases} \sum_{piv} \phi(src, piv)\,\phi(piv, tgt) & \text{int} \\ \max_{piv} \phi(src, piv)\,\phi(piv, tgt) & \text{max} \end{cases} \qquad \textbf{(7-5)}$$

La qualité du système de traduction (en termes du score BLEU) dans le cas utilisant les corpus parallèle multilingue est meilleure que celle dans le cas utilisant les corpus indépendants. L'une des raisons est qu'il y a 77 % d'entrées communes entre les deux tables de traduction dans le cas utilisant les corpus parallèle multilingue ; cependant, il n'y a que 44 % d'entrées communes dans l'autre cas. L'intégration des scores donne par ailleurs une meilleure performance que le maximum des scores.

Toujours dans ce même article de [Bertoldi 2008b], l'approche de traduction synthétisée est évaluée. Les auteurs proposent une nouvelle technique pour former le corpus synthétisé. Le corpus source – cible est synthétisé à partir d'un corpus parallèle source – pivot. Chaque phrase du côté en langue pivot est traduite en N-meilleures hypothèses en langue cible avec le score de traduction *P(hypothèse|source)*. Un ensemble de *M* phrases est choisi depuis N-meilleures hypothèses selon leur score. Une hypothèse peut être choisie plusieurs fois, et le nombre de fois qu'une hypothèse est choisie est dépendant de son score de traduction. Ces M phrases sont appariées avec la phrase source pour former le corpus synthétisé.

Les expériences pour le corpus parallèle multilingue et les corpus indépendants ont été réalisées. Les performances de la méthode de traduction consécutive du type *NxN* (N=100) et la nouvelle méthode de traduction synthétisée semblent être comparables ; et les deux surpassent la performance de la méthode de traduction consécutive du type *1x1* et celle de la méthode de combinaison des tables de traduction.

En plus des approches de combinaison par l'union des équivalences entre deux tables de traductions présentées ci-dessus, récemment, [Chen 2008, 2009] ont proposé l'utilisation de l'intersection entre des équivalences de deux tables. L'idée est que l'intersection serait plus précise et plus compacte que l'union. En outre, l'intersection peut réduire la taille de la table de traduction, et ainsi diminuer le temps et les coûts de l'espace requis pour la tâche de traduction. Tout d'abord, une table de traduction directe source – cible est construite à partir d'un corpus parallèle source – cible existant. Deux autres tables sont alors construites entre les couples de langues source – pivot et pivot – cible. Ensuite, ces deux tables sont utilisées pour filtrer la table de traduction directe. Essentiellement, les auteurs conservent une entrée dans la table directe lorsque cette entrée peut être induite à partir de deux tables de traduction source – pivot et pivot – cible. Les probabilités de cette entrée restent les mêmes. Deux techniques spécifiques de filtrage sont proposées qui requièrent la correspondance entière ou partielle entre les groupes de mots. Une partie des entrées est retirée de la table. Sur les expériences réalisées, la taille d'une table de traduction peut être réduite à moins de 30 % de la table initiale. De plus, les auteurs présentent des améliorations significatives de qualité avec la table de traduction réduite [Chen 2008, 2009].

7.2. Utilisation d'une langue pivot pour l'extraction de corpus parallèles

Comme présenté dans la section 7.1, la triangulation nous permet d'améliorer la qualité des systèmes de traduction automatique probabiliste. Dans notre cas, nous voulons aborder le problème de l'utilisation de la triangulation pour améliorer le processus d'extraction de données parallèles. Les corpus multilingues sont de plus en plus disponibles sur l'Internet. Notre hypothèse est que les processus d'extraction appliqués pour des paires de langues différentes peuvent s'améliorer l'un l'autre. Par exemple, les données extraites vietnamiennes – anglaises peuvent être utilisées dans le processus d'extraction d'un corpus vietnamien – français, et vice-versa. Ainsi, nous suggérons que cette combinaison, en utilisant la triangulation, nous permet d'améliorer non seulement la qualité des données extraites mais aussi la qualité du système de traduction.

En plus, l'application de triangulation présentée dans la section 7.1 est appropriée dans le cas où le corpus parallèle pour la paire de langue $S – C$ n'est pas disponible (ou en faible quantité), mais les corpus parallèles des paires de langue $S – P$ et $P – C$ existent en grande quantité (Figure 7-1a) (dans ce cas on peut appliquer l'approche de traduction consécutive et l'approche de combinaison des tables de traduction qui nécessitent deux systèmes de traduction $SMT_{S\text{-}P}$ et $SMT_{P\text{-}C}$). Notre cas est un peu différent : il n'existe que des corpus comparables pour les paires de langue $S – C_1$ et $S – C_2$ (en faible quantité), mais un corpus parallèle pour $C_1 – C_2$ existe en grande quantité (Figure 7-1b). Notre but est d'améliorer la qualité des données extraites pour les paires de langues $S – C_1$ et $S – C_2$, en utilisant la paire de langues $C_1 – C_2$. Dans nos travaux, nous nous concentrons sur la deuxième approche de triangulation : l'approche de traduction synthétisée.

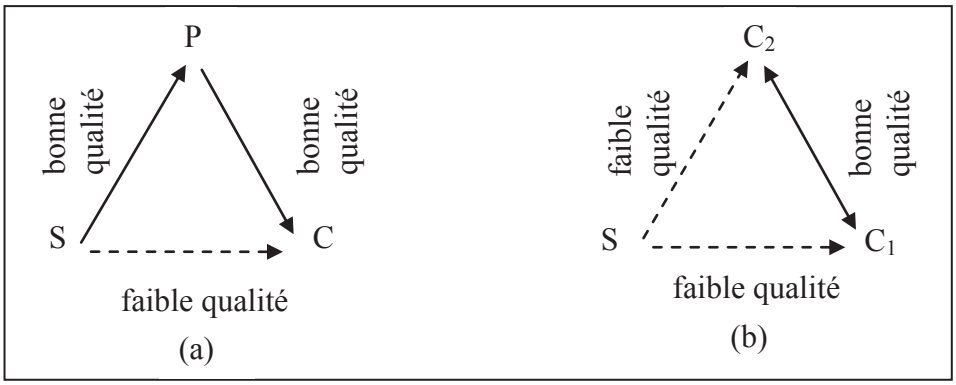


Figure 7-1 : Relation entre les langues dans l'application de triangulation : (a) les applications présentées dans la section 7.1; (b) notre application

Dans un premier temps, nous présentons un test pour évaluer si les données synthétisées peuvent aider le processus d'extraction non supervisée. Après cela, nous proposons l'utilisation de la triangulation dans une combinaison de deux processus d'extraction, pour améliorer simultanément la qualité d'extraction de deux paires de langues différentes.

7.2.1. Expérience préliminaire sur les données synthétisées

Dans ce test, les données synthétisées d'une troisième langue sont ajoutées dans le processus d'extraction non supervisée d'une paire de langues. Nous rappelons que dans la deuxième approche de triangulation, le côté en langue pivot du corpus $C_{S\text{-}P}$ est traduit en langue cible par le système de traduction $SMT_{P\text{-}C}$. Les sorties de traduction sont appariées avec le côté source du corpus $C_{S\text{-}P}$ pour former le corpus parallèle synthétisé pour la paire de langue source – cible. Nous appliquons ce type de triangulation dans le système d'extraction non supervisé pour ajouter des nouvelles données au module de traduction.

Nous supposons que nous avons un corpus de phrases parallèles pour la paire de langue $S - C_2$: $X_{S\text{-}C2}$ (si nous n'avons qu'un corpus comparable de la paire de langue $S - C_2$, nous pouvons appliquer la méthode d'extraction non supervisée pour extraire les paires de phrases parallèles). Par ailleurs, nous supposons que le système de traduction $SMT_{C2\text{-}C1}$ de la langue C_2 vers la langue C_1 existe et est disponible. Donc, nous voulons faire usage du corpus $X_{S\text{-}C2}$ ainsi que du système de traduction $SMT_{C2\text{-}C1}$ afin d'améliorer le processus d'extraction d'un corpus comparable $D_{S\text{-}C1}$. Pour cela, les données en langue C_2 du corpus parallèle $X_{S\text{-}C2}$ sont traduites en langue C_1 en utilisant le système $SMT_{C2\text{-}C1}$. Alors, les nouvelles données (synthétiques donc probablement bruités) $X_{S\text{-}C1}$ obtenues sont ajoutées au processus d'extraction non supervisé pour la paire de langue $S - C_1$ (à l'étape d'extraction). Ces données supplémentaires $X_{S\text{-}C1}$ peuvent être ajoutées au module de traduction. Puis, le processus d'extraction non supervisé (chapitre 6) est appliqué comme d'habitude. Nous appelons ce test *M3 – Test* (résumé dans la Figure 7-2).

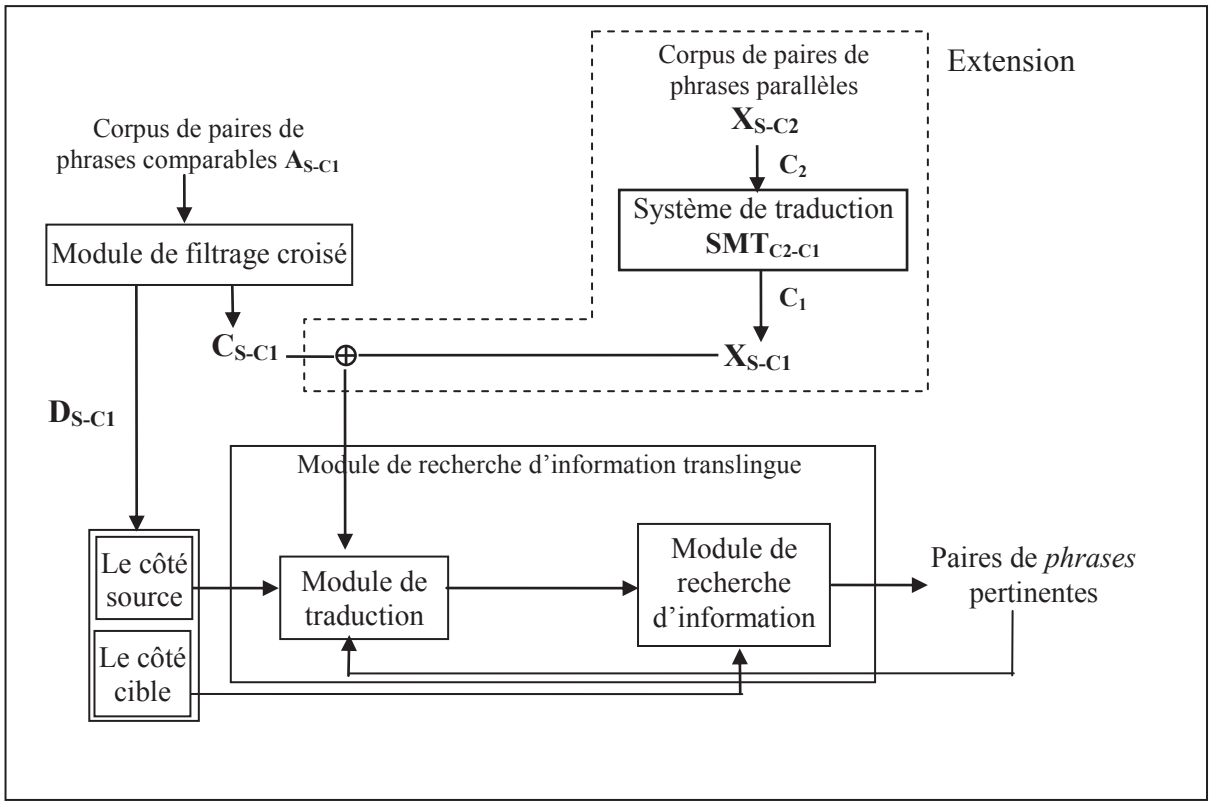


Figure 7-2 : Extension de la méthode d'extraction non supervisée en utilisant la triangulation (M3 – Test)

La question à laquelle nous allons essayer de répondre ici est alors : si la qualité du module de traduction dans le processus d'extraction est améliorée, est-ce que la qualité des paires de phrases extraites est améliorée également ?

Nous avons fait un test sur les données vietnamiennes, françaises et anglaises du site Web du VNA. Nous avons utilisé les données parallèles vietnamiennes (S) – anglaises (C_2) pour améliorer le processus d'extraction non supervisée du corpus comparable vietnamien (S) – français (C_1).

Le système de traduction de la langue anglaise vers la langue français SMT_{en-fr} peut être n'importe quel système commercial existant comme Systran, mais nous avons décidé de construire notre propre système, en utilisant l'outil MOSES, à partir des corpus *Europarl* et *News Commentary* qui sont fournis pour différentes campagnes d'évaluation internationales (WMT, IWSLT). Ce système de traduction a été évalué sur l'ensemble de test fourni dans la tâche de traduction de la campagne d'évaluation (nous utilisons les données de WMT 2009 (*Workshop of statistical machine translation*)). Le score BLEU obtenu du système SMT_{en-fr} est 23,74. Ceci peut être considéré comme très acceptable en comparaison avec d'autres systèmes de traduction anglais–français présentés à WMT 2009[1].

Le processus d'extraction des paires de phrases pertinentes vietnamiennes – anglaises à partir du corpus de VNA a déjà été présenté dans la section 6.4.3. Au total, nous avons 25 466 paires de phrases pertinentes. Les paires extraites sont converties vers les données synthétisées par le système de traduction SMT_{en-fr}. Cependant, les paires de phrases extraites contiennent des paires de phrases parallèles, comparables et bruitées. Pour assurer la qualité du module de traduction, les données extraites vietnamiennes – anglaises doivent être « plus propres ». Dans un test

[1] http://matrix.statmt.org/matrix/systems_list/659. Il y a 18 systèmes de traduction anglais–français présentés à WMT 2009, avec le score BLEU qui varie de 14,8 à 25,6 (10 systèmes ont un score moins grand que 23,74 ; et 8 systèmes ont un score plus grand que 23,74).

précédent, nous avions un ensemble de 8 218 paires de phrases vietnamiennes – anglaises obtenues avec seuil SWR plus grand que 0,4[1]. Nous supposons que cet ensemble est suffisant pour ce test.

Ainsi, le processus d'extraction présenté dans la Figure 7-2 est réalisé avec :

- $X_{S\text{-}C2}$ qui contient 8 218 paires de phrases vietnamiennes–anglaises
- $C_{S\text{-}C1}$ qui contient 4 076 paires de phrases vietnamiennes–françaises après le filtrage croisé
- $D_{S\text{-}C1}$ qui contient 341 178 paires de phrases vietnamiennes–françaises

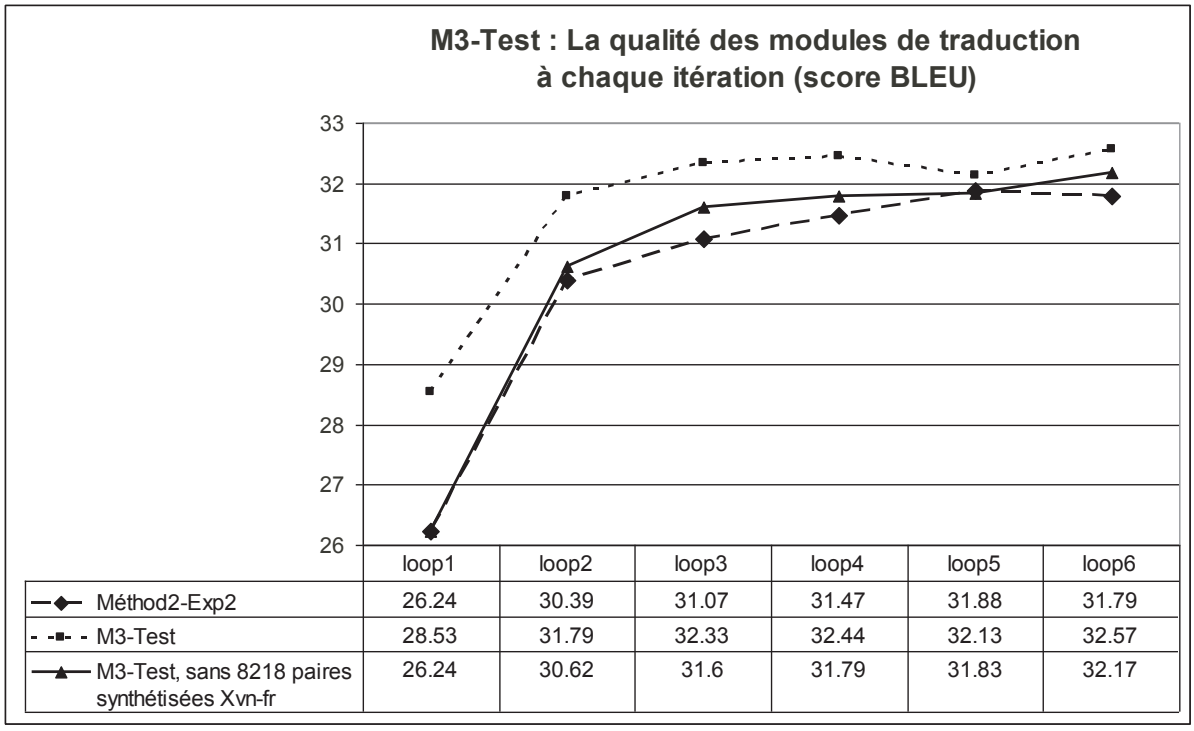
Le processus d'extraction itératif a été réalisé. Le nombre de paires de phrases vietnamiennes – françaises extraites du test *M3 – Test* est présenté dans le Tableau 7-1, en comparaison avec le résultat de la *Méthode 2 – Exp 2* déjà présentée au chapitre précédent. Le nombre de paires extraites du test *M3 – Test* est légèrement inférieur à celui de la *Méthode 2 – Exp 2*.

Tableau 7-1 : Le nombre de paires de phrases extraites à chaque itération pour la Méthode2-Exp2 (S0: C=4 076) et le test M3 – Test (S0: C=4 076 + X=8 218)

itération	Méthode2 – Exp2	M3 – Test
1	6 780	6 798
2	10 016	9 892
3	11 672	11 488
4	12 734	12 575
5	13 524	13 340
6	14 100	13 872
7	14 560	14 350
8	14 969	14 767

La qualité des modules de traduction (en term du score BLEU) après chaque itération est présentée dans la Figure 7-3 (estimée sur le même corpus de test de 400 paires de phrases présenté au chapitre 6). La ligne en pointillés larges présente la qualité des modules de traduction de la *Méthode 2 – Exp 2* (dont le corpus d'apprentissage comprend l'ensemble *C* et les phrases extraites à chaque itération), et la ligne en pointillés fins présente celle du test *M3 – Test* (dont le corpus apprentissage comprend l'ensemble *C*, l'ensemble *X* et les phrases extraites à chaque itération). De plus, pour comparer uniquement la qualité des paires extraites, des modules de traduction construits seulement avec l'ensemble C et les phrases extraites à chaque itération du test *M3 – Test* (sans utiliser l'ensemble *X*) sont expérimentés et présentés en ligne continue.

[1] Dans le test précédent, le filtrage croisé du processus d'extraction vietnamien–anglais est réalisé avec un seuil de SWR égal à 0,3. Le nombre de paires de phrases de $C_{vn\text{-}en}$ est grand mais la qualité de paires de phrases extraites à la fin est plus faible que le cas présenté dans la section 6.4.3, donc nous ne présentons pas ce test ici. Mais l'ensemble de 8 212 paires de phrases vietnamiennes–anglaises (avec le score SWR plus grand que 0,4) peut être considéré comme propre pour ce test. Le test précédent est présenté dans la publication [Do T.N.D. 2010]



	loop1	loop2	loop3	loop4	loop5	loop6
Méthod2-Exp2	26.24	30.39	31.07	31.47	31.88	31.79
M3-Test	28.53	31.79	32.33	32.44	32.13	32.57
M3-Test, sans 8218 paires synthétisées Xvn-fr	26.24	30.62	31.6	31.79	31.83	32.17

Figure 7-3 : La qualité des modules de traduction à chaque itération – M3 – Test

En comparant la ligne en pointillés larges (*Méthode 2 – Exp 2*) et la ligne continue (*M3 – Test*, sans $X_{vn\text{-}fr}$), nous voyons que, bien que le nombre de paires de phrases extraites de deux processus *Méthode 2 – Exp 2* et *M3 – Test* soit comparable, la qualité des paires de phrases extraites du test *M3 - Test* est un peu plus élevée que celle de la *Méthode 2 – Exp 2*, grâce aux données synthétisées de la troisième langue, l'anglais. Nous voyons par ailleurs que la méthode de triangulation, qui permet d'ajouter un nombre significatif de données bilingues au départ, améliore les performances de traduction, même si ces données sont synthétiques.

7.2.2. Notre méthode de triangulation pour intégrer la troisième langue dans le processus d'extraction non supervisée

Dans le test précédent, les données synthétisées d'un corpus de phrases parallèles $S\text{-}C_2$ étaient utilisées pour aider le processus d'extraction pour une paire de langues $S\text{-}C_1$. Maintenant, supposons qu'au début nous n'ayons que deux corpus comparables des paires de langues $S\text{-}C_1$ et $S\text{-}C_2$. Nous proposons ici d'utiliser la triangulation pour combiner et améliorer simultanément les deux processus d'extraction pour les paires de langues $S\text{-}C_1$ et $S\text{-}C_2$. Par exemple, les données extraites du processus d'extraction vietnamien – anglais peuvent être utilisées dans le processus d'extraction d'un corpus vietnamien – français, et les données extraites du processus d'extraction vietnamien – français, de leur côté, sont utilisées dans le processus d'extraction d'un corpus vietnamien – anglais. Ainsi, nous suggérons que cette combinaison nous permet d'améliorer non seulement la qualité des données extraites mais aussi la qualité des deux systèmes de traduction.

Nous proposons donc la combinaison de deux processus d'extraction, intégrée avec la triangulation. Nous l'appelons la méthode de triangulation ou la *Méthode 3*. La méthode de triangulation est présentée dans la

Figure 7-4. Au début, il y a deux corpus comparables $A_{S\text{-}C1}$ et $A_{S\text{-}C2}$ indépendants (qui ne sont pas parallèles en trois langues) qui sont dérivés de la même source de données (un site Web multilingue par exemple). La méthode d'extraction non supervisée (présentée dans le chapitre 6) est appliquée pour fouiller les paires de phrases pertinentes à partir des deux corpus $A_{S\text{-}C1}$ et

$A_{S\text{-}C2}$. Nous appelons les deux processus d'extraction $Process_{S\text{-}C1}$ et $Process_{S\text{-}C2}$. A chaque itération, non seulement les paires de phrases extraites des itérations précédentes mais aussi les paires extraites de l'autre processus sont ajoutées à S_{i+1} pour créer le nouveau module de traduction :

- les paires de phrases extraites $X_{S\text{-}C2}$ à l'itération i^{eme} du $Process_{S\text{-}C2}$ sont converties vers les données synthétisées $X^*_{S\text{-}C1}$ en traduisant le côté en langue C_2 vers la langue C_1 par un système de traduction $SMT_{C2\text{-}C1}$ existant ;
- les données synthétisées $X^*_{S\text{-}C1}$ sont ensuite ajoutées au module de traduction du processus $Process_{S\text{-}C1}$ à l'itération $i+1^{eme}$.

Pour la première itération, les données extraites après le filtrage croisé $C_{S\text{-}C2}$ sont traduites vers les données synthétisées $C^*_{S\text{-}C1}$ et ajoutées au premier module de traduction S_0.

Donc, à l'itération $i+1^{eme}$, le module de traduction S_{i+1} du processus $Process_{S\text{-}C1}$ est entraîné par les données d'apprentissage de l'itération i^{eme}, les paires de phrases extraites à l'itération i^{eme} $X_{S\text{-}C1}$ et les données synthétisées $X^*_{S\text{-}C1}$ à l'itération i^{eme} du $Process_{S\text{-}C2}$.

La même méthode de combinaison est appliquée au $Process_{S\text{-}C2}$.

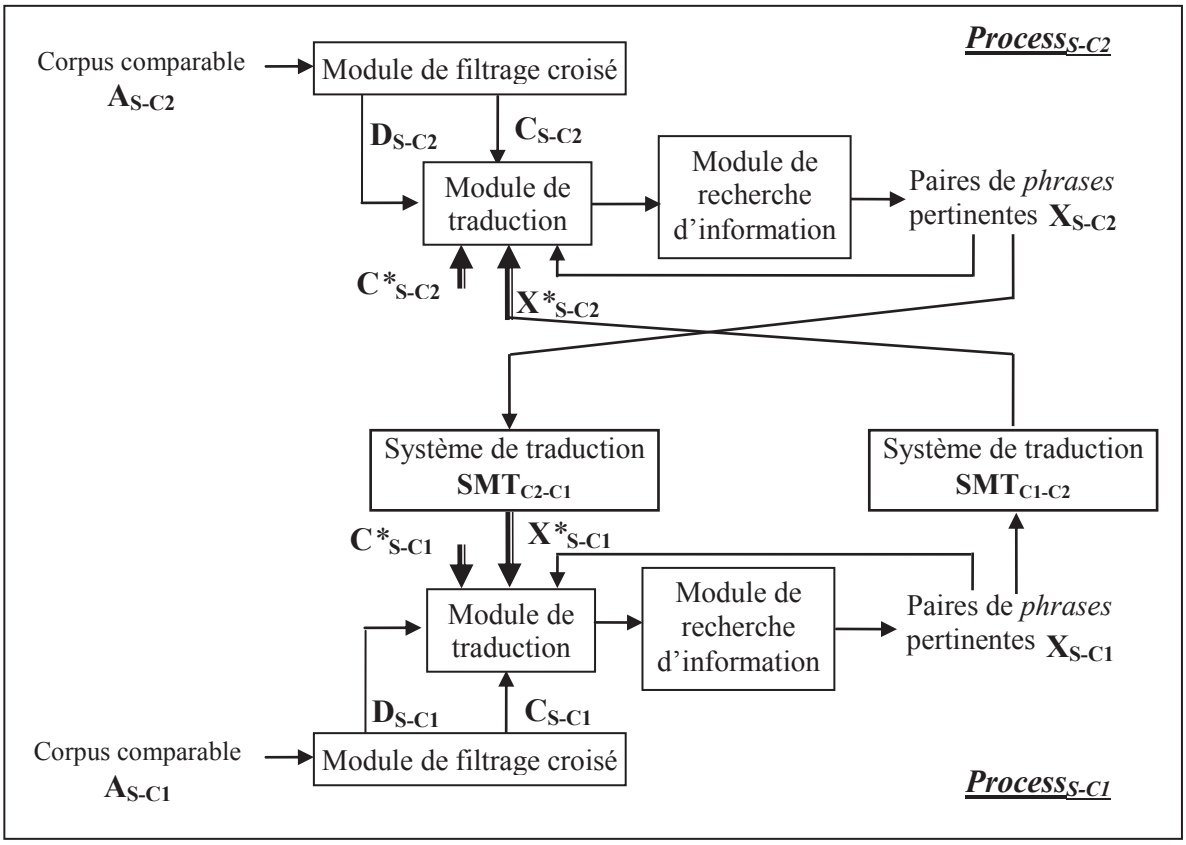
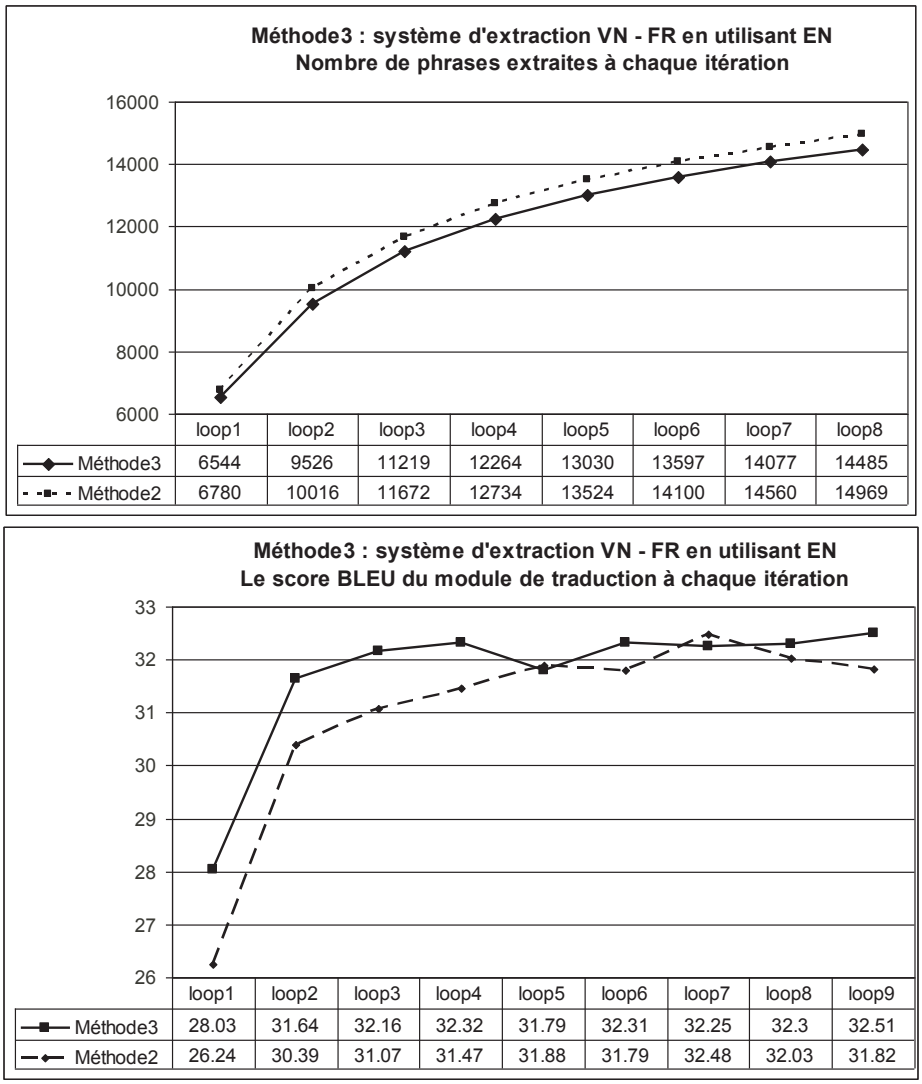


Figure 7-4 : La méthode de triangulation – Méthode 3 : l'intégration de la troisième langue dans la méthode non supervisée

7.3. Expériences sur l'intégration de la troisième langue dans la méthode d'extraction non supervisée

Dans le chapitre 6, la méthode d'extraction non supervisée (*Méthode 2*) est appliquée sur deux corpus comparables vietnamien–français et vietnamien–anglais du corpus VNA. Les résultats d'extraction sont présentés dans Figure 6-14, Figure 6-16, Tableau 6-4 et Tableau 6-7. Pour évaluer la méthode de triangulation (*Méthode 3*), les même corpus comparables vietnamien–

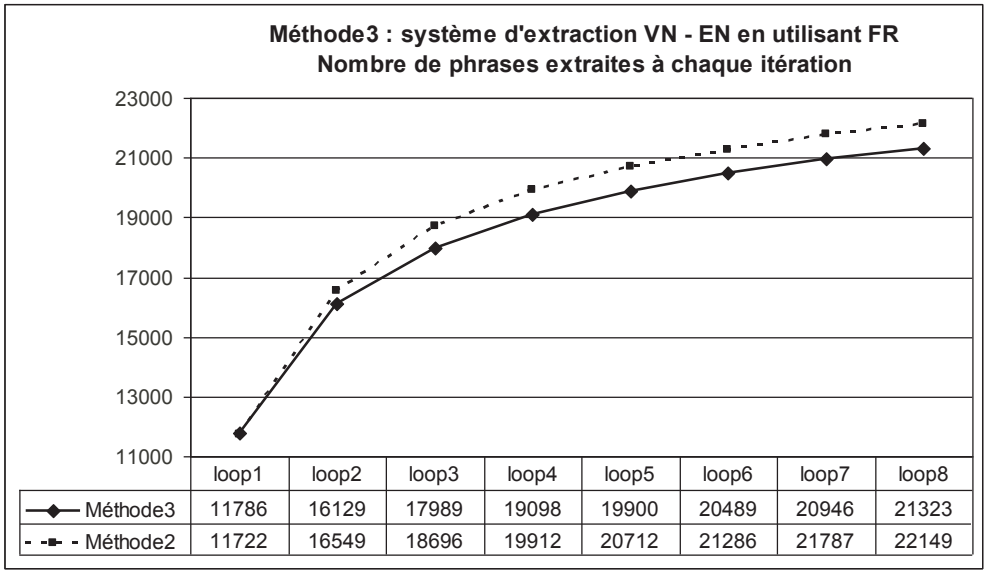
français et vietnamien–anglais sont utilisés. Les résultats d'extraction de la *Méthode 3* seront comparés avec ceux de la *Méthode 2*. Les systèmes de traduction de la langue anglaise vers la langue française $SMT_{en\text{-}fr}$ et de la langue française vers la langue anglaise $SMT_{fr\text{-}en}$ sont construits à partir des corpus *Europarl* et *News Commentary* de la campagne d'évaluation WMT 2009 en utilisant l'outil MOSES (comme présenté dans la section 7.2.1). Le score BLEU obtenu pour le système $SMT_{en\text{-}fr}$ est de 23,74, et celui du système $SMT_{fr\text{-}en}$ est 22,97 (évalués avec l'ensemble de test de WMT 2009). La méthode d'extraction non supervisée avec l'intégration de la triangulation est appliquée, et les résultats d'extraction pour les paires de langues vietnamienne–française et vietnamienne–anglaise sont présentés dans les Figure 7-5 et Figure 7-6 ci-dessous (en comparaison avec la *Méthode 2*).



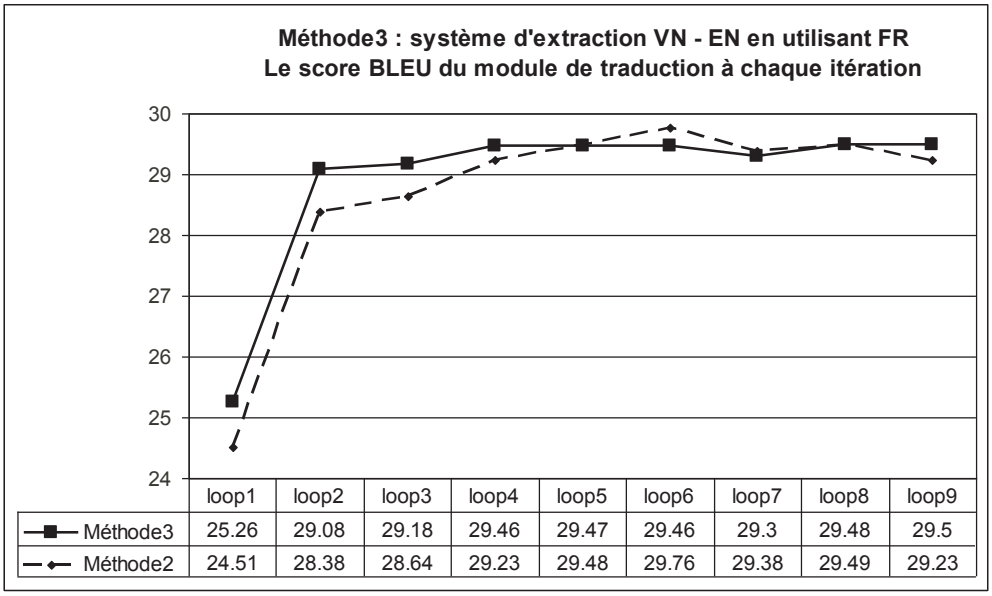
	loop1	loop2	loop3	loop4	loop5	loop6	loop7	loop8
Méthode3	6544	9526	11219	12264	13030	13597	14077	14485
Méthode2	6780	10016	11672	12734	13524	14100	14560	14969

	loop1	loop2	loop3	loop4	loop5	loop6	loop7	loop8	loop9
Méthode3	28.03	31.64	32.16	32.32	31.79	32.31	32.25	32.3	32.51
Méthode2	26.24	30.39	31.07	31.47	31.88	31.79	32.48	32.03	31.82

Figure 7-5 : Méthode 3 - Le nombre de paires de phrases extraites et les scores BLEU du module de traduction à chaque itération dans le système VN-FR



	loop1	loop2	loop3	loop4	loop5	loop6	loop7	loop8
Méthode3	11786	16129	17989	19098	19900	20489	20946	21323
Méthode2	11722	16549	18696	19912	20712	21286	21787	22149



	loop1	loop2	loop3	loop4	loop5	loop6	loop7	loop8	loop9
Méthode3	25.26	29.08	29.18	29.46	29.47	29.46	29.3	29.48	29.5
Méthode2	24.51	28.38	28.64	29.23	29.48	29.76	29.38	29.49	29.23

Figure 7-6 : Méthode 3 - Le nombre de paires de phrases extraites et les scores BLEU du module de traduction à chaque itération dans le système VN-EN

Grâce aux données supplémentaires de la troisième langue, les scores BLEU du module de traduction à chaque itération sont significativement plus élevés que ceux de la *Méthode 2*. Un résultat intéressant est que le nombre de paires de phrases extraites à chaque itération est réduit en comparaison avec la *Méthode 2*.

Nous avons aussi évalué la qualité des paires de phrases extraites sur un ensemble de 100 paires de phrases choisies au hasard parmi celles extraites à chaque itération. Premièrement, la précision du processus d'extraction (le pourcentage de paires extraites parallèles et comparables) est calculée pour chaque itération. Le résultat est présenté dans le Tableau 7-2.

Tableau 7-2 : La précision du processus d'extraction après chaque itération dans les systèmes VN – FR et VN-EN

	Méthode 2		Méthode 3	
	# de paires extraites	**Précision**	**# de paires extraites**	**Précision**
VN – FR				
itération 1	6 780	88	6 544	94
itération 2	10 016	88	9 526	93
itération 3	11 672	87	11 219	91
itération 4	12 734	86	12 264	91
itération 5	13 524	85	13 030	90
itération 6	14 100	86	13 597	89
itération 7	14 560	85	14 077	88
itération 8	14 969	84	14 485	86
VN – EN				
itération 1	11 722	82	11 786	86
itération 2	16 549	82	16 129	84
itération 3	18 696	81	17 989	84
itération 4	19 912	80	19 098	83
itération 5	20 712	78	19 900	81
itération 6	21 286	78	20 489	79
itération 7	21 787	78	20 946	78
itération 8	22 149	76	21 323	77

Malgré le nombre de paires extraites réduites, la *Méthode 3* nous donne une précision plus élevée. Au cours des premières itérations, la précision du processus d'extraction de la *Méthode* 3 est beaucoup plus grande que celle de la *Méthode 2*. Mais dans les itérations suivantes, elle est réduite et semble égale à la précision du processus d'extraction de la *Méthode 2*. Une des raisons peut être qu'un grand nombre de paires de phrases non pertinentes est accumulé dans le module de traduction, non seulement des paires de phrases en langue C_1 (resp. C_2) non pertinentes extraites directement mais aussi des paires de phrases synthétisées à partir des paires de phrases non pertinentes en langue C_2 (resp. C_1).

Finalement, nous n'utilisons que les paires de phrases extraites de deux méthodes pour construire deux nouveaux systèmes de traduction S_{M2} et S_{M3}, et comparons les qualités de ces systèmes. S_{M2} est entraîné par des paires de phrases extraites par la *Méthode 2*, et S_{M3} est entraîné par des paires de phrases extraites par la *Méthode 3*. Les scores BLEU des deux systèmes de traduction sont estimés sur le même corpus de test de 400 paires de phrases.

Tableau 7-3 : La qualité des systèmes de traduction entraînés par des paires de phrases extraites avec les Méthode 2 et 3

		Nombre de paires extraites par Méthode 2	**Nombre de paires extraites par Méthode 3**	**Nombre de paires en commun**	**Score BLEU du S_{M2}**	**Score BLEU du S_{M3}**
VN – FR	itération 5	13 524	13 030	10 199	31,25	31,41
	itération 8	14 969	14 485	11 530	31,28	31,86
VN – EN	itération 5	20 712	19 900	15 702	29,19	29,45
	itération 8	22 149	21 323	16 971	29,11	29,36

Dans la tâche d'extraction des phrases pertinentes vietnamiennes – françaises, après 8 itérations, la *Méthode 2* nous donne 14 969 paires de phrases (avec une précision de 84 %), et la *Méthode 3* nous donne 14 485 paires de phrases (avec une précision de 86 %). Il y a 11 530 paires de phrases en commun. Le score BLEU du système de traduction construit avec les données extraites de la *Méthode 2* est de 31,28, et celui du système de traduction construit avec les données extraites de la *Méthode 3* est de 31,86. Malgré le fait que le nombre de paires de phrases

extraites à partir de la *Méthode 3* est inférieur à celui de la *Méthode 2*, la *Méthode 3* nous donne le plus grand score BLEU.

La même tendance est observée pour la tâche d'extraction les phrases pertinentes vietnamiennes–anglaises. Après 8 itérations, la *Méthode 2* nous donne 22 149 paires de phrases vietnamiennes–anglaises (avec une précision de 76 %), et la *Méthode 3* nous donne 21 323 paires de phrases (avec une précision de 77 %). Il y a 16 971 paires de phrases communes. Le score BLEU du système de traduction construit avec les données extraites de la *Méthode 2* est 29,11, et celui du système de traduction construit avec les données extraites de la *Méthode 3* est 29,36.

Tableau 7-4 : Un exemple de la traduction d'une phrase vietnamienne vers français par la Méthode 1, Méthode 2 et Méthode 3 à quelques itérations

Référence en français	le chef d'état vietnamien a proposé que le vietnam et l'australie renforcent les dialogues politiques de haut rang , ainsi que les contacts
Méthode 1	le président vietnamien a demandé ***X*** le vietnam et l'australie **renforcer les dialogues politiques** de haut rang , ainsi que des rencontres
Méthode 2-itér1	le président **du** vietnam ***X*** **sur** le vietnam et l'australie d'intensifier les **dialogue politique** de haut rang , ainsi que des rencontres
Méthode 2-itér2	le président vietnamien a proposé que le vietnam et l'australie **de renforcer** davantage leur dialogue politique de haut rang , ainsi que des rencontres
Méthode 2-itér7	**le président de l'état vietnamien** a proposé ***X*** le vietnam et l'australie **,** de renforcer davantage leur dialogue politique de haut rang ainsi que des rencontres
Méthode 2-itér8	**le président de l'état vietnamien** a demandé **au** vietnam et l'australie d'intensifier **les dialogues politiques** de haut rang , ainsi que des rencontres
Méthode 3-itér1	le président vietnamien a demandé ***X*** le vietnam et l'australie d'intensifier **les dialogues politiques** de haut rang ***X*** ainsi que des rencontres
Méthode 3-itér2	le président vietnamien a demandé ***X*** le vietnam et l'australie d'intensifier **les dialogues politiques** de haut rang , ainsi que des rencontres
Méthode 3-itér4	le président vietnamien a demandé au vietnam et ***X*** l'australie d'intensifier **les dialogues politiques** de haut rang , ainsi que des rencontres
Méthode 3-itér7	le président vietnamien a **proposé** au vietnam et ***X*** l'australie d'intensifier **les dialogues politiques** de haut rang , ainsi que des rencontres

Le Tableau 7-4 montre le même exemple dans le Tableau 6-6 avec la traduction de la *Méthode 3*. La traduction à la première itération de la *Méthode 3* est déjà meilleure que la traduction de la *Méthode 2* à la même itération, grâce aux données supplémentaires en anglais. La traduction à l'itération 4 de la *Méthode 3* est comparable avec celle de la *Méthode 2* à l'itération 7.

Les résultats présentés dans le Tableau 7-2, Tableau 7-3 et Tableau 7-4 peuvent nous convaincre que la qualité des paires de phrases extraites de la méthode de triangulation est meilleure que celle de la méthode non supervisée, grâce aux données supplémentaires de la troisième langue. En tout cas, la méthode de triangulation a montré son efficacité dans l'extraction des données pertinentes à partir d'un corpus comparable.

7.4. Conclusion

La méthode d'extraction non supervisée s'est avérée être efficace dans le chapitre 6, mais elle peut être améliorée grâce à la disponibilité de données supplémentaires d'une autre langue. En même temps que la triangulation est utilisée, les données extraites pour une paire de langues S – C_1 peuvent améliorer le processus d'extraction pour la paire de langues S – C_2 et vice-versa.

Cette méthode peut être appliquée largement pour fouiller un corpus multilingue pour extraire à la fois plusieurs paires de langues différentes.

On peut faire le rapprochement entre notre méthode de triangulation appliquée simultanément sur deux couples de langues *S – C1* et *S – C2* et le processus de co-apprentissage (*co-training*), un concept déjà utilisé dans d'autres tâches du TALN [Blum 1998, Callison-Burch 2002, Guz 2007]. Dans le processus de co-apprentissage, deux classificateurs, qui sont entraînés par deux ensembles de caractéristiques distinctes d'objet, sont utilisés en même temps pour classifier des objets. Dans notre méthode de triangulation, nous avons deux systèmes (d'extraction pour les paires de langues $S - C_1$ et $S - C_2$) indépendants (obtenus à partir de deux corpus comparables indépendants $A_{S\text{-}C1}$ et $A_{S\text{-}C2}$) qui se renforcent l'un l'autre.

On peut envisager d'étendre notre méthode de triangulation pour extraire un corpus parallèle multilingue à partir des corpus monolingues avec la technique de co-apprentissage. Par exemple, au début, nous avons 3 corpus monolingues indépendants en langue S_1, S_2, S_3, et nous voulons trouver les paires de traductions parallèles en trois langues S_1-S_2-S_3. Un groupe de phrases S_1-S_2-S_3 est considéré comme un ensemble de traductions parallèles si les paires S_1-S_2 et S_1-S_3 sont des paires de phrases parallèles. Lorsque les systèmes pour S_1-S_2 et S_1-S_3 sont bien entraînés, nous pouvons les utiliser pour généraliser au groupe de phrases S_1-S_2-S_3. Dans le futur, nous envisageons de construire et valider le processus de co-apprentissage pour extraire un corpus parallèle multilingue à partir des corpus monolingues indépendants (voir la Figure 7-7).

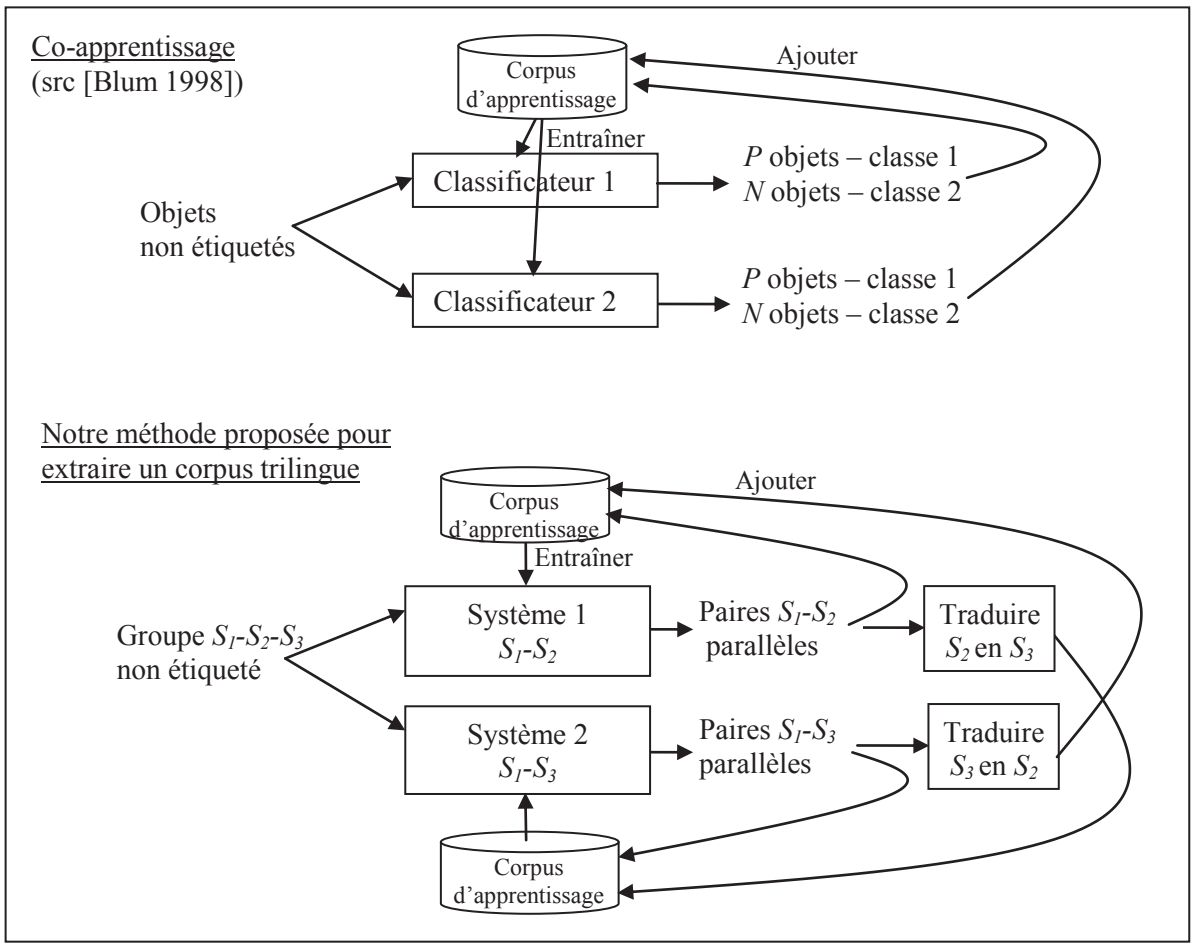


Figure 7-7 : Le processus de co-apprentissage proposé pour extraire un corpus parallèle multilingue à partir des corpus monolingues indépendants

Conclusions et perspectives

Conclusions

Notre manuscrit a présenté notre méthodologie d'extraction de corpus parallèle pour construire un grand corpus d'apprentissage dans le but de développer un système de traduction automatique probabiliste. La nouveauté de notre travail réside dans le fait que nous nous concentrons sur les langues peu dotées, où le corpus d'apprentissage parallèle n'existe pas toujours, ou bien il n'existe qu'avec une quantité faible de données.

Dans ce travail, nous avons proposé trois méthodes d'extraction de données parallèles à partir d'un corpus comparable qui est disponible, riche et diversifiée sur le Web : les sites web multilingues de nouvelles journalistiques. A partir d'un site web multilingue, avec peu ou pas de données parallèles, nos méthodes nous permettent d'extraire un corpus d'apprentissage pour construire rapidement un système de traduction automatique.

Après avoir présenté les études théoriques sur la traduction automatique, l'état de l'art sur l'approche de traduction automatique probabiliste (l'approche de traduction que nous utilisons) et les caractéristiques de la langue vietnamienne (la langue peu dotée que nous avons choisie pour appliquer les méthodes proposées), nous exposons nos contributions en trois méthodes d'extraction.

Les divers niveaux de parallélisme d'une paire de phrases, d'une paire de documents et d'une paire de corpus sont bien définis. La première méthode basée sur des informations lexicales peut extraire à la fois les documents comparables et les phrases parallèles. Cependant, cette méthode requiert des données supplémentaires sur la paire de langues (telles qu'un dictionnaire bilingue, une liste de mots d'arrêt, etc.). Lorsque ces données supplémentaires sont disponibles, nous pouvons appliquer cette méthode pour toute paire de langues. La première méthode est appliquée aux données en vietnamien et français récupérées depuis un site Web multilingue de nouvelles journalistiques, le *Vietnam News Agency*. Nous avons extrait environ de 50 milles paires de phrases, avec 42 % de paires de phrases parallèles, 38 % de paires de phrases comparables, et 20 % de paires de phrases non pertinentes. Nous avons construit aussi des systèmes de TA probabiliste à partir des données extraites. La combinaison des informations entre les mots et les syllabes vietnamiennes peut aussi être utile pour améliorer les performances des systèmes de traduction. Selon des expériences complémentaires, nous voyons que pour les langues peu dotées, en plus de l'extraction de paires de phrases parallèles, l'extraction de paires de phrases comparables est aussi avantageuse pour la construction d'un système de TA probabiliste.

Dans le cas où les données supplémentaires nécessaires pour la première méthode ne sont pas disponibles, nous proposons une deuxième méthode d'extraction, non supervisée, qui peut s'appliquer dans le cas de langues peu dotées. Le processus d'extraction ne requiert aucune donnée supplémentaire sur les deux langues considérées. L'entrée de cette méthode est le corpus comparable à fouiller seul. Le processus d'extraction contient deux étapes : l'étape

d'initialisation et l'étape d'extraction. La différence entre cette méthode et d'autres disponibles dans la communauté se trouve dans l'étape d'initialisation et le score proposé (SWR) qui mesure le parallélisme d'une paire de phrases. Les données pour construire le système d'extraction initial sont extraites directement à partir du corpus comparable à l'entrée, par un processus de filtrage croisé original. Un processus itératif a été utilisé pour augmenter le nombre de paires de phrases extraites et améliorer la qualité du système d'extraction. Les expériences réalisées montrent que notre méthode d'extraction entièrement non supervisée peut être réellement appliquée dans le cas du manque de données parallèles. La méthodologie a été validée sur les même corpus comparables vietnamien – français et vietnamien – anglais du corpus VNA. Nous avons extrait environ de 18 milles paires de phrases vietnamien – français, avec 55 % de paires de phrases parallèles, 33 % de paires de phrases comparables, et 12 % de paires de phrases non pertinentes. Et 25,5 milles paires de phrases vietnamien – anglais, avec 44 % de paires de phrases parallèles, 34 % de paires de phrases comparables, et 22 % de paires de phrases non pertinentes. En plus, malgré le fait que cette méthode non supervisée ne nécessite pas de données supplémentaires, la qualité du module de TA construit est comparable à celle de la première méthode qui nécessite des données de meilleure qualité comme un dictionnaire bilingue, une liste de mots d'arrêt, des heuristiques, etc.

Avec la disponibilité de corpus multilingues sur l'Internet, nous avons proposé, à la fin du manuscrit, l'extension de la méthode d'extraction non supervisée en utilisant la triangulation par une troisième langue. En d'autre termes, les données extraites pour une paire de langues $S - C_1$ peuvent améliorer le processus d'extraction pour la paire de langues $S - C_2$ et vice-versa. Notre méthode de triangulation est appropriée dans le cas où il n'existe que des corpus comparables pour les paires de langue $S - C_1$ et $S - C_2$ (en faible quantité), mais un corpus parallèle pour $C_1 - C_2$ existe en grande quantité. La troisième méthode a été validée sur les même corpus comparables vietnamien – français et vietnamien – anglais du corpus VNA. Nous avons extrait environ de 17,6 milles paires de phrases vietnamien – français, avec 53 % de paires de phrases parallèles, 37 % de paires de phrases comparables, et 10 % de paires de phrases non pertinentes. Et 25,2 milles paires de phrases vietnamien – anglais : 46 % de paires de phrases parallèles, 34 % de paires de phrases comparables, et 20 % de paires de phrases non pertinentes. La troisième méthode a été comparée avec la méthode non supervisée et les résultats montrent une amélioration des la qualité des données parallèles collectées dans tous les cas.

Perspectives

A court terme, nous envisageons d'appliquer et de valider nos méthodes proposées sur d'autres types de ressources, et d'autres langues peu dotées. Et nous avons l'intention d'appliquer cette méthode à une plus grande échelle pour exploiter un plus grand flux de données comparables extraites du Web.

Premièrement, nous voulons d'appliquer nos méthodes sur d'autres ressources de données pour valider les méthodes dans plusieurs domaines. En plus du site Web de *Vietnam News Agency*, nous trouvons d'autres sites Web de nouvelles journalistiques[1] qui possèdent des articles en trois langues vietnamienne, française, anglaise. Donc, le premier travail à court terme est d'appliquer nos méthodes d'extraction avec les données récupérées à partir de ces sites Web. Ce travail nous permettra de collecter plus de données d'apprentissage pour construire les systèmes de traduction depuis et vers la langue vietnamienne.

[1] Les sites Web de nouvelles journalistiques en plusieurs langues : http://www.voanews.com/, http://www.rfi.fr/, http://www.bbc.co.uk/, etc.

Deuxièmement, nous voulons valider nos méthodes sur d'autres langues peu dotées. Une des langues peu dotées qui nous intéresse particulièrement est le khmer, la langue principale et officielle du Cambodge, qui est parlé par environ 14 millions de personnes. Un des difficultés pour travailler avec la langue khmère est le traitement de l'écriture du khmer : les mots sont écrits sans espace et il n'existe pas de ponctuation claire entre les phrases. Heureusement, des travaux ont déjà été menés sur le traitement de corpus de textes en khmer, la segmentation d'une phrase en mots ou en syllabes [Le V.B. 2006, Seng 2010]. Nous trouvons aussi des sites Web de données qui contiennent des articles écrits en langues khmer, français et anglais[1]. Nous souhaitons aussi valider notre méthodologie pour une langue très peu dotée ayant un système d'écriture différent telles quel le khmer.

La validation de nos méthodes avec plusieurs types de données et de paires de langues permettra sans doute de proposer des adaptations et/ou améliorations pour augmenter la performance de la méthode d'extraction.

A long terme, dans la suite de notre travail, nous souhaitons continuer la recherche sur l'extraction de données parallèles au niveau des fragments de phrases (une séquence de mots consécutifs). Nos méthodes d'extraction permettent d'extraire des paires de phrases pertinentes, des paires de phrases parallèles et des paires de phrases comparables. Donc la recherche de paires de fragments de mots parallèles peut augmenter la performance du système de traduction automatique probabiliste et peut aider du processus d'extraction les données parallèles. La recherche de fragments parallèles de mots non consécutifs est aussi novatrice et peut s'avérer intéressante pour les méthodes de traduction hiérarchiques (présence de symboles non terminaux dans les segments).

Par ailleurs, comme présenté à la fin du chapitre 7, nous envisageons de développer et valider le processus de co-apprentissage pour extraire un corpus parallèle multilingue à partir des multiples (>3) corpus monolingues indépendants.

En plus, nous souhaitons de compléter le modèle de traduction probabiliste en ajoutant des règles sur la temporalité. Ce travail est en collaboration avec Nicolas Boffo dans sa thèse nommée « Formalisation de la temporalité en vietnamien pour la traduction automatique »[2]. L'utilisation des règles sur la temporalité pourrait grandement améliorer la traduction des temps des verbes (passé, présent et futur) et les relations temporelles entre différentes parties d'une phrase ou d'un paragraphe (par exemple, la phrase « *je t'ai dit que demain j'irai au laboratoire* » contient une partie « *au passé* » et une partie « *au futur* »).

Pour finir, d'un point de vue de l'animation de la recherche, j'envisage de lancer un pôle de travail autour de la traduction automatique au laboratoire MICA (Hanoï), en collaboration avec le LIG (via la participation à des projets communs de recherche en TA, à des campagnes d'évaluation, etc.).

[1] L'exemple des sites Web en khmer et d'autres langues (anglais, français) http://www.everyday.com.kh/, http://postkhmer.com/, http://www.dap-news.com/, http://ka-set.info/, etc.

[2] La thèse en cotutelle entre Laboratoire Praxiling UMR 5267 (CNRS - Université Montpellier III) et le Centre de recherche MICA - CNRS/UMI2954

Bibliographie

Abréviations utilisées dans la bibliographie

ACL	Association for Computational Linguistics
AMTA	Conference of the Association for Machine Translation in the Americas
ASRU	Automatic Speech Recognition and Understanding workshop
CIKM	Conference on Information and Knowledge Management
COLING	International conference on Computational Linguistics
COLT	Annual conference on Computational Learning Theory
EACL	European chapter of the Association for Computational Linguistics
EAMT	European Association for Machine translation
EMNLP	Conference on Empirical Methods on Natural Language Processing
ENLG	European workshop on Natural Language Generation
ESSLLI	European Summer School in Logic, Language and Information
Eurospeech	European Conference on Speech Communication and Technology
Interspeech	Conference of the International Speech Communication Association
IWSLT	International Workshop on Spoken Language Translation
LATA	International Conference on Language and Automata Theory and Applications
LREC	International conference on Language Resources and Evaluation
HLT/NAACL	Human Language Technologies, Annual conference of the North American chapter of the ACL
NLPRS	Natural Language Processing Pacific Rim Symposium
RANLP	International Conference on Recent Advances in Natural Language Processing
RIAO	Recherche d'Informations Assisté par Ordinateur
SIGIR	Conference on Research and Development in Information Retrieval (Special Interest Group on Information Retrieval)
TALN	Traitement Automatique des Langues Naturelles
TAPD	Tabulation in Parsing and Deduction workshop
TMI	International Conference on Theoretical and Methodological Issues in Machine Translation
VLSP	Vietnamese Language and Speech Processing Workshop
WMT	Workshop on statistical Machine Translation

[Abdul-Rauf 2009] Abdul-Rauf S., H. Schwenk, *On the use of comparable corpora to improve SMT performance*, In proceedings of EACL, Athens (Greece), p.16-23, 1-3 April 2009

[Ahn 2006] Kisuh Ahn and Matthew Frampton, *Automatic generation of translation dictionaries using intermediary languages*, In proceedings of Cross Language Knowledge Induction Workshop, EACL, Trento Italy 2006

[Al-Onaizan 1999] Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky, *Statistical machine translation*, Final report, Johns Hopkins University summer workshop, 1999

[Banerjee 2005] Satanjeev Banerjee and Alon Lavie, *METEOR: An automatic metric for MT evaluation with improved correlation with human judgement*, In proceedings of Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or summarization, ACL 2005

[Baroni 2004] M. Baroni, and S. Bernardini, *BootCaT: Bootstrapping corpora and terms from the web*, In proceedings of LREC 2004

[Berment 2004] V. Berment, *Méthodes pour informatiser des langues et des groupes de langues peu dotées*, Thèse de doctorat, Université Joseph Fourier, 2004

[Bertoldi 2008a] Bertoldi, Nicola, and Marcello Federico, *A new decoder for spoken language translation based on confusion networks*, In proceedings of ASRU 2005

[Bertoldi 2008b] Nicola Bertoldi, Madalina Barbaiani, Marcello Federico, Roldano Cattoni, *Phrase-based statistical machine translation with pivot languages*, In proceedings of IWSLT 2008

[Blum 1998] Blum A. and Mitchell T, *Combining labeled and unlabeled data with co-training*, In proceedings of COLT 1998

[Boitet 2008] Christian Boitet, *Les architectures linguistiques et computationnelles en traduction automatique sont indépendantes*, In proceedings of TALN 2008

[Borin 2000] Lars Borin, *You'll take the high road and I'll take the low road: Using a third language to improve bilingual word alignment*, In proceedings of COLING 2000.

[Brown 1991] Brown P.F., Lai J.C. et Mercer R.L, *Aligning sentences in parallel corpora*, In proceedings of ACL 1991

[Brown 1993] Brown P.F., Pietra S.A.D., Pietra V.J.D., Mercer R.L, *The mathematics of statistical machine translation: parameter estimation*, Computational Linguistics. Vol. 19, no. 2, 1993

[Callison Burch 2002] Chris Callison Burch, *Co-training for Statistical Machine Translation*, Master of science, 2002

[Callison-Burch 2005] Chris Callison-Burch, Philipp Koehn, *Introduction to Statistical Machine Translation*, Introductory course at ESSLLI, 2005.

[Callison-Burch 2006] Chris Callison-Burch, Miles Osborne, Philipp Koehn, *Re-evaluating the role of BLEU in machine translation research*, In the proceedings of EACL 2006.

[Callison-Burch 2007] Chris Callison-Burch, *Machine translation: Word-based and the EM algorithm*, Tutorial at Johns Hopkins University, December 3, 2007.

[Cettolo 2010] Cettolo M., Federico M., Bertoldi N, *Mining parallel fragments from comparable texts*, In proceedings of IWSLT 2010.

[Chen 1993] Chen S.F, *Aligning sentences in bilingual corpora using lexical information*, In proceedings of ACL 1993

[Chen 1998] Stanley F. Chen and Joshua Goodman, *An empirical study of smoothing techniques for language modeling*, Technical report TR-10-98, Computer science group, Harvard University, Aug 1998

[Chen 2000] Chen Jiang et Jian-Yun Nie, *Parallel Web text mining for cross language information retrieval*, In proceedings of RIAO 2000

[Chen 2008] Yu Chen, Andreas Eisele, Martin Kay, *Improving Statistical Machine Translation Efficiency by Triangulation*, In proceedings of LREC 2008

[Chen 2009] Yu Chen, Martin Kay, Andreas Eisele, *Intersecting multilingual data for faster and better statistical translations*, In proceedings of HLT/NAACL 2009

[Chiang 2007] David Chiang, *Hierarchical phrase-based translation*, Computational Linguistics, 33(2):201-228, 2007

[Cohn 2007] Trevor Cohn, Mirella Lapata, *Machine Translation by Triangulation: Making Effective Use of Multi-Parallel Corpora*, In proceedings of ACL 2007

[Collier 1997] Collier N., Hirakawa H., Kumano A., *Creating a noisy parallel corpus from newswire articles using cross-language information retrieval*, Transactions of information processing society of Japan, 1997

[Collins 1997] M. Collins, *Three generative, lexicalized models for statistical parsing*, In proceedings of ACL 1997

[De Gispert 2006] Adrià de Gispert, José B. Mariño, *Catalan-English statistical machine translation without parallel corpus: bridging through Spanish*, In proceedings of LREC 2006

[De Schryver 2002] G. -M. De Schryver, *Web for/as corpus: a perspective for the African languages*, in Nordic Journal of African Studies, no 2, vol. 11, 2002

[Dempster 1977] A.P. Dempster, N.M. Laird, D.B. Rubin, *Maximum likelihood from incomplete data via the EM algorithm*, in Journal of the Royal Statistical Society. Series B (Methodological), Vol.39, No.1.(1977), pp 1-38.

[Dinh D. 2001] D. Dinh, K. Hoang, V-T. Nguyen, *Vietnamese word segmentation*, In proceedings of NLPRS 2001

[Dinh D. 2002] Dien Dinh, *Building a training corpus for word sense disambiguation in English-to-Vietnamese Machine Translation*, In proceedings of Workshop on Machine translation in Asia, COLING 2002

[Dinh D. 2003a] Dinh Dien, Hoang Kiem, *POS-tagger for English-Vietnamese bilingual corpus*, In proceedings of Workshop on Building and using parallel texts: data driven machine translation and beyond, HLT-NAACL 2003

[Dinh D. 2003b] Dinh Dien, Nguyen Luu Thuy Ngan, Do Xuan Quang, Van Chi Nam, *A hybrid approach to word order transfer in the English to Vietnamese machine translation*, In proceedings of MTS 2003

[Dinh D. 2003c] D. Dinh, P-H. Pham, Q-H. Ngo, *Some lexical issues in building electronic Vietnamese dictionary*, Papillon Workshop, 2003

[Dinh D. 2004] Dinh Dien, Thuy Ngan, Xuan Quang, Chi Nam, *The parallel corpus approach to building the syntactic tree transfer set in the English-to-Vietnamese machine translation*, In the proceedings of International Conference on Electronics, Information, and Communications, 2004

[Do T.N.D. 2010] Thi-Ngoc-Diep Do, Laurent Besacier, Eric Castelli. *Improved Vietnamese French parallel corpus mining using English language*, In proceedings of IWSLT 2010

[Doan N.H. 2001] Doan N.H, *Generation of Vietnamese for French-Vietnamese and English-Vietnamese machine translation*, In proceedings of ENLG 2001

[Doan T.T. 1999] Đoàn Thiện Thuật, *Ngữ âm tiếng Việt (in English: Vietnamese phonetics)*, book, Hanoi national university publishing house, 1999

[Doddington 2002] G. Doddington, *Automatic evaluation of machine translation quality using n-gram co-occurrence statistics*, In proceedings of HLT/NAACL 2002

[Dyer 2010] Chris Dyer, Adam Lopez, Juri Ganitkevitch, Johnathan Weese, Ferhan Ture, Phil Blunsom, Hendra Setiawan, Vladimir Eidelman, and Philip Resnik, *Cdec: A decoder, alignment, and learning framework for finite-state and context-free translation models*, In proceedings of ACL (Demo session) 2010

[Dymetman 2010] Marc Dymetman, Nicola Cancedda, *Intersecting hierarchical and phrase-based models of translation, formal aspects and algorithms*. In proceedings of Workshop on syntax and structure in statistical translation, COLING 2010

[Eck 2004] Matthias Eck, Stephan Vogel, and Alex Waibel, *Language model adaptation for statistical machine translation based on information retrieval*, In proceedings of LREC 2004.

[Eisele 2006] A. Eisele, *Parallel corpora and phrase-based statistical machine translation for new language pairs via multiple intermediaries*, In proceedings of LREC 2006

[Fletcher 2004] W. H. Fletcher, *Making the web more useful as a source for linguistic corpora*. Reference in U. Connor & T. Upton (Eds.), Applied corpus linguistics: a multidimensional perspective, Amsterdam/New York: Rodopi Publishers, pp. 191–205, 2004

[Fung 2004a] P. Fung, P. Cheung, *Mining very-non-parallel corpora: parallel sentence and lexicon extraction via bootstrapping and EM*. In proceedings of EMNLP, 2004

[Fung 2004b] P. Fung, P. Cheung, *Multi-level bootstrapping for extracting parallel sentences from a quasi-comparable corpus*, In proceedings of COLING, 2004

[Gale 1993] W.A. Gale, K.W. Church, *A program for aligning sentences in bilingual corpora*, In proceedings of ACL 1993

[Germann 2001] Germann, U., Jahr, M., Knight, K., Marcu, D., and Yamada, K., *Fast decoding and optimal decoding for machine translation*. In Proceedings of ACL 2001

[Ghani 2001] Ghani R., Jones R., Mladenic D., *Mining the web to create minority language corpora*, In proceedings of CIKM 2001

[Gollins 2001] Tim Gollins, Mark Sanderson, *Improving cross language information retrieval with triangulated translation*, In proceedings of SIGIR 2001

[Graff 1994] David Graff, *UN Parallel Text (Complete)*. Linguistic Data Consortium, Philadelphia, 1994.

[Guz 2007] Guz U., Cuendet S., Hakkani-Tür D. & Tur G., *Co-training Using Prosodic and Lexical Information for Sentence Segmentation*, In proceedings of Interspeech 2007

[Hayashi 2010] Katsuhiko Hayashi, Hajime Tsukada, Katsuhito Sudoh, Kevin Duh, Seiichi Yamamoto, *Hierarchical phrase-based machine translation with word-based reordering model*, In proceedings of COLING 2010

[Ho T.B. 2005] T.B. Ho, *Current status of machine translation research in Vietnam, towards asian-wide multilanguage machine translation project*, In proceedings of VLSP Workshop 2005

[Ho T.B. 2008] Ho, T.B., Pham, N.K., Ha, T.L., Nguyen, P.T., *Issues and first phase development of the English Vietnamese translation system Evsmt1.0*, Special Issue in

Journal of Science, Natural Sciences and Technology, Vol. 24, N3S, Vietnam National University- Hanoi Publishers, 2008

[Hoang P. 2002] Hoàng Phê, *Từ điển tiếng Việt (in english: Vietnamese Dictionary),* Lexicography Centre - Maison d'édition Danang, Vietnam, 2002

[Hoang T.C. 2004] Hoàng Thị Châu, *Phương ngữ tiếng Việt (en français : Les dialectes vietnamiens),* Editions de l'Univerité Nationale de Hanoi 2004

[Huang 2007] Liang Huang and David Chiang, *Forest rescoring: faster decoding with integrated language models*, In proceedings of ACL 2007

[Hutchins 2001] W.J. Hutchins, *Machine translation over fifty years*. Histoire, epistémologie, langage. ISSN 0750-8069, 2001

[Imamura 2002] K. Imamura, *Application of translation knowledge acquired by hierarchical phrase alignment for pattern-based MT*, In proceedings of TMI 2002

[Istvan 2009] Varga Istvan, Yokoyama Shoichi, *Bilingual dictionary generation for low-resourced language pairs*, In proceedings of EMNLP 2009

[Kaplan 1982] Kaplan, R. et Bresnan, J., *Lexical-functional grammar: a formal system for grammatical representation*, In the mental representation of grammatical relations, MIT Press, Cambridge, Massachusetts (pages 173–281), 1982

[Khang B.H. 2004] Khang B.H., *Báo cáo tổng kết khoa học và kỹ thuật - Đề tài nghiên cứu phát triển công nghệ nhận dạng, tổng hợp và xử lý ngôn ngữ tự nhiên, Chương trình KC-01 (Science and technical report on "Research and development focus on recognition, synthesis and natural language processing techniques"*), Project KC-01, 2004

[Kilgarriff 2003] Kilgarriff A, Grefenstette G., *Introduction to the special issue on the Web as corpus*, Computational Linguistics, volume 29, 2003

[Kirchhoff 2005] Katrin Kirchhoff and Mei Yang, *Improved language modeling for statistical machine translation*, In proceedings of Workshop on building and using parallel texts, ACL 2005

[Knight 2003] Kevin Knight and Philipp Koehn, *What's New in Statistical Machine Translation*, Tutorial at HLT/NAACL 2003

[Koehn 2003a] Philipp Koehn, *Noun Phrase Translation*, PhD thesis, University of Southern California, 2003

[Koehn 2003b] Philipp Koehn, Franz Josef Och, and Daniel Marcu, *Statistical phrase-based translation*, In proceedings of HLT/NAACL 2003

[Koehn 2004] Philipp Koehn, *Pharaoh: a beam search decoder for phrase-based statistical machine translation models*, In proceedings of AMTA 2004

[Koehn 2005] Philipp Koehn, *Europarl: a parallel corpus for statistical machine translation*, Machine Translation Summit 2005

[Koehn 2007a] Philipp Koehn, Hoang H., Birch A., Callison-Burch C., Zens R., Federico M., Bertoldi N., Cowan B., Shen W., Moran C., *Moses: open source tool-kit for statistical machine translation*, In proceedings of ACL 2007

[Koehn 2007b] Philipp Koehn and Hieu Hoang, *Factored translation models*, In proceedings of EMNLP 2007

[Kumano 2007] Kumano, T., Tanaka H., Tokunaga T., *Extracting phrasal alignments from comparable corpora by using joint probability SMT model*, In proceedings of TMI 2007

[Kumar 2007] Shankar Kumar and Franz Och and Wolfgang Macherey, *Improving word alignment with bridge languages*, In proceedings of EMNLP 2007

[Lavecchia 2010] Caroline Lavecchia, *Les triggers inter-langues pour la traduction automatique statistique*, Thèse de doctorat, Université de Nancy 2

[Le H.P. 2006] Le H.P., *Some issues in Vietnamese Language Processing*, Workshop of Asean Applied NLP for Linguistics Diversity and Language Resource Development, 2006

[Le H.P. 2008] H. P. Le, T. M. H. Nguyen, A. Roussanaly, T. V. Ho, *A hybrid approach to word segmentation of Vietnamese texts*, In proceedings of LATA 2008

[Le K.H 2003a] Lê Khánh Hùng, *Văn phạm phụ thuộc phạm vi (in English: Scope-dependent Grammar),* Vietnamese symposium on Research, Development and Application of Information and Communication Technology, ICT.rda 2003

[Le K.H 2003b] Lê Khánh Hùng, *Một phương pháp dịch máy liên ngữ (in English: One interlingual machine translation method),* Vietnamese symposium on Research, Development and Application of Information and Communication Technology, ICT.rda 2003

[Le M.H. 1997] Le Manh Hai, Asanee Kawtrakul, Yuen Poovorawan. *Phrasal transfer model for Vietnamese-English machine translation,* In proceedings of Workshop on Multilingual Information Processing, NLPRS, 1997

[Le M.H. 2009] Le Manh Hai, Phan Thi Tuoi, *Three algorithms for word-to-phrase machine translation*, In proceedings of International Asian Language Processing, IALP 2009

[Le M.H. 2010] Le Manh Hai, Phan Thi Tuoi, *Lexical gap in English - Vietnamese machine translation: What to do?,* In proceedings of International Asian Language Processing, IALP 2010

[Le V.B. 2003] Le V.B., Bigi B., Besacier L. et Castelli E., *Using the Web for fast language model construction in minority languages*, In proceedings of Eurospeech 2003

[Le V.B. 2006] Lê Việt Bắc, *Reconnaissance automatique de la parole pour des langues peu dotées*, Thèse de doctorat en informatique, Université Joseph Fourier

[Lehtokangas 2002] Raija Lehtokangas, Eija Airio, *Translation via a pivot language challenges direct translation in CLIR*, In proceedings of Workshop I, Cross-language information retrieval: a research map, SIGIR 2002

[Li 2009] Zhifei Li, Chris Callison-Burch, Chris Dyer, Juri Ganitkevitch, Sanjeev Khudanpur, Lane Schwartz, Wren Thornton, Jonathan Weese, and Omar Zaidan, *Joshua: open source toolkit for parsing-based machine translation*. In proceeding of WMT, EACL 2009

[Lopez 2007] Adam Lopez, *A survey of statistical machine translation*, Technical report, Institute for Advanced computer studies, University of Maryland, 2007

[Ma 1999] Ma X. et Liberman M., *Bits: A method for bilingual text search over the web*, In proceeding of Machine Translation Summit 1999

[Ma 2006] Ma X., *Champollion: A robust parallel text sentence aligner*, In proceeding of LREC 2006

[Ma 2008] Ma, Y., Ozdowska, S., Sun, Y. and Way, A. *Improving word alignment using syntactic dependencies*, In proceedings of the Workshop on Syntax and Structure in Statistical Translation (SSST), ACL 2008

[Mai N.C. 1997] Mai Ngọc Chừ, Vũ Đức Nghiệu, Hoàng Trọng Phiến. *Cơ sở ngôn ngữ học và tiếng Việt (La base de linguistique et de la langue vietnamienne)*, Maison d'édition d'éducation, Vietnam 1997

[Marcu 2002] Daniel Marcu and William Wong, *A phrase-based, joint probability model for statistical machine translation*. In proceeding of EMNLP 2002.

[Moore 2002] Moore, *Fast and accurate sentence alignment of bilingual corpora*, In proceedings of AMTA 2002

[Munteanu 2005] Munteanu D.S., Marcu D., *Improving machine translation performance by exploiting non-parallel corpora,* Computational Linguistics 2005

[Munteanu 2006a] Munteanu D.S., *Exploiting Comparable Corpora*, PhD thesis, University of Southern California 2006

[Munteanu 2006b] Munteanu D.S., Marcu D., *Extracting parallel sub-sentential fragments from non-parallel corpora,* In proceedings of ACL 2006

[Nagao 1984] Makoto Nagao, *A framework of a mechanical translation between Japanese and English by analogy principle*, In proceedings of Artificial And Human Intelligence (A. Elithorn and R. Banerji, editors), Elsevier Science Publishers, B.V, 1984.

[Nakov 2009] Preslav Nakov, Hwee Tou Ng, *Improved statistical machine translation for resource-poor languages using related resource-rich languages*, In proceedings of EMNLP 2009

[Nguyen C.T. 2007] Cam-Tu Nguyen and Xuan-Hieu Phan, *JVnSegmenter: A Java-based Vietnamese Word Segmentation Tool*, http://jvnsegmenter.sourceforge.net

[Nguyen H.Q. 2008] Nguyen H.Q., *Reconnaissance automatique de la parole continue à grand vocabulaire en vietnamien*, Thèse de doctorat, Université d'Avignon et des Pays de Vaucluse, 2008

[Nguyen L.T. 2010] Nguyen Lan Trung, *A propos du système lexical du vietnamien*, Synergies Pays riverains du Mékong n1 - pp. 53-71, 2010

[Nguyen T.B. 2004] T.B. Nguyen, T.M.H. Nguyen, L. Romary, X.L. Vu, *Developing tools and building linguistic resources for Vietnamese morpho-syntactic processing*, In proceedings of LREC 2004

[Nguyen T.C. 1978] Nguyễn Tài Cẩn, *Nguồn gốc và quá trình hình thành cách đọc Hán Việt (L'origine et la formation de la lecture de sino-vietnamienne),* Maison d'édition d'Université nationale de Hanoi, 1978

[Nguyen T.H.N. 2008] Nguyen Thi Hong-Nhung and Dien Dinh. *A syntactic-based word re-ordering for English Vietnamese statistical machine translation system*. Lecture Notes in Computer Science, 2008, Volume 5351, PRICAI 2008: Trends in Artificial Intelligence

[Nguyen T.M.H. 2006] Nguyen T.M.H., *Outils et ressources linguistiques pour l'alignement de textes multilingues français-vietnamiens*, Thèse de doctorat, Université Henri Poincaré, Nancy 1 2006

[Nguyen T.P. 2007] Nguyen P.T., A. Shimazu, M.L. Nguyen, V.V. Nguyen, *A syntactic transformation model for statistical machine translation*, In proceedings of International Journal of Computer Processing of Oriental Languages, 2007

[Nguyen T.P. 2008] Thai Phuong Nguyen, Akira Shimazu, Tu-Bao Ho, Minh Le Nguyen, Vinh Van Nguyen. *A tree-to-string phrase-based model for statistical machine translation*. In proceedings of the Twelfth Conference on Computational Natural Language Learning, 2008.

[Nguyen M.C. 2005] Nguyen My Chau, Ikeda Takashi. *Translation of adnominal modification structures in Japanese Vietnamese machine translation*. In proceedings of Natural Language Processing Vol.12. No.3, pp.145-182, 2005.

[Nguyen M.C. 2006] Nguyen My Chau, Tanaka Yuki, Ikeda Takashi, *Translation of structure [N1 no N2] in Japanese-Vietnamese machine translation*. In journal of Natural Language Processing, 2006.

[Niessen 2004] Sonja Niesen and Hermann Ney, *Statistical machine translation with scarce resources using morpho-syntactic information*, Computational Linguistics, 30(2): 181-204, 2004

[Nikoulina 2010] Vassilina Nikoulina, *Modèle de traduction statistique à fragments enrichis par la syntaxe*, Thèse de doctorat, Université de Grenoble

[Och 1999] Och, Franz-Josef, Christoph Tillmann, and Hermann Ney, *Improved alignment models for statistical machine translation*. In Proceedings of the Joint Conference on EMNLP and Very Large Corpora (EMNLP/VLC) 1999

[Och 2000] Franz Josef Och, Hermann Ney, *Improved Statistical Alignment Models*, In Proceedings of ACL, 2000.

[Och 2001] Och, Franz-Josef, Nicola Ueffing, and Hermann Ney, *An efficient A* search algorithm for statistical machine translation,* In proceedings of the Data-Driven Machine Translation Workshop, ACL 2001

[Och 2002] Franz Josef Och, Hermann Ney, *Discriminative training and maximum entropy models for statistical machine translation.* In proceedings of ACL 2002

[Och 2003a] F. Och, *Minimum error rate training in statistical machine translation*. In proceedings of ACL 2003

[Och 2003b] Och, F.J., Gildea, D., Khudanpur, S., Sarkar, A., Yamada, K., Fraser, A., Kumar, S., Shen, L.,Smith, D., Eng, K., Jain, V., Jin, Z. and Radev, D., *Syntax for statistical machine translation*, Technical report, Summer Workshop, John Hopkins University 2003

[Och 2004] Franz Josef Och and Hermann Ney, *The alignment template approach to machine translation.* Computational Linguistics, volume 30, 2004

[Och 2005] F. Och, *Statistical machine translation: foundations and recent advances*, Tutorial at MT Summit 2005

[Och 2006] Franz J. Och, *Challenges in Machine Translation*, Invited Talk at TC-STAR Workshop on Speech-to-Speech Translation, 2006

[Oepen 2007] Oepen, S.,Velldal, E., Lonning, J.T., Meurer, P., Rosén, V. and Flickinger, D., *Towards hybrid quality-oriented machine translation on linguistics and probabilities in MT*, In proceedings of TMI 2007

[Ogilvie 2001] Ogilvie, Paul and Jamie Callan, *Experiments using the lemur toolkit*. In the proceedings of the Text Retrieval Conference, TREC 2001

[Papineni 2002] Papineni K., Roukos S., Ward T., Zhu W., *BLEU: a method for automatic evaluation of machine translation*. In proceedings of ACL 2002

[Patry 2005] Patry A., Langlais P., *Paradocs: un système d'identification automatique de documents parallèles*. In proceedings of TALN 2005

[Popovic 2006] Maja Popovic and Hermann Ney, *Statistical machine translation with small amounts of bilingual training data,* In proceedings of SALTMIL Workshop on Minority Languages, LREC 2006

[Quirk 2007] Quirk C., Udapa R., Menezes A., *Generative models of noisy translations with applications to parallel fragment extraction*, In proceedings of MT Summit, EAMT 2007

[Resnik 1998] Philip Resnik, *Parallel Strands: a preliminary investigation into mining the web for bilingual text*, In proceedings of AMTA 1998

[Resnik 1999] Philip Resnik, *Mining the Web for bilingual text*, In proceedings of ACL 1999

[Resnik 2003] Philip Resnik and Noah A. Smith, *The Web as a parallel corpus*, Computational Linguistics, vol. 29, no. 3, 2003

[Robertson 1994] S. E. Robertson and S. Walker, *Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval*, In proceedings of SIGIR 1994

[Rosenfeld 2000] Roni Rosenfeld, *Two decades of statistical language modeling: where do we go from here?*, In proceedings of IEEE,88 (8):1270–1278, Aug 2000

[Sanchez-Martinez 2008] Felipe Sánchez-Martínez, *Using unsupervised corpus-based methods to build rule-based machine translation systems*, PhD thesis, Universitat d'Alicante, 2008

[Sarikaya 2009] Sarikaya R., Maskey S., Zhang R., Jan E., Wang D., Ramabhadran B., Roukos S., *Iterative sentence–pair extraction from quasi–parallel corpora for machine translation*, In proceedings of Interspeech 2009

[Sato 1990] Sato, S., and Nagao, M., *Toward memory-based translation*, In proceedings of COLING 1990

[Scannell 2007] Scannell, K. P., *The Crúbadán Project: Corpus building for under-resourced languages*. In C. Fairon, H. Naets, A. Kilgarriff & G.-M. De Schryver (Eds.), Building and exploring web corpora (pp. 5–15), 2007

[Schwenk 2006] Holger Schwenk, Daniel Dchelotte, Jean-Luc Gauvain, *Continuous space language models for statistical machine translation*, In proceedings of COLING/ACL2006

[Schwenk 2007] Holger Schwenk, *Continuous space language models*. Computer Speech and Language 21(2007) 492–518, 2007

[Schwenk 2008] Holger Schwenk, *Investigations on large-scale lightly-supervised training for statistical machine translation*, In proceedings of IWSLT 2008

[Schwenk 2010] Holger Schwenk, *Continuous space language models for statistical machine translation,* The Prague Bulletin of Mathematical Linguistics, (93):137-146, 2010

[Seng 2010] Sopheap SENG, *Vers une modélisation statistique multiniveau du langage, application aux langues peu dotées*, Thèse de doctorat en informatique, Université Joseph Fourier, 2010

[Shen 2004] Shen, L., *Discriminative reranking for machine translation*, In proceedings of HLT/NAACL 2004

[Shi 2006] Shi L. Niu C., Zhou M., Gao J., *A DOM tree alignment model for mining parallel data from the web*, In proceedings of Computational Linguistics / ACL 2006

[Snover 2006] Matthew Snover, Dorr B., Schwartz R., Micciulla L., Makhoul J., *A study of translation edit rate with targeted human annotation,* In proceedings of AMTA 2006

[Snover 2009] Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz*, Fluency, Adequacy, or HTER? Exploring Different Human Judgments with a Tunable MT Metric,* In proceedings of the Fourth WMT at the 12th EACL, Athens, Greece, 2009

[Steinberger 2006] Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Daniel Varga, *The JRC-Acquis: a multilingual aligned parallel corpus with 20+ languages*, In proceedings of LREC 2006

[Stolcke 2002] Stolcke A., *SRILM an extensible language modeling toolkit*, International Conference on Spoken Language Processing 2002

[Tanaka 1994] Kumikko Tanaka, Kyoji Umemura, *Construction of a bilingual dictionary intermediated by a third language*, In proceedings of COLING 1994

[Tillmann 1997] Tillmann C., Vogel S., Ney H., Zubiaga A., Sawaf H., *Accelerated DP-based search for statistical translation,* In proceedings of Eurospeech 1997

[Tillmann 2003] Christoph Tillmann et Hermann Ney, *Word reordering and a dynamic programming beam search algorithm for statistical machine translation*. In proceedings of ACL 2003

[Tillmann 2009] Tillmann C., Xu J., *A simple sentence-level extraction algorithm for comparable data*, In proceedings of HLT/NAACL 2009

[Tran D.D. 2007] Trần Đỗ Đạt, *Synthèse de la parole à partir du texte en langue vietnamienne*, Thèse de doctorat, l'INP Grenoble

[Uchida 2001] Hiroshi Uchida, Meiying Zhu, *The universal networking language beyond machine translation,* UNDL Foundation, Japan, 2001

[Utiyama 2003] Utiyama M. and H. Isahara, *Reliable measures for aligning Japanese-English news articles and sentences*, In proceedings of ACL 2003

[Utiyama 2007] Utiyama M. and H. Isahara, *A comparison of pivot methods for phrase-based statistical machine translation*, In proceedings of NAACL-HLT 2007

[Vauquois 1968] Vauquois, B., *A survey of formal grammars and algorithms for recognition and transformation in machine translation*. IFIP Congress-68, Edinburgh, 254–260; reprinted in Ch. Boitet (ed.) *Bernard Vauquois et la TAO: Vingt-cinq Ans de Traduction Automatique* – Analectes, 201–213, Grenoble (1988): Association Champollion. 1968.

[Venugopal 2005] Venugopal, Ashish and Stephan Vogel, *Considerations in maximum mutual information and minimum classification error training for statistical machine translation*. In proceedings of EAMT 2005

[Vilar 2010] D. Vilar, D. Stein, M. Huck, and H. Ney, *Jane: open source hierarchical translation, extended with reordering and lexicon models*. In proceedings of WMT and Metrics MATR, ACL 2010

[Vogel 1996] Stephan Vogel, Hermann Ney, and Christoph Tillmann, *HMM-based word alignment in statistical translation*. In proceedings of COLING 1996

[Vu H. 2008] Vu Hoang, Mai Ngo, Dien Dinh, *A dependency-based word reordering approach for statistical machine translation*. In proceedings of Research, Innovation and Vision for the Future, RIVF, IEEE International Conference, 2008

[Wang 1997] Y. Wang and A. Waibel, *Decoding algorithm statistical machine translation*, In proceedings of ACL 1997

[Wang 2006] Haifeng Wang , Hua Wu , Zhanyi Liu, *Word alignment for languages with scarce resources using bilingual corpora of other language pairs*. In proceedings of COLING/ACL 2006

[Wu 1994] Wu D., *Aligning a parallel English-Chinese corpus statistically with lexical criteria*, In proceedings of ACL 1994

[Wu 2007] Wu H., H. Wang, *Pivot language approach for phrase-based statistical machine translation,* In proceedings of ACL 2007

[Yamada 2001] Yamada, K. and Knight, K., *A syntax-based statistical translation model*, In proceedings of ACL 2001

[Yang 2002] Yang C.C. et Li P.W., *Mining english/chinese parallel documents from the world wide web*. In proceedings of the 11th International World Wide Web Conference, 2002

[Zhao 2002] Zhao B., Vogel S., *Adaptive parallel sentences mining from Web bilingual news collection*, In proceedings of IEEE International Conference on Data Mining 2002

Bibliographie personnelle

1. Viet-Bac Le, Laurent Besacier, Sopheap Seng, Brigitte Bigi, **Thi-Ngoc-Diep Do**. *Recent advances in Automatic Speech Recognition for Vietnamese*. In proceedings of the International Workshop on Spoken Languages Technologies for Under-resourced Languages, SLTU 2008. Hanoi, Vietnam.

> This paper presents our recent activities for automatic speech recognition for Vietnamese. First, our text data collection and processing methods and tools are described. For language modeling, we investigate word, sub-word and also hybrid word/sub-word models. For acoustic modeling, when only limited speech data are available for Vietnamese, we propose some crosslingual acoustic modeling techniques. Furthermore, since the use of sub-word units can reduce the high out-of-vocabulary rate and improve the lack of text resources in statistical language modeling, we propose several methods to decompose normalize and combine word and sub-word lattices generated from different ASR systems. Experimental results evaluated on the VnSpeechCorpus demonstrate the feasibility of our methods.

2. **Thi-Ngoc-Diep Do**, Viet-Bac Le, Brigitte Bigi, Laurent Besacier, Eric Castelli. *Mining a comparable text corpus for a Vietnamese - French statistical machine translation system*. In proceeding of WMT, EACL 2009, Athens, Greece.

> This paper presents our first attempt at constructing a Vietnamese-French statistical machine translation system. Since Vietnamese is an under-resourced language, we concentrate on building a large Vietnamese-French parallel corpus. A document alignment method based on publication date, special words and sentence alignment result is proposed. The paper also presents an application of the obtained parallel corpus to the construction of a Vietnamese-French statistical machine translation system, where the use of different units for Vietnamese (syllables, words, or their combinations) is discussed.

3. **Thi-Ngoc-Diep Do**, Viet-Bac Le, Brigitte Bigi, Laurent Besacier, Eric Castelli. *Exploitation d'un corpus bilingue comparable pour la création d'un système de traduction probabiliste Vietnamien - Français*. In proceeding of TALN 2009, Senlis, France.

> Cet article présente nos premiers travaux en vue de la construction d'un système de traduction probabiliste pour le couple de langue vietnamien-français. La langue vietnamienne étant considérée comme une langue peu dotée, une des difficultés réside dans la constitution des corpus parallèles, indispensable à l'apprentissage des modèles. Nous nous concentrons sur la constitution d'un grand corpus parallèle vietnamien-français. Une méthode d'identification automatique des paires de documents parallèles fondée sur la date de publication, les mots spéciaux et les scores d'alignements des phrases est appliquée. Cet article présente également la construction d'un premier système de traduction automatique probabiliste vietnamien-français et français-vietnamien à partir de ce corpus et discute l'opportunité d'utiliser des unités lexicales ou sous-lexicales pour le vietnamien (syllabes,

mots, ou leurs combinaisons). Les performances du système sont encourageantes et se comparent avantageusement à celles du système de Google.

4. **Thi-Ngoc-Diep Do**, Laurent Besacier, Eric Castelli. *Unsupervised SMT for a low-resourced language pair*, In proceedings of the International Workshop on Spoken Languages Technologies for Under-resourced Languages - SLTU 2010, Penang, Malaysia.

This paper presents an unsupervised method in application of extracting parallel sentence pairs from a comparable corpus. A translation system is used to mine the comparable corpus and to withdraw the parallel sentence pairs. An iteration process is implemented not only to increase the number of extracted parallel sentence pairs but also to improve the quality of translation system. A comparison between this unsupervised method and a semi-supervised method is also presented. The unsupervised extracting method was tested in a hard condition: the parallel corpus did not exist and the comparable corpus contained up to 50% of non parallel sentence pairs. However, the result shows that the unsupervised method can be really applied in the case of lacking parallel data.

5. **Thi-Ngoc-Diep Do**, Laurent Besacier, Eric Castelli. *A Fully Unsupervised Approach for Mining Parallel Data from Comparable Corpora*. In proceeding of EAMT 2010, Saint-Raphaël, France.

This paper presents an unsupervised method for extracting parallel sentence pairs from a comparable corpus. A translation system is used to mine the comparable corpus and to detect parallel sentence pairs. An iterative process is implemented not only to increase the number of extracted parallel sentence pairs but also to improve the overall quality of the translation system. A comparison between this unsupervised method and a semi-supervised method is also presented. The unsupervised method was tested in a hard condition: no available parallel corpus to bootstrap the process and the comparable corpus contained up to 50% of non parallel data. The experiments conducted show that the unsupervised method can be really applied in the case of lacking parallel data. While preliminary experiments are conducted on French-English translation, this unsupervised method is also applied successfully to a low e-resourced language pair (French-Vietnamese).

6. **Thi-Ngoc-Diep Do**, Laurent Besacier, Eric Castelli. *Apprentissage non supervisé pour la traduction automatique : application à un couple de langues peu doté*. In proceeding of TALN 2010, Montréal, Canada.

Cet article présente une méthode non-supervisée pour extraire des paires de phrases parallèles à partir d'un corpus comparable. Un système de traduction automatique est utilisé pour exploiter le corpus comparable et détecter les paires de phrases parallèles. Un processus itératif est exécuté non seulement pour augmenter le nombre de paires de phrases parallèles extraites, mais aussi pour améliorer la qualité globale du système de traduction. Une comparaison avec une méthode semi-supervisée est présentée également. Les expériences montrent que la méthode non-supervisée peut être réellement appliquée dans le cas où on manque de données parallèles. Bien que les expériences préliminaires soient menées sur la traduction français-anglais, cette méthode non-supervisée est également appliquée avec succès à un couple de langues peu doté : vietnamien-français.

7. **Thi-Ngoc-Diep Do**, Laurent Besacier, Eric Castelli. *Improved Vietnamese-French Parallel Corpus Mining Using English Language*. In proceeding of IWSLT 2010, Paris, France.

This paper improves our unsupervised method for extracting parallel sentence pairs from a comparable corpus presented in [1]. In this former paper, a translation system was used to mine a comparable corpus and to detect French-Vietnamese parallel sentence pairs. An iterative process was implemented to increase the number of extracted parallel sentence pairs which improved the overall quality of the translation. This paper validates the unsupervised approach on a new under-resourced language pair (Vietnamese-English) and it also addresses the problem of using triangulation through a third language to improve the

parallel data mining process. An extension of the unsupervised method is proposed to make use of triangulation. Two ways to include the additional data from triangulation are carried out. The experiments conducted on Vietnamese – French show that using triangulation through English can improve the quality of the extracted data and slightly improve the quality of the translation system measured with BLEU.

8. Laurent Besacier, Haithem Afli, **Do Thi Ngoc Diep** , Hervé Blanchon, Marion Potet. *LIG Statistical Machine Translation Systems for IWSLT 2010*. In proceeding of IWSLT 2010, Paris, France.

This paper describes the systems developed by the LIG laboratory for the 2010 IWSLT evaluation. We participated to the AE BTEC task and to the new TALK task. For AE BTEC task we developed two different systems: a statistical phrase-based system and a hierarchical phrase-based system using the Moses toolkit. The combination of these systems, which improves the results on different development sets, makes our final submission. This year, we concentrated on the new TALK task. The development of a reference translation system, as well as an ASR output translation system, is presented. For this latter task, re-punctuating the ASR output, before translation, seems to be very useful, while segmenting the ASR flow, which is also discussed in this paper, has shown to be less useful. Unsuccessful attempts to exploit ASR lattices instead of ASR 1best are also presented at the end of this article.

9. **Thi-Ngoc-Diep Do**, Eric Castelli, Laurent Besacier. *Mining Parallel Data from Comparable Corpora via Triangulation*. In proceeding of International Conference on Asian Language Processing, IALP 2011, Penang, Malaysia.

This paper improves an unsupervised method for extracting parallel sentence pairs from a comparable corpus by using the triangulation through a third language. Before, an unsupervised method for extracting parallel sentence pairs from a comparable corpus has been proposed. This method is based on technique of cross-language information retrieval with iterative process and requires no more additional parallel data. The method has been validated on the Vietnamese-French and Vietnamese-English bilingual data. In this paper, we address the problem of using triangulation through a third language to improve the parallel data mining processes: English is used in the Vietnamese-French parallel data mining process, and French is used in the Vietnamese-English parallel data mining process. The experiments conducted show that using triangulation can improve the quality of the extracted data and the quality of the translation system as well.

Annexes

Annexe 1 : Software for sentence alignment

Some of the software for sentence alignment can be found, such as

- GMA: Geometric Mapping and Alignment http://nlp.cs.nyu.edu/GMA
- Bilingual sentence aligner (Microsoft) : http://research.microsoft.com/research/downloads/default.aspx
- Align: http://www.cs.unt.edu/~rada/wa/tools/aberger
- Vanilla: An implementation of the Gale&Church sentence alignment algorithm http://nl.ijs.si/telri/Vanilla/
- UPlug: http://stp.ling.uu.se/cgi-bin/joerg/Uplug
- CTK: Champollion Tool Kit: http://champollion.sourceforge.net
- Hunalign: http://mokk.bme.hu/resources/hunalign

Here we present a simple test of performance between two sentence aligners, the Champollion Tool Kit and the Hunalign, on the comparable data.

Hunalign:

- Use Gale-Church sentence-length information
- In the presence of a dictionary, hunalign uses it, combining this information with Gale-Church sentence-length information. In the absence of a dictionary, it first falls back to sentence-length information, and then builds an automatic dictionary based on this alignment. Then it realigns the text in a second pass, using the automatic dictionary.
- Hunalign does not deal with changes of sentence order: it is unable to come up with crossing alignments, i.e., segments A and B in one language corresponding to segments B' A' in the other language.

Champollion:

- It assumes a noisy input, i.e. that a large percentage of alignments will not be 1:1 alignments
- The number of deletions and insertions will be significant.
- Non-lexical measures, such as sentence length can and should still be used, but they should only play a supporting role when lexical evidence is present.
- It assigns weights to translated words, Uses a function to compute the similarity between any two segments, each of which consists of one or more sentences, Uses a dynamic programming method to find the optimal alignment which maximizes the similarity of the source text and the translation

The input of this test is a tokenized and sentence-segmented text in two languages: Vietnamese and French. The output presents pairs of aligned sentences. On the other hand, the text is aligned

manually to test the performance of these aligners. In the following table, we present two small sample texts with the alignment output. The output in **bold** is wrong result.

SRC sentences	TGT sentences
1. dedicace 2. a léon werth 3. je demande pardon aux enfants d'avoir dédié ce livre à une grande personne 4. j'ai une excuse sérieuse cette grande personne est le meilleur ami que j'ai au monde 5. j'ai une autre excuse cette grande personne peut tout comprendre même les livres pour enfants 6. j'ai une troisième excuse cette grande personne habite la france où elle a faim et froid 7. elle a besoin d'être consolée 8. si toutes ces excuses ne suffisent pas je veux bien dédier ce livre à l'enfant qu'a été autrefois cette grande personne 9. toutes les grandes personnes ont d' abord été des enfants 10. mais peu d'entre elles s'en souviennent je corrige donc ma dédicace 11. a léon werth quand il était petit garçon 12. premier chapitre 13. lorsque j'avais six ans j'ai vu une fois une magnifique image dans un livre sur la forêt vierge qui s'appelait histoires vécues 14. ca représentait un serpent boa qui avalait un fauve 15. voilà la copie du dessin 16. on disait dans le livre les serpents boas avalent leur proie tout entière sans la mâcher	1. gửi léon werth 2. tôi xin lỗi các bé con vì đã đề tặng cuốn sách này cho một người lớn 3. tôi có một lẽ chân xác để tự bào chữa và xin được thứ lỗi người lớn nọ là người bạn chí thiết trong đời tôi 4. tôi còn một lẽ nữa người lớn nọ có thể hiểu hết mọi sự ngay cả những cuốn sách viết cho bé con người ấy cũng hiểu nốt 5. tôi còn một lẽ thứ ba để được tha thứ người lớn nọ hiện sống ở nước pháp và đang chịu đói và rét 6. y thật cần được an ủi 7. nếu tất cả những lẽ đó không đủ để bào chữa cho mình thì tôi rất muốn đề tặng cuốn sách này cho đứa con mà xưa kia người lớn nọ vốn đã từng là nó vậy 8. mọi người lớn ban sơ đều đã từng là những bé con 9. nhưng ít người trong số đó ghi nhớ điều kia 10. vậy tôi xin sửa chữa lời đề tặng 11. gửi léon werth 12. thuở ông ta còn là bé con 13. chương i 14. thuở lên sáu một lần nọ tôi thấy một bức tranh lộng lẫy trong một cuốn sách viết về rừng thẳm nhan đề sự tích đã sống 15. bức tranh đó họa một con trăn đương nuốt con mãnh thú 16. trên đây là bản đồ mô phỏng bức họa kia 17. trong cuốn sách người ta nói "giống trăn nuốt toàn thể con mồi không nhai nghiền gì cả

Expected result	Hunalign result	Champollion result
1 <=>		1 <=>
2 <=>1	2 <=> 1	2 <=> 1
3<=>2	3 <=> 2	3 <=> 2
4<=>3	4 <=> 3	4 <=> 3
5<=>4	5 <=> 4	5 <=> 4
6<=>5	6 <=> 5	6 <=> 5
7<=>6	7 <=> 6	**omitted <=> 6**
8<=>7	8 <=> 7	**7,8 <=> 7**
9<=>8	9 <=> 8	9 <=> 8
10<=>9,10	**10 <=> 9**	10 <=> 9,10
11<=>11, 12	**11 <=> 10~11**	11 <=> 11,12
12<=>13	**12 <=> 12**	12 <=> 13
13<=>14	**13 <=> 13~14**	13 <=> 14

14<=>15 15<=>16 16<=>17	14 <=> 15 15 <=> 16 16 <=> 17	14 <=> 15 15 <=> 16 16 <=> 17

SRC sentences	**TGT sentences**
26. séminaire de sethserey sam, titre : analyse de la langue khmère en vue de la synthèse de la parole, soutenance de fin de projet master, le sept septembre deux mille sept, centre mica	25. báo cáo bảo vệ cao học của anh sethserey sam 26. nhan đề phân tích tiếng khmère phục vụ tổng hợp tiếng nói 27. ngày bảy tháng chín hai nghìn không trăm linh bảy 28. địa điểm trung tâm mica
31. date : le quatorze septembre deux mille six, seize h zéro zéro, lieu mica center	36. thời gian mười sáu h không không ngày mười bốn tháng chín hai nghìn không trăm linh sáu 37. địa điểm trung tâm mica
32. le centre mica est umi deux mille neuf cents cinquante quatre du cnrs, signature de la convention tripatite entre l' ip de hanoi le cnrs et l'inp grenoble le seize mai deux mille six	38. trung tâm mica trở thành trung tâm nghiên cứu quốc tế hỗn hợp umi hai nghìn chín trăm năm mươi tư thuộc cnrs 39. lễ ký kết thỏa thuận được tổ chức sáng ngày mười sáu không năm hai nghìn không trăm linh sáu
33. la convention a été signée en présence de ml' ambassadeur de france	
34. copyright deux mille quatre mica	40. bản quyền của trung tâm mica hai nghìn không trăm linh tư
35. commentaire pour le site	41. góp ý với chúng tôi

Expected result	**Hunalign result**	**Champollion result**
26 <=> 25,26,27,28 31 <=> 36,37 32,33 <=> 38,39 34 <=> 40 35 <=> 41	26 <=>25,26,27,28 **31<=> 36** **32<=> 37,38** **33<=> 39** 34<=> 40 35<=> 41	26 <=> 25,26,27,28 31 <=> 36,37 32,33 <=> 38,39 34 <=> 40 35 <=> 41

→ After considering a large number of sample test texts, we decided to use Champollion toolkit in our work, because it seem to be suitable with the comparable input text.

Annexe 2 : MOSES, How it trains the data

"*Moses is a statistical machine translation system that allows you to automatically train translation models for any language pair. All you need is a collection of translated texts (parallel corpus). An efficient search algorithm finds quickly the highest probability translation among the exponential number of choices*". This is the introduction of MOSES on its official web site http://www.statmt.org/moses/. At this address, you can find out all information concerning about how to get MOSES, how to install MOSES (in local machine and server), how to use MOSES to train the data, how to turn the development data and how to decode a new data. Here we present our work on analysis what MOSES does to learn the translation model from the training data (how it builds the phrase table from the training data). This work is based on the coding file *train_factored_phrase_table.pl* in MOSES' scripts.

TRAIN_FACTORED_PHRASE_TABLE.PL: apply to Foreign to English translation direction

Nine Steps

(1) Prepare corpus
(2) Run GIZA to get word alignments
(3) Merge word alignment in two directions
(4) Learn lexical translation
(5) Extract phrases
(6) Score phrases
(7) Learn reordering model
(8) Learn generation model
(9) Create decoder config file

Options

Required:

- 'root-dir=s' : path to working directory
- 'corpus=s' : path to training data
- 'f=s' : source language
- 'e=s' : target language
- 'lm=s' : list of Language Models

Not required:

Set up path, file name etc.:

- 'corpus-dir=s' : Default: root-dir/corpus
- 'corpus-compression=s' : if training data is zipped -> s = ".gz"
- 'model-dir=s' : Default s= root-dir/model
- 'verbose' : It is not used in script
- 'parallel' : If it is set, run some processes on parallel
- 'help' :

- 'debug' :
- 'nodebug' :
- 'dont-zip' : Default = not set
- 'input-factor-max=i' :,
- 'decoding-steps=s' : used to write to moses.ini, Default = "t0"
  Ex. "t0,g0,t1,g1:t2" : 2 paths.
  1st path: translate step 1, generate step 1, translate step 2, generate step 2.
  2nd path: translate step 3.
- 'scripts-root-dir=s' : path to training scripts. automatically updated, allow to override,
- 'factor-delimiter=s' : delimiter used in training data. Default = "|"
- 'config=s' : Default = root-dir/model/moses.ini

Work with lexical files:

- 'lexical-file=s' :Default s= root-dir/model/lex,
- 'no-lexical-weighting' : if set, don't create files : root-dir/corpus/*.vcb

Work with Giza:

- 'alignment-factors=s' : factor use to align words (Giza). "0,1,2–0,1". One factor: "$0,1,2_{(SRC\ factors)}$–$0,1_{(TGT\ factors)}$". Default = "0-0"
- 'bin-dir=s' : path to *giza++, mkcls, snt2cooc.out*, automatically updated, allow to override
- 'giza-e2f=s' : Default: s = root-dir/giza.e-f
- 'giza-f2e=s' : Default: s= root-dir/giza.f-e
- 'giza-extension=s' : Default: s = A3.final
- 'first-step=i' : Default =1
- 'last-step=i' : Default =9
- 'giza-option=s' : Additional options for Giza++
- 'hmm-align' : if it is set, giza-extension is set to "Ahmm.5", options for Giza++ are changed
- 'parts=i' : run Giza on part. Default =1 => run all
- 'direction=i' : Default i=0. i=1: run Giza for direction Fr -> En, i=2: run Giza for direction En->Fr, i=0: both directions
- 'only-print-giza' : only print command, don't execute Giza command,

Work with phrase alignments:

- 'max-phrase-length=i' : Default i = 7: maximum phrase length in an phrase alignment.
- 'alignment=s' : The way to create phrase alignments from word alignments. Default s=grow-diag-final. Other ways: union/intersert/grow/srttotgt/tgttosrc + [diag][final][final-and]
- 'alignment-file=s' : Default s= root-dir/model/aligned
- 'extract-file=s' : Default s=root-dir/model/extract
- 'translation-factors=s' : factors used to create phrase-table. "0-0+0,1-0,1". Factor 1: "0-0", Factor 2: "0,1-0,1". Default = "0-0"
- 'phrase-translation-table=s' : list of phrase-tables. Used to override the built phrase-tables. Listed in translation factors' order.

Work with reordering tables:

- 'reordering-factors=s' : factors used to create reordering table and extract phrase. "0-0+0,1-0,1". Factor 1: "0-0", Factor 2: "0,1-0,1". Default = "0-0"
- 'reordering=s' : type of reordering concerned. Default = distance. Other type: =msd(orientation)/monotonicity [+ bidirectional] + f/fe
- 'reordering-smooth=s' : Default = 0.5u,
- 'reordering-table=s' : list of reordering tables. Used to override the built reordering tables. Listed in reordering factors' order.
- 'extract-file=s' : Default s=root-dir/model/extract (Use root-dir/model/extract.o.gz)

Work with generation tables:

- 'generation-factors=s' : factors used to create generation table. "0-0+0,1-0,1". Factor 1: "0-0", Factor 2: "0,1-0,1". Default = unDefault
- 'generation-table=s' : list of generation tables. Used to override the built generation tables. Listed in generation factors' order
- 'generation-type=s' : list of generation types, *single* or *double*. Listed in generation factors' order.

Step (1): prepare corpus

1. Create ./corpus directory
2. Extract alignment factors (0-0) from training file: input: trn.fr, trn.en, output: trn.0-0.fr, trn.0-0.en
3. Make classes by running commands:
   + mkcls -c50 -n2 -ptrn.0-0.fr -Vfr.vcb.classes opt
   + mkcls -c50 -n2 -ptrn.0-0.en -Ven.vcb.classes opt
   (-c: number of classes, -n: number of optimization runs, -p: file training, -V: file output)
4. Get vocabulary: count occurrence numbers for each word in training files. output: fr.vcb, en.vcb
5. Numberize the training files: replace word by its vcb_id
   input: fr.vcb, trn.0-0.fr, en.vcb, trn.0-0.en
   output: fr-en-int-train.snt, en-fr-int-train.snt

Step (2): run GIZA

Default: run for both directions F-E and E-F. Work with alignment factors

1. Setup giza's default options
   + po=0.999, m1=5, m2=0, m3=3, m4=3, o=giza, nodumps=1, onlyaldumps=1, nsmooth=4, model1dumpfrequency=1, model4smoothfactor=0.4, t=fr.vcb [en.vcb], s=en.vcb [fr.vcb], c=fr-en-int-train.snt [en-tr-int-train.snt], coocurrenceFile=./fr-en.cooc [./en-fr.cooc], o=./fr-en [./en-fr]
   + If hmm-align is set -> m3=0, m4=0, hmmiterations=5, hmmdumpfrequency=5, nodumps=0
   + Combine with "-giza-option" option
2. Run command
   + snt2cooc.out en.vcb fr.vcb fr-en-int-train.snt > ./fr-en.cooc
   + snt2cooc.out fr.vcb en.vcb en-fr-int-train.snt > ./en-fr.cooc
3. Run giza : *giza giza_option*
   Result : *.A3.final

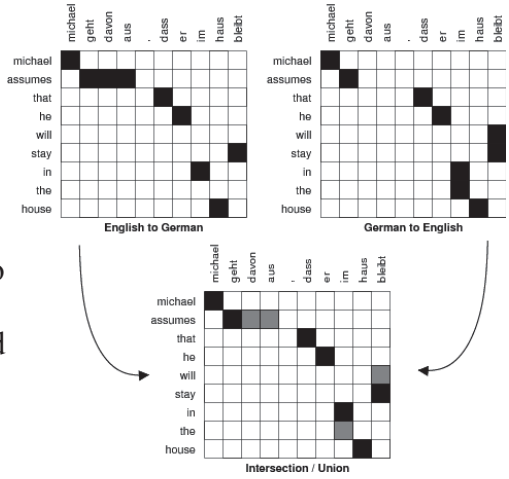
Step (3): align words

Combine word alignments from *.A3.final files. Work with alignment factors

1. Run command: *giza2bal.pl -d en-fr.A3.final -i fr-en.A3.final*
   + Read alignments from two files A3.final.
   + Write to standard output: for each alignment :
   "1
   nbr_of_EN_words EN_sentence # a[1 .. nbr_of_EN_words]"
   nbr_of_FR_words FR_sentence # b[1 .. nbr_of_FR_words]
   "
   where a[i] is the position of FR word which is aligned with this i^{th} EN word
   b[i] is the position of EN word which is aligned with this i^{th} FR word
2. Build arguments for *symal* command,
   + input: output file of command *giza2bal.pl*.
   + '-alignment' option: Default = "grow-diag-final"
   - type: "union/intersect/grow/srctotgt/tgttosrc"
   - neighbor hood: "diag"

- /final/final-and.

3. Run command *"symal -alignment="type" -diagonal="yes/no" -final="yes/no" -both="yes/no"*
   output: aligned-type-diag-final.gz



For each bi-alignment
- Read word alignments from input file
- Create matrix of word alignments
- Points to add to Destination
+ if type = intersection: just intersection points
+ if type = union: point in one of the bi-alignment
+ if type = srctotgt: points in Fr side which are aligned to EN words
+ if type = tgttosrc: points in EN side which are aligned to FR words
+ if type = grow:
- Find out the intersection points
- Create neighbor mask:
-- if diag = false : neighbor mask in co-ordinate [-1,0], [1,0], [0, 1] and [0, -1]
-- if diag = true: add co-ordinates [-1, 1] [1, 1] [-1, 1] and [1, -1]
- Destination = intersection points
- Loop: Add a point to Destination if it is the neighbor of one point in Destination + it is in the union set + it is not covered. Loop to no more point added.
- If final = yes and both = no : Add the rest points in one of two sides which are not covered.
- If final = yes and both = yes : Add the rest points in both sides which are not covered
- Print all points in Destination to file aligned-type-final-both.gz : "$0_{(Fr\_pos)}$-$0_{(EN\_pos)}$ 1-3 2-5...."

Step (4): learn lexical translation

Work with translation factors
input: trn.factor.fr, trn.factor.en, aligned-type-final-both.gz

1. Read line in Fr, En and alignment file
2. For each FR[EN] word, count number of times this FR[EN] word is aligned to an EN[FR] word, number of times this FR[EN] word is aligned to a NULL word.
3. Write to output file
   lex.factor.f2n: "EN_word FR_word (number of times this FR word is aligned to this EN word) / (number of FR word)"
   lex.factor.n2f: "FR_word EN_word (number of times this FR word is aligned to this EN word) / (number of EN word)"

Step (5): extract phrases

Work with translation factors and reordering factors if reordering_lexical=1
Execute training/phrase-extract/extract.cpp
+ input: file trn.factor.fr (rename aligned.factor.fr), trn.factor.en (rename aligned.factor.en), aligned-type-final-both.gz, max_phrase_length=7, orientation, output=extract.factor.gz
+ Read FR sentence, EN sentence and Alignment sentence.
+ For each EN sentence,
-- For each EN word sequence (1 .. 7 words) in EN sentence
- Find max, min position of FR words which are aligned with this EN sequence.
- if (max – min) < max_phrase_length
-- If all FR words from min position to max position are aligned only to this EN sequence OR to Null word --> extend FR sequence from min to max. Then extend FR sequence to left and to right by combining with un-aligned FR words.

- Write down to output file extract.factor.gz, extract.factor.inv.gz, extract.factor.o.gz

-- extract.factor.gz "FR_sequence ||| EN_sequence ||| FR_pos-EN_pos FR_pos-EN_pos..." (alignment position in aligned-type-final-both.gz – offset of this sequence)

-- extract.factor.inv.gz "EN_sequence ||| FR_sequence ||| EN_pos-FR_pos EN_pos-FR_pos ..." (alignment position in aligned-type-final-both.gz – offset of this sequence)

-- extract.factor.o.gz "FR_sequence ||| EN_sequence ||| mono/swap/other (previous) mono/swap/other (following)

(Check whether this alignment and the previous/following alignment are swapped or not ?)

Step (6): score phrases

Work with translation factors

1. Sort extracted files: extract.factor, extract.factor.inv
2. Create half table: ./phrase-table.half.f2n

   Input: extract.factor.sorted, flag_inverted=0

   + Split the input file into parts if it is longer than 10.000.000 lines. (extract.part1 --> phrase-table.half.f2n.part1,...)

   + For each part: call *training/phrase-extract/score.cpp extract.part* lex.factor.f2n phrase-table.half.f2n.part**

   - Load lexical file

   - File extract.part* is sorted, Extract all phrase pairs (FR_sequence EN_sequence Alignment) with the same FR_sequence. Loop for each FR_sequence (Process pairs with the same FR_sequence):

	-- Loop for all EN_sequences which are aligned with this FR_sequence:
extract.sorted	For each EN_sequence: If there is more than one way of Alignment =>
Fr1 ||| En1 ||| A1	mark the pair with Alignment which appears mostly.
Fr1 ||| En1 ||| A1	-- Re-loop for all EN_sequences which are aligned with this
Fr1 ||| En1 ||| A1	FR_sequence: For each EN_sequence:
Fr1 ||| En1 ||| A2	--- Calculate lex_score: lex_score = 1
Fr1 ||| En1 ||| A2	- Get Alignment which is marked for this EN_sequence. For all words in
Fr1 ||| En1 ||| A3	this EN_sequence:
Fr1 ||| En2 ||| A4	-- If this word is aligned to NULL => lex_score *= prob_in_lex_table
Fr1 ||| En2 ||| A4	(NULL)(this word)
Fr1 ||| En2 ||| A5	-- else: this_word_score += prob_in_lex_table(Fr_word)(this word) for all
Fr1 ||| En3 ||| A6	Fr_words aligned to this word;; lex_score *= this_word_score / number of
Fr1 ||| En4 ||| A7	Fr_words aligned to this word.
….	--- Calculate translation score = number of times this EN_sequence is aligned to this FR_ sequence / number of times ALL EN_sequences are aligned to this FR_ sequence
	--- Get alignment for a FR[EN] sequence: "(x,x,x) (x,x) (x,x,x) ..." :
	.. (x,x,x) alignment at each FR[EN] position
	.. x: position of aligned EN[FR] word in EN[FR] sequence.
	--- print to phrase-table.half.f2n.part*: "FR_sequence ||| EN_sequence ||| Alignment for FR sequence ||| Alignment for EN sequence ||| Translation_score Lex_score (For the direction FR--> EN) "

   + Cat all phrase-table.half.f2n.part* into phrase-table.half.f2n

3. Create half table: ./phrase-table.half.n2f

   phrase-table.half.n2f: "FR_sequence ||| EN_sequence ||| BLANK ||| BLANK ||| Translation_score Lex_score (For the direction EN--> FR)"

4. Sort and Combine two half table into phrase-table.half.*
   phrase-table: "FR_sequence ||| EN_sequence ||| Alignment for FR sequence ||| Alignment for EN sequence ||| Translation_score$_{(FR--> EN)}$ Lex_score $_{(FR--> EN)}$ Translation_score$_{(EN--> FR)}$ Lex_score $_{(EN--> FR)}$ 2.718"

Step (7): learn reordering model

Work with reordering factors

1. Reordering type = *distance, msd (orientation)/monotonicity [+ bidirectional] + f/fe*
2. If type != distance : continue.
3. Use extract.factor.o.sorted
   + Read all lines from
   file
   + Count number of times mono/swap/other appears in previous/following (number of times an alignment Fr-En is swapped/not swapped/other with the previous/following alignment)
   + Close file
4. Calculate the initial scores
   + initial_score*_previous/following*_for*_mono/swap/other* = 0.5 * [(number of times *mono/swap/other* appears in *previous/following*) + 0.1]/ (number of times ALL types mono+swap+other appear in previous + following)
5. Reopen extract.factor.o.sorted file, read lines from file
   + If type ~= "fe": count until reaching a different EN_sequence. If type ~= "f": count until reaching a different FR_sequence
   + Count number of times *mono/swap/other* appears in previous/following -> score*_mono/swap/other_previous/following* = initial_score*_previous/following*_for*_mono/swap/other* + number of times *mono/swap/other* appears in *previous/following* for these FR_EN alignments
   + Store to file . If type ~= "f" -> store "Fr_sequence |||" else If type ~= "fe" -> store "Fr_sequence ||| En_sequence |||". Then append:
   -- If type = msd: store for previous only. "score_mono_previous / total_previous score_swap_previous / total_previous score_other_previous / total_previous"
   -- If type = msd-bidirectional: store for previous + following. "score_mono_previous / total_previous score_swap_previous / total_previous score_other_previous / total_previous score_mono_following / total_ following score_swap_following / total_ following score_other_following / total_following"
   -- If type = monotonicity: "score_mono_previous / total_previous (score_swap_previous+score_other_previous) / total_previous"
   -- If type = monotonicity-bidirectional: "score_mono_previous / total_previous (score_swap_previous+score_other_previous) / total_previous score_mono_following / total_ following (score_swap_following+score_other_following) / total_following"

Step (8): learn generation model

Work with generation factors (0,1–1,2 + ...), type = *double/single*

1. factor = 0,1–1,2; factor_e_source = 0,1; factor_e = 1,2
2. Read line from trn.fr, trn.en: For each word:
   + Extract factors in factor_e_source to $source, factors in factor_e to $target
   + count number of time this "$source - $target" appears , number of time this "$source" appears, number of time this "$target" appears.
3. Print to file
   + if type = *double* then print to output "$source $target (number of time this "$source - $target" appears) / (number of time "$source" appears) (number of time this "$source - $target" appears) / (number of time "$target" appears) \n"

+ if type = *single* then print to output "$source $target (number of time this "$source - $target" appears) / (number of time "$source" appears) \n"

Step (9): create moses.ini

- [input-factors]: translation factors, [mapping] : decoding step
- [ttable-file], [lmodel-file], [ttable-limit] (def=20 0 0 ..)
- [distortion-file], w = 1 (if distance), 3 (if msd), 6 (if msd-bidirectional), 1 (if mono), 2 (if mono-bidirectional). print "factor type w reordering_table \n"
- [weight-d] = 0.6 / number_of_reordering_models. Loop for d = 1 + [Total w] for 1st factor + [Total w] for 2nd factor +...
- [weight-l] = 0.5/ number_of_language_models, [weight-t] = 0.2 , loop 5 times
- [weight-generation] = 0.5 if type=double/ 0 if type=single. Loop for number of factors
- [weight-w] = -1, [distotion-limit] = 6

File format

trn.fr, trn.en	Training file, each sentence per line. Each word is represented in factors. "$word0_{Factor0}$|$word0_{Factor1}$|$word0_{Factor2}$ $word1_{Factor0}$|$word1_{Factor1}$|$word1_{Factor2}$ …."
trn.0-0.fr, trn.0-0.en	Extract Factor0 for FR, Factor 0 for EN.
*.vcb.classes	'word1 class_id1 word2 class_id2...'
*.vcb	'1 UNK 0 vcb_id_1 word_1 freq_1 vcb_id_2 word_2 freq_2 ...'
fr-en-int-train.snt, en-fr-int-train.snt	trn.0-0.fr, replace "word" by "vcb_id"
*.A3.final	#Sentence pair ... source length ... target length ... alignment score ... *Target sentence* NULL ({...}) *word_1_of_src_sent* ({*position_in_tgt_sent, ...*}) *word_2_of_src_sent* ({*position_in_tgt_sent, ...*}) ....
aligned-type-diag-final.gz	"0-0 1-3 2-5...." (Fr_pos EN_pos) Merge 2 files A3.final, expand by 'alignment=s' option. Print out all "aligned points"
lex.factor.f2n	"EN_word FR_word (number of times this FR word is aligned to this EN word) / (number of FR word)"
lex.factor.n2f	"FR_word EN_word (number of times this FR word is aligned to this EN word) / (number of EN word)"
extract.factor.gz	"FR_sequence ||| EN_sequence ||| FR_pos-EN_pos FR_pos-EN_pos ..." (alignment position in aligned-type-final-both.gz – offset of this sequence)
extract.factor.inv.gz	"EN_sequence ||| FR_sequence ||| EN_pos-FR_pos EN_pos-FR_pos ..." (alignment position in aligned-type-final-both.gz – offset of this sequence)
extract.factor.o.gz	"FR_sequence ||| EN_sequence ||| mono/swap/other (for previous) mono/swap/other (for following) " Check whether this alignment and the previous/following alignment are swapped or not?
phrase-table	"FR_sequence ||| EN_sequence ||| Alignment for FR sequence ||| Alignment for EN sequence ||| Translation_score Lex_score (For the direction FR--> EN) Translation_score Lex_score (For the direction

	EN--> FR) 2.718”
reordering model (depend on type)	“Fr_sequence ||| En_sequence ||| score_mono_previous/total_previous score_swap_previous/total_previous score_other_previous/total_previous [score_mono_following/total_ following score_swap_following/total_ following score_other_following/total_following]”
generation model	“$source $target (number of time this “$source - $target” appears) / (number of time “$source” appears) (number of time this “$source - $target” appears) / (number of time “$target” appears) \n”