Méthodes en caractères pour le traitement automatique des langues

Etienne Denoual

Université Joseph Fourier

ETIENNE DENOUAL

Méthodes en caractères pour le traitement automatique des langues

Thèse
présentée pour obtenir le grade de
Docteur de l'université Joseph Fourier
spécialité INFORMATIQUE
défendue publiquement le 21 septembre 2006
devant le jury composé de :

M. Jean CAELEN,	président,
M. Christian BOITET,	directeur,
M. Yves LEPAGE,	co-directeur,
M. Andrei POPESCU-BELIS,	rapporteur,
M. Martin RAJMAN,	rapporteur,
M. Pierre ZWEIGENBAUM,	rapporteur,
M. Dominique VAUFREYDAZ,	examinateur.

Remerciements

Alors que prend fin l'écriture de ce manuscrit, je tiens à remercier Yves Lepage qui m'a accompagné de sa bienveillance pendant toute la durée de ces travaux, et sans qui depuis le départ rien, mais vraiment rien, n'aurait été possible. Nos conversations quotidiennes passionnées ont été autant d'occasions pour moi de bénéficier du savoir et de l'érudition d'un homme généreux, et pédagogue. Plus qu'un directeur, Yves est et sera toujours pour moi le maitre qu'il a été tout le long de l'écriture de cette thèse.

Je voudrais aussi remercier Christian Boitet, sans qui je ne serais tout simplement pas en ce moment en train d'écrire ces lignes, et qui malgré la distance et un emploi du temps chargé s'est toujours montré extrêmement disponible et enthousiaste. Le temps que nous avons passé ensemble au Japon ou en conférence m'a permis de découvrir en lui un véritable esprit critique, doublé d'une grande intégrité scientifique.

Je remercie bien sûr les différents chefs de département qui m'ont accueilli pendant mon séjour à ATR : Hiromi Nakaiwa, qui a bien voulu croire en moi alors qu'il me connaissait à peine, Yutaka Sasaki, et Eiichiro Sumita. Je remercie tout particulièrement Satoshi Nakamura, qui fût mon premier responsable à ATR dans le domaine de la reconnaissance automatique de parole, et qui fût appelé depuis à exercer de plus hautes responsabilités.

Je tiens enfin à remercier Mathieu, Marion et Benoît pour m'avoir supporté tous les jours pendant l'écriture de ce manuscrit, et pour m'avoir enseigné ce qu'est la véritable classe. J'ai bien sûr une pensée amicale pour Alexandre et Damien, mes chaleureux compagnons d'expatriation, dont l'appui indéfectible me va droit au cœur.

Je tiens enfin à remercier mes parents, qui malgré l'éloignement m'ont toujours encouragé, dans tout ce que j'ai pu entreprendre.

Résumé

Le traitement automatique des langues fondé sur les données a récemment assimilé de nombreuses techniques et perspectives héritées du domaine de la reconnaissance automatique de parole, au nombre desquelles les méthodes statistiques et les mesures en mots, qui ont conduit à un souci plus important de quantification.

L'adoption du mot comme unité atomique de traitement a cependant le désavantage que les méthodes sont difficilement transposables aux systèmes d'écriture sans séparateur orthographique, comme ceux du chinois, du japonais, du thaï, du lao, etc Des méthodes théoriquement aveugles à la nature des données se révèlent donc paradoxalement, au contact de ces langues, n'être pas multilingues.

Le présent travail se veut guidé par un souci d'universalité des méthodes et de multilinguisme et promeut donc l'utilisation en traitement automatique des langues de méthodes travaillant au niveau du « signal » de l'écrit, c'est à dire des caractères. Cette unité permet de se passer de segmentation en mots, étape actuellement incontournable pour des langues comme le chinois ou le japonais. L'avantage est que le caractère est immédiatement accessible pour n'importe quelle langue informatisée.

Dans un premier temps, nous transposons et appliquons en caractères une méthode d'évaluation objective de la traduction automatique, BLEU, dont l'application est bien établie en mots.

Les résultats encourageants obtenus sur BLEU nous permettent dans un deuxième temps d'aborder d'autres tâches de traitement des données linguistiques. Tout d'abord, le filtrage de la grammaticalité ; ensuite, la caractérisation automatique de la similarité et de l'homogénéité des ressources linguistiques. Dans toutes ces tâches, le traitement en caractères obtient des résultats acceptables, et tout à fait comparables à ceux obtenus en mots.

Dans un troisième temps, nous abordons des tâches de production de données linguistiques : le calcul analogique sur les chaines de caractères permet la production automatique de paraphrases aussi bien que la traduction automatique. Ce travail montre qu'on peut construire un système complet de traduction automatique ne nécessitant pas de segmentation, a fortiori pour traiter des langues sans séparateur orthographique.

Mots-clés

Informatique multilingue, unités de traitement, opérations sur les chaines de caractères, évaluation de la traduction automatique, filtrage de la grammaticalité, méthodes entropiques, caractérisation de ressources linguistiques, modélisation stochastique de langue, calcul analogique, production de paraphrases, traduction automatique par l'exemple.

Résumé en anglais

Over the years, data-driven natural language processing has integrated a number of techniques and viewpoints from the field of speech recognition. Statistical methods, and the use of the word unit have lead to a greater focus on quantification.

Using the word unit makes it difficult to transpose methods to languages with no orthographic separators, for instance Chinese, Japanese, Thai, Lao, etc. Methods which are theoretically operating blindly on any type of data, are paradoxically hardly applicable in a multilingual context.

The present work aims at universal and multilingual methods, and therefore promotes the use of character-based methods for natural language processing. Although the word based processing of non-segmenting languages such as Chinese or Japanese requires a segmentation step, using the character unit makes it unnecessary. The character is an immediately accessible unit in all languages in their electronic form.

We first transposed from word units to character units a well-known automatic evaluation measure for machine translation, BLEU.

Data processing and data generation are two complementary sides of natural language processing. The satisfying results obtained on BLEU lead us to consider other tasks in the field of linguistic data processing : grammatical filtering, and automatic data profiling of the similarity and homogeneity of linguistic resources. Character based processing lead to satisfying results, comparable to those obtained when using words.

Last, we considered tasks in data generation : proportional analogy on character strings allows the automatic generation of paraphrases, as well as machine translation (MT). This work shows that a complete MT system may be built which does not require any segmentation of linguistic data, and which may therefore handle non-segmenting languages with no preprocessing.

Table des matières

Liste des tableaux

Liste des figures

Note

Nous suivons autant que possible dans ce mémoire les rectifications orthographiques préconisées par le Conseil supérieur de la langue française dans sa réforme du 3 mai 1990. Il ne faudra donc pas que le lecteur s'étonne de lire par exemple *entrainer* au lieu de *entraîner*, *cout* au lieu de *coût*, ou encore *ambigüité* au lieu de *ambiguïté*.

Dans la nouvelle orthographe, les accents circonflexes sont en général omis sur les lettres *i* et *u*, à l'exception des mots qui sans accent seraient homographes, et dans le cas de la 1^{re} et 2^{e} personnes du pluriel du passé simple et de la 3^{e} personne du singulier du subjonctif imparfait. Quant au tréma, il est placé dans un mot comme *aigüe* sur la lettre *u*, qui devrait être sourde s'il était absent, et non plus sur la lettre *e* comme auparavant (ancienne orthographe : *aiguë*).

L'intégralité des rectifications orthographiques a été publiée au Journal officiel de la Republique française le 6 décembre 1990. Un vade-mecum de la nouvelle orthographe est consultable dans sa deuxième édition à l'adresse suivante :

http://www.fltr.ucl.ac.be/fltr/rom/vdm.html

Introduction

Situation et motivation

Le domaine du traitement automatique des langues a intégré de façon assez récente l'apport de méthodes du traitement du signal, inspirées de la reconnaissance automatique de parole. Or, alors qu'en reconnaissance de la parole on a coutume de travailler sur les phonèmes, en traitement automatique des langues on a tendance à travailler sur les mots. De plus en plus de tâches du traitement automatique des langues ont ainsi été traitées. Depuis environ une dizaine d'années, on essaye d'imaginer et d'implémenter des méthodes purement fondées sur les données, qui deviennent la seule connaissance accessible aux systèmes. Ces données se présentent sous la forme de ressources linguistiques le plus souvent textuelles : des corpus, des bases d'exemples. Bien que la disponibilité de ces ressources croisse sans cesse, le domaine du traitement automatique des langues bute actuellement sur des problèmes qui y sont directement, ou indirectement liés :

- premièrement, il existe des problèmes de domaine, et d'adéquation des données à celui-ci : si on cherche à traiter un grand domaine, la quantité de données linguistiques est insuffisante pour assurer une couverture satisfaisante dans le cadre de méthodes purement fondées sur les données. La phase d'entrainement de telles méthodes est très couteuse en ressources, pour des résultats pas assez spécialisés. Si l'on s'intéresse en revanche à des domaines restreints, la quantité de données linguistiques à disposition est trop importante et on ne dispose pas de méthode pour les caractériser. On sait mal identifier les redondances, ou les données qui nuisent à la performance globale du système.

- deuxièmement, il est nécessaire d'effectuer des prétraitements sur les données, qui introduisent des biais importants : en effet, les « atomes » traités sont le plus souvent des formes « orthographiques » telles qu'on les visualise intuitivement dans les langues utilisant des systèmes d'écriture avec séparateurs, comme la plupart des langues européennes[1]. Cette façon de faire est cependant difficilement transposable au cas de langues utilisant des systèmes d'écriture dénués de séparateurs, par exemple le chinois, le japonais, le coréen, le thaï, le lao, etc. Même pour les langues utilisant des systèmes d'écriture avec séparateurs, on ne peut trouver un consensus absolu en ce qui concerne la segmentation de la langue écrite.

- enfin, l'unité de traitement utilisée pour des méthodes fondées sur les données, et sur laquelle on peut effectuer des calculs en temps « raisonnable », est

[1] On peut même avancer qu'actuellement la majorité des techniques est transposée depuis la langue anglaise, langue qui focalise une bonne partie de l'attention de la recherche internationale actuelle en traitement automatique des langues, principalement pour des raisons économiques.

limitée à des phrases ou à des énoncés très courts : on se retrouve ainsi à un autre niveau de prétraitement, qui n'est pas plus approprié que celui des mots. Les performances de telles méthodes sont la plupart du temps en demi-teinte : si on prend l'exemple de la traduction automatique fondée sur l'approche statistique, ou encore des classificateurs à apprentissage supervisé tels que les machines à vecteurs-supports, les performances se situent dans les 70-90%, et sont donc plutôt élevées sans être entièrement satisfaisantes[2].

Cette thèse vise en partie à contourner ou à résoudre les problèmes méthodologiques, liés aux données utilisées et aux méthodes actuelles, en proposant d'utiliser un atome universel, immédiatement accessible, plus petit que celui actuellement utilisé : le caractère. Nous voulons ainsi montrer l'utilité de cette unité dans le domaine du traitement automatique des langues, cet atome de traitement permettant une application élégante de méthodes déjà éprouvées, dans un contexte multilingue.

Intérêt de notre étude

L'intérêt du travail exposé dans cette thèse est de montrer que, par un simple changement d'échelle du mot au caractère, on peut arriver à des performances équivalentes à celles obtenues en utilisant le mot comme unité, tout en évitant un certain nombre de problèmes liés à la définition controversée de cette unité. Nous avons pour objectif de proposer plusieurs méthodes nouvelles, simples, et efficaces, utilisant le caractère comme unité, et qui permettent de traiter élégamment des tâches classiques du traitement automatique des langues, en traitement et en production de données linguistiques : le filtrage de données en fonction de leur grammaticalité, la caractérisation de données, la production automatique de paraphrases, et la traduction automatique.

L'originalité de notre travail est aussi de montrer l'application d'un tel changement d'échelle dans le cadre de langues pour lesquelles il existe de grandes difficultés à cerner la notion de mot, comme c'est le cas en chinois et en japonais par exemple, mais aussi en lao, en thaï, etc.

Enfin, à travers l'application de résultats de recherche d'Yves Lepage[3] sur l'analogie, nous montrons l'application possible d'une méthode générale utilisant l'unité de caractère aux problèmes de la production automatique de paraphrases, et de la traduction automatique.

Plan de la thèse

Dans la partie I, nous faisons un état de l'art des différentes méthodes utilisées en traitement automatique des langues, en particulier en traduction automatique, ce qui nous permet de dégager les évolutions récentes et les nouveaux problèmes qui leur sont liés. Le développement des méthodes statistiques en traitement automatique des langues a provoqué d'importants changements dans l'attitude adoptée vis-à-vis des données linguistiques : alors qu'on se contentait de traiter des corpus,

[2] Remarque de Stephan Oepen à la table ronde de l'atelier intitulé *Linguistically interpreted corpora*, LINC-2005, Jeju, Corée.

[3] Pour une vue d'ensemble des travaux d'Yves Lepage sur l'analogie, se référer à son mémoire d'habilitation à diriger des recherches : Lepage, *De l'analogie rendant compte de la commutation en linguistique*, 2003.

l'évolution technologique liée au développement des méthodes statistiques a poussé le domaine à produire du corpus. La « tradition » linguistique est de constater des phénomènes dans des corpus, de traiter des données linguistiques en les considérant comme une donnée initiale intouchable. En revanche, l'approche statistique introduit des méthodes qui nécessitent de produire des données : par exemple, les mesures d'évaluation objective, qui nécessitent des références. Elle introduit des outils qui permettent de chiffrer des questions de domaine (par l'entropie croisée, par exemple), qu'on avait auparavant du mal à quantifier. Il s'établit une véritable dualité entre le traitement des données, et la production de données. C'est pourquoi nous articulons la suite de notre étude en deux parties, portant tout d'abord sur le traitement de données (partie II), et ensuite sur la production de données (partie III).

Nous nous intéressons à des méthodes atomiques utilisant une unité plus petite que celle du mot qui est l'unité actuellement la plus couramment utilisée : cette unité plus petite est le caractère. Nous montrons dans une étude préliminaire (partie I, chapitre 2, section 2.3) que la tâche d'évaluation automatique de la traduction automatique peut être traitée élégamment avec cette unité, et ce de façon plus universelle qu'avec le mot comme unité.

Dans la partie II, nous abordons l'utilisation de méthodes en caractères pour des tâches de traitement de données linguistiques : nous procédons en annexe C à une introduction et à des rappels en modélisation statistique de langue, qui servent dans la suite de cette partie. L'utilisation d'une modélisation statistique afin de traiter des données linguistiques, bien qu'elle soit aveugle aux données, nécessite de découper les énoncés en petites parties : afin de contourner les problèmes liés à la découpe de la langue écrite en mots, nous montrons qu'elle peut tout aussi bien être effectuée en caractères.

Nous considérons ensuite le problème de la grammaticalité en langue écrite, et montrons que des techniques simples utilisant l'unité de caractère permettent d'arriver très vite à des résultats satisfaisants. Les résultats obtenus se révèlent particulièrement utiles par la suite de notre étude, pour contraindre la production automatique de données linguistiques.

Enfin, nous proposons une approche pour la caractérisation automatique de ressources linguistiques. De telles ressources sont de plus en plus nombreuses, et volumineuses. Elles sont disponibles, entre autres via Internet, et sont souvent multilingues. Il se pose un réel problème de caractérisation de telles données : nous proposerons une méthode de profilage rapide fondée sur la modélisation statistique de langue en caractères. Elle permet de quantifier la similarité de la ressource considérée relativement à des références. Nous proposons par extension une méthode pour quantifier l'homogénéité des ressources linguistiques.

Dans la partie III, nous abordons l'utilisation de méthodes en caractères pour des tâches de production de données linguistiques. Nous présentons en annexe D une méthode nouvelle, provenant des travaux d'Yves Lepage : l'analogie sur les chaines de caractères.

Nous appliquons l'analogie à deux problèmes. Tout d'abord, nous nous intéressons à la production automatique de paraphrases. La génération proprement dite est effectuée par calcul analogique, et nous la contraignons par la méthode de détection de la grammaticalité exposée en partie II afin d'en améliorer la qualité.

Enfin, nous nous intéressons au problème de la traduction automatique : là encore, nous montrons qu'il est possible de mettre à profit l'analogie sur les chaines de caractères pour construire un système complet de traduction automatique n'utilisant

aucun prétraitement, même dans des langues dont le système d'écriture n'a pas de séparateur de mots.

Pour conclure, nous faisons un bilan général des résultats de notre recherche et des problèmes actuels du traitement automatique des langues auxquels ils répondent, et dégageons plusieurs pistes pour nos travaux futurs : la segmentation de documents en plus petites unités significatives, et une approche multiniveaux.

Partie I

État de l'art et problèmes essentiels liés à l'unité de traitement des données

Introduction

Le traitement automatique des langues est un domaine pluridisciplinaire, qui a connu un essor important depuis le début des années 1950. Le domaine a évolué en intégrant à plusieurs périodes des influences d'autres domaines voisins : par exemple celui de l'intelligence artificielle dans les années 1970, ou de la reconnaissance automatique de parole dans les années 1980. D'autre part, il a pu bénéficier des avancées, ou subir les contraintes techniques du moment, ce qui a entrainé historiquement une grande diversité des méthodes.

Dans le chapitre 1, nous effectuons une classification des méthodes utilisées en traitement automatique des langues, sans privilégier d'application particulière, en mettant l'accent sur la diversité des niveaux de représentation qu'utilisent les différentes approches. Nous faisons ainsi une distinction entre les méthodes basées sur les théories et connaissances associées, et les méthodes fondées sur les données. En particulier, nous différencions les méthodes nécessitant l'application de prétraitements sur les données de celles, minoritaires, qui peuvent être appliquées sur les données brutes, non modifiées. Nous montrons que le développement des méthodes statistiques a provoqué d'importants changements dans l'attitude adoptée vis-à-vis des données linguistiques : alors que suivant l'approche traditionnelle linguistique on traitait les données comme telles, le développement des méthodes statistiques a poussé le domaine à produire des données, établissant une dualité entre le traitement, et la production de données.

Dans le chapitre 2, nous mettons en relief les problèmes liés au choix et à l'utilisation d'une unité de travail pour le traitement automatique des langues, ainsi que des axes prometteurs pour les résoudre. En particulier, nous examinons la nécessité d'appliquer des prétraitements sur les données, et montrons qu'elle découle de contraintes sur l'unité utilisée. Les méthodes actuelles du traitement automatique des langues utilisent généralement le mot comme unité. Après avoir précisé les problèmes qu'entraine la segmentation en mots, nous proposons d'utiliser une unité plus petite que celui-ci, le caractère. Dans une étude preliminaire portant sur l'évaluation automatique de la traduction automatique, nous montrons qu'utiliser le caractère et non le mot comme unité permet de traiter le problème élégamment et de façon plus universelle.

Chapitre 1

Classification des méthodes en traitement automatique des langues

Introduction

Pour brosser un panorama actuel du traitement automatique des langues, on peut en catégoriser les différentes méthodes par les applications, c'est-à-dire par les tâches auxquelles elles sont appliquées. Si on s'intéresse aux méthodes que le traitement automatique des langues met en jeu, on aboutit à une vision orthogonale à la précédente : en effet, quelle que soit l'application concernée, les méthodes se recoupent et évoluent historiquement de façon cohérente.

Le traitement automatique des langues est un domaine pluridisciplinaire, à la frontière de la linguistique et de l'informatique, et fait collaborer de nombreux acteurs : linguistes, informaticiens, lexicographes, statisticiens et traducteurs cherchent à appliquer des traitements efficaces à la langue et aux données linguistiques, afin de traiter automatiquement ou semi-automatiquement diverses tâches. Si l'on se cantonne uniquement au traitement des données linguistiques textuelles, par opposition au traitement de la langue parlée, on peut citer un certain nombre d'applications importantes du traitement automatique des langues :

- la correction orthographique et grammaticale, qui est de nos jours présente dans la grande majorité des logiciels bureautiques d'entreprise ;
- la recherche d'informations, qui a connu un essor sans précédent à mesure que se sont développés lors des vingt dernières années les réseaux globaux de communication, mettant en relation un grand nombre d'utilisateurs ;
- le résumé automatique, la fouille de texte et les systèmes de question-réponse, qui demandent d'appréhender des aspects sémantiques plus profondément que dans la recherche d'information actuelle ;
- la génération automatique de textes, qui permet de produire automatiquement des énoncés de la langue à partir d'une certaine représentation de l'information ;
- et enfin la première application historique du traitement automatique des langues, la traduction automatique, qui est utilisée quotidiennement à des fins de veille par des millions d'utilisateurs à travers le monde.

Il serait faux de croire que la variété des applications a entrainé un cloisonnement des méthodes de traitement. Ce serait en effet prendre le problème à l'envers. La diversité des méthodes utilisées en traitement automatique des langues a deux fondements : le premier est historique. Dans un premier temps, les contraintes matérielles ont privilégié les approches fondées sur la connaissance, et c'est uniquement depuis les années 1970 que les approches fondées sur les données ont connu un essor. La deuxième raison de cette diversité des approches en traitement automatique des langues est liée aux différents niveaux de représentation de l'information linguistique, et donc à la nature des traitements qu'il faudra appliquer aux données de la langue afin de passer d'une représentation à une autre. Au delà de l'aspect historique, la diversité des méthodes correspond ainsi à la diversité des niveaux de représentation qu'elles traitent. Cette diversité des approches comporte à la fois des aspects scientifiques et idéologiques importants. Cet état de fait est illustré par le débat qui oppose depuis le début des années 1990 partisans de l'approche linguistique « traditionnelle » et ceux de l'approche statistique, qui s'inspire du domaine de la reconnaissance de parole[1]. Nous constatons enfin que la tendance actuelle est aux méthodes hybrides, brouillant les frontières entre ces deux approches a priori antagonistes.

Dans la suite de cette introduction, nous détaillons les diverses méthodes utilisées en traitement automatique des langues, en les rattachant à leur développement historique respectif, sans privilégier une application en particulier. Comme on l'a mentionné plus haut, il est intéressant de différencier les méthodes fondées sur les théories et la connaissance, de celles fondées sur les données. Nous verrons que, bien qu'elles soient en théorie aveugles à la nature des données, des méthodes telles que celles qui sont fondées sur l'approche statistique nécessitent la plupart du temps la mise en œuvre de prétraitements spécifiques, qui vont à l'encontre du caractère universel qu'on leur prête. Nous montrons enfin que les méthodes tendant à utiliser peu ou pas de prétraitement sont récentes, et peu nombreuses.

1.1 Méthodes fondées sur des théories et des connaissances associées

Les méthodes fondées sur des théories linguistiques et les connaissances associées sont historiquement les premières : en effet, on avait dans les années 1950 peu de données sous forme informatique, peu de place pour les stocker, et enfin peu de puissance de calcul pour les traiter[2]. Il était donc logique de traiter des problèmes de langue par les connaissances théoriques, ou construites par des linguistes à partir d'ensembles plus limités de données. Dans les sections suivantes, nous détaillons les méthodes par règles, et les méthodes fondées sur la connaissance, venues du domaine de l'intelligence artificielle.

[1] Le célèbre débat *Rationalisme contre Empirisme* qui avait animé la conférence TMI (*Technological and Methodological Issues in Machine Translation*) en 1992 est toujours bien vivant dans les mémoires.

[2] On notera que ce manque de données est encore d'actualité pour beaucoup de langues : dans Kathol *et al.*, *Speech translation for low-resource languages: The case of pashto*, 2005, par exemple, devant le manque de données disponibles lors de l'élaboration d'un système de traduction automatique anglais-pachtoune, les concepteurs font appel à des linguistes et des locuteurs du pachtoune.

1.1.1 Méthodes par règles et approches linguistiques

Alors que la traduction automatique est historiquement la première application envisagée en traitement automatique des langues, la linguistique informatique est réellement apparue avec les premiers analyseurs syntaxiques au début des années 1950. Dès cette époque, Bar-Hillel[3] proposa de déterminer automatiquement la structure d'une phrase, grace à une représentation formelle de la syntaxe de la langue en machine. On notera le développement des méthodes par automates : Yngve[4] proposa une méthode par automate à états finis, traitant les phrases sous une forme réduite, sous forme de suite de classes de mots, comprenant l'information grammaticale et syntaxique. Il mit au point le premier langage spécialisé, dédié au traitement des chaines de caractères et fondé sur la reconnaissance de formes, COMIT.

Les modèles à grammaires servirent eux aussi à l'élaboration d'analyseurs syntaxiques : ainsi l'analyseur de Hays[5] fondé sur la grammaire de dépendance de Tesnière[6], et la grammaire stratificationnelle de Lamb[7]. Par la suite, les années 1980 virent le développement de plusieurs formalismes déclaratifs, tels que les DCG[8] (*Definite Clause Grammars*, pour grammaires de clauses définies), les FUG[9] (*Functional Unification Grammars*, pour grammaires fonctionnelles d'unification), ou encore le langage spécialisé PATR-II[10] (*Parse and Translate*, soit analyse et traduction), ainsi que la convergence avec des formalismes linguistiquement motivés, tels que les LFG[11] (*Lexical Functional Grammars*, pour grammaires lexico-fonctionnelles), les grammaires GPSG[12] (*Generalized Phrase Structure Grammars*, pour grammaires syntagmatiques généralisées), puis HPSG (*Head-driven Phrase Structure Grammars*) et les TAG[13] (*Tree Adjunct Grammars*, pour grammaires d'arbres adjoints).

Dans une approche par règles, l'information est représentée sous la forme de règles, décrites manuellement par des linguistes, ou apprises semi-automatiquement. Les connaissances lexicales et grammaticales sont généralement séparées[14]. Comme on l'a mentionné plus haut, l'avantage premier d'une représentation explicite des connaissances est son utilisabilité avec peu de ressources : ainsi, là où un système par règles utilisera un dictionnaire de 400 000 entrées, soit l'équivalent de 16 000 pages de texte, un système fondé sur l'approche statistique telle qu'exposée dans la section 1.2.3 nécessitera couramment entre 50 et 200 millions de mots, soit entre 200 000 et 800 000 pages de texte, il faudra donc 50 fois plus de données à un tel système pour être opérationnel. Un autre avantage des méthodes par règles, en plus de leur élégance formelle et de leur incrémentalité (rajouter des règles plus précises améliore les performances jusquà un certain point) est la qualité des résultats obtenus sur des domaines restreints.

[3] Bar-Hillel, *The present state of research on mechanical translation*, 1953.
[4] Yngve, *Syntax and the problem of multiple meaning*, 1955.
[5] Hays, *Automatic language-data processing*, 1962.
[6] Tesnière, *Eléments de syntaxe structurale*, 1959.
[7] Lamb, *On the mechanization of syntactic analysis*, 1962.
[8] Pereira & Warren, *Definite clause grammars for language analysis*, 1980.
[9] Kay, *Functional grammar*, 1979.
[10] Shieber, *Criteria for designing computer facilities for linguistic analysis*, 1985.
[11] Kaplan & Bresnan, *Lexical-functional grammar: a formal system for grammatical representation*, 1982.
[12] Gazdar & Pullum, *Generalized phrase structure grammar: A theoretical synopsis*, 1982.
[13] Joshi *et al.*, *Tree adjunct grammar*, 1975.
[14] Ce n'est pas le cas dans les formalismes dits *lexicalisés* comme par exemple les grammaires lexico-fonctionnelles.

Un désavantage des méthodes par règles est en revanche leur manque de souplesse lors d'une transposition de domaine : au delà des règles structurales de base d'une langue, l'élaboration de règles spécifiques à un domaine est nécessaire pour assurer une couverture satisfaisante. Àu partir d'une certaine taille critique, se posent des problèmes d'extensions des grammaires : les ajouts s'accompagnent d'une perte en terme de robustesse, due aux contradictions créées entre les connaissances initiales et les ajouts.

Le désavantage majeur des méthodes par règles réside surtout dans le temps nécessaire à l'élaboration manuelle, à l'écriture et à la vérification de ces règles. La constitution d'une grammaire est une opération longue, complexe, et fastidieuse. La plupart des systèmes de traitement automatique des langues fondés sur les règles et atteignant des performances respectables avec une bonne couverture sont le résultat de l'accumulation d'années de travail linguistique effectué à la main.

Même si ce travail parait extrêmement couteux, il est à relativiser. Prenons l'exemple d'un système de traduction automatique fondé sur l'approche statistique, et ayant besoin au bas mot de 200 000 pages de texte pour fonctionner : la création d'un tel corpus nécessiterait environ 250 000 heures de travail, soit plus de 150 hommes-années. À titre de comparaison, c'est quatre fois plus de travail que n'en nécessite la création d'un nouveau couple de langue dans un système de traduction automatique par règles comme Reverso de Softissimo[15].

1.1.2 Méthodes fondées sur la connaissance

Le domaine du traitement automatique des langues a reposé jusqu'à la fin des années 1970 sur l'intelligence artificielle et les systèmes experts. Les méthodes du traitement automatique des langues fondées sur la connaissance ont en effet subi de nombreux apports du domaine de l'intelligence artificielle[16]. Le fait que la machine doive traiter la langue implique qu'elle en appréhende dans une certaine mesure l'aspect sémantique. La recherche en traitement automatique des langues fit ce constat dès le début des années 1960, mais il fallut attendre une dizaine d'années pour que des systèmes se dotent de véritables composantes sémantiques. L'intelligence artificielle, si elle est apparue comme un domaine de recherche à part entière à la fin des années 1940, peina encore dans les années 1960 à dépasser le stade des systèmes de résolution ad hoc de problèmes particuliers. C'est dans les années 1970 qu'elle apportera au traitement automatique des langues des modèles de connaissance et de représentation du savoir, qui amorceront dans le domaine le développement des aspects sémantiques de la représentation textuelle.

De telles représentations posent un problème de modélisation des connaissances, et nécessitent une définition des symboles par rapport à des ontologies, si possible explicites et informatisées (mais en pratique presque toujours implicites). Pour les applications où l'on ne peut appliquer une telle modélisation des connaissances, on a recours à des heuristiques, qui sont une forme de représentation de la connaissance acquise par un expert au bout de nombreuses expériences. Le résultat de ces expériences est interprété puis généralisé sous la forme de règles heuristiques, qui s'appliquent ensuite aux données du problème à traiter.

On peut citer l'exemple d'une tentative de description symbolique exhaustive, avec le projet CYC. Ce projet, amorcé en 1984, construit une base de connaissances

[15] Voir http://www.softissimo.com .

[16] Sabah, *L'intelligence artificielle et le langage*, 1989.

qui comporte plusieurs centaines de milliers d'éléments factuels et de règles, sur lesquels on peut appliquer plusieurs moteurs d'inférence. Dans le domaine de la traduction automatique, on peut citer le travail mené au sein de l'équipe GETA[17] pour intégrer les systèmes experts et les techniques issues de l'intelligence artificielle aux moteurs de traduction, puis le système crée par l'université Carnegie Mellon pour Caterpillar (Kant, puis Catalyst).

1.2 Méthodes fondées sur les données, plus ou moins prétraitées

Les méthodes fondées sur les données ont connu un essor plus récent que celles fondées sur les théories linguistiques et les connaissances, en premier lieu pour des raisons d'ordre technique. Bien que des méthodes empiriques aient été envisagées et étudiées dès le début des années 1950[18], on n'avait en réalité avant les années 1970 ni les moyens de stocker les quantités massives de données requises dans une telle approche ni les moyens de les traiter, ces traitements prenaient plus de temps qu'une approche par la connaissance. L'utilisation aux USA de méthodes empiriques comme de méthodes par règles très primitives pour la traduction automatique produisit dans le contexte de la guerre froide des années 1960 des résultats décevants, et dans un contexte de réorientation massive vers la linguistique formelle et l'intelligence artificielle, le domaine connut aux USA près d'une vingtaine d'années de passage à vide[19], recentrant ainsi les études sur la traduction semi-automatique et l'analyse syntaxique. Ailleurs, différents systèmes continuaient à se développer, par exemple le système TAUM[20] qui nait dès 1967, ou encore le SFB-100 de Sarrebruck vers 1968, ou encore le système du CETA[21] vers la même période.

Le domaine a bénéficié d'apports importants depuis le domaine de la reconnaissance automatique de parole, et a dès lors connu un véritable renouveau, pour devenir un courant majeur de la fin des années 1990. Dans les sections suivantes, nous détaillons les approches par modèles de Markov, les méthodes de classification, et les méthodes statistiques.

1.2.1 Modèles de Markov (HMM)

Les méthodes par modèles de Markov cachés, souvent abrégés en HMM pour *Hidden Markov Models*, ont d'abord connu un vif succès dans le domaine de la reconnaissance automatique de parole. L'information linguistique y est représentée entièrement par les données, sous leur forme compilée en modèles de Markov. Les méthodes fondées sur les données ont ainsi besoin de quantités importantes de données d'apprentissage sous la forme de grands corpus. Par rapport aux méthodes par règles, l'intérêt des méthodes par modèles de Markov est de nécessiter moins de travail

[17] Boitet & Gerber, *Expert systems and other new techniques in MT systems*, 1984.

[18] Voir par exemple Kaplan, *An experimental study of ambiguity in context*, 1950. pour la résolution d'ambigüités par des méthodes statistiques.

[19] Voir le célèbre rapport : Bar-Hillel, *The present status of automatic translation of languages*, 1960, ainsi que ALPAC, *Language and machines. Computers in translation and linguistics*, 1966. Ces deux rapports furent décriés, principalement pour ALPAC dans un contre rapport :Pankowicz, *Commentary on ALPAC report*, 1966. Bar-Hillel produisit en 1972 un autre rapport, positif.

[20] Chandioux, *METEO: un systeme opérationnel pour la traduction automatique des bulletins météorologiques*, 1976

[21] Vauquois, *La traduction automatique à Grenoble*, 1975

humain : la compilation des données est effectuée automatiquement. En revanche, selon la taille des données utilisées, cette compilation, accompagnée des inévitables optimisations, peut s'avérer tout aussi couteuse en terme de temps.

Le développement des méthodes fondées sur les modèles de Markov est ainsi historiquement lié à la construction des premiers grands corpus en langue anglaise, financée en vue de l'évaluation par l'agence américaine DARPA[22]. Les systèmes développés dans les années 1971-1976 dans le cadre du projet ARPA-SUR[23] représentent une avancée décisive dans le domaine. Le système HARPY[24] de CMU compile les connaissances phonétiques et lexico-syntaxiques dans un réseau totalisant environ 15 000 états. Atteignant une performance de 95% sur un lexique d'un millier de mots, le système afffiche les meilleures performances des systèmes de l'époque, et incite une grande partie des recherches en reconnaissance automatique de parole à se concentrer sur les réseaux et modèles de Markov. Le relatif succès de ces méthodes va par la suite influencer les recherches en traitement automatique des langues.

Des approches mixtes ont ensuite été tentées, comme par exemple l'utilisation de modèles de Markov dans une analyse gauche-droite (HMM-LR, LR étant l'abbréviation de *Left to Right scanning, rightmost derivation in reverse*) décrite par Kita[25], mais se sont révélées moins convaincantes. Les méthodes utilisant des modèles de Markov présentent des gains en performance considérables sur les approches par règles. On peut citer le cas du système BBN HWIM (Bolt Beranek and Newman Inc. :*Hear What I Mean*), qui utilise en plus de réseaux de types markoviens des contraintes grammaticales réalisées par des ATN, pour *Augmented Transition Network Grammar*[26], ou encore celui du système HearSay[27] de l'université Carnegie Mellon, composé de sources de connaissance modulaires.

Des systèmes plus récents, comme SPHINX en 1985, utilisent les modèles de Markov avec une approche stochastique[28]. SPHINX était considéré comme le meilleur système de reconnaissance automatique de parole continue conçu à l'époque. Actuellement, la plupart des produits commerciaux de reconnaissance automatique de parole, tels que ceux d'IBM, DragonSystems, France Telecom, Philips ou Microsoft, utilisent toujours des méthodes fondées sur les modèles de Markov.

1.2.2 Méthodes de classification

Les méthodes de classification appliquées au traitement automatique des langues sont en quelque sorte la résultante de l'application de techniques d'intelligence artificielle dans une approche fondée sur les données. Dans celles-ci, les tâches du traitement automatique des langues sont vues comme de simples tâches d'apprentissage automatique[29]. Les réseaux neuronaux sont des systèmes d'apprentissage discriminants, utilisés en reconnaissance automatique de parole depuis 1985. Leur application au traitement automatique des langues comme méthode de classification, dans

[22] Pallett *et al.*, *DARPA ATIS test results*, 1990.

[23] *Advanced Research Projects Agency of the Department of Defense - Speech Understanding and Recognition*, voir Klatt, *Review of the ARPA speech understanding project*, 1977.

[24] Lowerre, *The HARPY speech recognition system*, 1976.

[25] Kita *et al.*, *HMM continuous speech recognition using predictive LR parsing*, 1989.

[26] Woods, *Transition network grammars for natural language analysis*, 1970.

[27] Lesser & Erman, *A retrospective view of the HEARSAY-II architecture*, 1977.

[28] Voir à ce sujet Rabiner, *A tutorial on hidden Markov models and selected applications in speech recognition*, 1989.

[29] En anglais, *machine learning*.

des applications d'étiquetage, ou d'apprentissage de gaussiennes pour les modèles de Markov, produit cependant des résultats décevants. De la même façon, l'application d'algorithmes génétiques en traitement automatique des langues n'a pas produit de résultats réellement convaincants. On peut noter cependant une application croissante des machines à vecteurs-supports (SVM pour *Support Vector Machines*) depuis la fin des années 1990, ainsi que des approches bayésiennes.

Là encore, l'information est entièrement contenue dans les données. Cependant, il est nécessaire d'extraire des données des traits caractéristiques, qui servent à la classification. Un certain nombre de prétraitements, et de choix de représentation concernant la constitution des différents traits est donc requise. L'avantage des méthodes de classification peu ou pas supervisées en traitement automatique des langues, est de proposer des méthodes où la connaissance est représentée uniquement dans les vecteurs de traits constitués à partir des données, et à l'application rapide pour peu que l'extraction des différents traits soit automatisée.

La mise en place d'un classificateur binaire comme une machine à vecteurs-supports nécessite donc la sélection manuelle d'exemples de l'une ou l'autre classe, ainsi que des traits à considérer dans la classification. Cette mise en œuvre est un processus couteux. Tout le reste est effectué automatiquement. Un classificateur peut donc être construit rapidement à partir de données nouvelles.

Le désavantage est que les performances de systèmes fondés sur de telles méthodes sont généralement bien en deçà de celles obtenues avec des méthodes par règles, plus efficaces mais aussi plus couteuses à élaborer.

Pour illustrer cela de façon plus précise, nous présentons dans la partie II, chapitre 1, l'application des machines à vecteurs-supports comme méthode de classification dans une tâche de filtrage de la grammaticalité d'énoncés produits automatiquement.

1.2.3 Méthodes statistiques

Les méthodes statistiques furent connues dès les années 1950 et réellement utilisées en recherche d'information et en traitement automatique des langues en analyse syntaxique[30] dans les années 1970. Cependant, bien que ces techniques fussent alors utilisées en reconnaissance[31] et en compréhension[32] de parole, elles ne s'imposèrent réellement qu'au cours des années 1980. Les méthodes statistiques faisant un usage intensif de grandes quantités de données d'apprentissage, leur développement a là encore été lié historiquement à la construction des premiers grands corpus de données linguistiques.

Le groupe d'IBM de Yorktown-Heights réalisa en 1978 le langage PLNLP, dont l'application à la langue anglaise sous le nom de PEG (pour *PLNLP English Grammar*) connut un grand succès[33]. Bien que l'on ne puisse parler d'approche statistique à proprement parler, les statistiques servaient alors à ajuster les poids dans les grammaires hors-contexte.

Introduite en traduction automatique par Brown[34] en 1990, l'approche statistique est devenue en une dizaine d'années une des approches dominantes du do-

[30] Andreewsky *et al.*, *Computational learning of semantic lexical relations for the generation and automatical analysis of content*, 1977.
[31] Jelinek, *Continuous speech recognition by statistical methods*, 1976.
[32] Baker, *Stochastic modeling for automatic speech understanding*, .
[33] Voir Langendoen & Barnet, *Plnlp: A linguist's introduction*, 1986.
[34] Brown *et al.*, *A statistical approach to machine translation*, 1990.

maine. La particularité des méthodes statistiques est de baser les calculs sur les N-grammes, c'est-à-dire sur l'utilisation de statistiques sur les séquences de mots pour prédire celui qui suit à l'aide de la construction de modèles probabilistes. L'utilisation des modèles N-grammes ainsi que de la théorie des probabilités différencie ces méthodes de celles fondées purement sur les modèles de Markov.

Dans les méthodes statistiques, l'information est tirée entièrement des données, et représentée sous la forme de modèles compilés à partir des donnés. Cependant, tout comme les méthodes fondées sur les modèles de Markov, *les méthodes du traitement automatique des langues fondées sur une approche statistique utilisent fondamentalement des modèles de génération pour faire de l'analyse.*

Une caractéristique commune des méthodes statistiques est qu'elles demandent généralement pour être efficaces des prétraitements intensifs sur une quantité de données la plus importante possible. L'avantage, et d'une certaine façon le désavantage majeur des méthodes statistiques, est qu'elles sont aveugles à la spécificité des données linguistiques, qui sont traitées comme n'importe quel autre type de données informatiques. L'intégralité de la phase d'apprentissage peut ainsi être réalisée de façon automatique, non supervisée, et les sytèmes font preuve d'une couverture généralement assez large dépendant essentiellement de la nature des données utilisées. La transposition à un autre domaine est plus aisée que dans le cas d'un système utilisant une approche par règles, puisqu'il suffit de disposer des données appropriées. Cependant, l'application de techniques en N-grammes présuppose la découpe en unités de traitement : cette découpe requiert elle-même un certain nombre d'hypothèses et de prétraitements, qui, par contre, doivent en général être linguistiquement motivés.

Nous étudierons dans cette thèse les problèmes pratiques et théoriques que crée le traitement en mots, et proposerons une alternative à celle-ci. Les autres désavantages des méthodes statistiques sont d'une part leurs grosses exigences en termes de calcul, et d'autre part leurs performances généralement peu satisfaisantes dans des domaines ou des tâches particulières, face à des systèmes spécialisés fondés sur les règles. Cela explique par exemple le fait que l'on ne trouve aucun produit commercial actuel faisant usage de l'approche statistique[35] dans le secteur de la traduction automatique.

1.2.4 Méthodes avec peu ou pas de prétraitement

Il existe des méthodes du traitement automatique des langues fondées sur les données et qui n'utilisent pas ou peu de prétraitements. L'information est ainsi représentée uniquement dans les données, sur lesquelles n'est effectuée aucune compilation ni extraction. La totalité ou presque des calculs est ainsi effectuée à la volée, sans qu'interviennent de connaissances extérieures aux données, ni d'éléments linguistiques.

On peut citer l'application intéressante de la technique d'apprentissage « paresseux »[36] (*lazy learning*) pour l'étiquetage automatique des catégories morphosyntaxiques. Le principe est de baser les calculs sur les occurrences observées en temps réel ainsi que sur celles observées et emmagasinées par le passé, plutôt que de compiler des règles à l'avance uniquement à partir des observations mémorisées. Le

[35] Si l'on fait exception du système commercialisé par *Language Weaver*, voir http://www.languageweaver.com

[36] Aha, *Lazy learning: Special issue editorial*, 1997.

calcul est retardé jusqu'au moment où l'observation est faite. Dans le domaine des mémoires de traduction, Planas[37] propose une méthode avec peu de prétraitements des données d'origine : on se contente de calculer à l'avance des index, puis la sous-mémoire sélectionnée est traitée à la volée. Och[38] propose lui une approche de la traduction automatique résolument statistique, mais où tous les calculs de modélisation de langage sont lancés à la volée, en fonction de la phrase à traduire et au moment où elle est soumise au système : la méthode, qui a donné de bons résultats pratiques lors de la campagne d'évaluation NIST2005[39] est cependant si gourmande en ressources qu'une phrase met en pratique plusieurs heures à être traduite[40], beaucoup plus donc que le temps nécessaire à un traducteur humain pour effectuer la même tâche, et naturellement avec une moins bonne qualité.

Nous développons à notre tour dans cette étude d'une part l'idée d'absence de prétraitement dans une approche fondée sur les données, en contournant le problème de découpe en mots par un travail au niveau du caractère, et d'autre part en utilisant une méthode universelle, l'analogie, utilisée au niveau des chaines de caractères.

1.3 Méthodes hybrides

La tendance actuelle est à la convergence entre les méthodes fondées uniquement sur les données et les méthodes fondées sur les théories linguistiques : on assiste ainsi à une explosion des méthodes hybrides. Hybrides, ces méthodes peuvent en fait l'être de deux manières : au niveau du calcul, ou au niveau des représentations.

À l'origine, les méthodes statistiques ont pour vocation de n'utiliser aucune représentation intermédiaire, pourtant dans la pratique beaucoup de systèmes en ont réintroduit : on peut citer par exemple le système MASTOR d'IBM[41], qui réalise une analyse sémantique et syntaxique des énoncés avant l'application de méthodes statistiques, celui de Ney[42] pour le projet Verbmobil, qui apprend des grammaires de surface, ou encore celui de l'équipe de l'ISI[43], qui introduit une approche statistique fondée sur la syntaxe, et l'alignement automatique d'arbres d'analyse dans plusieurs langues. On peut aussi concilier une approche linguistique avec une méthode non statistique fondée sur les données : le système de traduction par l'exemple de Microsoft Research[44] extrait des règles à partir des données, qui permettent de s'abstraire en partie de la structure spécifique des langues à traduire. Il est aussi possible d'apprendre des automates d'arbres[45] automatiquement dans le cadre d'une approche statistique.

---

[37] Planas, *TELA : Structure et algorithmes pour la traduction fondée sur la mémoire*, 1998.

[38] Och, *Statistical machine translation: Foundations and recent advances*, 2005.

[39] Przybocki, *The 2005 NIST machine translation evaluation plan (MT-05)*, 2004.

[40] De l'aveu même de Franz Joseph Och, la méthode est extrèmement gourmande : disposant d'une ferme de 1 000 ordinateurs, il fait traduire chacune des 1 000 phrases du jeu de test par l'un des ordinateurs.

[41] Voir http://domino.watson.ibm.com/comm/research.nsf/pages/r.uit.innovation.html

[42] Ney *et al.*, *Statistical translation of spoken dialogues in the Verbmobil system*, 2000.

[43] Germann *et al.*, *Fast decoding and optimal decoding for machine translation*, 2003.

[44] Brockett *et al.*, *English-Japanese example-based machine translation using abstract linguistic representations*, 2002.

[45] Thatcher & Wright, *Generalized finite automata theory with an application to a decision problem of second-order logic*, 1968.

Conclusion

Nous avons brossé dans cette introduction un panorama des différentes méthodes employées en traitement automatique des langues. Nous avons vu que la raison de la diversité des méthodes était d'une part historique, et d'autre part théorique, car liée au niveau de représentation linguistique considéré. Dans l'approche fondée sur les règles, la connaissance linguistique est explicitée sous la forme de grammaires ou d'automates et de dictionnaires, alors qu'à l'autre extrême, dans l'approche fondée sur les données, elle est extraite des données via une modélisation statistique, indépendamment de tout phénomène linguistique proprement dit, les données pouvant être aussi bien numériques que linguistiques. C'est principalement sur ce point que s'affrontent les tenants des approches symboliques, et ceux des approches fondées sur les données. Un tel débat est particulièrement actuel dans le domaine de la traduction automatique, qui après avoir été longtemps dominé par les approches par règles, a été le témoin d'importants tranferts depuis le domaine de la reconnaissance automatique de parole dans les années 1980.

Les méthodes pouvant être appliquées sur les données brutes, sans nécessité de prétraitement, sont minoritaires. Les méthodes fondées sur les données, comme l'approche statistique, travaillent en effet sur des unités, en général les mots, qu'il faut extraire des données. Cette extraction nécessite des connaissances pour être efficace, et va à l'encontre du caractère « universel » que visent les méthodes fondées sur les données. Dans le chapitre suivant, nous mettons en relief les différents problèmes liés à cette nécessité de découper les données linguistiques en unités de traitement, et proposons d'y remédier en utilisant un atome plus petit et immédiatement accessible : le caractère.

Chapitre 2

Mise en relief des problèmes actuels et des axes prometteurs

Introduction

Comme nous l'avons mentionné dans l'introduction, le traitement automatique des langues fondé sur les données bute actuellement sur un certain nombre de problèmes difficiles. Ces problèmes sont liés aux données utilisées, de façon à la fois quantitative et qualitative.

Il y a tout d'abord un problème de pertinence vis-à-vis du domaine que l'application doit traiter. Une application de traitement automatique des langues est conçue pour opérer dans des domaines où la langue se comporte de manière plus ou moins libre. Des applications peuvent avoir pour ambition de couvrir des domaines vastes comme des domaines restreints (on dit alors qu'on évolue dans un ou des sous-langages au sens de Kittredge[1]). On rencontre alors deux problèmes majeurs : soit on vise à couvrir un grand domaine de la langue, et la quantité de données linguistiques disponibles est insuffisante pour assurer au système de traitement automatique des langues une couverture nécessaire (problème bien connu de « rareté des données ») ; soit on vise à couvrir des domaines restreints de la langue (donc le système évolue dans un sous-language), et la quantité de données linguistiques nécessaires est trop importante, les données sont trop générales pour obtenir des performances optimales dans de tels domaines, car on ne dispose pas de méthode pour les caractériser ou les trier.

Le deuxième problème que l'on rencontre est fondamental car il est méthodologique : lorsqu'on traite un document, un prétraitement est nécessaire afin de le découper en unités ou atomes de base[2]. Habituellement, dans l'approche du traitement automatique des langues fondée sur les données, le texte est découpé en ce qu'on a coutume d'appeler des mots. Le traitement réservé à la ponctuation varie, mais si le plus souvent la phrase est considérée comme une entité bien délimitée, il est fait assez peu de cas de sa ponctuation interne[3]. Cette approche est à notre sens malheureuse, car la ponctuation constitue une information linguistique non

[1] Kittredge & Lehrberger, *Sublanguage. Studies of language in restricted semantic domains*, 1982, c'est-à-dire des domaines limités en termes de variations lexicales, syntaxiques, et sémantiques.

[2] Une *tokenization*, en anglais.

[3] Les campagnes d'évaluation de systèmes de traduction automatique IWSLT-2004 et 2005 exigeaient des phrases sans ponctuation en sortie des systèmes !

négligeable sur la structure du texte.

D'autre part, la question de la division en mots est sujette à débat : le mot est une unité significative de la grammaire traditionnelle, empiriquement liée à la forme écrite de la langue considérée. Sur le plan linguistique, on peut tout à fait penser qu'il n'existe pas un mais des mots à différents niveaux de représentation : le mot « oral » phonétique (ou groupe accentuel) correspond mal au critère de séparabilité fonctionnelle, et aux critères de délimitation intonative ; le mot « écrit » graphique, s'il peut paraitre plus accessible dans des langues comme le français ou l'anglais par exemple, ne l'est pas dans les langues dépourvues d'espace comme le chinois ou le japonais ou dans d'autres dépourvues de ponctuation.

Si le mot orthographique peut représenter à première vue la brique de base qui relie le signifiant au signifié, l'atome d'une analyse en constituants immédiats[4], il est en réalité impossible à définir de façon satisfaisante et générale. Tout au plus pourra-t-on tenter de le faire dans le cadre d'une langue en particulier.

C'est sur un tel problème que nous nous proposons de diriger notre attention tout au long de cette étude.

2.1 Les problèmes inhérents à une segmentation en mots

Les linguistes informaticiens et chercheurs en traitement automatique des langues, qui utilisent de manière quotidienne l'ordinateur pour étudier ou traiter la langue, font constamment face à un problème : le plus souvent, ils étudient et traitent uniquement la forme écrite de la langue, alors qu'elle est orale par essence.

Bien que le système d'écriture d'une langue ne soit pas lié à la structure interne de la langue considérée, la forme écrite de la langue ne peut pour autant être négligée. Elle est souvent la seule donnée à notre disposition, par exemple sous la forme de données « moissonnées » sur le web, ou tirées de grandes bases de données linguistiques telles que celles distribuées par FRANTEXT[5], le LDC[6] ou l'ELDA[7]. En traitement automatique des langues fondé sur les données, on travaille habituellement au niveau de ce qu'on appelle le mot, conformément à l'idée intuitive que l'on en a dans les langues à segmentation claire, c'est-à-dire dont le système d'écriture inclut des séparateurs spécifiques. Ces séparateurs segmentent le document en de plus petites parties, en délimitant des chaines de lettres ou d'idéogrammes. À l'opposé, dans les langues dont le système d'écriture n'admet pas de séparateur, une phrase, un paragraphe ou des documents entiers peuvent être écrits en une séquence continue de lettres ou d'idéogrammes. Cela pose donc un problème de méthode.

Illustrons ce que nous venons de dire. Le français, l'anglais et l'allemand sont des langues dont le système d'écriture admet des séparateurs : l'espace et la ponctuation. Par exemple, dans la phrase suivante en français :

Le chat mange une souris.

on peut compter 5 mots. Dans son équivalent en anglais :

The cat eats a mouse.

[4] Les grammaires hors-contexte sont habituellement fondées sur des catégories affectées aux mots, le mot est donc une unité de départ.

[5] Voir http://www.frantext.fr .

[6] Voir http://www.ldc.upenn.edu .

[7] Voir http://www.elda.org .

on peut aussi compter 5 mots. Enfin, dans son équivalent en allemand :

Die Katze ißt eine Maus.

on peut encore compter 5 mots. Cependant, dans son équivalent en japonais :

猫が鼠を食べている。
/nekoganezumiwotabeteiru./

on n'est pas en mesure de dénombrer visuellement plusieurs mots. Dans ce cas précis, on se heurte au fait que dans le système d'écriture de la langue japonaise, il n'y a pas de frontière entre les mots induite par des séparateurs clairs[8] (dans le cas des langues européennes citées en exemple, l'espace et la ponctuation). Ce phénomène est loin d'être exceptionnel : la majorité des langues orientales telles que le chinois, le thaï, le lao s'écrivent sans espace, tout comme par ailleurs le latin et le grec ancien à une certaine époque en Europe. Dans l'exemple qui nous intéresse, on pourrait toutefois contourner un tel problème en proposant une segmentation en mots selon des règles définies à l'avance : par exemple,

猫 ‖ が ‖ 鼠 ‖ を ‖ 食べて ‖ いる ‖。
/neko‖ga‖nezumi‖wo‖tabete‖iru‖./

Ou alors, on pourrait aussi accepter :

猫 ‖ が ‖ 鼠 ‖ を ‖ 食べている ‖。
/neko‖ga‖nezumi‖wo‖tabeteiru‖./

Ou encore, on pourrait choisir une segmentation en bunsetsus :

猫が ‖ 鼠を ‖ 食べている ‖。
/nekoga‖nezumiwo‖tabeteiru‖./

que trouveraient plus cohérente les personnes ayant des notions de grammaire japonaise.

On peut tirer trois enseignements de l'exemple cité ci-dessus :

- il n'existe pas une mais plusieurs façons apparemment correctes, et justifiées, de découper la langue en mots, dans le cas où le système d'écriture de la langue considérée n'admet pas de séparateur. Il existe donc un désaccord au niveau de la découpe ;
- d'autres unités que le mot (telles que le bunsetsu dans l'exemple en langue japonaise) semblent tout aussi appropriées que le mot pour découper la langue, si ce n'est plus ;
- une pratique courante en traitement automatique des langues, est de transposer des approches depuis des langues dont le système d'écriture admet des séparateurs à des langues dont le système d'écriture n'en admet pas, alors qu'elles ne sont pas nécessairement appropriées ni intuitives dans celles-ci. Nous développons ce point dans la section 2.2.

[8] On pourra tout de même remarquer qu'une séparation est définie dans le cas du *romaji*, transcription de la langue utilisant l'alphabet latin, ou encore dans le cas des livres pour enfants où le système d'écriture utilise uniquement l'alphabet syllabique dit *hiragana*. Dans ce dernier cas, la segmentation est en bunsetsus.

Une séparation graphique entre les mots ne pouvant être définie dans le cadre de la théorie linguistique générale, le mot graphique lui-même ne peut être défini grace au système d'écriture d'une langue particulière, et n'est ainsi pas une unité linguistique à valeur générale.

Pour Mounin, bien que dans certaines langues on dispose d'indices phonétiques (comme par exemple le changement de la consonne initiale en contexte dans le cas du breton) la notion de mot est empiriquement liée à sa forme écrite : ce qui peut alors sembler clair dans le système d'écriture de beaucoup de langues européennes, où un mot est une unité délimitée par deux blancs, un signe de ponctuation et un blanc ou l'inverse, ne l'est pas dans d'autres langues. Mounin conclut ainsi[9] :

> *Le mot n'est pas une réalité de linguistique générale.*

Ainsi, le mot graphique peut être encadré par deux termes dotés d'une réalité linguistique. D'un côté, le terme de *lexème*, porteur de sens, qui devient *monème* s'il est indécomposable en unités plus petites ; de l'autre, le terme de *lexie*, unité de surface du lexique et qui comprend le lexème, ses dérivés affixaux et ses composés. Mounin cite l'exemple suivant : *pomme*, *pommier* et *pomme de terre* sont des lexies alors que seul *pomm(e)* est un lexème[10].

Même dans le cas où une unité de mot peut être précisée dans le cadre d'une langue en particulier, l'application de critères rigoureux produit des analyses dont le résultat s'écarte bien souvent considérablement du sens commun du terme. En effet, on peut retrouver ce même problème dans des langues dont le système d'écriture admet des séparateurs. Martinet[11] cite lui-même les cas suivants :

- le cas du génitif en anglais : ainsi, dans *The King of England's*, on ne sait pas dire si l'on dénombre 4 ou 5 mots graphiques ;
- dans le mot composé en français *Bonne d'enfant*, on dénombre 1 ou 3 mots, alors que dans sa traduction allemande *Kindermädchen*, on peut se demander de façon légitime si l'on doit compter 1 ou 2 mots, puisque que l'on retrouve indépendamment *Kinder* et *Mädchen*.

La segmentation des mots composés en allemand n'est pas évidente, comme le montre l'exemple extrême suivant[12] : *Lebensversicherungsgesellschaftsangestellter*, qui signifie en français *Employé de compagnie d'assurance vie*. Un tel exemple nous renvoie à la réalité : la langue est d'abord et avant tout orale. Oralement, on peut ne pas faire de pause dans un tel mot composé, alors qu'on sépare les mots perçus comme tels. De ce fait, un mot composé est un mot, au même titre que ses composés sont des mots, lorsqu'ils sont pris comme tels. Est un mot, ce qui a un sens autonome.

Martinet préfère au terme *mot* le terme *syntagme autonome* : le syntagme est chez lui une combinaison d'unités réalisée par un sujet parlant. Le fait qu'il soit autonome implique que sa fonction ne dépend pas de sa place dans l'énoncé. Ainsi, il propose la définition suivante :

> Un syntagme autonome formé de monèmes non séparables est ce qu'on appelle communément un mot. On étend toutefois cette désignation aux

[9] Mounin, *Dictionnaire de la linguistique*, 1974, p. 222.

[10] Pommier est formé de deux lexèmes, *pomm-* désignant le fruit, et le foncteur *-ier*, signifiant « arbre à X ».

[11] Martinet, *Éléments de linguistique générale*, 1970, p. 116.

[12] Voir Manning & Schütze, *Foundations of statistical natural language processing*, 1999, p.129 .

> monèmes autonomes comme *hier*, *vite*, ainsi qu'aux monèmes non autonomes, fonctionnels comme *le*, *livre*, *rouge*, dont l'individualité phonologique est généralement bien marquée encore que leur séparabilité ne soit pas toujours acquise [...].

Opérer avec une unité significative plus vaste que le monème et qu'on appelle *mot* ne pose dès lors pas d'inconvénient, tant que l'on garde à l'esprit que le terme de *mot* recouvre dans chaque langue des types particuliers de relations syntagmatiques.

Nous allons donc nous intéresser à une autre unité possible pour découper la langue, afin de traiter en particulier les langues dont le système d'écriture n'admet pas de séparateur : le caractère, tel qu'utilisé pour écrire les textes électroniques.

2.2 Utilisation du caractère comme unité de base du traitement automatique des textes

Afin de contourner les problèmes exposés précédemment, nous proposons l'utilisation d'une autre unité que le mot comme unité de base du traitement automatique des langues. Est-il possible d'utiliser un atome qui soit plus petit, et plus accessible que le mot, tout en permettant un traitement élégant des données linguistiques avec de bonnes performances ?

Le caractère, sous sa forme électronique écrite, est une unité immédiatement accessible dans toute langue informatisée. En fonction de la langue et du codage utilisé, la taille d'un caractère peut varier entre un et plusieurs octets. Cette taille étant connue, nous proposons que la langue sous sa forme électronique écrite soit traitée en tant que chaine de caractères plutôt qu'en tant que chaine de mots. Recenser des séparateurs tels que l'espace ou la ponctuation dans les langues européennes n'est alors plus nécessaire. Les espaces et signes de ponctuation sont traités dans le texte électronique comme ce qu'ils sont : des caractères comme les autres, qui font partie d'une plus grande chaine de caractères.

Cette unité présente un autre avantage : si l'on connait les caractéristiques d'un caractère électronique dans un codage donné, alors les méthodes du traitement automatique des langues utilisant une telle unité deviennent indépendantes de la langue considérée, au moins d'une façon pratique (plus besoin de prétraitement spécifique à la langue qu'on traite pour la segmenter en atomes).

Enfin, bien qu'il faille des chaines de longueur plus grande pour englober ce qui serait autrement une séquence de mots, cela est compensé par le fait que la taille du vocabulaire diminue. On peut le vérifier en effectuant une rapide expérience sur un corpus anglais/japonais, tel que celui utilisé en partie III, section 2.4 pour les campagnes IWSLT-2004 et 2005 d'evaluation de la traduction automatique. Sur 20 000 lignes en langue anglaise, un vocabulaire de 8 191 mots se ramène à un vocabulaire de 61 caractères seulement (avec en moyenne $3,91$ caractères par mot orthographique, i.e. toute suite délimitée par l'espace ou la ponctuation). De façon similaire, sur 20 000 lignes en langue japonaise segmentées automatiquement, un vocabulaire de 9 506 mots se ramène à un vocabulaire de 1 871 caractères seulement (avec en moyenne $1,87$ caractères par mot).

Concernant ce problème de la taille, mesurée en nombre de caractères, nécessaire pour couvrir l'équivalent en mots, on peut rapporter par ailleurs un résultat surprenant : Och[13] montre que dans le cadre d'un système d'alignement statistique de

[13] Och, *Statistical machine translation: Foundations and recent advances*, 2005.

mots (GIZA++), un mot anglais peut être remplacé par ses N premiers caractères sans détériorer la qualité de l'alignement de façon significative. Il montre ainsi que l'alignement conserve une performance similaire lorsqu'on ne garde que les $N = 4$ premiers caractères des mots à aligner.

Dans cette étude, nous examinerons l'utilisation du caractère comme atome, unité de base du traitement automatique des langues. Reprenons les considérations que nous avons faites sur le problème du caractère arbitraire de la découpe en mots. D'habitude, tout le monde transpose des méthodes en mots pour une langue où l'on « voit » les mots, à des langues sans mots visuels. On tronque en mots « artificiels » des textes écrits dans des langues dont le système d'écriture d'admet pas de séparateur graphique.

Nous proposons dans cette étude d'inverser la démarche : nous montrons que certaines méthodes peuvent s'appliquer en caractères sur une langue comme l'anglais, et donc qu'on peut les appliquer en caractères sur des langues sans séparateur de mots (voir figure 2.1).

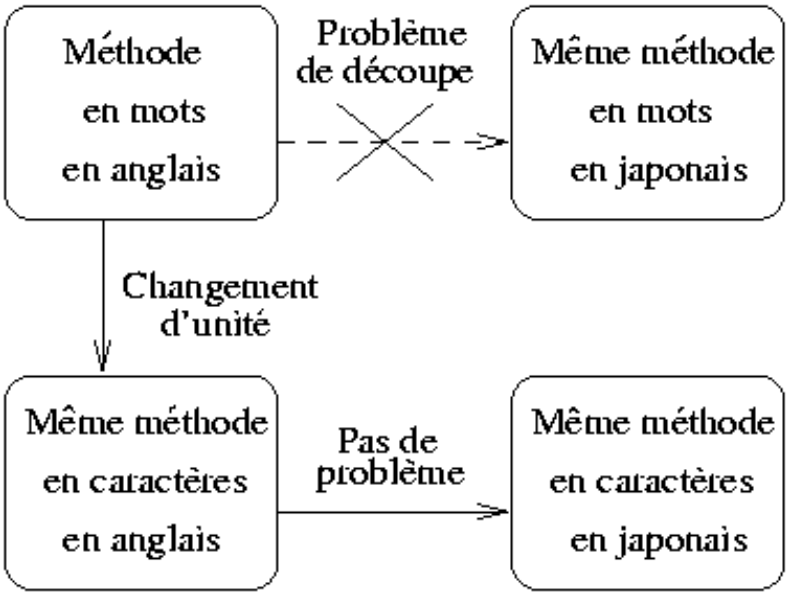

Figure 2.1: Transposition de méthodes en mots vers des méthodes en caractères afin de contourner le problème de découpe.

Nous allons dans un premier temps illustrer l'utilité pour le traitement automatique des langues de méthodes utilisant le caractère come unité, en nous intéressant à une application en particulier : l'évaluation de la traduction automatique.

2.3 Illustration préliminaire de l'intérêt des méthodes en caractères

2.3.1 Introduction au problème général de la tâche d'évaluation de la traduction automatique

L'évaluation est une tâche essentielle dans le domaine de la traduction automatique[14], mais qui, dans les conditions où elle est mise en œuvre actuellement, se révèle particulièrement couteuse. Des jugements qualitatifs produits par des humains sont couteux en terme de temps, et sont subjectifs : c'est pourquoi les méthodes d'évaluation fondées sur des jugements humains sont dites *subjectives*. Afin d'accélérer le processus et ainsi de réduire les couts, tout en réduisant la subjectivité sur la qualité des jugements, on a cherché à produire des mesures automatiques, dites *objectives*. Malheureusement, bien que ces mesures permettent d'accélérer considérablement l'évaluation proprement dite des systèmes, elles sont beaucoup plus couteuses à préparer.

Le rapport ALPAC[15] aborda le problème de l'évaluation de la traduction automatique et jeta les bases d'une méthodologie pour l'évaluation subjective de systèmes. Cette méthodologie sera reprise, nous le verrons, jusque dans les méthodes objectives proposées du début des années 1990 à nos jours. En dehors de critères externes tels que la vitesse de traduction, il s'agit d'évaluer la performance linguistique des systèmes.

Actuellement, deux critères indépendants sont souvent utilisés afin d'évaluer une traduction : *intelligibilité* et *fidélité*, qui sont notées selon une échelle de 9 notes. Les données d'évaluation sont constituées de phrases sélectionnées au hasard, prises séparément, retirées de leur contexte. Elles sont traduites par des systèmes d'une part et des traducteurs humains d'autre part, et deux comparaisons sont effectuées : la première entre la phrase en langue source et la phrase traduite par un système (par des juges bilingues), la deuxième uniquement en langue cible entre la phrase traduite par un traducteur humain et la phrase traduite par un système (par des juges parlant uniquement la langue cible). Pour chaque phrase, un juge doit mesurer son intelligibilité, sa fidélité, ainsi que le temps nécessaire pour lire et mesurer l'intelligibilité de la phrase traduite.

Avec le recul, de vives critiques se firent jour sur la méthodologie adoptée par le rapport ALPAC, qui en lui-même ne proposait ni ne réalisait aucune évaluation : la fidélité[16] n'était pas définie de façon précise, les phrases étaient jugées hors de leur contexte[17]. L'approche contre-intuitive adoptée alors désavantageait les systèmes face aux traducteurs. Il était entre autres reproché à la méthodologie de favoriser implicitement la qualité grammaticale, au détriment de la qualité informationnelle des traductions, alors qu'une tâche de veille scientifique donne plus d'importance à ce dernier aspect.

Au début des années 1990, un deuxième rapport réalisé par la JEIDA[18] (Japanese Electronic Industry Development Association) proposa beaucoup plus d'axes d'a-

[14] À tel point que certains prétendent que l'évaluation de la traduction automatique a fait couler plus d'encre que la traduction automatique elle-même (remarque attribuée à Yorick Wilks, voir King *et al.*, *FEMTI: Creating and using a framework for MT evaluation*, 2003).

[15] ALPAC, *Language and machines. Computers in translation and linguistics*, 1966.

[16] King, *Evaluating natural language processing systems*, 1996.

[17] Pankowicz, *Commentary on ALPAC report*, 1966.

[18] JEIDA, *Methodology and criteria on machine translation evaluation*, 1992.

nalyse que le rapport ALPAC, la méthodologie permettant entre autres l'évaluation de la qualité des traductions fournies par un système de traduction automatique. La méthode était fondée sur l'utilisation d'un jeu de test, construit en fonction des problèmes que présentaient les traductions effectuées par le système. Un jeu de test devait ainsi contenir tous les phénomènes concernés par l'évaluation, et seulement ceux-ci. Beaucoup d'autres facteurs sont pris en compte : par exemple le cout d'évolution des dictionnaires, la capacité d'amélioration, ou l'environnement ergonomique d'utilisation.

La campagne ARPA[19], qui eut lieu entre 1992 et 1994, et visa à faire participer un grand nombre de systèmes commerciaux et académiques, fut responsable de l'introduction de deux critères de qualité, *compréhensibilité* et *qualité de traduction*, ainsi que de la production d'un corpus de jugements numériques de la qualité de traduction, par rapport à un ensemble de références. C'est le cout relativement élevé lié à la production de tels jugements qui a poussé à la recherche de méthodes permettant d'estimer automatiquement la qualité de systèmes de traduction automatique.

Dès lors, la plupart des méthodes envisagées calculent d'une manière ou d'une autre la similarité entre les sorties de système de traduction automatique, et au moins une traduction référence. Les premières approches comparant traductions candidates et traductions de référence sont fondées sur plusieurs idées intuitives : par exemple, celle que le score de similarité doit être proportionnel au nombre de mots en commun[20], ou encore celle que des mots correspondants présents dans le même ordre devraient produire un score supérieur à celui de mots présents dans le désordre[21]. Dernièrement, les mesures automatiques sont fondées sur une idée encore plus simple : plus la somme des longueurs des sous-chaines contigües, communes à la chaine candidate et à la chaine référence est grande, plus le score devrait être élevé.

Des mesures d'évaluation automatiques apparues au cours des 5 dernières années, telles que BLEU[22], NIST[23] ou mWER[24] sont en grande partie fondées sur cette idée simple. Elles sont maintenant largement utilisées par la communauté des chercheurs en traduction automatique statistique afin d'évaluer leurs prototypes au jour le jour[25].

Nous avons vu que ces méthodes reposent globalement sur une idée commune : comparer les sorties des systèmes de traduction automatique à un certain nombre de traductions de référence, produites à l'avance par des traducteurs humains. La comparaison est effectuée en termes de comptage de courtes séquences de mots[26].

[19] White *et al.*, *ARPA MT evaluation methodologies : evolution, lessons and further approaches*, 1994.

[20] Melamed, *Automatic evaluation and uniform filter cascades for inducing N-best translation lexicons*, 1995.

[21] Rajman & Hartley, *Automatically predicting MT systems rankings compatible with fluency, adequacy or informativeness scores*, 2001.

[22] Papineni *et al.*, *BLEU: a method for automatic evaluation of machine translation*, 2001.

[23] Doddington, *Automatic evaluation of machine translation quality using N-gram co-occurrence statistics*, 2002.

[24] Och, *Minimum error rate training in statistical machine translation*, 2003.

[25] En revanche, elles ne sont pas couramment utilisées par les fournisseurs industriels de traduction automatique, qui doivent évaluer des dizaines de systèmes opérant sur couples de langues différents, et pour des tâches bien définies. Nous abordons le problème des couples de langues plus loin.

[26] Ce que l'on appelle par abus de language des *N-grammes de mots*, sans rapport avec un *modèle statistique N-gramme de langue*. En toute logique, une séquence de N mots devrait être appelée

Cependant, quand on a affaire à du chinois ou à de l'arabe par exemple, le découpage en mots n'est pas celui de l'anglais. Par conséquent, bien que de telles méthodes d'évaluation de la traduction automatique soient en principe indépendantes de la langue considérée, elles sont en pratique appliquées uniquement en mots, et donc habituellement appliquées uniquement lorsque la langue cible est la langue anglaise. Les organisateurs de campagnes d'évaluation internationales NIST[27], TIDES[28] ou IWSLT[29] évaluent des sorties de systèmes de traduction automatique qui sont déjà auparavant segmentées en mots. Les campagnes d'évaluation visant d'autres langues cibles que l'anglais, par exemple dans le sens anglais-japonais ou anglais-chinois, sont rarement effectuées[30].

2.3.2 Utilisation du caractère pour une méthode d'évaluation de la traduction automatique telle que BLEU

Problème de la segmentation en mots

Quand un système de traduction automatique utilisant l'approche statistique traduit de l'anglais vers le chinois ou le japonais, il est capable de produire des sorties segmentées en mots, puisque son fonctionnement est fondé sur une telle unité, via son modèle lexical[31]. Mais cela n'est pas nécessairement le cas lorsqu'il s'agit de systèmes commerciaux. Par exemple, Systran ne produit pas de texte segmenté lorsqu'il traduit de l'anglais en chinois ou en japonais. L'évaluation comparative de systèmes de traduction automatique dont la langue cible ne présente pas de segmentation évidente en mots est donc problématique. Elle se heurte à un problème de temps et de cout, puisqu'une telle segmentation doit être effectuée à la main, par des êtres humains. On pourrait certes appliquer des outils de segmentation automatique sur de telles sorties, par exemple le segmenteur de l'université de Pékin pour le chinois[32] ou ChaSen[33] pour le japonais, puis appliquer l'évaluation automatique. Cependant, les scores obtenus seraient alors biaisés par les taux d'erreur des outils de segmentation appliqués sur des sorties de systèmes de traduction automatique[34], sorties qui diffèrent considérablement de textes standard. La segmentation de telles sorties donnerait lieu à une performance différente de celle de textes standard.

Par conséquent, on peut difficilement comparer de façon équitable des scores obtenus pour un système produisant des phrases non segmentées à des scores obtenus pour un système produisant des phrases déjà segmentées en mots. En revanche, tout texte sous sa forme électronique constitue une chaine de caractères, il présente donc nécessairement une segmentation immédiate en caractères, que l'on peut utiliser sans pré-traitement.

N-séquence de mots, ou plus simplement une chaine de N mots.

[27] Przybocki, *The 2005 NIST machine translation evaluation plan (MT-05)*, 2004.

[28] `http://www.nist.gov/speech/tests/mt/mt_tides01_knight.pdf`

[29] Akiba *et al.*, *Overview of the IWSLT04 evaluation campaign*, 2004.

[30] Dans le cas d'IWSLT05 par exemple, 4 couples dont la langue cible est l'anglais sont évalués : chinois-anglais, japonais-anglais, coréen-anglais, arabe-anglais. Un seul couple a pour langue cible une autre langue, le couple anglais-chinois.

[31] Brown *et al.*, *A statistical approach to machine translation*, 1990.

[32] Duan *et al.*, *Chinese word segmentation at Peking University*, 2003.

[33] Matsumoto *et al.*, *Morphological analysis system ChaSen version 2.2.9*, 2002.

[34] De tels taux d'erreurs se situent habituellement entre 5% et 10% pour des textes standards. Une évaluation complète des outils de segmentation sur des sorties de systèmes de traduction automatique serait par ailleurs requise.

On peut alors se poser la question de la transposition de la méthode BLEU en caractères, conformément à l'argumentation exprimée dans la section 2.2 : à cause du problème de la découpe en mots, on ne peut appliquer directement la mesure BLEU sur des textes écrits en japonais. On peut en revanche montrer que la mesure fonctionne lorsqu'elle est transposée en caractères, sur une langue pourvue de séparateurs comme l'anglais. Ensuite, elle est applicable immédiatement en caractères sur une langue dépourvue de séparateurs, comme le japonais (voir figure 2.1).

Nous proposons donc dans cette étude une transposition de la méthode BLEU d'évaluation automatique de la traduction automatique en unité de mot vers l'unité de caractère. Les raisons pour lesquelles notre choix s'est arrêté sur la mesure BLEU sont les suivantes : tout d'abord, c'est actuellement la mesure la plus utilisée par la communauté scientifique effectuant des recherches et du développement dans le domaine de la traduction automatique statistique, à tel point par exemple que seule la mesure BLEU est utilisée lors des campagnes d'évaluation NIST. D'autre part, BLEU est représentative d'une classe de mesures fondée sur l'attestation de chaines de N unités[35].

BLEU en caractères

Indépendamment du problème de segmention en mots exposé précédemment, il est indéniable que des méthodes comme BLEU et NIST ont été largement et très rapidement adoptées par la communauté internationale de la traduction automatique. Nous présentons en détail les méthodes BLEU et NIST en annexe B. Bien qu'elles ne soient pas exemptes de critiques[36], elles s'intègrent comme des composantes automatiques dans un cadre d'évaluation plus large. Ces deux mesures rendent compte de caractéristiques complémentaires des traductions jugées : *fluidité* et *informativité*[37]. Bien qu'imparfaites, elles ont l'avantage d'être automatiques, rapides et donc peu onéreuses dans leur application, la concentration des couts se faisant majoritairement sur le temps de préparation des traductions références[38]. C'est pour cette raison qu'il est illusoire d'exiger de la communauté de la traduction automatique qu'elle abandonne le savoir-faire pratique lié à l'utilisation de ces mesures. Nous estimons donc qu'il est préférable de trouver une équivalence avec des mesures bien établies plutôt que de montrer une corrélation satisfaisante avec des jugements humains, ce qui serait équivalent à proposer une nouvelle mesure d'évaluation de la traduction automatique. Voilà ce qui nous conduit à proposer l'adaptation, la transposition d'une méthode déjà existante et bien établie d'évaluation automatique de la traduction automatique depuis l'unité de mot vers l'unité de caractère.

L'objectif avoué de cette étude n'est donc pas de rechercher une corrélation avec le jugement humain, mais d'établir une équivalence entre des scores BLEU obtenus de deux manières : en mots et en caractères. Intuitivement, on s'attend à trouver une corrélation élevée. Cependant, elle demande à être démontrée, quantifiée, et

[35] On pourrait imaginer, sous réserve de vérification, que la démonstration d'une transposition réussie en unité de caractère soit reproductible dans le cas d'une autre mesure de cette même classe, par exemple NIST. On peut toutefois légitimement penser que celà marcherait moins bien, pour des raisons que nous explicitons dans la conclusion de l'annexe B portant sur les mesures BLEU et NIST.

[36] Blanchon, *Comment définir, mesurer, et améliorer la qualité, l'utilisabilité et l'utilité des systèmes de TAO de l'écrit et de l'oral*, 2004.

[37] Voir Akiba *et al.*, *Overview of the IWSLT04 evaluation campaign*, 2004, p.7 .

[38] Nous proposons dans la partie III, chapitre 1 la réduction du cout de production de ces références par une méthode de génération automatique.

il faut déterminer les nombres optimaux de caractères et de mots pour lesquels la meilleure corrélation est obtenue. C'est ce à quoi nous nous attachons dans les sections suivantes.

2.3.3 Expériences

Nous rappelons brièvement la méthode de calcul d'un score BLEU, avant de présenter les données qui nous serviront à étudier le lien entre BLEU en mots et en caractères. Une description plus approfondie de la méthode BLEU est donnée en annexe B.1.

BLEU représente une classe de méthodes d'évaluation automatique jugeant la précision. Brill[39] la compare a ROUGE[40], une méthode couramment employée pour l'évaluation des systèmes de résumé automatique. D'après lui, ROUGE représenterait plutôt un exemple de classe de méthodes d'évaluation automatique jugeant le rappel[41]. Le résumé automatique est une tâche pour laquelle la notion de rappel est plus pertinente qu'en traduction automatique : le but est d'éliminer un maximum d'informations redondantes, en gardant toutefois l'intégralité de l'information essentielle. En revanche, BLEU privilégie la précision, en mesurant le recouvrement d'une traduction à juger par rapport à des phrases de référence.

Calcul du score BLEU et données expérimentales

Nous adoptons la notation proposée par Babych[42] : on note BLEU_{wN} un score BLEU calculé en mots, avec un ordre maximal N ; on note BLEU_{cM} un score BLEU calculé en caractères, avec un ordre maximal M. Pour un ordre maximal N, un score BLEU_{wN} est le produit de deux termes : une pénalité BP fonction de la brièveté de la phrase jugée, et la moyenne géométrique des précisions modifiées à l'ordre n, notée p_n, calculée pour toutes les longueurs de chaines jusqu'à N (voir l'annexe B.1 pour plus de détails). On choisit habituellement la valeur pratique de $N = 4$ pour l'évaluation en langue anglaise, car c'est cette valeur qui a donné dans l'étude originale les meilleures corrélations avec un jugement humain.

$$\text{score BLEU}_{wN} = BP \times \sqrt[N]{\prod_{n=1}^{N} p_n}$$

En transposant la méthode des mots vers les caractères, nous ne modifions pas la formule originale de BLEU : nous l'appliquons à des chaines de M caractères, au lieu de chaines de N mots.

Pour cette étude, nous avons besoin d'une langue dans laquelle la segmentation en mots orthographiques est claire[43]. C'est évidemment le cas de la langue anglaise, où l'on peut s'appuyer sur l'usage de l'espace et de la ponctuation pour obtenir une segmentation en mots a priori. Afin de montrer l'équivalence entre BLEU en mots et en caractères, nous nous appuierons donc sur l'anglais. Les expériences présentées ici se basent sur un ensemble de test de 510 phrases japonaises tirées du corpus

[39] Brill & Soricut, *A unified framework for automatic evaluation using N-gram co-occurence statistics*, 2004.

[40] Lin & Hovy, *Automatic evaluation of summaries using N-gram co-occurence statistics*, 2003.

[41] ROUGE remplace en effet la précision modifiée à l'ordre n, notée p_n dans la formulation de BLEU, par une mesure de couverture inspirée du rappel, et notée C_n.

[42] Babych & Hartley, *Modelling legitimate translation variation for automatic evaluation of MT quality*, 2004.

[43] Tout du moins étant donné des règles définies à l'avance.

BTEC[44], et traduites en anglais par 4 systèmes différents de TA, soit au total 2 040 traductions candidates. On dispose pour chaque phrase candidate d'un ensemble de 13 références produites auparavant à la main.

La traduction en anglais de la phrase japonaise :

濃いコーヒーが飲みたい。
/koi koohii ga nomitai/

par l'un des 4 systèmes de traduction automatique est :

I'd like to have some strong coffee.

Elle constitue l'une des 2 040 phrases candidates, et sera évaluée par rapport aux 13 références suivantes préparées à l'avance[45] :

01- I'd like some strong coffee.
02- I want some strong coffee.
03- I want to drink some strong coffee.
04- I'd like a strong coffee.
05- I'd like to drink some strong coffee.
06- I'd like to have a cup of strong coffee.
07- I want some strong coffee.
08- Strong coffee would taste good right about now.
09- I'd like a strong cup of coffee.
10- I'd like a good strong cup of coffee.
11- What I need right now is some strong coffee.
12- I want a strong cup of coffee.
13- I want a cup of strong coffee.

Pour chacune des 2 040 phrases, nous avons calculé les scores BLEU en mots et en caractères. Le tableau 2.1 donne les caractéristiques des données candidates et références.

Tableau 2.1: Caractéristiques numériques des données utilisées pour calculer les scores BLEU.

	candidats	références
caractères / phrase	30,65 ± 15,95	31,58 ± 18,02
mots / phrase	6,31 ± 3,26	7,08 ± 3,31
caractères / mot	3,84 ± 2,10	3,80 ± 2,07

Équivalence BLEU_{wN} / BLEU_{cM}

Afin d'établir l'équivalence entre BLEU en mots et en caractères, nous allons utiliser trois méthodes différentes pour étudier l'équivalence entre BLEU_{wN} et BLEU_{cM} : nous recherchons la meilleure corrélation, le meilleur accord de jugement entre les deux mesures, et le meilleur comportement d'après une propriété intrinsèque de BLEU.

[44] Nous donnons un aperçu du corpus BTEC et de l'ensemble de test de 510 phrases en annexe A.1.

[45] Les références sont préparées à l'avance par des traducteurs professionnels. Pressés de produire un grand nombre de paraphrases, ils produisent parfois des constructions peu naturelles, comme par exemple dans le cas de la 8e référence. C'est un des travers des méthodes d'évaluation basées sur la comparaison avec des références.

Meilleure corrélation La corrélation linéaire entre deux variables $X = (x_1, \ldots, x_n)$ et $Y = (y_1, \ldots, y_n)$ est le rapport entre la covariance et le produit des écarts-types. Si on note $\overline{x}$ la moyenne de X, et σ_x l'écart-type de X, alors on peut écrire le coefficient de corrélation sous la forme suivante :

$$r_{XY} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sigma_x \sigma_y}$$

Pour un ordre donné N, nous voulons déterminer la valeur de M pour laquelle les scores BLEU_{cM} (donc calculés en caractères) ont la corrélation la plus forte avec les scores BLEU_{wN}.

Pour cela, nous calculons pour tous les N variant de 1 à 4 et tous les M variant de 1 à 25, toutes les corrélations entre BLEU_{wN} et BLEU_{cM}. Nous déterminons alors pour chaque N la valeur de M qui donne la meilleure corrélation. Les résultats obtenus sont résumés dans le tableau 2.2. Pour la valeur pratique de $N = 4$ mots utilisée habituellement pour l'anglais, le meilleur M est de 17 caractères. On pourra remarquer que le rapport moyen M/N obtenu est de $4,14$, une valeur proche de la longueur moyenne de caractères par mot de $3,84$ (voir tableau 2.1).

Tableau 2.2: Valeurs de N et M équivalentes pour BLEU_{wN} et BLEU_{cM} obtenues par plusieurs méthodes.

	BLEU_{w1}	BLEU_{w2}	BLEU_{w3}	BLEU_{w4}
Corrélation (meilleur M)	0,89 (5)	0,90 (8)	0,85 (10)	0,83 (17)
Kappa (meilleur M)	0,17 (5)	0,29 (9)	0,34 (14)	0,35 (18)
meilleur M pour un comportement similaire à l'ordre $(N-1)$ (seuil = 90%)		(9)	(14)	(18)

Meilleur accord de jugement Le cœfficient Kappa[46] mesure l'accord entre deux juges, dont les jugements sont appariés. Ces jugements doivent être réalisés sur une échelle commune de r notes.

Si on note n_i le nombre de points auquel les deux juges ont accordé la même note r alors on peut écrire P_o, la proportion d'accord observée, et P_e, la proportion d'accord aléatoire :

$$P_o = \frac{1}{n}\sum_{i=1}^{n} n_i$$

$$P_e = \frac{1}{n^2}\sum_{i=1}^{n} n_i^2$$

Le Kappa s'écrit ainsi :

$$K = \frac{P_o - P_e}{1 - P_e}$$

Nous utilisons cette technique pour mesurer l'accord entre BLEU_{wN} et BLEU_{cM}. De la même manière que précédemment, nous calculons pour tous les M et N,

[46] Cohen, *A coefficient of agreement for nominal scales*, 1960.

tous les cœfficients Kappa entre BLEU_{wN} et BLEU_{cM}, et nous déterminons pour chaque N la valeur de M qui maximise Kappa. Afin de confronter les « jugements » que produisent les scores en BLEU, nous utilisons la méthode suivante : tout score BLEU est compris entre 0 et 1. Nous pouvons donc lui attribuer une note de la façon suivante : on choisit arbitrairement $r = 10$ notes, de 0 à 9, pour découper l'intervalle $[\,0\,;\,1\,]$ en 10 plus petits intervalles. D'autres découpes sont bien sûr possibles, mais nous avons préféré éviter d'employer une découpe plus fine afin de garder un plus grand nombre de points dans chacun des intervalles. Une note de 0 correspond alors à l'intervalle $[\,0\,;\,0,1\,[$, et ainsi de suite jusqu'à la note 9, qui correspond à $[\,0,9\,;\,1\,]$. Par exemple, une phrase avec un score BLEU de $0,435$ se verra attribuer une note de 4. Tout score BLEU devient donc de fait un jugement constitué par une note entière. Calculer un score BLEU en mots puis en caractères pour une même phrase est alors équivalent à demander à deux juges différents de juger une même phrase.

Le maximum du Kappa est atteint pour les valeurs[47] indiquées dans le tableau 2.2. Pour $N = 4$ mots, on trouve un meilleur M de 18 caractères.

Meilleur comportement vis-à-vis du rang inférieur De par sa définition, BLEU dépend de la moyenne géométrique des précisions modifiées à l'ordre n, notées p_n. Nécessairement, on ne peut trouver une chaine de longueur n donné dans une phrase si aucune des deux sous chaines de longueur $(n-1)$ n'est trouvée dans cette même phrase. Par exemple, si on ne peut trouver les chaines *abc* ou *bcd* dans une chaine, alors on ne pourra pas y trouver *abcd*.

Nous pouvons donc énoncer pour BLEU la propriété suivante :

> Quel que soit N, quel que soit le candidat, quel que soit l'ensemble de références,
>
> $$\text{BLEU}_{wN} \leq \text{BLEU}_{w(N-1)}$$

Le graphe de la figure 2.2 vérifie cette propriété expérimentalement. Il montre la correspondance entre scores BLEU_{w4} et BLEU_{w3} pour les données dont nous disposons : tous les points sont bien placés sur la diagonale, ou en dessous.

En utilisant la propriété énoncée ci-dessus, nous cherchons à trouver expérimentalement la valeur M pour laquelle $\text{BLEU}_{cM} \leq \text{BLEU}_{w(N-1)}$ est vrai dans la majorité des cas. Une telle valeur, M, pourra alors être considérée comme équivalente à la valeur de N en mots. Nous recherchons incrémentalement le M qui permettra à BLEU_{cM} d'avoir un comportement similaire à BLEU_{wN} : nous choisissons de fixer ce M lorsqu'un seuil de 90% des points sont sur la diagonale ou en dessous. Pour $N = 4$, comme l'indique le graphe en figure 2.3, le seuil est atteint pour $M = 18$.

Le graphe de la figure 2.4 montre la correspondance des scores pour les données dont nous disposons (bien que ce soit difficilement visible à l'œil nu, 90% des points sont bien sur la diagonale ou en dessous). Ce résultat tend à confirmer que le M pour lequel BLEU_{cM} montre un comportement similaire à BLEU_{w4} se situe aux environs de 18.

[47] À l'exception de $N = 3$ mots pour lequel la valeur obtenue est de $M = 14$ caractères, très différente de celle de 10 obtenue dans le cas de la corrélation, les autres valeurs de M diffèrent par 1 au maximum.

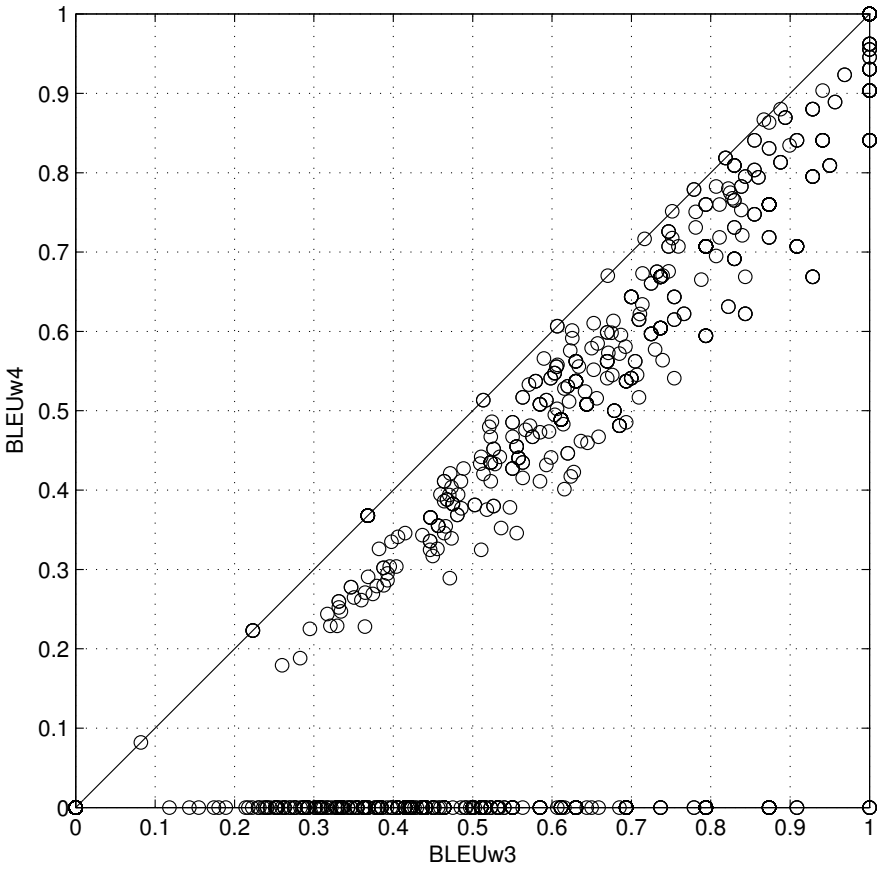

Figure 2.2: Scores BLEU_{w4} en fonction de BLEU_{w3}.

Conclusion Nous avons vu dans les paragraphes précédents comment, pour un N donné, on peut trouver un M correspondant : on détermine M tel que la corrélation soit élevée, qu'il existe un bon accord de jugement, et que le BLEU_{cM} ainsi défini exhibe un comportement similaire au BLEU_{wN} donné.

Pour la valeur pratique de $N = 4$ mots utilisée habituellement en langue anglaise, nous avons trouvé comme meilleures valeurs de M 17 pour la corrélation, 18 pour l'accord de jugement et 18 pour la similarité de comportement. Nous pouvons donc choisir la valeur $M = 18$ pour le nombre de caractères correspondant à $N = 4$ mots.

Comparaison et classement de systèmes de TA

Nous pouvons maintenant recalculer les scores en BLEU_{w4} et BLEU_{c18} des 4 systèmes de traduction automatique différents utilisés précédemment. Le tableau 2.3 indique pour chaque système les valeurs globales obtenues en moyenne sur les 510 phrases de l'ensemble de test.

On remarque qu'en passant du mot au caractère, les scores baissent en moyenne de $0,047$. Nous pouvons trouver à ce fait une justification : une phrase de moins de N unités a nécessairemment un score BLEU de 0 pour des N-grammes de l'unité correspondante. Le tableau 2.4 montre clairement que dans nos données, il y a plus de phrases de moins de 18 caractères (350) que de phrases de moins de 4 mots (302). Il y a donc plus de scores à 0 en caractères, ce qui explique le décalage des scores

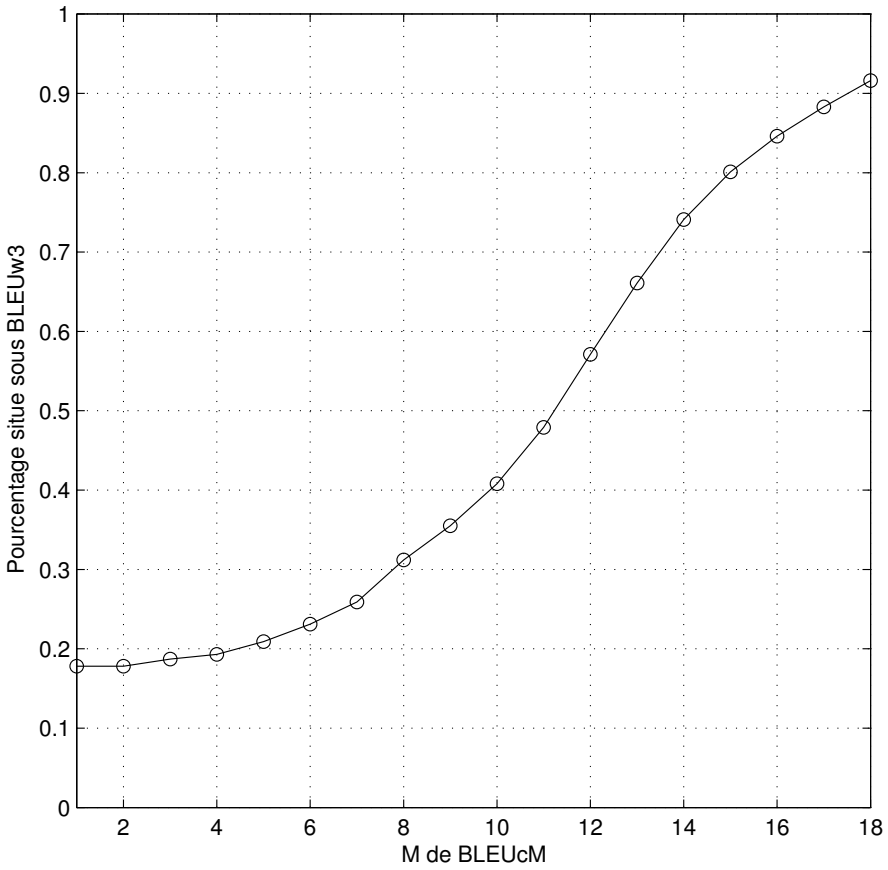

Figure 2.3: Proportion des scores BLEU_{cM} situés sous BLEU_{w3}, pour M variant de 1 à 18.

Tableau 2.3: Scores moyens pour les 4 systèmes de traduction automatique utilisés, en BLEU_{w4} et BLEU_{c18}.

	système 1		système 2		système 3		système 4
score BLEU_{w4}	$0,349$	$>$	$0,312$	$\sim$	$0,305$	$>$	$0,232$
score BLEU_{c18}	$0,292$	$>$	$0,267$	$\sim$	$0,279$	$>$	$0,183$
difference en score	$-0,057$		$-0,045$		$-0,036$		$-0,049$

lorsqu'on passe du mot au caractère.

Tableau 2.4: Distribution des longueurs des 510 phrases de l'ensemble de test, en mots et en caractères.

longueur	<4 mots	≥ 4 mots	total
<18 caractères	266	84	**350**
≥ 18 caractères	37	123	160
total	**302**	208	510

Le tableau 2.3 indique cependant que le passage du mot au caractère ne boule-

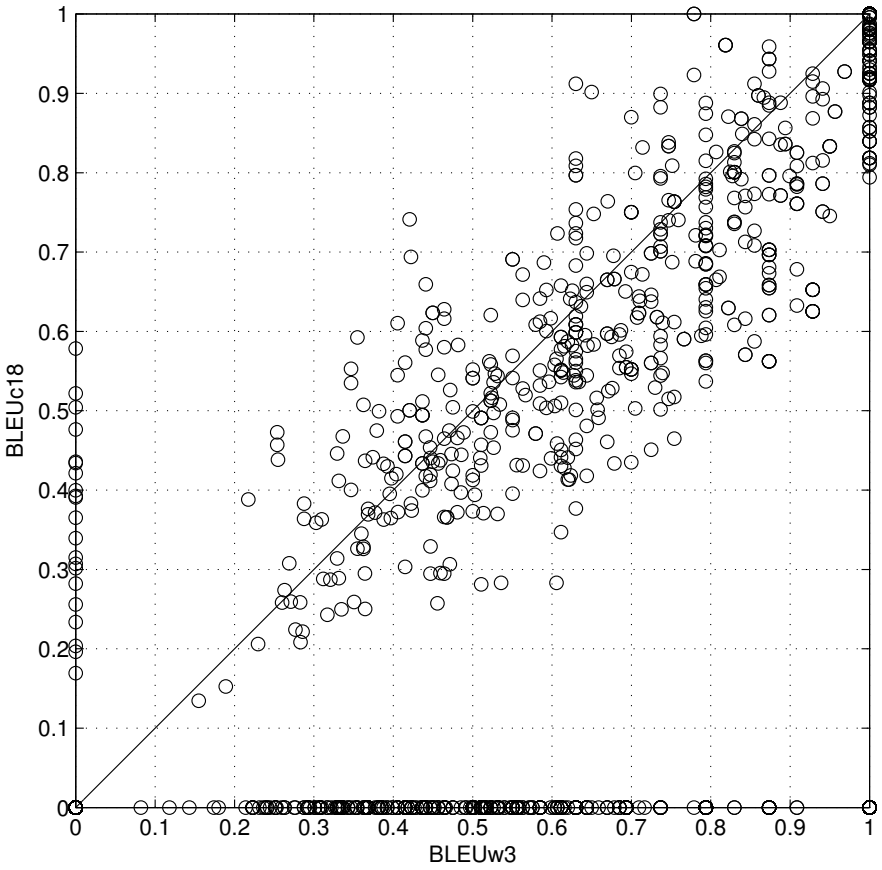

Figure 2.4: Scores BLEU_{c18} en fonction de BLEU_{w3}.

verse pas le classement général des systèmes.

L'application de la méthode standard BLEU dans ces deux unités[48] montre que les intervalles de confiance sont d'environ 2% pour BLEU (dans notre cas ± 0.01), de sorte que les systèmes 2 et 3 ne sont pas distingués par BLEU_{wN}. Cette proximité de scores pour ces deux systèmes peut s'expliquer par le fait qu'il s'agit de deux versions d'un même système de traduction automatique paramètrées différemment.

Réflexion sur la granulation des scores BLEU

Granulation des scores en fonction de l'unité choisie Nous nous intéressons dans cette section à un phénomène que nous nommons par la suite *granulation*[49] : par *granulation*, nous désignons ici l'accumulation de scores autour de certaines valeurs. Après avoir observé le phénomène, nous cherchons à préciser la nature de telles accumulations.

Nous nous intéressons en particulier ici au changement de granulation qu'entraine le passage du mot au caractère. On peut illustrer ce phénomène d'accumulation en

[48] Zhang *et al.*, *Interpreting BLEU/NIST scores: how much improvement do we need to have a better system?*, 2004.

[49] Nous avions tout d'abord pensé désigner ce phénomène par le terme de *granularité*. Après avoir constaté que le terme *granularité* était vraisemblablement un anglicisme (il est présent dans Wikipedia, mais absent du Trésor de la langue française informatisé), nous lui avons préféré le terme de *granulation*.

prenant tout d'abord un cas extrême : la figure 2.5, p. 52 montre les scores BLEU_{w1} et BLEU_{c3} obtenus pour chacune des 2 040 phrases, en mots en ordonnée, et en caractères en abscisse. On observe que des scores de valeurs très proches en BLEU_{w1} semblent s'agglutiner autour de valeurs en abscisse. Ces accumulations semblent se répéter de façon périodique. On ne peut en revanche identifier un tel phénomène d'accumulation sur les ordonnées.

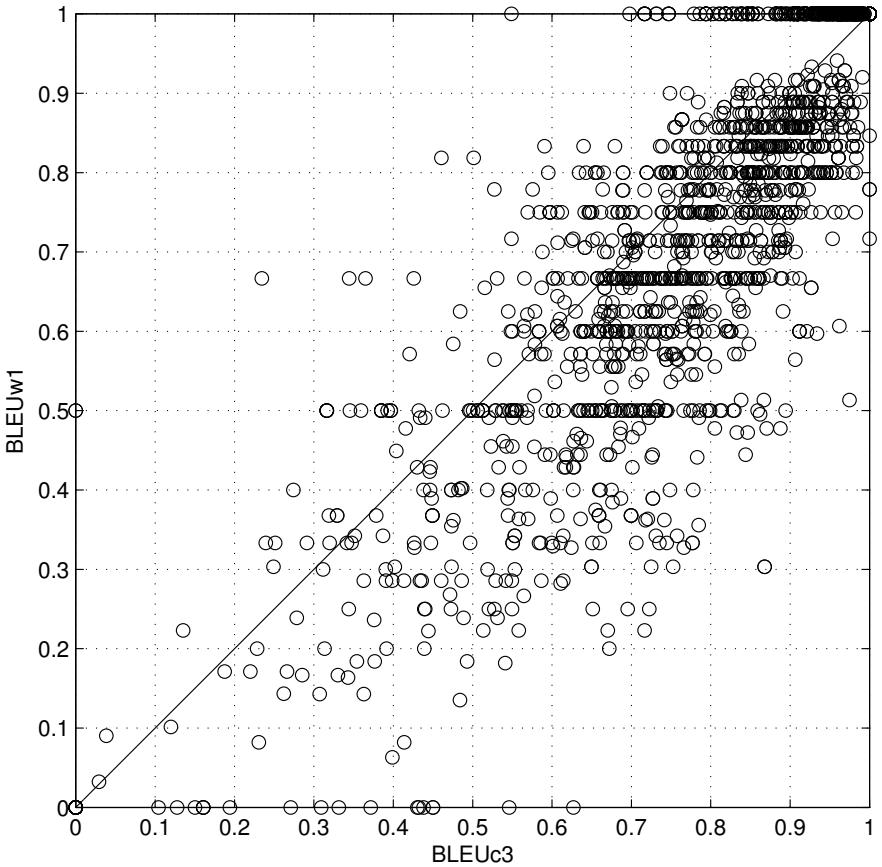

Figure 2.5: Granulation de BLEU_{wN} : scores BLEU_{w1} en fonction de BLEU_{c3}.

Considérons à présent les deux mesures BLEU_{w4} et BLEU_{c18} : le graphe de la figure 2.6 p. 53 montre les scores BLEU obtenus pour chacune des 2 040 phrases, en mots en ordonnée, et en caractères en abscisse.

Là encore, quoique de façon moins prononcée, on observe que des scores de valeurs très proches en BLEU_{w4} semblent s'agglutiner autour de valeurs périodiques. Ces accumulations sur l'axe horizontal et leur absence sur l'axe vertical sont mises en évidence dans la figure 2.7 p. 54, qui montre les distributions des scores pour chacune des deux unités. La périodicité des accumulations sur l'axe horizontal apparait plus nettement en prenant la transformée de Fourier de ces distributions : pour BLEU_{w4}, le premier formant se situe à la valeur 20, indiquant une périodicité[50] de $1/20 = 0,05$. En revanche, une inspection des valeurs de la transformée de Fourier de la distribution des scores BLEU_{c18} ne montre aucun maximum relatif. La courbe est

[50]On relève une périodicité identique des valeurs de BLEU_{w1} dans le cas des mesures BLEU_{w1} et BLEU_{c3}

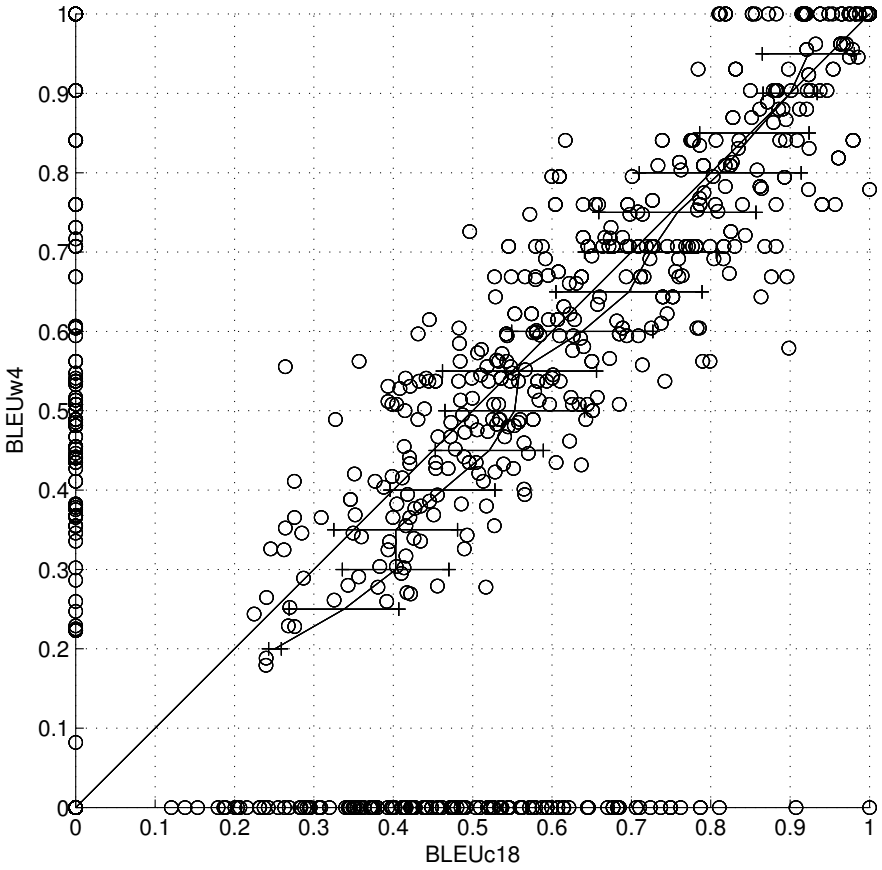

Figure 2.6: Granulation de BLEU_{wN} : scores BLEU_{w4} en fonction de BLEU_{c18}.

strictement décroissante.

Nous faisons de ces observations l'interprétation suivante. La mesure en mots BLEU_{w4} présente un phénomène intrinsèque de granulation : en effet, les scores qu'elle produit ne sont pas distribués de façon homogène et aléatoire sur l'axe de 0 à 1, mais ont tendance à tomber préférentiellement sur des valeurs qui sont des multiples de $0,05$. En revanche, on ne retrouve pas ce phénomène dans le cas des scores produits par la mesure en caractères BLEU_{c18} : les scores sont distribués de façon plus homogène sur l'intervalle $[\,0\,,\,1\,]$. En résumé, lorsqu'on passe de la mesure en mots à la mesure en caractères, la granulation intrinsèque de la mesure en mots disparait.

Conversion de BLEU_{w4} en BLEU_{c18} Le phénomène de granulation peut être exploité de la manière suivante : en exploitant le phénomène de granulation de BLEU_{w4}, on peut tenter de proposer une conversion de BLEU_{w4} en BLEU_{c18} pour les scores calculés individuellement par phrase. On considère la moyenne et l'écart-type des scores BLEU_{c18} tombant dans le même intervalle de granulation de BLEU_{w4} (voir figure 2.6). La moyenne tend à se rapprocher de celle du score BLEU_{w4} correspondant à mesure qu'on se rapproche de 1, reflétant le fait que plus les scores sont élevés, plus ils sont corrélés. La moyenne des écarts-types étant de $0,078$, un score x en BLEU_{w4} est donc équivalent à $x \pm 0,078$ en BLEU_{c18}, pour

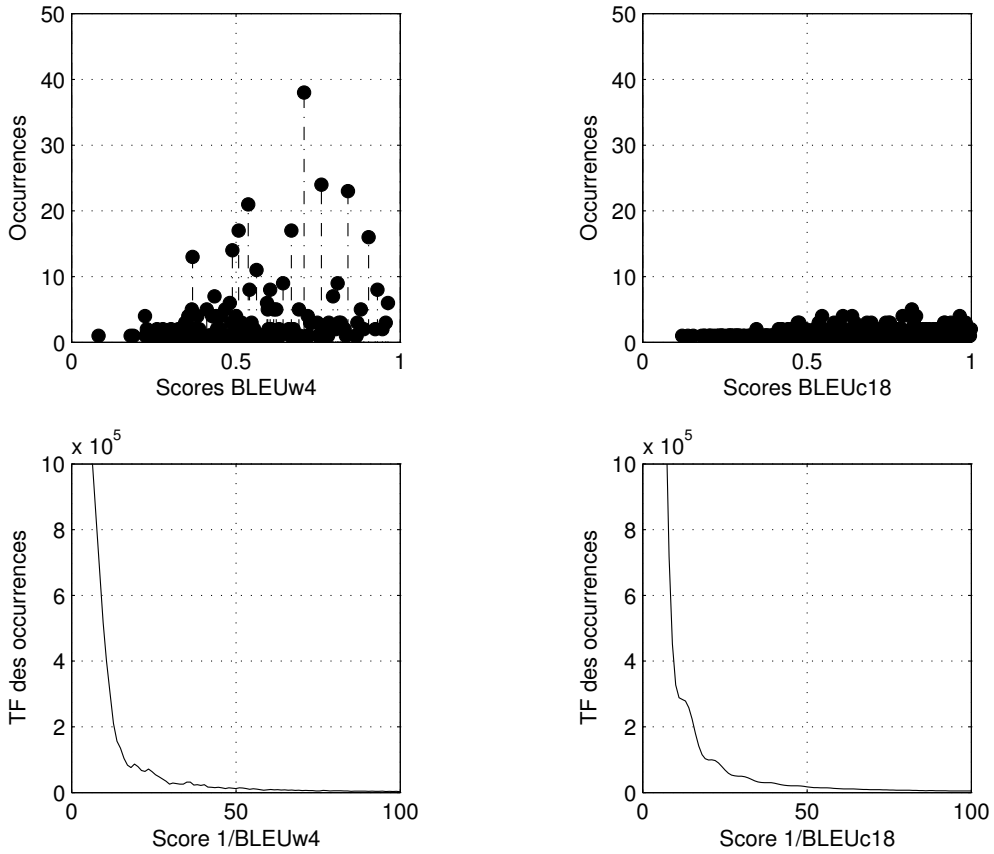

Figure 2.7: Distribution des scores BLEU_{w4} et BLEU_{c18}, et leurs transformées de Fourier respectives.

une phrase prise individuellement.

Conversion de BLEU_{c18} en BLEU_{w4} Pour une conversion dans l'autre sens, on ne peut utiliser la granulation des scores BLEU_{c18}, puisque, comme on l'a vu dans la section plus haut, il n'y en a pas. On choisit donc de découper l'intervalle $[\,0\,,\,1\,]$ en 10 intervalles de taille égale. Pour chacun de ceux-ci, on calcule l'écart-type en termes de score BLEU_{w4} (voir le tableau 2.5). On remarque que l'écart-type est faible pour les scores très faibles ou très élevés. Ainsi, il semble que passer de BLEU en mots à BLEU en caractères soit plus sûr pour les scores très faibles ou très élevés. Expérimentalement, on est satisfait de constater d'autre part que 85% des scores pour des phrases prises individuellement font moins de $0,2$ ou plus de $0,6$ en caractères, pour lesquels l'écart-type en BLEU_{w4} est de $0,125$. Il est important de garder à l'esprit qu'on parle ici de scores calculés sur des phrases individuelles. On constate de fait que les scores moyens de systèmes de traduction automatique tombent généralement dans l'intervalle $[\,0,3\,;\,0,6\,]$, mais il s'agit de scores moyens calculés sur plusieurs centaines de phrases test. Comme nos expériences le montrent, de tels scores sont bien souvent la moyenne de deux populations distinctes de scores sur des phrases individuelles : des scores individuels très faibles, et des scores individuels très élevés.

Tableau 2.5: Conversion de scores BLEU_{c18} en scores BLEU_{w4}.

intervalles de BLEU_{c18}	]0 ; 0.2[	[0.2 ; 0.3[	[0.3 ; 0.4[	[0.4 ; 0.5[	[0.5 ; 0.6[	[0.6 ; 0.7[	[0.7 ; 0.8[	[0.8 ; 0.9[	[0.9 ; 1]
% de points en BLEU_{c18}	-	2,36%	4,55%	10,46%	15,68%	12,98%	12,31%	15,01%	26,64%
% de points en BLEU_{w4}	-	2,70%	5,56%	8,26%	14,50%	10,79%	13,49%	7,96%	36,59%
écart-type	-	$\pm$ 0,102	$\pm$ 0,052	$\pm$ 0,113	$\pm$ 0,073	$\pm$ 0,058	$\pm$ 0,060	$\pm$ 0,062	$\pm$ 0, 033

Application aux sorties non segmentées

Comme nous l'avons mentionné dans l'introduction de cette étude préliminaire, la transposition d'une méthode d'évaluation automatique de la traduction automatique telle que BLEU de la division en mots à la division en caractères avait deux motivations : la première était de vérifier que l'on pouvait montrer une équivalence entre la méthode appliquée en mots, et en caractères. Nous avons cherché à montrer cette équivalence sur une langue où la segmentation en mots est graphiquement claire, prenant en compte la ponctuation et les espaces. C'est pourquoi nous avons choisi de travailler sur l'anglais. Changer d'unité en passant du mot au caractère ne revient pas à, par exemple, passer du mètre au centimètre : un mot est constitué d'un nombre variable de caractères. Le passage n'est donc pas trivial, et il fallait s'assurer qu'une équivalence pouvait être trouvée entre la méthode en mots et la méthode en caractères.

Cependant, cette vérification effectuée, la deuxième motivation de cette transposition est bien entendu de rendre possible l'application d'une méthode fondée sur l'attestation de chaines de N unités, telle que BLEU, dans des langues où une segmentation en mots n'est pas graphiquement claire.

Puisque nous avons effectivement montré une équivalence entre BLEU en mots et en caractères sur une même langue, nous proposons en préambule à la conclusion de cette expérience préliminaire quelques résultats de l'application de la méthode BLEU en caractères sur des sorties de traduction automatique, dans une langue n'admettant pas de segmentation immédiate.

Un des systèmes ayant traduit les 510 phrases de l'ensemble de test est bidirectionnel, et permet donc de traduire de l'anglais vers le japonais. À l'aide de la méthode BLEU en caractères que nous venons d'exposer, nous sommes désormais en mesure d'évaluer la qualité de traductions en japonais[51].

Ces résultats sont présentés à titre illustratif, car ils ne peuvent être confrontés pour l'instant à d'autre mesures automatiques : il n'en existe en effet à notre connaissance aucune pour l'instant, qui soit applicable sur de telles langues. Il faudrait ainsi confronter de tels résultats au jugement humain afin de montrer la validité de la méthode, de la même façon que dans l'article original[52] proposant la méthode

[51] Nous disposons là encore, pour chacune des 510 phrases en japonais, de 13 traductions de référence produites par des traducteurs professionnels.

[52] Papineni *et al.*, *BLEU: a method for automatic evaluation of machine translation*, 2002.

BLEU.

Nous appliquons donc la méthode BLEU en caractères sur les 510 phrases traduisant de l'anglais vers le japonais. Cependant, bien que nous ayons déterminé une longueur équivalente de $M = 18$ pour l'anglais, nous devons de la même manière choisir une longueur de M-gramme adéquate pour le japonais. Malheureusement, nous ne disposons pas d'autres scores automatiques ou de jugements humains sur des phrases en japonais, qui nous permettraient comme précédemment de déterminer un M adéquat[53]. Afin d'appliquer tout de même la méthode, nous nous proposons de déterminer une valeur de M pour la langue japonaise en procédant à une approximation : nous allons considérer qu'en disposant d'une ressource bilingue anglais-japonais, on peut déterminer M par une règle de trois. Si on note M_j la longueur recherchée en caractères japonais, N_a la longueur habituelle de 4 mots en langue anglaise, L_j la taille en caractères de la ressource japonaise et L_a la taille en mots de la ressource anglaise, alors on peut écrire :

$$M_j = N_a \times \frac{L_j}{L_a}$$

Bien entendu, cette approximation est critiquable. En effet, la traduction humaine d'un texte tend à faire gonfler sa taille. Pour que les longueurs des ressources japonaises et anglaises soient équivalentes en terme de taille, il faudrait que l'on soit sûr que la moitié en ait été constituée par traduction anglais-japonais, et l'autre par traduction japonais-anglais, ce qui est probablement faux[54]. Nous ne prétendons pas non plus que la valeur de M_j calculée soit généralement équivalente à la valeur standard $N_a = 4$. Au contraire, la valeur M_j calculée est ici entièrement liée aux caractéristiques de la ressource bilingue utilisée dans l'expérience.

Dans le bicorpus BTEC, on a d'un côté $1,1$ million de mots en langue anglaise, et de l'autre $2,43$ millions de caractères en langue japonaise. La règle de trois s'écrit donc :

$$M_j = 4 \times \frac{2,43}{1,1} \simeq 9$$

Quatre mots en anglais seraient donc équivalents à neuf caractères japonais.

Cette longueur nous permet de calculer un score BLEU de $0,233$ en BLEU_{c9} sur les 510 phrases traduites.

En utilisant la mesure BLEU standard en mots, un tel score n'aurait pas pu être produit sans une segmentation préalable des sorties de traduction automatique. Mais une telle segmentation est en général difficile à obtenir de façon fiable, du fait de la nature fortement bruitée des sorties de systèmes de traduction automatique.

2.3.4 Discussion

Dans cette étude préliminaire, nous avons étudié l'application d'une méthode bien connue d'évaluation de la traduction automatique en prenant le caractère au lieu du mot comme unité[55]. Nous avons recherché une équivalence entre une méthode

[53] C'est même là que réside l'intérêt de notre étude : transposer une méthode d'évaluation automatique dans une langue où elle était jusque là inapplicable.

[54] La ressource BTEC étant constituée de couples de phrases recueillis dans des livrets publiés au Japon, il est en effet probable que l'intégralité du contenu ait été traduit du japonais à l'anglais.

[55] Denoual & Lepage, *BLEU in characters: towards automatic MT evaluation in languages without word delimiters*, 2005.

appliquée en chaines de N mots, puis la même en chaines de M caractères. Cela nous a permis de mettre en évidence une forte corrélation, un bon accord de jugement grâce au calcul du Kappa, ainsi qu'une similarité de comportement vis-à-vis du rang inférieur, pour des valeurs correspondantes de M et N.

Pour la valeur la plus couramment utilisée pour l'anglais $N = 4$, nous avons déterminé une valeur correspondante en caractères de $M = 18$. D'autre part, en examinant la granulation apparente de la mesure en mots, nous avons déterminé de façon expérimentale une procédure de conversion pour des phrases individuelles :

$$\boxed{\text{BLEU}_{c18} \simeq \text{BLEU}_{w4} \pm 0,078}$$

Cette étude préliminaire ouvre donc le chemin à l'application de la méthode BLEU d'évaluation de la traduction automatique à des langues dépourvues de segmentation immédiate en mots, telles que le chinois, le japonais, ou le thaï. Cela ouvre aussi des perpectives de travaux futurs : au delà des résultats donnés en fin de section 2.3.3, il restera à évaluer de manière extensive et dans chaque langue cible considérée la corrélation entre jugement humain et la méthode BLEU en caractères. Il serait d'autre part intéressant de vérifier la possibilité d'une transposition similaire des autres mesures automatiques fondées sur l'attestation de chaines de N unités, telles que NIST ou mWER[56].

Conclusion

Dans ce chapitre, après avoir fait une revue des problèmes liés à l'atomicité des données en traitement automatique des langues, nous avons examiné une tâche précise au cours d'une expérience préliminaire. Pour la tâche d'évaluation automatique de la traduction automatique, notre étude a montré que l'utilisation d'un atome plus petit que le mot, en l'occurrence le caractère, permettait de contourner les problèmes de segmentation du texte à traiter tout en produisant des résultats corrélés à ceux obtenus en unité de mot. Ces résultats encourageants justifient donc une étude plus large sur d'autres applications en traitement automatique des langues.

[56] En ce qui concerne une transposition de la méthode NIST, voir les réserves exprimées en conclusion de l'annexe B.

Partie II

Traitement de données

Introduction

Nous avons vu dans la partie I que les applications du traitement automatique des langues sont diverses, et qu'elles mettent en relation des compétences pluridisciplinaires. À travers une classification des méthodes utilisées en traitement automatique des langues, nous avons mis en évidence des différences méthodologiques importantes au sein même des méthodes fondées sur les données : ces différences tiennent notamment aux divers degrés de prétraitement qui sont appliqués aux données utilisées.

Les méthodes par modèles de Markov, les méthodes classificatoires, ainsi que les méthodes statistiques nécessitent toutes des prétraitements importants sur les données avant d'être mises en œuvre : compilation pour les modèles de Markov, extraction de traits pour les méthodes classificatoires, et phase d'apprentissage pour les méthodes statistiques. Une idée répandue est que les méthodes fondées sur les données sont moins couteuses en temps et en travail humain que les méthodes fondées sur la connaissance, puisqu'elles visent à être entièrement automatiques, et non supervisées. Pourtant, bien qu'il soit vrai que ces méthodes soient moins couteuses en intervention humaine lors de l'exécution, les prétraitements qu'elles nécessitent sont en revanche extrêmement couteux. En premier lieu, ces prétraitements nécessitent une découpe en unités textuelles de base. Nous avons montré que la nécessité d'une telle découpe engendre plusieurs problèmes méthodologiques, et avons proposé l'utilisation d'une autre unité plus petite afin de les contourner : le caractère. Dans une expérience préliminaire portant sur l'évaluation automatique de la traduction automatique, nous avons montré l'intérêt de l'utilisation du caractère qui, parce qu'elle produit des résultats comparables à ceux obtenus avec l'unité traditionnelle du mot, élimine donc la nécessité de prétraiter les données.

Nous proposons dans cette partie d'appliquer l'unité de caractère au traitement automatique des données linguistiques. L'étude préliminaire en évaluation de la traduction automatique présentant des résultats prometteurs, nous essayons d'élargir nos résultats prometteurs à d'autres tâches. Nous examinons donc deux autres tâches de traitement des données linguistiques : le filtrage de la grammaticalité, et la caractérisation automatique de données linguistiques.

Ces deux applications utilisent l'unité de caractère dans le cadre des méthodes N-grammes. À cet effet, nous faisons tout d'abord en annexe C une introduction et des rappels en théorie de l'information appliquée au traitement des langues : nous exposons ainsi l'intérêt que peut avoir une modélisation statistique en traitement automatique des langues, et clarifions des méthodes souvent utilisées mais parfois mal comprises. Ensuite, nous étudions l'utilisation de l'unité de caractère dans le cadre du filtrage automatique de grammaticalité : nous montrons que des techniques simples fondées sur l'unité de caractère permettent d'arriver à des performances satisfaisantes, tout en réduisant le problème de la rareté des données et en permettant l'application de la technique indépendamment de la langue considérée, sans nécessité de prétraitement. Les résultats de cette étude sont appliqués par la suite, dans le cadre de la génération automatique de données linguistiques particulières, des paraphrases (voir partie III, chapitre 1).

Enfin, nous nous intéressons à un problème souvent négligé alors même que la tendance actuelle en traitement automatique des langues est d'utiliser intensivement de grandes quantités de données : la caractérisation automatique et multilingue de données linguistiques. Nous montrons que l'application de techniques en unité de

caractère permet le profilage rapide de grandes quantités de données sur des langues comme l'anglais ou le japonais. Nous définissons une mesure de similarité des ensembles de données textuelles, que nous comparons à des mesures déjà existantes et qui opèrent en mots. Nous montrons ainsi qu'en plus d'être équivalente en terme de performance, la mesure proposée a l'avantage d'être applicable à toute langue sans nécessité de prétraitement, à la différence des autres techniques, qui nécessitent une segmentation préalable des textes à comparer en mots ou en lexèmes. Cette approche de la quantification de la similarité de ressources textuelles est étendue à celle de l'homogénéité interne des grandes bases de données. Nous étudions en dernier lieu l'influence de l'homogénéité de telles ressources sur la performance de plusieurs systèmes de traitement automatique des langues fondés sur les données.

Chapitre 1

Filtrage de la grammaticalité

1.1 Introduction au problème du filtrage de la grammaticalité

Grace au développement d'Internet, il est devenu de plus en plus facile de collecter de grandes ressources linguistiques à partir du Web, de forums et de groupes de discussion. La collecte peut même être automatisée : un logiciel robot pourra collecter et sauvegarder des milliers de pages de données de façon partiellement supervisée ou complètement automatisée. Ces données devront par la suite être nettoyées de leurs balises XML, standardisées au niveau du codage, etc. Une telle collecte s'apparente à de l'extraction.

Des données de la langue peuvent néanmoins être produites sans avoir recours à une simple extraction. Les données peuvent en effet être générées de diverses manières et à différentes fins : par patrons, par l'intermédiaire d'une représentation intermédiaire ; pour un dialogue homme-machine, une mise en forme de l'information[1] ou la constitution d'une ressource de paraphrases telle qu'elle sera effectuée par la suite dans ce mémoire dans la partie III.

Cependant, un écueil sous-jacent à cette constitution automatique ou semi-automatique de données est que la ressource construite n'est jamais dénuée d'erreurs à 100%. Erreurs de codage, du choix de la langue, balises XML orphelines, ou plus simplement, erreurs dans la grammaticalité des donnés telles qu'elles ont été écrites par l'utilisateur dont elles proviennent, dans le cas de la collecte automatisée sur le Web, ou erreurs d'un processus de génération a priori imparfait.

Avant même de penser à corriger de telles erreurs, il est important de savoir les détecter. Si on émet l'hypothèse que les données sont dépourvues d'erreurs d'encodage ou d'incohérences flagrantes au niveau, par exemple de la langue utilisée, on peut assimiler cette tâche de détection à une tâche de détection de grammaticalité[2].

Nous allons montrer que pour cette tâche, des techniques simples utilisant une découpe en caractères peuvent amener rapidement à de bons résultats. La méthode à utiliser dépend grandement de la qualité finale de la détection, face à la quantité de données retenues : en somme, tout dépendra de l'application visée et de ses exigences

[1] On pensera particulièrement à des systèmes qui compilent de grandes quantités de donnés numériques et produisent un diagnostic financier, ou médical sous une forme facilement compréhensible par un être humain. On pourra voir par exemple Hallett & Scott, *Structural variation in generated health reports*, 2005 pour une illustration dans le contexte médical.

[2] Et même pour être rigoureux, de détection d'agrammaticalité.

(son cahier des charges) en termes de qualité des données linguistiques.

Dans le cadre de ce mémoire, nous présentons deux approches à une telle détection, fondées sur des traitement en chaines de caractères.

Une première approche pour la détection de grammaticalité dans des données linguistiques consiste à utiliser des éléments de modélisation des langues tels qu'exposés en annexe C, afin de caractériser les données, sous leur forme originale ou éventuellement transformées, par un certain nombre de traits. Ces traits ainsi définis sont utilisés dans le cadre d'une méthode d'apprentissage supervisée : les machines à vecteurs-supports, ou SVM[3] en abrégé. Après une phase d'apprentissage supervisée, l'utilisation de ces vecteurs de traits pourra permettre de détecter si une phrase est ou pas grammaticale.

L'autre approche que nous présentons se veut à la fois plus simple et plus extrême : nous montrons que le dénombrement de chaines de caractères attestées dans une ressource extérieure peut servir à obtenir une détection très précise au détriment du rappel.

En résumé, les deux approches sont très différentes en terme de complexité et de temps de mise en œuvre, mais elles ont pour point commun l'utilisation probante de méthodes fondées sur le traitement en caractères. Nous présentons successivement ces deux méthodes, et en comparons les performances respectives.

1.2 Approches pour la détection

1.2.1 Détection par machines à vecteurs-supports, fondée sur des résultats de modélisation du langage

Introduction à la méthode par SVM

Les machines à vecteurs-supports sont une classe d'algorithmes d'apprentissage : le principe général est qu'à partir de données choisies par l'utilisateur et présentées sous la forme d'un vecteur de traits, on construit un classificateur. Ces vecteurs de traits exemples sont transformés, de façon à trouver la séparation optimale (c'est-à-dire minimisant l'erreur de classification).

Pour ce faire, les exemples sont d'abord transformées en vecteurs d'un espace F (de dimension non nécessairement finie). L'algorithme détermine alors un hyperplan qui sépare les données le mieux possible, en minimisant les erreurs de classification. Il est donc nécessaire de disposer d'une suite d'exemples représentatifs du problème de classification qu'on espère traiter. Dans ce sens, le classification des exemples de départ étant effectuée à la main, à l'extérieur du système, on dira des machines à vecteurs-supports qu'elles sont une classe d'algorithmes d'apprentissage supervisé.

Il est nécessaire de structurer le texte autour de la notion de traits : il faut donc s'intéresser à la transformation d'un exemple en un vecteur de traits caractéristiques. Dans notre cas, nous cherchons à identifier la valeur d'une variable X liée à chacune des chaines de caractères examinées et qui constituent une ressource linguistique : $X = 1$ si la chaine constitue une phrase de la langue, grammaticalement correcte, ou $X = 0$ dans le cas contraire. On cherche donc des traits dont le calcul peut être effectué de façon automatique, et qui permettront au classificateur d'établir la séparation la plus nette possible grace à l'information qu'ils contiennent.

[3] « Support Vector Machines » en langue anglaise.

Utilisation de l'entropie croisée en N-grammes de caractères comme trait caractéristique

Dans le cas où l'on dispose à l'origine d'une ressource linguistique dont la qualité est connue, il peut être intéressant d'évaluer l'entropie croisée d'une chaine de caractères sur un modèle de langage construit sur cette même ressource. Intuitivement, on espère qu'un modèle construit sur des données dont la qualité grammaticale est généralement bonne produira des entropies croisées faibles lorsque la chaine jugée sera elle même grammaticalement correcte, et plus élevées en revanche lorsque la chaine jugée ne sera pas grammaticale.

En notant $s_i = \{c_1^i..c_{|s_i|}^i\}$ une phrase de $|s_i|$ caractères, nous pouvons reformuler l'entropie croisée $H_T(A)$ d'un modèle N-gramme p construit sur un corpus d'entrainement T, sur un corpus de test $A = \{s_1, .., s_Q\}$ de Q phrases, sous la forme suivante :

$$H_T(A) = \frac{-\sum_{i=1}^{Q}[\sum_{j=1}^{|s_i|}\log p_j^i]}{\sum_{i=1}^{Q}|s_i|} \tag{1.1}$$

où $p_j^i = p(c_j^i|c_{j-N+1}^i..c_{j-1}^i)$.

Si un corpus de test A se limite à une seule chaine $A = s_0 = \{c_1^0..c_{|s_0|}^0\}$ à juger, on calcule donc son entropie en N-grammes pour différentes valeurs de N, sur le modèle p construit à partir de T.

Cependant, afin de rajouter de l'information supplémentaire, nous allons nous intéresser de surcroit aux entropies croisées de données transformées, calculées sur des modèles construits à partir de données elle-même transformées.

Nous exposons dans la section suivante les deux transformations qui ont été retenues.

Transformation des données

Repli des chaines de caractères L'idée intuitive est de saisir des dépendances entre parties éloignées dans la chaine, et qu'un modèle N-gramme construit sur des données non-transformées ne serait pas en mesure de saisir, parce que le contexte N n'est pas assez grand. Nous effectuons donc une transformation simple, et réversible : la chaine de caractères est « pliée » à mi-longueur, afin que les caractères soient entremêlés de la manière suivante.

Si l'on note C une chaine de L caractères telle que $C = c_1c_2...c_{L-1}c_L$, alors sa transformée repliée C' sera :

$$C' = \begin{cases} c_1c_Lc_2c_{L-1}...c_{\frac{L}{2}} & \text{si } y \text{ est pair.} \\ c_1c_Lc_2c_{L-1}...c_{\frac{L+1}{2}} & \text{si } y \text{ est impair.} \end{cases}$$

Cela donne en pratique, par exemple sur une phrase en langue anglaise :

`Can␣I␣have␣your␣room␣number,␣please?`

→

`C?aens␣aIe␣lhpa␣v,er␣eybomuurn␣␣␣rmoo`

L'exemple exposé ci-dessus montre le fait qu'on espère saisir des dépendances à longue distance dans la phrase, telles que « Can [..], please ? » ou « Could [..], please ? » en langue anglaise (une phrase commençant par un auxiliaire modal est en effet souvent une interrogative).

Transformation de Burrows-Wheeler La transformation de Burrows-Wheeler[4] (qu'on abrégera en TBW) est une transformation réversible qui réarrange l'ordre des caractères d'une chaine. Elle est utilisée couramment dans le domaine de la compression de données sans perte, conjuguée à d'autres algorithmes. La transformation en elle-même réordonne et regroupe les caractères semblables d'une chaine dans un ordre qui empiriquement produit de meilleurs résultats de compression par des algorithmes simples, fondés sur des modèles de langage en unité de caractère par exemple. La transformation a tendance à regrouper les caractères semblables dans la chaine, elle est en pratique souvent utilisée en préparation à une compression de données sans perte.

La transformation de Burrows-Wheeler n'est pas une transformation intuitive. On classe toutes les rotations de la même chaine (il y en a autant que de caractères dans la chaine en question), puis on produit la dernière colonne du tableau formé par toutes les rotations successives.

Si on reprend l'exemple de la phrase précédente, cela donne :

`Can␣I␣have␣your␣room␣number,␣please?`

→

`nIm,rere?␣Cehmvslb␣poua␣ory␣ue␣anoa`

Afin de disposer d'un exemple visuellement plus démonstratif, examinons la phrase suivante, constituée de mots commençant tous par *th* :

`The␣thoughtless␣theocrat␣thwarted␣that␣theory.`

→

`dstetywhroehthhlutTtttgtteehcaoseaar␣␣␣␣␣hohr.`

On remarque en effet une tendance des caractères à se regrouper en suites de caractères semblables (les 5 espaces sont par exemple tous regroupés en une séquence contigüe).

Vecteur de traits

Comme on l'a exposé plus haut, toute phrase représentée par une chaine de caractères est modélisée par un vecteur de traits caractéristiques, qui va servir lors de la classification par la machine à vecteurs supports. Chaque vecteur contient :

- les valeurs des entropies croisées E_{o_i} de la chaine calculées sur un modèle de langage construit à partir de la ressource linguistique originale, l'ordre i allant de 2 à L, taille maximum du contexte.
- les valeurs des entropies croisées E_{o_i} de la chaine repliée, calculées sur un modèle de langage construit à partir de la ressource linguistique repliée (dont toutes les chaines ont été repliées), l'ordre i allant là encore de 2 à L.
- les valeurs des entropies croisées E_{o_i} de la transformation de Burrows-Wheeler, calculées sur un modèle de langage construit à partir de la ressource linguistique elle aussi formée de toutes les transformation de Burrows-Wheeler des chaines de la ressource, l'ordre i allant là encore de 2 à L.

[4] Burrows & Wheeler, *A block-sorting lossless data compression algorithm*, 1994

- la taille de la chaine en caractères, S.
- le nombre de caractères différents apparaissant dans la chaine, N.

Un vecteur de traits peut donc s'écrire sous la forme suivante :

$$(E_{o_2} \cdots E_{o_L}\ E_{s_2} \cdots E_{s_L}\ E_{bwt_2} \cdots E_{bwt_L}\ S\ N) \tag{1.2}$$

où L est la taille maximum du contexte considéré. Cette taille est, pour des raisons matérielles, fixée à 15 pour des systèmes d'écriture codant un caractère sur un octet (tels que le codage ASCII) et à 10 pour des systèmes codant un caractère sur deux octets (par exemple pour les systèmes destinés à coder le japonais ou le chinois).

Expériences

Données Nos expériences ont été menées sur le corpus multilingue BTEC en anglais, japonais et chinois : 3 600 nouvelles phrases ont été produites automatiquement à partir de la ressource selon la méthode expliquée en partie III, chapitre 1 ; elles sont relues par des locuteurs de naissance de la langue correspondante, qui attribuent à chacune la mention « correcte » ou « erronée ». Afin de minimiser les erreurs de jugement, des règles précises et laissant le moins de part possible aux ambiguïtés[5] sont données aux relecteurs. Le tableau 1.1 montre la proportion de phrases signalées comme « erronées » dans les données brutes, 3 600 phrases dans chaque langue.

Tableau 1.1: Total des phrases automatiquement produites qui sont signalées comme erronées.

Langue	japonais	chinois	anglais
Phrases	2 794	2 957	3 044
Proportion	77,61%	82,14%	84,56%

Pour construire un SVM, on doit disposer de données exemples qui vont implicitement caractériser les données erronées, ainsi que les données correctes. On sépare donc de façon aléatoire les 3 600 phrases en deux sous-ensembles : l'un est constitué de 3 000 phrases dont on extrait manuellement toutes les phrases erronées, afin d'entrainer le système ; l'autre constitué des 600 phrases restantes, est utilisé pour évaluer la performance du système. Le tableau 1.2 montre la proportion de phrases erronées dans chacun des deux sous-ensembles de données.

Tableau 1.2: Nombre et proportion de phrases erronées dans les sous-ensembles 1 (3 000 phrases) et 2 (600 phrases).

Langue	japonais	chinois	anglais
Sous-ensemble 1	2 378(79,27%)	2 469(82,30%)	2 530(84,33%)
Sous-ensemble 2	416(69,33%)	531(88,50%)	514(85,67%)

[5]Les ambiguïtés sont principalement liées à la ponctuation : des phrases déclaratives peuvent couramment être utilisés comme interrogatives dans un contexte oral, tout simplement en remplaçant le point final par un point d'interrogation. Nous souhaitons cependant demeurer autant que possible dans le registre de la langue écrite.

Afin de disposer de phrases exemples correctes en quantité équilibrée pour construire le SVM, on laisse de côté 3 000 phrases du BTEC, qui ne seront pas utilisées pour l'entrainement des modèles de langage. L'entrainement des modèles se fait donc sur une ressource de 159 318 phrases seulement.

Le caractère arbitraire du choix des seuils pour la taille du contexte des modèles de langage peut légitimement être critiqué, mais nous justifions ce choix par les chiffres suivants. En moyenne, on dénombre dans les données utilisées :

- $5,24$ caractères par mot dans le BTEC anglais. Un contexte de taille 15 recouvre donc en moyenne approximativement 3 mots anglais[6].
- $4,99$ caractères par bunsetsu dans le BTEC japonais. Un contexte de taille 10 recouvre donc en moyenne approximativement 2 bunsetsus.
- 11 caractères par phrase dans le BTEC chinois. Un contexte de taille 10 recouvre donc en moyenne une phrase.

De contextes de taille 15 en anglais et de 10 en japonais ou chinois peuvent donc fournir une information intéressante sur l'organisation syntagmatique de la phrase dans chacune des langues considérées. Il est évident qu'idéalement, on aimerait être en mesure d'utiliser des contextes de taille encore supérieure. Malheureusement, la construction de modèles d'ordre élevé engendre des temps de calcul prohibitifs.

Réglage des paramètres du SVM On désire retenir uniquement le nombre minimal de traits qui conduiront à une classification optimale des phrases à juger : les paramètres influant sur les marges de détection sont simplement réglés de façon dichotomique, l'unique prérequis étant de ne tolérer aucune phrase erronée lors du processus de classification. Dès lors, nous cherchons à maximiser le taux d'acceptation des phrases correctes. Puisque notre objectif est de laisser passer uniquement les phrases correctes, on s'attend à sur-éliminer des données correctes. On privilégie donc ici la précision en phrases correctes au détriment du rappel.

Les réglages sont effectués en filtrant le sous-ensemble 2 décrit auparavant, dans les trois langues considérées. On utilise une descente de gradient : à chaque étape, le trait ajouté qui engendre la meilleure performance globale est retenu[7]. Ce processus de sélection prend fin lorsque l'ajout d'un trait ne fait que dégrader la performance, ou lorsque l'ajout consécutif de trois traits ne l'influence plus.

Le tableau 1.3 montre les taux d'acceptation des phrases correctes (le rappel) en fin d'optimisation, pour les trois langues considérées, sur un ensemble de test de 600 phrases pour chaque langue.

Les réglages ont montré que :

- dans le cas du japonais, 5 traits sont retenus :
  - 8-grammes de données brutes,
  - 3, 4, et 7-grammes de données repliées,
  - taille de la phrase ;

[6] On considère souvent que la taille moyenne du groupe syntagmatique élémentaire en anglais est de 3 mots.

[7] La simplicité de la méthode contraste malheureusement avec la lourdeur des calculs nécessaires à une telle optimisation.

Tableau 1.3: Taux d'acceptation de phrases correctes, et nombre correspondant de traits requis pour atteindre 100% de rejet de phrases erronées.

Langue	japonais	chinois	anglais
Taux d'acceptation	14,13%	23,21%	6,98%
Nombre de traits	5	2	6

- dans le cas du chinois, seuls 2 traits sont retenus :
  - 2-grammes de données brutes,
  - 3-grammes de données repliées ;
- dans le cas de l'anglais, 6 traits sont retenus :
  - 4-grammes de données brutes,
  - 3 et 4-grammes de données repliées,
  - 2, 7 et 8-grammes de données ayant subi une transformation de Burrows-Wheeler.

L'importance attestée de contextes relativement courts dans le cas de modèles construits sur des données repliées confirme notre intuition première : les dépendances globales entre début et fin de phrase sont riches en information permettant de classer les phrases.

La contribution de la transformation de Burrows-Wheeler est limitée dans le cas des langues à ecriture idéographique ou semi-idéographique, le chinois et le japonais. Le fait de réordonner des caractères pour former des séquences de caractères semblables a peu d'effet en chinois, où deux caractères semblables ne sont que rarement répétés dans les phrases courtes des textes dont nous disposons. La transformation de Burrows-Wheeler semble détruire l'information linguistique locale, mais son application apporte des gains significatifs en anglais[8]. D'autres expériences sur l'anglais avec la même procédure indiquent que la performance maximale plafonne à $4,65\%$ environ sans transformation de Burrows-Wheeler.

1.2.2 Détection par chaines de caractères de longueur N

Méthode

Nous proposons maintenant une autre approche de la détection, à la fois plus simple dans son principe, et plus extrême dans son application : l'élimination pure et simple de phrases contenant des séquences de caractères non attestées dans une ressource linguistique extérieure.

Dans cette nouvelle méthode, nous vérifions pour chacune des sous-chaines de caractères de longueur N contenues dans la phrase à juger si elle est attestée dans une ressource linguistique existante. Si une sous-chaine ne l'est pas, alors on considère que la phrase n'est pas grammaticale. Bien entendu, cette hypothèse est en réalité souvent fausse : nous adoptons délibérément un comportement sévère, afin de

[8] On peut remarquer au passage que le BTEC en anglais comporte un ≪ vocabulaire ≫ réduit à une soixantaine de caractères distincts, alors que celui du chinois est constitué de 2 975 caractères idéographiques et de ponctuation distincts !

pouvoir contrôler le couple rappel-précision grace au seul paramètre N. Si par exemple on souhaite obtenir une très grande précision des phrases filtrées au détriment du rappel (beaucoup de phrases grammaticalement correctes seront éliminées par erreur), on peut augmenter N. Inversement, réduire N augmentera le nombre des phrases retenues (le rappel), mais détériorera la précision, des phrases erronées pouvant se glisser parmi les phrases retenues à cause d'un filtrage insuffisant.

Dans une expérience sur un grand nombre de phrases générées automatiquement, nous comparons la qualité en terme de rappel-précision d'une détection effectuée avec plusieurs valeurs de N.

Qualité de la détection

L'expérience exposée dans cette partie reprend en détail la démarche exposée dnas la partie III, section 1.2.4, p. 111, et intitulée *limitation de la combinatoire par des contraintes sur la contigüité des chaines de caractères*. On dispose d'un ensemble de 4 495 266 phrases en langue anglaise produites automatiquement, parmi lesquelles certaines ne sont pas grammaticalement correctes. Un jugement humain sur un échantillon de 400 phrases permet d'estimer que $23,6\%$ d'entre elles sont correctes, avec un taux de confiance de $98,81\%$. Autrement dit environ 1 060 883 des phrases sont correctes sur les 4 495 266 phrases de départ.

Nous appliquons la procédure de détection sur l'intégralité des phrases produites pour plusieurs valeurs de N. Le filtrage retient un certain nombre de phrases candidates, dont la qualité est évaluée manuellement par échantillonage. Une estimation du nombre de phrases correctes est ensuite produite, ce qui permet de calculer la précision, et le rappel de la méthode de détection. Enfin, nous calculons également la F-mesure[9], qui combine précision et rappel en une seule valeur numérique et donne un aperçu de la performance globale de la méthode. Les résultats sont consignés dans le tableau 1.4.

Tableau 1.4: Performances du filtrage de grammaticalité par détection de chaines de caractères de longueur N.

N	7	9	11	13	15	17	20
Nbre de phrases retenues	697 031	296 403	157 002	88 980	53 181	33 453	17 862
Estimation du nbre de phrases correctes	557 625	278 619	152 291	86 311	52 117	32 784	17 863
Rappel	0,53	0,26	0,14	0,08	0,05	0,03	0,02
Précision	0,80	0,94	0,97	0,97	0,98	0,98	0,99
F-mesure	0,63	0,41	0,25	0,15	0,09	0,06	0,03

La valeur de la F-mesure montre que les meilleurs compromis entre le rappel et la précision sont obtenus pour de faibles longueurs de N. L'intérêt d'une telle

[9]La F-mesure est la moyenne harmonique de la précision P et du rappel R:

$$F = \frac{1}{\frac{1}{2}(\frac{1}{P} + \frac{1}{R})} = \frac{2PR}{P+R} \tag{1.3}$$

détection est qu'elle permet d'influencer directement le couple rappel-précision à l'aide du seul paramètre N. Si dans un procédé le rappel n'a que peu d'importance mais si en revanche on a une exigence en terme de précision, on est libre d'augmenter la longueur de chaine N. C'est par exemple le cas dans l'expérience présentée au chapitre 1, où le nombre de phrases générées dans la première phase est très grand, mais où la contrainte est d'obtenir une qualité des productions finales très élevée, au moins égale à celle de la ressource de départ.

Comparaison avec la méthode par machine à vecteurs-supports

Afin de comparer les deux méthodes de détection présentées dans ce chapitre, nous reproduisons la même expérience avec un SVM tel qu'exposé ci-dessus, et optimisé pour l'anglais. On peut influer sur le couple rappel-précision de la détection en modifiant un paramètre deu SVM appelé *facteur de cout* (*cost factor*). Ce facteur représente la tolérance accordée aux erreurs de classification. Nous faisons varier ce facteur, et consignons les résultats dans le tableau 1.5.

Tableau 1.5: Performances du filtrage de grammaticalité par la méthode par machine à vecteurs-supports.

Nbre de phrases retenues	950 470	707 140	498 550	275 216
Estimation du nbre de phrases correctes	578 180	458 830	395 180	254 610
Rappel	0,55	0,43	0,37	0,24
Précision	0,61	0,65	0,79	0,93
F-mesure	0,58	0,52	0,50	0,38

Avec un SVM, nous ne sommes pas en mesure d'obtenir une précision supérieure à 93%. Si l'on compare rappel, précision et F-mesure pour les valeurs correspondantes avec la méthode de détection par chaines de longueur N, on trouve des résultats comparables : 94% de précision avec la détection par chaines pour 93% avec la méthode par SVM ; 26% de rappel avec la détection par chaines pour 24% avec la méthode par SVM ; enfin, une F-mesure à 41% avec la détection par chaines pour 38% avec la méthode par SVM. La meilleure F-mesure est obtenue avec la détection par chaines : $0,63$ contre $0,58$ avec la méthode par SVM.

L'avantage de la méthode de détection par chaines est qu'elle permet matériellement d'atteindre une précision meilleure, même si c'est toujours au détriment du rappel. On la privilégiera donc naturellement dans une application où la précision est prépondérante.

Conclusion

Nous avons présenté dans ce chapitre deux approches de la détection de la grammaticalité, fondées sur des traitement en chaines de caractères. La détection permet de différencier automatiquement des phrases grammaticalement correctes parmi un grand nombre de phrases éventuellement incorrectes.

La première méthode se base sur l'utilisation de mesures d'entropie croisée avec des modèles stochastiques de langue (tels qu'exposés en annexe C), construits sur

des données linguistiques brutes et transformées, au niveau des caractères. Ces valeurs d'entropie croisée servent alors de traits caractéristiques dans le contexte de l'utilisation de machines à vecteurs-supports. L'optimisation d'une telle machine permet d'identifier les traits les plus riches en information discriminante pour les langues anglaise, chinoise, et japonaise.

La deuxième méthode utilise un principe très simple, en privilégiant le taux de rejet des productions incorrectes : les phrases contenant des chaines de caractères de longueur N non attestées dans une ressource linguistique extérieure sont éliminées, le paramètre N servant à régler la sévérité du système en terme de rappel-précision.

Les deux méthodes atteignent des performances similaires, mais la détection par chaines de caractères permet d'atteindre des précisions plus élevées que la méthode par SVM, au détriment bien sûr du rappel. Ces deux approches très différentes montrent l'utilisation possible de méthodes fondées sur le traitement au niveau des caractères pour le traitement automatique des données linguistiques.

Détecter la grammaticalité ou l'agrammaticalité des productions revient à dire si elles sont dans la langue, ou en dehors. Nous allons affiner cette détection grossière dans le chapitre suivant, en cherchant à caractériser des données de la langue en fonction du sous-langage auxquelles elles appartiennent. Nous continuons à appliquer des techniques inspirées de la modélisation statistique des langues pour traiter cette tâche, qui reste relativement peu étudiée dans le domaine. Nous verrons que travailler sur les caractères avec de telles méthodes permet de les appliquer indifféremment aux langues dont le système d'écriture admet une segmentation graphique en mots, et à celles dont le système d'écriture ne le permet pas.

Chapitre 2

Similarité et homogénéité de corpus

2.1 Introduction au problème de la caractérisation de données

Avant propos

Tant en linguistique computationnelle qu'en traduction automatique par l'exemple ou statistique fondée sur l'utilisation de grandes quantités de données, les corpus ne sont pas de simples outils : les ressources linguistiques sont au centre de chaque domaine. Fondamentalement, on utilise des corpus pour leur représentativité : un corpus peut par exemple être représentatif d'un domaine, ou d'un style.

En linguistique computationnelle, un corpus peut par exemple être utilisé pour étudier la fréquence de phénomènes linguistiques, ou pour l'élaboration d'une grammaire ; en traduction automatique par l'exemple, on déduira du corpus des opérations élémentaires et des règles de transfert ; en traduction automatique fondée sur des méthodes statistiques, on déduira du corpus des modèles stochastiques de langue, et des modèles de traduction.

Paradoxalement au fait que les ressources linguistiques sont souvent les matériaux de base dans plusieurs domaines du traitement automatique des langues, relativement peu de travaux ont été produits sur la caractérisation automatique de telles ressources. Pourtant, une telle caractérisation serait utile afin de vérifier la représentativité qu'on prête aux ressources. On aimerait pouvoir préciser et confirmer ou infirmer des intuitions linguistiques a priori, liées par exemple à la méthode de constitution d'une ressource, et à l'origine des documents dont elle est constituée.

D'autre part, toute caractérisation subjective est hasardeuse dès lors que les données sont trop importantes pour être lues par un être humain en un temps raisonnable[1]. Selon les sources ou les modes d'acquisition d'un corpus, on rencontrera des descriptions subjectives comme : *transcriptions de dialogues en contexte informel*, ou *corpus de productions orales spontanées*, ou bien encore *transcriptions de dialogues fortement scénarisés*. De telles étiquettes sont, on en conviendra, de bien peu de secours à l'utilisateur ou au concepteur, qui voudront avoir une idée

[1] Les corpus utilisés en traduction automatique statistique peuvent par exemple comporter des millions de phrases alignées dans plusieurs langues, soit plusieurs dizaines ou centaines de milliers de pages dans chaque langue.

plus fine des caractéristiques spécifiques d'un corpus, et des changements en termes de performances que provoquera l'usage de telle ou telle ressource.

Afin de pouvoir caractériser plus finement les ressources linguistiques dont nous disposons, nous proposons ici une méthode permettant de quantifier la similarité d'un corpus relativement à deux autres corpus de référence, qui sont choisis au préalable.

Travaux passés et motivation

Il y a de nos jours un vaste choix pour le chercheur ou l'utilisateur qui voudrait utiliser une ressource linguistique. Des organismes comme ELRA/ELDA[2] ou LDC[3] par exemple sont spécialisés dans la mise à disposition de ressources ; seules des contraintes financières liées à leur utilisation pourront poser problème. La constitution de corpus a même tendance à devenir une « activité de recherche » comme une autre pour certains groupes. L'avènement du réseau Internet ou des disques optiques à forte capacité a rendu possible la reproduction et la dissémination rapide de quantités considérables de données.

Le problème principal qui se pose n'est donc plus la disponibilité, ni l'accès aux ressources, mais bien la difficulté d'analyse et de compréhension de grandes quantités de données. Dans ce domaine, on recense peu de travaux passés. Cependant, des études ont été effectuées sur ce qu'on rencontre parfois sous le nom de *profilage de corpus* (*corpus profiling* en langue anglaise). Parmi les travaux les plus connus en la matière, on notera ceux de Biber[4], qui étudie les variations en termes de registre entre langue orale et langue écrite dans divers documents, par analyse en composantes principales, dans le but de caractériser des axes généraux[5]. Kilgarriff et Rose[6] se penchent sur les notions de similarité et d'homogénéité des données textuelles, et plaident pour un investissement plus marqué de la recherche en TAL dans les études sur le profilage, faute de quoi il sera de plus en plus difficile à l'avenir de comparer la portée de travaux similaires, basés sur l'utilisation de corpus différents. Ils définissent, construisent et comparent des corpus de similarité connue (*Known Similarity Corpora*, abrégé par *KSC*) à l'aide de mesures de perplexité et de comptages fréquentiels de mots ou de lexèmes, en particulier fondées sur un test de χ^2 qui se révèle être le plus robuste.

Cependant, l'usage de ces KSC requiert que les mots, ou les lexèmes les plus fréquents des deux corpus comparés soient pratiquement les mêmes. Intuitivement, on pourrait croire que cette condition tend à devenir automatiquement vérifiée à mesure qu'on compare des ensembles de données importants. Pourtant, Liebscher[7] montre qu'il n'en est rien en comparant des listes de comptages extraits du très grand ensemble de données que constituent les groupes de discussion Google. Un autre désavantage d'une telle mesure est qu'elle donne certes une idée de la similarité

[2]Voir http://www.elra.info et http://www.elda.org/ .

[3]Voir http://www.ldc.upenn.edu/ .

[4]Biber, *Variation across speech and writing*, 1988 et Biber, *Dimensions in register variation*, 1995.

[5]Axes dont l'interprétation n'est pas toujours intuitive.

[6]Kilgarriff, *Using word frequency lists to measure corpus homogeneity and similarity between corpora*, 1997, Kilgarriff & Rose, *Measures for corpus similarity and homogeneity*, 1998 et Kilgarriff, *Comparing corpora*, 2001

[7]Liebscher, *New corpora, new tests, and new data for frequency-based corpus comparisons*, 2003.

de tel corpus vis-à-vis de tel autre, mais qu'elle ne permet pas d'en comparer, ni d'en ordonner plusieurs (sauf bien sûr dans le cas dégénéré des KSC).

Mesurer la similarité à l'aide de fréquences de mots ou de lexèmes introduit une autre difficulté : ceci présuppose en effet que le mot est une unité bien définie. Nous avons vu que ce n'est bien sûr pas le cas dans beaucoup de langues, en chinois ou en japonais par exemple, où la segmentation en mots est sujette à discussion[8]. Dans l'état, il est donc impossible de mettre en œuvre de telles méthodes basées sur la comparaison de fréquences de mots ou de lexèmes sur le japonais ou le chinois, sans recourir à des prétraitements.

Plutôt que de s'interroger sur la similarité de deux corpus, peut-être serait-il plus intuitif, plus simple, et plus utile de répondre à la question suivante : si on est en mesure de définir une échelle de similarité entre deux corpus de référence A et B, où se placera un troisième corpus C sur cette même échelle ?

Suivant une idée similaire, Biber met en évidence dans ses travaux sept dimensions de variations en comptant des phénomènes linguistiques choisis au préalable dans des documents, et montre que tout document peut se voir affecter un score selon chacune de ces dimensions. Conformément à cette idée, mais avec pour objectif d'éliminer la nécessité de sélectionner manuellement des phénomènes linguistiques, nous nous servirons d'éléments de théorie de l'information pour définir une échelle de similarité entre deux corpus. Sur celle-ci, tout autre corpus pourra se voir affecter un score, que nous appellerons *coefficient de similarité*.

2.2 Une quantification de la similarité

2.2.1 Entropie croisée en N-grammes

Pourquoi utiliser un critère entropique pour aborder des données linguistiques ?

Notre intuition de départ est que les différences entre plusieurs sous-langages au sens de Kittredge[9], c'est-à-dire au sens d'une utilisation de la langue dans un contexte restreint, peuvent être saisies implicitement par des modèles stochastiques de langue. Les mesures entropiques au niveau des caractères ont l'avantage d'être utilisables de façon immédiate et « aveugle » sur tout type de données textuelles électroniques. Leur utilisation, par opposition à des mesures au niveau des mots, élimine donc un biais bien réel dans le cas de langues sans segmentation en mots. La segmentation est introduite artificiellement afin de faciliter le traitement informatique.

Ici, nous considérerons des entropies croisées en N-grammes de caractères, par opposition à des N-grammes de mots. Dunning a montré sur l'identification automatique de langues[10] l'avantage de tels modèles : on arrive rapidement à de bons résultats avec des tailles de corpus relativement réduites. Il montre entre autres l'identification réussie entre anglais et espagnol dans $99,9\%$ des cas, avec 50 kilo-octets de données d'entrainement seulement et sur 500 octets de données de test, ce score pouvant encore être augmenté en augmentant la taille des données de test.

[8] On lui préfèrera souvent, par exemple dans le cas du japonais, le bunsetsu, qui est une unité beaucoup plus facilement accessible car constituée d'un ou plusieurs mots pleins suivis éventuellement d'un ou plusieurs mots vides. Il va de soi qu'il existe proportionnellement un plus grand nombre d'unités de ce genre que de simples mots, ou lexèmes.

[9] Kittredge & Lehrberger, *Sublanguage. Studies of language in restricted semantic domains*, 1982, il est à noter qu'il emploie « sublanguages » dans le texte.

[10] Dunning, *Statistical identification of language*, 1994.

Reprenons la formulation de l'entropie croisée de la page 65. En notant $s_i = \{c_1^i..c_{|s_i|}^i\}$ une phrase de $|s_i|$ caractères, l'entropie croisée $H_T(A)$ d'un modèle N-gramme p construit sur un corpus d'entrainement T, sur un corpus de test $A = \{s_1, .., s_Q\}$ de Q phrases, est :

$$H_T(A) = \frac{-\sum_{i=1}^{Q}[\sum_{j=1}^{|s_i|} \log p_j^i]}{\sum_{i=1}^{Q} |s_i|} \tag{2.1}$$

où $p_j^i = p(c_j^i | c_{j-N+1}^i..c_{j-1}^i)$.

Nous construisons ici plusieurs modèles N-grammes en caractères sur des corpus de référence, et en estimons l'entropie croisée sur des corpus de test.

2.2.2 Coefficient de similarité

Définition

Considérons deux ressources références T_1 et T_2. Nous souhaitons quantifier la similarité d'un corpus T_3 par rapport à ce couple, et ainsi pouvoir répondre ła question suivante : T_3 est-il plus proche en langue de T_1, ou de T_2 ? Nous allons utiliser des modèles de langue construits sur T_1 et T_2 pour définir entre eux une échelle de similarité, sur laquelle tout corpus T_3 à caractériser se verra affecter un coefficient de similarité. Les modèles stochastiques de langue permetent en effet de saisir des régularités propres aux ressources sur lesquelles ils ont été construits. Les entropies croisées des modèles N-grammes de caractères construits sur T_1 et T_2 sont estimées sur T_3. Nous les notons respectivement $H_{T_1}(T_3)$ et $H_{T_2}(T_3)$, conformément aux notations de l'équation 2.1. Les entropies croisées de chaque référence pour un modèle construit sur elle-même, et sur l'autre référence, sont ensuite à leur tour estimées, c'est-à-dire $H_{T_1}(T_2)$ et $H_{T_1}(T_1)$, ainsi que $H_{T_2}(T_1)$ et $H_{T_2}(T_2)$, de manière à obtenir les poids W_1 et W_2 des références T_1 et T_2 :

$$W_1 = \frac{H_{T_1}(T_3) - H_{T_1}(T_1)}{H_{T_1}(T_2) - H_{T_1}(T_1)} \tag{2.2}$$

et :

$$W_2 = \frac{H_{T_2}(T_3) - H_{T_2}(T_2)}{H_{T_2}(T_1) - H_{T_2}(T_2)} \tag{2.3}$$

Nous considérons que W_1 et W_2 sont les poids du barycentre des références choisies. Nous définissons donc le **coefficient de similarité** entre les ensembles de référence T_1 et T_2 par :

$$I(T_3) = \frac{W_1}{W_1 + W_2} = \frac{1}{1 + \frac{W_2}{W_1}} \tag{2.4}$$

Par définition, il est facile de voir que $I(T_1) = 0$ et $I(T_2) = 1$. Il est aussi facile de voir que tout corpus T_3 se verra assigner un score compris entre $I(T_1) = 0$ et $I(T_2) = 1$. Nous considérons que deux corpus sont similaires lorsque l'un d'entre eux peut être complètement prédit étant donné la connaissance de l'autre (c'est-à-dire, étant donné un modèle de langue construit sur l'autre). Nous étendons cette idée à trois corpus, deux d'entre eux servant de références, le troisième étant l'objet de l'étude.

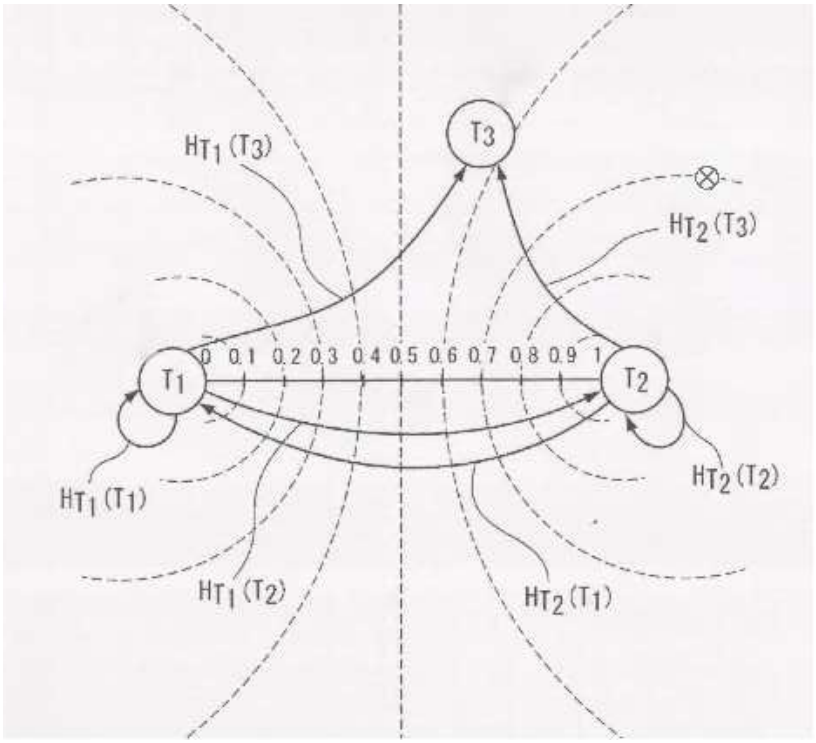

Figure 2.1: Illustration graphique de la méthode.

Étude des cas dégénérés

La figure 2.1 illustre le principe de la méthode. Les lignes en traits pointillés montrent les lieux de coefficients de même valeur (0 indique une plus grande similarité au corpus T_1, 1 au corpus T_2, le $\bigotimes$ est expliqué dans la section 2.2.4). Comme on peut le voir, $I(T_3)$ ne peut prendre que trois valeurs limites $(0, \frac{1}{2}, 1)$ lorsque la taille $n-1$ de la longueur du contexte gauche pris en compte dans le modèle de langue tend vers l'infini. Si B est la valeur en bits qu'il faut pour coder un caractère dans un codage donné, alors on a trois cas :

- cas où T_3 tend à être très similaire à T_2 et très peu similaire à T_1. Avec un long contexte gauche $n-1$, T_3 est entièrement déterminé par un modèle de langue construit sur T_2, mais complètement indéterminé par un modèle de langue construit sur T_1, donc :

$$\lim_{n\to\infty} H_{T_1}(T_3) = B \qquad \lim_{n\to\infty} H_{T_2}(T_3) = 0 \tag{2.5}$$

  T_1 et T_2 sont supposés être non similaires, donc :

$$\lim_{n\to\infty} H_{T_2}(T_1) = \lim_{n\to\infty} H_{T_1}(T_2) = B \tag{2.6}$$

  De la même manière, un modèle utilisé pour prédire le corpus à partir duquel il a été construit donnera une incertitude minimale pour de longs contextes gauches :

$$\lim_{n\to\infty} H_{T_1}(T_1) = \lim_{n\to\infty} H_{T_2}(T_2) = 0 \tag{2.7}$$

Nous pouvons donc simplifier les expressions de W_1 et W_2 :

$$\lim_{n\to\infty} W_1 = \lim_{n\to\infty} \frac{H_{T_1}(T_3) - H_{T_1}(T_1)}{H_{T_1}(T_2) - H_{T_1}(T_1)} = \frac{B-0}{B-0} = \frac{B}{B} = 1 \tag{2.8}$$

$$\lim_{n\to\infty} W_2 = \lim_{n\to\infty} \frac{H_{T_2}(T_3) - H_{T_2}(T_2)}{H_{T_2}(T_1) - H_{T_2}(T_2)} = \frac{0-0}{B-0} = 0 \tag{2.9}$$

Par conséquent, si la taille $n-1$ du contexte gauche tend vers l'infini, alors :

$$\lim_{n\to\infty} I(T_3) = 1 \tag{2.10}$$

- cas opposé, où T_3 tend à être très similaire à T_1 et très peu similaire à T_2. Avec un long contexte gauche $n-1$, T_3 est entièrement déterminé par un modèle de langue construit sur T_1, mais complètement indéterminé pour un modèle de langue construit sur T_2, donc :

$$\lim_{n\to\infty} W_1 = 0 \qquad \lim_{n\to\infty} W_2 = 1 \qquad \lim_{n\to\infty} I(T_3) = 0 \tag{2.11}$$

- cas où T_3 tend à être aussi similaire à T_2 qu'à T_1 :

$$\lim_{n\to\infty} H_{T_1}(T_3) = \lim_{n\to\infty} H_{T_2}(T_3) = H \tag{2.12}$$

Et donc :

$$\lim_{n\to\infty} W_1 = \frac{H}{B} = \lim_{n\to\infty} W_2 \tag{2.13}$$

$$\lim_{n\to\infty} I(T_3) = \frac{1}{1 + \frac{H/B}{H/B}} = \frac{1}{1+1} = \frac{1}{2} \tag{2.14}$$

Nous venons de montrer ici que I tend à ne prendre qu'une des trois valeurs $(0, \frac{1}{2}, 1)$ lorsque la taille du contexte gauche tend vers l'infini.

2.2.3 Comparaison avec d'autres mesures de similarité

Motivation

Un certain nombre de mesures de similarité ont été étudiées dans des travaux passés. Ces mesures se servent de comptages de phénomènes linguistiques donnés, et mettent en relation des listes d'occurrences de mots ou de lexèmes.

Nous souhaitons comparer la méthode proposée plus haut avec deux mesures existantes, qui se basent sur le comptage de phénomènes linguistiques : χ^2 et le logarithme de la vraisemblance[11]. Ces mesures ont déjà été appliquées à la comparaison de corpus en langue anglaise. Elles sont symétriques, et comparent un document à un autre via une liste de fréquences de mots ou de lexèmes. La valeur qu'elles produisent est interprétée comme une mesure de la distance inter-documents.

[11] En langue anglaise, « Log-likelihood ».

Détails d'autres approches de la similarité : χ^2 et logarithme de la vraisemblance G^2

Pour la mesure du χ^2, Kilgarriff[12], formule l'hypothèse que les documents à comparer ont été extraits d'un même ensemble plus important, ce qui tient lieu d'hypothèse nulle pour le test. Le test du χ^2 permet de comparer les distributions issues d'une même population qu'on suppose parente, et de confirmer ou d'infirmer l'hypothèse nulle.

Plus la valeur de la statistique est faible, plus les distributions des deux documents seront proches, ce qui indique une plus grande similarité entre les deux documents. Inversement, la valeur du χ^2 augmente avec la dissimilarité des documents. À partir des listes de fréquences de mots ou de lexèmes extraits de chaque document, on calcule pour chaque élément le nombre d'occurrences qu'on s'attend à constater si l'hypothèse nulle était vérifiée, c'est-à-dire s'ils avait été extraits d'une même population.

Si on note les tailles des documents A et B respectivement N_A et N_B, et que le mot ou lexème w a été observé à une fréquence de $o_{w,A}$ dans A et de $o_{w,B}$ dans B, on fait alors l'hypothèse que la moyenne sur A est égale à la moyenne sur $(A+B)$. La valeur attendue $e_{w,A}$ est alors :

$$e_{w,A} = \frac{N_A(o_{w,A} + o_{w,B})}{N_A + N_B} \tag{2.15}$$

et on procède de même avec $e_{w,B}$ pour le document B. Le χ^2 consiste à calculer un écart constaté entre les observations o_i et les fréquences qu'on s'attendait à constater pour la paire de documents A and B. Le χ^2 est défini comme la somme sur tous les mots ou lexèmes w, et sur les documents A et B, de ces écarts au carré pondérés par la fréquence attendue. Son calcul est donc :

$$\chi^2 = \sum_w \frac{(o_{w,A} - e_{w,A})^2}{e_{w,A}} + \sum_w \frac{(o_{w,B} - e_{w,B})^2}{e_{w,B}} \tag{2.16}$$

la somme se faisant sur tous les éléments w des listes comparées.

La mesure du logarithme de vraisemblance au sens de Dunning[13], aussi appelé G^2, est une meilleure approximation d'une distribution binômiale que le χ^2 pour les éléments à faible fréquence d'apparition. Elle est plus robuste aux variations de taille des documents, et permet de mieux prendre en compte la comparaison des événements peu fréquents. La formule de G^2 est obtenue en prenant le logarithme de la fonction de vraisemblance d'une distribution binomiale ou multinomiale.

Tableau 2.1: Tableau de contingence pour un mot ou lexème w dans des documents A et B.

	Doc.A	Doc.B
w	a	b
$\neg\, w$	c	d

Dans le cas d'un tableau de contingence comme celui du tableau 2.1, où a,b,c et d sont associés à tout élément w de la liste des éléments à comparer (là encore, mots ou lexèmes), le logarithme de vraisemblance G^2_w d'un élément w est défini comme :

[12] Kilgarriff, *Comparing corpora*, 2001.

[13] Dunning, *Accurate methods for the statistics of surprise and coincidence*, 1993.

$$
\begin{aligned}
G_w^2 = 2(&a\log(a) + b\log(b) + c\log(c) + d\log(d) \\
&- (a+b)\log(a+b) - (a+c)\log(a+c) \\
&- (b+d)\log(b+d) - (c+d)\log(c+d) \\
&+ (a+b+c+d)\log(a+b+c+d))
\end{aligned} \tag{2.17}
$$

G^2 est alors la somme des logarithmes de vraisemblance G_w^2 des n éléments w :

$$
G^2 = \sum_{i=1}^{n} G_i^2 \tag{2.18}
$$

χ^2 et G^2 donnent des valeurs qui sont interprétées comme des mesures de la distance entre deux documents. Ces distances peuvent à leur tour être transposées à notre méthode, afin de définir des coefficients de similarité fondés sur χ^2 et G^2 (c'est-à-dire qu'on remplace la distance entropique de notre méthode par les mesures de χ^2 ou G^2).

Evaluation

S'il est relativement aisé de proposer une mesure de similarité, il est en revanche plus difficile de déterminer si elle est précise, et appropriée. Il serait utile en vue d'une évaluation, de disposer d'un ensemble de ressources dont la similarité serait connue à l'avance, par construction. Afin de comparer la méthode que nous proposons avec des mesures fondées sur χ^2 et G^2, nous utilisons la méthode des *Corpus de Similarité Connue* (ou *KSC*, comme mentionné dans la section 2.1) décrite par Kilgarriff[14].

Nous travaillerons sur la langue japonaise, pour laquelle il n'y a pas de segmentation en mots bien définie, afin de pouvoir appliquer notre méthode entropique sans prétraitement. Pour les coefficients de similarité à base de χ^2 et G^2, il faut segmenter les données au préalable. La méthode que nous proposons, basée sur l'entropie croisée en caractères, ne nécessite pas, elle, cette étape de segmentation. Les corpus utilisés dans cette évaluation sont présentés et détaillés à la section 2.2.4 et en annexe A.

La figure 2.2 montre comment construire artificiellement un ensemble de KSC entre un corpus T_1 et un autre T_2 de taille comparable, par tranches de 20% de données. Six corpus de similarité connue sont créés par cette méthode.

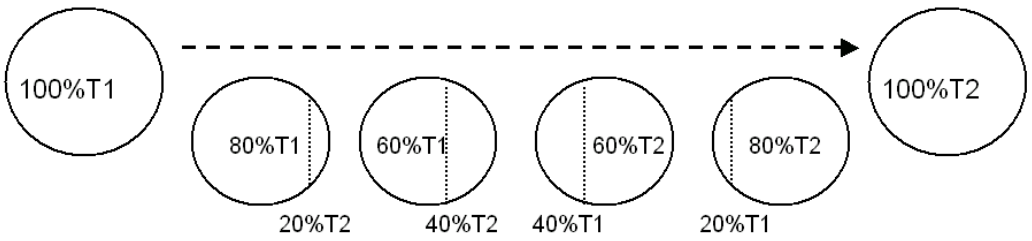

Figure 2.2: Construction d'un ensemble de corpus de similarité connue (KSC).entre deux corpus T_1 et T_2.

[14] Kilgarriff, *Comparing corpora*, 2001.

Pour nos expériences, nous utilisons un certain nombre de corpus décrits en annexe A. Il s'agit de SLDB, BTEC, et Nikkei. Nous construisons 3 ensembles de KSC avec chacun des couples formés par ces trois corpus : des tranches de 10 000 mots (ainsi que leur équivalent en données non-segmentées) sont extraites de chaque corpus et réarrangées de façon aléatoire. De la sorte, chaque ensemble de KSC consiste en des mélanges d'un couple de corpus.

Par exemple, l'ensemble de KSC de (SLDB,BTEC) comporte un sous-ensemble $s10b0$ constitué de 10 tranches de SLDB et aucune tranche de BTEC (100% SLDB, 0% BTEC), un sous-ensemble $s9b1$ constitué de 9 tranches de SLDB et une tranche de BTEC (90% SLDB, 10% BTEC), et ainsi de suite. Chaque sous-ensemble est constitué de 10 tranches et a donc une taille de 100 000 mots, sur lesquels nous pouvons produire un certain nombre de jugements de bon sens du type :

> « Le coefficient de $s10b0$ doit être plus faible que celui de $s9b1$ car toutes les données le constituant proviennent de SLDB »[15].

Chaque ensemble de KSC est donc constitué de 11 sous-ensembles de 100 000 mots, ou leur équivalent en données non segmentées. L'équivalent de 500 000 mots est laissé de côté pour être utilisé comme référence pour les échelles de similarité. Comme dans l'expérience présentée par Cavaglià[16], les listes d'occurrences de mots comparées par χ^2 et G^2 prennent en compte les 500 lexèmes les plus fréquents (Cavaglià a montré dans ses travaux des résultats optimaux pour des longueurs comprises entre 320 et 640 mots ou lexèmes).

Nous appliquons ensuite les différents coefficients de similarité aux ensembles de KSC construits. Un score compris entre 0 et 1 est affecté à chacun des sous-ensembles. Ceci permet de les classer et de les confronter aux jugements de bon sens. Nous utilisons deux méthodes classiques pour comparer le classement idéal et les classements expérimentaux : le calcul de coefficients Kappa, et le calcul de la corrélation de Spearman.

Les résultats sont consignés dans le tableau 2.2.

Tableau 2.2: Coefficients Kappa (10 intervalles) et corrélation de Spearman des rangs produits par les coefficients de similarité fondés sur l'entropie croisée en caractères, le χ^2 et le G^2, comparés aux rangs idéaux.

Kappa	$I_{Entropie}$	I_{χ^2}	I_{G^2}
SLDB-BTEC	0.5	0.7	0.8
SLDB-Nikkei	0.9	0.7	0.7
BTEC-Nikkei	0.6	0.9	0.9

Spearman	$I_{Entropie}$	I_{χ^2}	I_{G^2}
SLDB-BTEC	0.918	0.973	0.990
SLDB-Nikkei	1.000	0.936	0.990
BTEC-Nikkei	0.982	1.000	1.000

[15] Dans le cas où l'on suppose que des données plus similaires à SLDB auront des coefficients faibles, et des données plus similaires à BTEC, des coefficients forts.

[16] Cavaglià, *Measuring corpus homogeneity using a range of measures for inter-document distance*, 2002.

Discussion

Comme mentionné brièvement dans la section 2.1, il est important de noter que la méthode de comparaison par classement de KSC a ses limites : le fait de créer artificiellement les KSC fait qu'on a tendance à comparer des mélanges de différents registres/sous-langages plutôt que des registres/sous-langages différents. D'autre part, il est possible que la taille des tranches de données soit trop faible pour permettre une comparaison valable. En effet, il est possible qu'un corpus utilisé pour construire un ensemble de KSC comporte des parties non homogènes[17]. Cependant, les trois mesures considérées ici ont une forte corrélation avec les jugements de bon sens, ce qui confirme leur validité en tant qu'indicateurs de la similarité, tout du moins s'il s'agit de mesurer des mélanges de différentes variétés de langage.

Les meilleurs scores varient en fonction de l'ensemble de KSC considéré, ce qui ne permet pas de montrer qu'une des mesures est supérieure aux deux autres.

Les méthodes utilisant χ^2 et G^2 fonctionnent uniquement en mots, alors que la méthode entropique que nous proposons fonctionne sur les caractères. On ne peut donc appliquer les méthodes utilisant χ^2 et G^2 sur des données japonaises brutes, elles doivent être segmentées auparavant. En revanche, la méthode entropique que nous proposons peut être appliquée directement au japonais sans segmentation.

D'autre part, nous avons montré que les performances des trois méthodes étaient comparables. En pratique, notre méthode est plus intéressante, puisqu'elle peut être appliquée directement, sans prétraitement. Son domaine d'application est beaucoup plus grand : on pourrait par exemple l'appliquer sur des langues comme le lao ou le thaï, pour lesquelles il existe peu, ou pas de segmenteurs disponibles.

2.2.4 Expérience : une quantification de la littérarité d'un corpus

Motivation

Dans cette section, nous allons étudier l'application de la méthode proposée à la quantification de la « littérarité » dans un corpus. Nous formons le terme *littérarité* à partir de l'adjectif *littéraire* et le définissons comme suit : la littérarité désigne le caractère littéraire d'un texte, par opposition à son caractère oral. Dans la suite, de la même manière qu'on opposera l'oral à l'écrit, nous opposerons l'oralité à la littérarité d'un texte.

Notre objectif est de montrer qu'il est possible de retrouver de façon automatique des résultats de bon sens en utilisant la méthode de quantification de la similarité proposée ci-dessus. D'autre part, alors que caractériser les données par leur source est à notre sens le plus bel exemple possible de caractérisation en amont, nous proposons ici une approche globale, située en aval, de la caractérisation de données textuellles.

Remarques préliminaires sur l'homogénéité des ressources considérées

À ce stade de notre étude, nous ne disposons d'aucune certitude en ce qui concerne l'homogénéité de corpus. En effet, nous n'avons pas connaissance de travaux proposant une définition concrète et précise de ce que pourrait être ou représenter l'homogénéité de données textuelles. Il pourrait par exemple sembler stérile de tenter une comparaison de corpus de tailles très différentes, puisqu'a priori on n'a aucune

[17] Il est possible par exemple que dans un corpus d'articles de journaux on ait sélectionné par hasard un supplément économie, donnée qu'on pourra qualifier d'hétérogène face au reste du corpus.

idée de l'influence de la taille d'un corpus sur son homogénéité globale, ni même de l'influence de l'homogénéité d'un corpus sur les performances d'un système de traitement automatique des langues fondé sur l'utilisation de tels corpus.

Nous pouvons cependant proposer un certain nombre d'hypothèses à ce sujet. Un corpus peut être constitué de données textuelles provenant de sources très différentes : le bon sens voudrait qu'un corpus constitué d'un recueil de documentations logicielles soit plus homogène qu'un autre constitué d'un mélange de transcriptions téléphoniques et de passages de littérature[18]. Nous reviendrons sur ce problème dans la section 2.3.

Dans une première approche, nous nous assurons que les corpus utilisés dans notre étude proviennent d'un même domaine (par exemple, l'un est un recueil d'articles de journaux uniquement, un autre un recueil de transcriptions téléphoniques uniquement). Nous divisons ensuite les données de référence en n parties choisies au hasard et effectuons alors les calculs d'entropie croisée et moyennons les résultats[19].

Nous étudions par la suite dans la section 2.3 une approche possible de l'homogénéité envisagée à partir de la méthode de quantification de la similarité proposée plus haut.

Données d'entrainement et de test

Nous avons conduit une expérience sur des corpus en anglais et en japonais, afin de vérifier que la méthode a un comportement stable indépendamment de la langue.

Afin de borner notre échelle, nous devons donc choisir des corpus qui représentent au mieux la différence que nous nous faisons de la langue « parlée » (spontanée), et de la langue « écrite » (peu spontanée, mais qui peut cependant être utilisée dans un contexte oral fortement scénarisé). Pour avoir une amplitude importante, il est préférable de choisir des données provenant de sources très différentes.

- Comme référence de la langue parlée, nous avons utilisé en anglais et en japonais le corpus SLDB (Spontaneous Speech Database), une ressource multilingue de dialogues transcrits[20] ;
- Comme référence de la langue écrite, nous avons utilisé pour l'anglais une partie du corpus Calgary[21] contenant plusieurs œuvres de littérature anglaise contemporaine. Pour le japonais, nous avons utilisé un corpus d'articles du Nikkei Shinbun[22].

Les données ont été choisies dans les deux langues dans l'idée de mesurer la littérarité de deux corpus se situant dans le même domaine mais ayant la réputation,

[18] Kilgarriff & Rose, *Measures for corpus similarity and homogeneity*, 1998.

[19] Ce procédé est très couramment appelé en langue anglaise *n-fold cross-validation*, et est censé éviter que les résultats ne soient un artéfact de données malencontreusement non représentatives de la ressource dans son intégralité. A notre connaissance, le terme a été popularisé dans Charniak, *Statistical language learning*, 1993.

[20] Nakamura *et al.*, *Japanese speech databases for robust speech recognition*, 1996.

[21] Le corpus Calgary est disponible librement sur le réseau Internet par ftp à ftp.cpcs.ucalgary.ca/pub/projects/text.compression.corpus .

Il est couramment utilisé dans le domaine de la compression de données à des fins de comparaison entre algorithmes.

[22] Les œuvres japonaises, libres de droit d'auteur, remontent à plus de 50 ans. Le japonais littéraire a complètement changé après la guerre, il était donc impossible d'utiliser des œuvres littéraires libres de droit. Notre étude se limite donc à un usage contemporain de la langue.

entièrement subjective et non vérifiée, de différer en termes de littérarité. Ces deux ressources multilingues sont le corpus C-STAR BTEC, qui est une sous-partie d'un grand corpus multilingue, le BTEC (Basic Traveler's Expression Corpus), et le corpus MAD (Machine-translation-Aided bilingual spoken Dialogue corpus). Ce sont tous deux des recueils de phrases du domaine du voyage et du tourisme : MAD est un recueil de phrases transcrites de façon réaliste, mais épurées de tout phénomène oral tel que répétition ou redite, et le BTEC est un recueil de phrases provenant de guides de conversation pour voyageurs. Le corpus MAD, qui provient de transcriptions de l'oral, a la réputation d'être légèrement plus « oral » que le BTEC dans son ensemble[23]. C'est précisément cette hyptothèse que nous voulons vérifier.

Afin d'offrir une base de comparaison plus étendue, nous rajouterons aux deux corpus évoqués :

- en anglais, le corpus TIME est un recueil d'articles de journaux du magazine TIME, et le corpus SPAE (Spoken Professional American-English) un recueil de transcriptions de réunions d'affaires dans un contexte professionel ;
- en japonais, le corpus MAINICHI est un recueil d'articles du journal Mainichi Shinbun, et le corpus NHK un recueil de transcriptions de nouvelles diffusées sur le réseau NHK.

La nature des sources de chaque corpus voudrait que l'on retrouve classés du plus oral au plus écrit les corpus anglais :

- MAD (transcriptions de l'oral) ;
- BTEC (livrets de conversation) ;
- SPAE (transcription de réunions de travail) ;
- TIME (articles de journaux).

Similairement, en japonais, nous voudrions retrouver classés les corpus :

- MAD (transcriptions de l'oral) ;
- BTEC (livrets de conversation) ;
- NHK (transcription de bulletins de nouvelles hautement scénarisées) ;
- MAINICHI (articles de journaux).

Nous nous proposons de retrouver expérimentalement cette classification.

Les ressources sont présegmentées en phrases, une phrase par ligne. Dans la lignée de ce qui a déjà été fait en aval de la traduction automatique ou en caractérisation de corpus, nous voulons montrer que les mesures entropiques ont l'avantage de ne nécessiter aucune segmentation à quelque niveau que ce soit pour être appliquées. Du point de vue de l'unité choisie, il va donc de soi que notre étude aurait pu être menée sur des données segmentées en paragraphes, ou même en documents (au sens de groupement de paragraphes), bien que dans ces cas là, la

[23]Les deux ressources étant vues communément comme des ressources à forte teneur orale, le BTEC étant toutefois plus du « pseudo-parlé » que le résultat d'une réelle transcription.

quantité de données nécessaire pour entrainer les modèles de langue aurait dû être plus importante. Afin de limiter cette quantité, les calculs entropiques sont donc effectués ligne par ligne, puis moyennés.

Un résumé des abréviations utilisées dans cette section et au cours de la suite de cette étude, ainsi que des extraits d'entrées types des ressources textuelles présentées, sont disponibles en annexe A.2. Dans la section suivante, nous donnons plusieurs informations statistiques de surface sur les ressources utilisées.

Caractérisation de surface des données utilisées

Par caractérisation de surface, nous entendons des chiffres calculables de façon triviale sur les données considérées. Plusieurs caractéristiques des corpus présentés dans la section 2.2.4 sont consignées dans le tableau 2.3 pour la langue anglaise, et dans le tableau 2.4 pour la langue japonaise.

Tableau 2.3: Caractéristiques numériques de plusieurs corpus en langue anglaise.

Anglais	SLDB	MAD	BTEC	SPAE	TIME	Calgary
Mots/phrase (Moy.)	11,27	9,29	5,94	23,34	23,17	20,21
Mots/phrase (écart-type)	±6,85	±5,83	±3,25	±26,43	±15,32	±15,18
Car./phrase (Moy.)	64,51	44,86	31,15	126,11	131,74	107,70
Car./phrase (écart-type)	±35,95	±27,57	±17,02	±140,71	±92,38	±84,69
Car./mot	5,72	4,83	5,24	5,40	5,68	5,33
Nbre total de car.	1 037K	475K	5 026K	223K	1 515K	757K
Nbre total de mots	181,2K	98,5K	964,2K	41K	264,5K	142,2K
Nbre total de phrases	16 078	10 601	162 318	1 759	11 416	7 035

Tableau 2.4: Caractéristiques numériques de plusieurs corpus en langue japonaise.

Japonais	SLDB	MAD	BTEC	NHK	Mainichi	Nikkei
Car./phrase (Moy.)	32,61	26,87	14,45	65,39	37,73	44,21
Car./phrase (Écart-type)	±22,22	±14,07	±7,12	±39,16	±31,88	±28,34
Nbre total de car.	20 806K	290K	2 426K	2 772K	2 740K	2 772K
Nbre total de phrases	84 751	10 612	162 318	66 512	71 647	253 016

Il est intéressant de relever deux faits en ce qui concerne les données en langue anglaise : trois corpus, SPAE, TIME, et Calgary, ont des valeurs mots/phrase nettement plus importantes que les trois autres, MAD, BTEC, et la référence de la langue parlée SLDB. Les trois premiers, qu'on a supposé relever plus d'un style écrit, ont des phrases de 20 mots en moyenne, alors que les trois autres, qu'on a supposé relever plus d'un style parlé, ont des phrases de 10 mots en moyenne. Cela tend à conforter l'intuition bien connue qu'à l'écrit on a tendance à trouver des phrases plus longues qu'à l'oral.

Les valeurs caractères/mot sont comparables pour toutes les ressources, et ne dénotent aucune différence entre langue parlée et langue écrite[24].

On peut donc dire que seule une mesure du nombre de mots par phrase permettrait de différencier de façon grossière les documents ayant une origine orale de ceux ayant une origine écrite. C'est d'autant moins convaincant que dans le cas d'un corpus comme SPAE, qui est constitué de transcriptions de réunions de travail faites à l'oral, aucune différence ne peut être faite visuellement avec des corpus de documents de la langue écrite[25]. Pourtant, les données d'un tel corpus contiennent implicitement des indications stylistiques qui les font appartenir à la langue parlée. On peut donc conclure que des informations de surface telles que présentées dans le tableau 2.3 ne permettent pas de caractériser la littérarité de ces corpus.

Pour le japonais, puisqu'on n'est pas en mesure de définir une segmentation claire en mots, nous présentons ici uniquement des valeurs portant sur des segmentations immédiates : caractères et phrases. De façon similaire aux corpus anglais, le nombre de caractères par phrase des corpus supposés avoir un contenu plus « écrit » est nettement plus élevé que ceux supposés avoir un contenu plus « oral ». Cependant, de manière analogue au phénomène décrit pour l'anglais SPAE, il est impossible de différencier le corpus de transcriptions de nouvelles parlées NHK des corpus de nouvelles écrites Mainichi et Nikkei au seul vu de ces chiffres.

Calcul des entropies

La mise en place de l'échelle de similarité commence par le calcul des modèles N-grammes sur les corpus de référence. Les calculs sont effectués dans les deux langues pour N allant de 2 à 16. Pour chaque corpus à tester, les entropies croisées sont ensuite calculées sur des blocs de 250 000 caractères environ, choisis au hasard et sans recouvrement, puis moyennées. Les résultats sont consignés sur la figure 2.3 pour l'anglais, et sur la figure 2.4 pour le japonais. Pour la langue anglaise, l'entropie croisée est calculée sur des modèles de langue construits sur la référence de l'oral SLDB (à gauche) et sur la référence de l'écrit Calgary (à droite). Pour la langue japonaise, l'entropie croisée est calculée sur des modèles de langue construits sur la référence de l'oral SLDB (à gauche) et sur la référence de l'écrit Nikkei (à droite).

Les valeurs entropiques les plus faibles sont atteintes pour des valeurs de N comprises entre 4 et 6 (à l'exception de l'entropie croisée d'un modèle mesurée sur son corpus d'entrainement, pour laquelle la performance est optimale), puis les valeurs augmentent à mesure que l'ordre N augmente. En effet, à mesure que l'ordre N augmente, le nombre de N-grammes non vus augmente de façon exponentielle du fait de la rareté des données évoquée dans l'annexe C.2.2. De façon similaire à ce qui est fait dans le domaine de la compression de données, nous considérerons comme Teahan[26] et Dunning[27] qu'une longueur N comprise entre 3 et 7 produit des résultats significatifs.

Si l'on regarde les résultats du corpus SLDB sur un modèle construit sur ce même corpus en données anglaises, l'entropie est minimale. On dit d'un modèle qui est appliqué aux mêmes données qui ont servi à le construire qu'il est optimal.

[24] Elles sont plus une caractéristique intrinsèque de la langue anglaise.

[25] On peut expliquer la longueur des phrases du corpus SPAE par le fait que les phrases prononcées en réunion ont souvent été préparées, au moins informellement sous forme de notes. Le style oratoire se rapproche ainsi plus de l'écrit par la longueur des phrases.

[26] Cleary & Teahan, *Unbounded length contexts for PPM*, 1997.

[27] Dunning, *Statistical identification of language*, 1994.

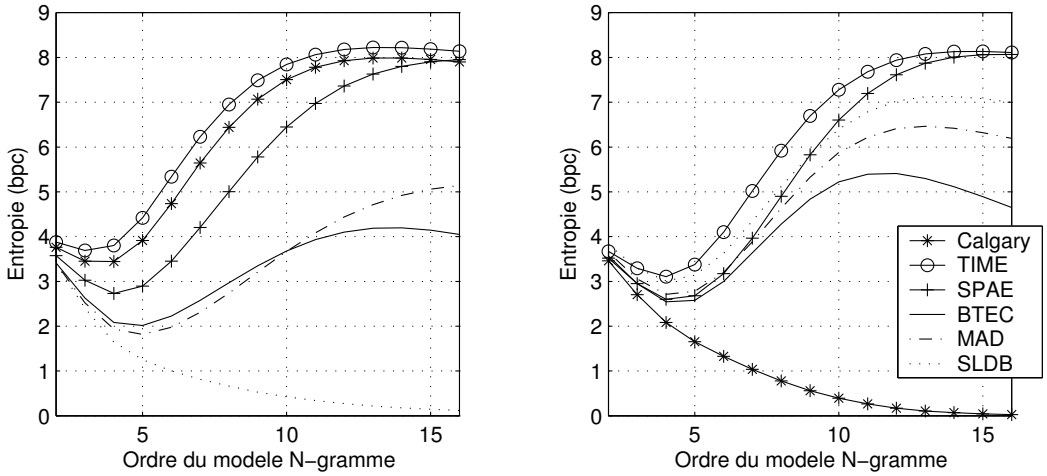

Figure 2.3: Entropies croisées de plusieurs corpus en langue anglaise.

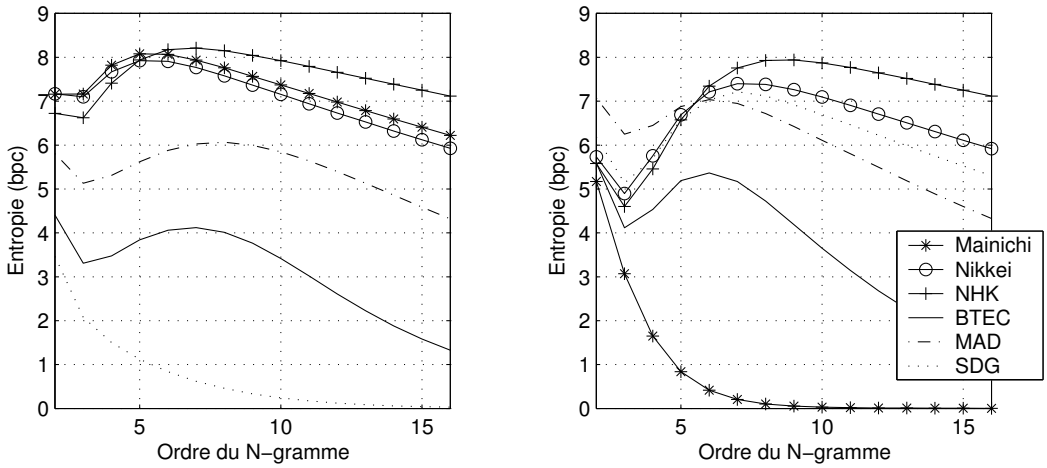

Figure 2.4: Entropies croisées de plusieurs corpus en langue japonaise.

Il n'est pas nécessaire que l'autre référence ait le score entropique le plus élevé : par exemple, sur la gauche de la figure 2.3, TIME a la plus grande valeur entropique (supérieure à celle de la référence de l'écrit Calgary) et sera donc dans la situation du point $\bigotimes$ de la figure 2.1, page 77. Sa projection sera dans l'intervalle [SLDB,Calgary].

Calcul du coefficient de littérarité

On procède au calcul du coefficient de littérarité de chaque corpus, pour plusieurs valeurs de l'historique N. Les figures 2.5 et 2.6 montrent les résultats pour l'anglais et pour le japonais respectivement. Pour l'anglais, 0 correspond à la référence de l'oral SLDB, 1 correspond à la référence de l'écrit Calgary. Pour le japonais, 0 correspond à la référence de l'oral SLDB, 1 correspond à la référence de l'écrit Nikkei. Etant donné le choix des ressources références, un score proche de 0 traduit une grande similarité à de la référence de l'oral (SLDB), alors qu'un score proche de 1 traduit une grande similarité à de la référence de l'écrit (Calgary pour la langue anglaise, Nikkei pour la langue japonaise).

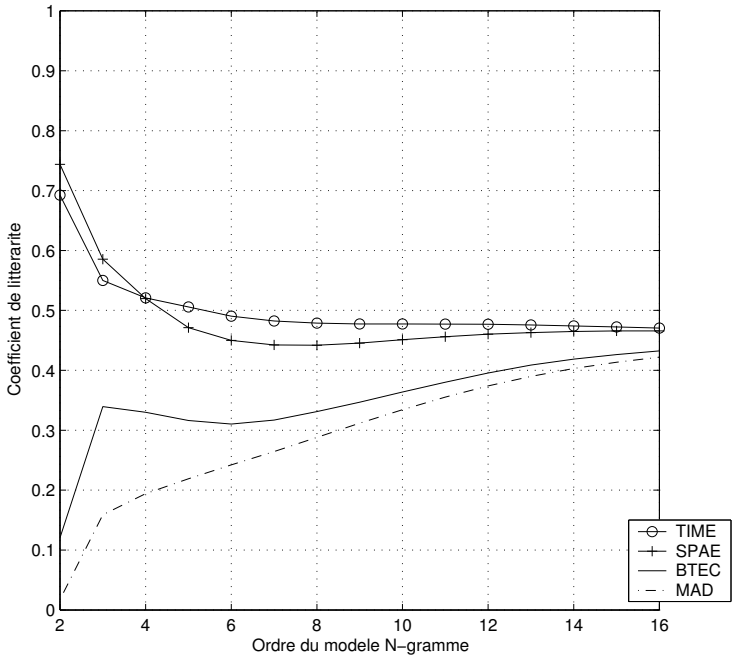

Figure 2.5: Coefficient de littérarité pour la langue anglaise.

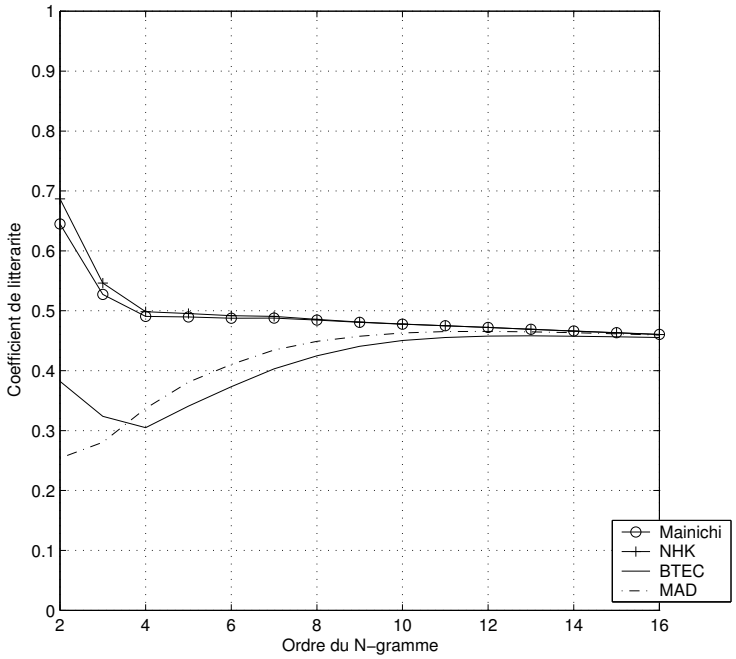

Figure 2.6: Coefficient de littérarité pour la langue japonaise.

La figure 2.7 montre les valeurs obtenues pour des modèles en 5-grammes de caractères. On remarque que MAD et BTEC ont des coefficients plus faibles que TIME et SPAE en anglais, et NHK et Mainichi en japonais. La séparation entre les deux « types » de ressources est visible. En ce qui concerne les ressources dont

	MAD	BTEC	SPAE/ NHK	TIME/ Mainichi
anglais	0,22	0,32	0,47	0,51
japonais	0,38	0,34	0,49	0,49

Figure 2.7: Coefficient de littérarité pour des modèles 5-grammes de caractères.

le coefficient reflète l'origine orale, le corpus MAD se voit attribuer les coefficients les plus faibles en langue anglaise ; cela conforte en somme sa réputation d'être une ressource dont le contenu est encore plus oral que le corpus BTEC. En revanche, cette séparation est moins évidente dans la langue japonaise où la courbe est plus écrasée, et où on a plus de mal à arriver à une conclusion claire sur la littérarité.

En ce qui concerne les ressources dont le coefficient reflète l'origine écrite, en langue anglaise TIME et SPAE ont des valeurs très proches pour N compris entre 3 et 5, le TIME affichant un coefficient supérieur à celui de SPAE à partir de $N = 4$. On est donc conforté dans l'intuition que TIME présente un caractère plus écrit que SPAE de par son origine purement écrite (textes de journaux) face à une ressource rassemblant des transcriptions de réunions dans un contexte professionnel.

En japonais, là encore les différences sont moins marquées : bien que la séparation soit claire entre ressources a priori orales et ressources a priori écrites, on ne peut émettre de jugement quant aux différences entre corpus écrits : les coefficients du corpus Mainichi (textes de journaux) et du corpus NHK (transcriptions d'informations lues à la télévision) sont très proches quel que soit N. Ce phénomène de tassement d'échelle en japonais (et, respectivement, d'étirement d'échelle en anglais) peut trouver sa cause dans le fait que nous avons été dans l'impossibilité d'utiliser des ressources de littérature contemporaine en langue japonaise.

La classification montrée ici semble relativement résistante et stable face aux variations de longueur d'historique N : les coefficients convergent vers une valeur proche de $0,5$ pour $N > 14$ en langue anglaise, et pour $N > 7$ en langue japonaise, ce qui peut être rapproché de la discussion sur les cas dégénérés menée dans la section 2.2.2. On avait alors montré qu'à mesure que N augmentait, le nombre de N-grammes attestés dans les ressources de référence diminuait, et que les coefficients tendaient vers la valeur théorique de $0,5$.

2.2.5 Discussion

On a montré dans cette section une manière d'approcher le problème de la similarité entre ressources linguistiques textuelles. Après avoir comparé le coefficient de similarité défini plus haut à des méthodes issues de travaux précédents, une expérience a été menée afin de vérifier l'utilité d'une telle approche dans le cadre d'une tâche de classification du caractère oral ou écrit de plusieurs ressources, en anglais et en japonais.

L'apport de cette méthode est double : en premier lieu, elle apporte une façon de profiler automatiquement et avec rapidité de grands documents textuels selon des références choisies par l'utilisateur. On peut donc imaginer son application future pour caractériser d'autres dimensions que la littérarité d'un texte. On citera par exemple l'étude du degré de politesse, de la période du niveau de langue (datations de textes par exemple), de la similarité entre le style de deux auteurs sur une œuvre

contestée[28], etc. Enfin, elle apporte une alternative aux méthodes fondées sur le comptage de phénomènes linguistiques en proposant de considérer les données en tant que chaines de caractères uniquement. Cela permet de se passer de segmentation préalable en mots ou en lexèmes, et ainsi de traiter élégamment les langues dont le système d'écriture ne comporte pas de séparateurs de mots.

Après avoir proposé une méthode pour la caractérisation globale de grands documents, on est tenté de vouloir en caractériser le contenu plus finement : en effet, s'il est possible de calculer une similarité entre les documents, on peut aussi s'intéresser à la similarité à l'intérieur des documents. De la même manière qu'on s'interroge sur la similarité du tout relativement à des références, on peut s'interroger sur la similarité des sous-parties d'une plus grande ressource. Dans la section suivante, nous exposons donc une application de la similarité en vue de caractériser l'homogénéité de ressources linguistiques.

2.3 Une représentation de l'homogénéité

2.3.1 Extension de la similarité à l'homogénéité

Motivation et intuition de départ

Est dite homogène, une chose dont tous les éléments sont de même nature ou présentent des similitudes de structure, de fonction, de répartition. Nous définissons donc l'homogénéité d'une ressource comme l'ensemble de ses variations internes. Par l'étude de l'homogénéité des ressources linguistiques, nous entendons ici l'étude des régularités et irrégularités en langue qui les caractérisent. Ces régularités ou irrégularités peuvent être de forme variée : par exemple, des différences de style, de registre, ou encore de phénomènes linguistiques internes.

Tout comme on a pu caractériser la similarité (externe) d'un document de façon automatique en le positionnant par rapport à des références, on désire pouvoir caractériser l'homogénéité (interne) d'un document. Comme nous l'avons dit plus haut, l'intuition de départ réside dans le fait que, si la similarité peut être calculée entre plusieurs documents, l'homogénéité est un calcul similaire, mais à l'intérieur des documents pour en caractériser la régularité. Il va sans dire qu'une ressource linguistique textuelle est rarement constituée de façon uniforme : comme il est intéressant pour le traitement automatique des langues de disposer de données en grandes quantités, des textes de différents auteurs, portant sur des sujets différents, ou employant un style différent, sont souvent présents dans une seule et même ressource, sans que de telles particularités soient explicitement signalées.

Si on peut parfois penser[29], comme le laisse croire l'approche statistique en traitement automatique des langues, que l'abondance de données est une solution à beaucoup de problèmes du domaine, il est indéniable que le choix des données utilisées est crucial. Certaines données, qui ne conviennent pas ou peu à la tâche à traiter, pénalisent par leur seule présence la performance globale du système. Il est donc crucial de pouvoir se faire une idée de l'homogénéité de grands ensembles de données, c'est-à-dire des variations internes qui s'y produisent, en termes par exemple de registre de langue, de domaine, etc.

[28] L'un de nos rêves est de l'appliquer au vieux débat Corneille/Molière/Racine, comme le traite élégamment Etienne Brunet dans Brunet, *Où l'on mesure la distance entre les distances*, 2004.

[29] Banko & Brill, *Scaling to very very large corpora for natural language disambiguation*, 2001.

Nous voyons donc ici l'homogénéité en terme de régularité, de la variation des parties par rapport au tout. Dans un premier temps, nous visualiserons les variations internes d'un corpus donné, au fil du texte et selon différentes découpes.

Expérience préliminaire : l'homogénéité du BTEC au fil du texte

Dans une première expérience, nous étudions l'homogénéité du corpus multilingue BTEC au fil du texte, c'est-à-dire sans modifier l'ordre des phrases à l'intérieur de la ressource. La ressource est découpée en sous-parties, et chacune se voit attribuer un coefficient de similarité selon les mêmes références que celles utilisées précédemment (SLDB pour l'oral et Calgary ou Nikkei pour l'écrit). En augmentant le nombre de parties, nous visons à obtenir une meilleure idée des variations locales du coefficient : de façon analogue à un changement de résolution, nous procéderons à une découpe en un grand nombre de petites parties (avec un lissage faible, plus de variations sont visibles mais elles sont moins significatives), et un petit nombre de grosses parties (avec un lissage important, les variations sont moins détaillées, mais elles sont plus significatives). L'expérience est menée sur le corpus BTEC en langues anglaise et japonaise. La figure 2.8 montre les variations du coefficient de similarité pour une division du corpus BTEC en 10 parties (figure de gauche) puis 100 parties (figure de droite) dans chaque langue. Comme précédemment, le coefficient est calculé sur des modèles en 5-grammes de caractères car c'est l'ordre qui a donné les meilleurs résultats lors des expériences précédentes, page 82.

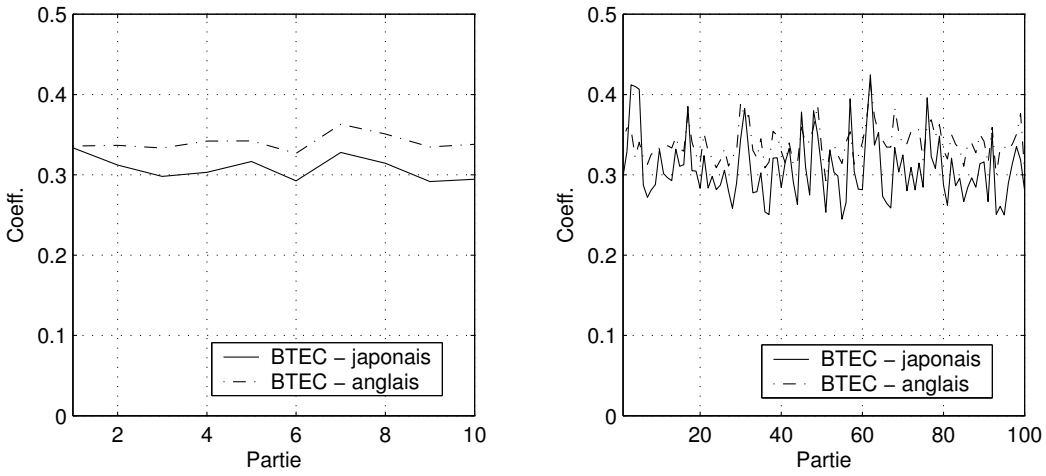

Figure 2.8: Variations du coefficient de similarité au sein du BTEC en anglais et japonais, pour 10 parties (gauche), et pour 100 parties (droite).

Le tableau 2.5 montre la corrélation entre les valeurs obtenues pour l'anglais et le japonais, et l'écart-type pour chacune des langues pour un nombre croissant de parties.

La corrélation entre les valeurs en anglais et en japonais augmente jusqu'à un maximum de $0,7$, ce qui correspond à une division en 100 parties (chaque partie contient de ce fait environ 1600 lignes). Cela tend à confirmer l'intuition que le degré de littérarité présente des variations similaires dans les deux langues de la même ressource alignée. D'autre part, cela est à rapprocher du fait que le BTEC est consitué d'environ 200 recueils de phrases de longueur variable, et que la

Parties	10	50	100	500	1000
Corrélation	0,59	0,61	0,70	0,69	0,67
Écart-type ang.	0,014	0,031	0,053	0,063	0,075
Écart-type jpn.	0,008	0,160	0,022	0,031	0,037

Tableau 2.5: Corrélation et écart-type pour un nombre croissant de parties.

valeur la plus proche en terme de découpe renvoie ainsi aux meilleures corrélations. Cette corrélation ne peut excéder un certain seuil maximal, du fait des différences inhérentes à la structure des langues examinées, ainsi que d'une division arbitraire : en effet, le corpus BTEC étant constitué d'une juxtaposition de lignes tirées de recueils de phrases utiles pour voyageurs, on peut imaginer qu'il y ait d'importantes différences en terme de style dans les différents recueils. En revanche l'écart-type, qui montre la variation interne moyenne du coefficient de similarité, nous fournit une quantification intéressante de l'homogénéité globale. L'écart-type augmente à mesure que le nombre de parties en lesquelles est divisé le corpus BTEC augmente : en effet, à mesure qu'on augmente la résolution, les irrégularités locales apparaissent de façon moins lissée.

Nous montrons dans la section suivante qu'il est possible de représenter l'homogénéité sous la forme d'une distribution de coefficients de similarité.

2.3.2 Représentation sous forme de distribution de coefficients de similarité

Intuition de départ et motivation

Une ressource linguistique est bien souvent composée de documents provenant de différentes sources. L'homogénéité d'une telle ressource est ainsi rarement accessible au delà de la simple connaissance de l'origine des documents la constituant. La diversité des sources, ainsi que le fait de rassembler un grand nombre de documents en un corpus entraine que son contenu présente des irrégularités. Ces irrégularités ayant un caractère multidimensionnel[30], on ne peut affirmer trivialement qu'une ressource est homogène ou hétérogène : des sous-langages différents exhiberont des variations aux niveaux lexical, syntaxique, sémantique et structurel[31].

Peu de travaux ont été menés auparavant sur l'homogénéité des ressources linguistiques, et sur ses applications éventuelles dans le TAL. Cependant, on relèvera ceux de Cavaglià et Rose :

- Cavaglià[32] reprend des mesures fondées sur des comptages de mots ou de lexèmes, et fait l'hypothèse que l'utilisation de ressources homogènes amène généralement à de meilleures performances des systèmes de traitement automatique des langues, mais ses expériences sur un catégoriseur automatique de textes sont peu probantes. Cela nous incite à vérifier puis mettre en doute la validité d'une telle hypothèse dans la suite de cette étude (section 2.3.3).

[30] Voir Biber, *Variation across speech and writing*, 1988, et Biber, *Dimensions in register variation*, 1995.

[31] Kittredge & Lehrberger, *Sublanguage. Studies of language in restricted semantic domains*, 1982.

[32] Cavaglià, *Measuring corpus homogeneity using a range of measures for inter-document distance*, 2002.

- Rose et Tucker[33] examinent la performance d'un système de reconnaissance de parole en fonction de la taille et du type des données utilisées pour construire le modèle de langue qui entre en jeu. Ils utilisent dans cette étude une petite ressource de départ, à laquelle ils rajoutent progressivement des données du même type à l'aide d'un critère de similarité fondé sur une corrélation de Spearman, critère proposé par Kilgarriff[34], et le logarithme de la vraisemblance G^2, défini par Dunning[35].

Comme on l'a vu au cours de l'expérience précédente, nous étudions l'homogénéité en fonction de deux sous-langages de référence, qui calibrent en quelque sorte la similarité associée. En cela il n'y a donc pas une, mais bien des homogénéités selon qu'on considère des axes bornés par des sous-langages différents (par exemple registre de politesse, domaine, etc.), car elles correspondent à des irrégularités différentes en fonction des données de référence utilisées. Par la suite et comme dans le protocole expérimental précédent, nous nous intéressons à l'homogénéité en termes de régularité et d'irrégularité entre registres de la langue orale et de la langue écrite. La connaissance permettant de détecter ces variations est ainsi incluse implicitement dans les données de référence.

Puisqu'on considère l'homogénéité d'une ressource comme la variation de la similarité de ses parties par rapport au tout, on peut imaginer la représenter par la distribution des coefficients de similarité de chacune de ses parties. Nous allons donc visualiser l'homogénéité du corpus BTEC sous forme de distribution. Nous examinerons ensuite l'influence de l'homogénéité des données sur un système de TAL en termes de perplexité, et de qualité des sorties dans le cas d'un système de traduction automatique, dans une expérience d'adaptation des données à une tâche à traiter.

L'homogénéité du BTEC sous forme de distribution de coefficients de similarité

Le corpus BTEC étant constitué d'une juxtaposition de recueils de phrases dans le domaine du tourisme, nous examinons la distribution de ses coefficients de similarité selon deux découpes intuitives : tout d'abord en conservant l'intégrité de chacun des recueils (à chaque recueil est associé un coefficient); puis phrase par phrase (à chaque phrase sera associé un coefficient). La figure 2.9 montre les distributions des coefficients de similarité en japonais et en anglais, à l'échelle du recueil et de la phrase, et le tableau 2.6 leurs valeurs moyennes et écart-type.

Tableau 2.6: Valeurs moyennes ± écarts-types des distributions des coefficients de similarité pour le japonais et pour l'anglais.

Coefficient	japonais	anglais
Recueil	0,330±0,020	0,288±0,027
Phrase	0,315±0,118	0,313±0,156

[33] Tony Rose & Tucker, *The effects of corpus size and homogeneity on language model quality*, 1997.

[34] Kilgarriff, *Using word frequency lists to measure corpus homogeneity and similarity between corpora*, 1997.

[35] Dunning, *Accurate methods for the statistics of surprise and coincidence*, 1993.

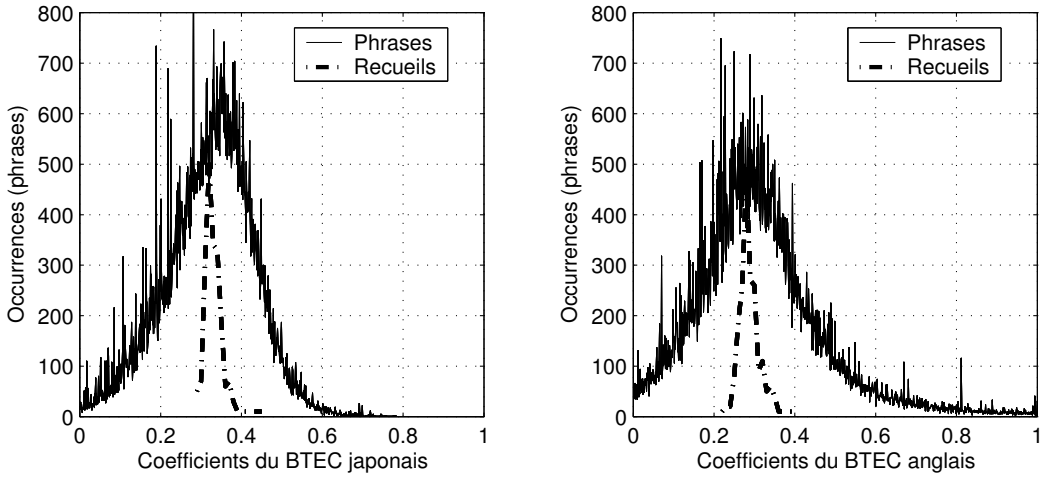

Figure 2.9: Distributions des coefficients de similarité en langue japonaise (gauche) et anglaise (droite), à l'échelle du recueil (trait gras pointillé) et de la phrase (trait fin continu).

Ces distributions donnent un aperçu de l'homogénéité du BTEC en terme de littérarité : elles font apparaitre les régularités et irrégularités entre registre oral et écrit au sein de la ressource. Alors que la moyenne apporte une information globale, l'écart-type des distributions renseigne sur la quantité de variations internes entre oral et écrit. On peut expliquer les différences de valeurs moyennes et d'écarts-types entre les deux découpes par le fait que tous les recueils ne contiennent pas le même nombre de phrases[36]. Bien que les distributions à l'échelle de la phrase et du recueil aient globalement la même allure gaussienne, les irrégularités au niveau de la phrase encouragent à utiliser une unité plus grande pour estimer les entropies croisées. Il est intéressant de noter qu'à l'échelle du recueil, la corrélation entre coefficients de similarité et longueur moyenne du recueil, ainsi qu'entre coefficients de similarité et longueur moyenne de la phrase de chaque recueil, est faible ($0,178$ et $0,278$, respectivement) : les irrégularités ainsi détectées ne sont pas fortement liées à des paramètres de surface triviaux. En revanche, il est satisfaisant de constater que la corrélation est forte[37] entre les coefficients en japonais et en anglais (0,781).

La mise en évidence dans les données d'irrégularités internes importantes en terme de registre oral ou écrit peut en outre permettre de les utiliser dans un système de traitement automatique des langues. En sélectionnant certaines données, on a la possibilité de modifier l'homogénéité du corpus : on peut par exemple choisir d'éliminer certaines irrégularités (ce qui revient à garder la même moyenne, mais à modifier l'écart-type). La section suivante présente une expérience d'adaptation des données utilisant les résultats obtenus ci-dessus.

[36]Les recueils du BTEC sont en effet longs en moyenne de 824 phrases, avec un écart-type important de 594 phrases.

[37]Il est en effet attendu que cette corrélation soit forte puisque les parties japonaises et anglaises sont simplement traductions l'une de l'autre.

2.3.3 Expérience : influence de l'homogénéité des données sur la performance d'un système de traitement automatique des langues

Méthode

Nous souhaitons vérifier l'hypothèse selon laquelle un système de traitement automatique des langues produira de meilleurs résultats quand les données qu'il utilise sont homogènes, et proches de la tâche en terme de similarité.

À cette fin, nous utilisons la représentation de l'homogénéité exposée ci-dessus pour sélectionner des données d'entrainement en fonction de leur coefficient de similarité, et par rapport à celui de la tâche. Nous comparons les performances d'un système construit sur de telles données avec celles d'un système construit sur des données choisies aléatoirement et en même quantité depuis la même ressource. En ne sélectionnant que des données similaires à la tâche, c'est-à-dire possédant un coefficient de similarité proche de celui de la tâche, nous modifions l'homogénéité des données : les irrégularités en terme de registre oral ou écrit sont laissées de côté. En terme de distribution, la moyenne reste la même alors que nous réduisons l'écart-type. Nous rendons artificiellement les données plus régulières,

La figure 2.10 illustre les deux façons de construire les données d'entrainement, par exemple ici sur une distribution en triangle.

Figure 2.10: Construction des données d'entrainement, à partir d'une distribution triangle : à gauche la réduction est faite par similarité croissante par rapport à la tâche, à droite la réduction est aléatoire.

Tout comme Cavaglià[38], nous voulons donc tester l'hypothèse selon laquelle plus le coefficient de similarité de données d'entrainement est proche de celui de la tâche, plus cette donnée est appropriée pour améliorer les performances d'un système.

En ce qui concerne la tâche, nous utilisons un ensemble de test constituée d'une suite de phrases extraites aléatoirement de la même ressource linguistique, le corpus BTEC. Cette tâche, jeu de test standard de 510 phrases en langue japonaise, est par ailleurs retirée des données d'entrainement. Les phrases constituant la tâche sont le résultat d'un tirage aléatoire de 510 phrases du corpus BTEC (pour plus de précisions quant à cet ensemble de test, voir l'annexe A.1).

Ce jeu de test sert tout d'abord à estimer la perplexité d'un modèle de langue construit sur les données d'entrainement, puis comme tâche de traduction pour un

[38] Cavaglià, *Measuring corpus homogeneity using a range of measures for inter-document distance*, 2002.

système de traduction automatique par l'exemple, utilisant les données d'entrainement comme unique base d'exemples.

Selon le même coefficient de similarité défini plus haut (donc doté des mêmes références choisies pour quantifier le registre oral ou écrit), la tâche a un coefficient $I_{tâche} = 0,331$. La tâche étant le résultat d'un tirage aléatoire de phrases du BTEC, le coefficient de la tâche devrait être proche de celui du BTEC. Nous vérifions bien que la moyenne des coefficients de similarité de chacun des recueils du BTEC est proche de cet indice : $I_{BTEC} = 0,330$. Cela confirme que la sélection aléatoire de phrases du BTEC a permis de constituer un ensemble très similaire au BTEC en terme de littérarité. La tâche est bien représentative du BTEC en terme de littérarité dans ce cas particulier :

$$I_{tâche} \simeq I_{BTEC} = 0,330\,.$$

Partant de ce cas particulier (la valeur moyenne des coefficients de la tâche et du BTEC est presque égale), il ne reste plus qu'à éliminer les données possédant les coefficients les plus éloignés.

Nous estimons tout d'abord la performance d'un système bâti sur de telles données en terme de perplexité, puis de qualité des sorties de traduction automatique. Puis nous la comparons, à quantité de données équivalente, à celle d'un système construit sur des données choisies aléatoirement dans le corpus BTEC.

Influence sur la perplexité - systèmes fondés sur les statistiques

Nous montrons dans l'annexe C que la perplexité est souvent utilisée pour comparer des modèles de langue : elle est représentative, dans une certaine mesure, de la performance des systèmes de traitement automatique des langues fondés sur l'approche statistique. Bien qu'elle ne soit pas systématiquement fortement corrélée avec la performance d'un système fondé sur un modèle de langue, elle donne une estimation du facteur de branchement moyen du modèle[39], ce qui donne une bonne idée de sa complexité et de ses capacités prédictives. La mesure de perplexité reste ainsi couramment utilisée par la communauté du traitement automatique des langues, principalement dans le domaine de la reconnaissance de parole : on considère que lorsque la perplexité baisse, la performance globale d'un système utilisant un modèle de langue augmente.

On procède au calcul des perplexités des modèles de langue en caractères construits sur une quantité de données croissante, d'une part choisie aléatoirement dans le BTEC, d'autre part choisie en priorité autour de $I_{tâche}$ (les seuils sont déterminés automatiquement pour chaque quantité de données à garder).

Les entropies croisées sont estimées sur la tâche, toutes les estimations sont effectuées 5 fois, et moyennées. La figure 2.11 montre les perplexités en caractères lissées pour des quantités de données d'entrainement croissantes et allant de $0,5\%$ à 100% du BTEC. Sur la figure de gauche, les perplexités sont calculées sur des quantités de données croissantes, choisies pour être proches de $I_{tâche}$, ou choisies aléatoirement. La figure de droite montre la variation de la perplexité lorsqu'on passe d'un modèle construit sur des données choisies aléatoirement à un modèle construit sur des donnés proches de $I_{tâche}$. Cette variation est la différence entre les deux courbes à gauche.

[39] Autrement dit, lorsque le modèle doit prédire le symbole qui suit une chaine donnée, il a le choix en moyenne parmi un nombre de symboles qui vaut sa perplexité.

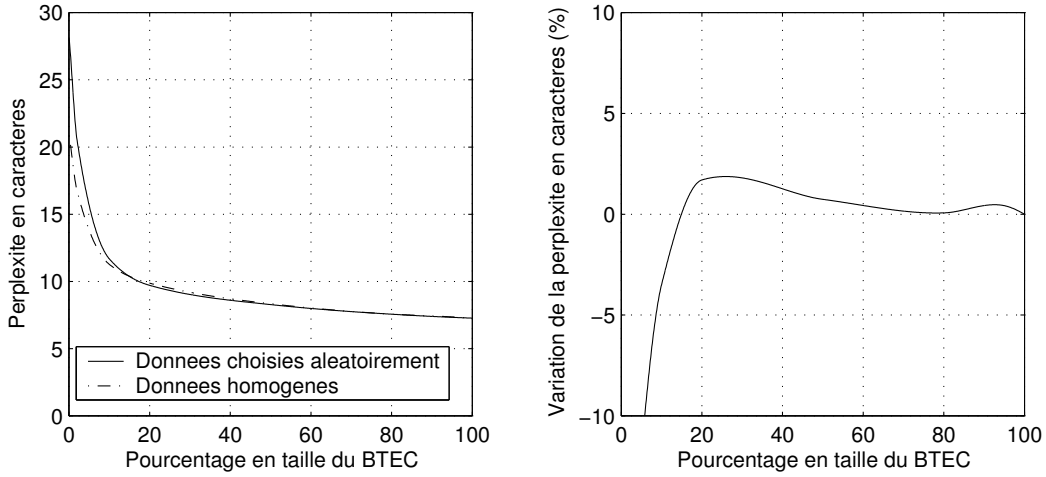

Figure 2.11: Perplexités en caractères des modèles de langue construits sur des quantités de données d'entrainement croissantes.

Comme on pouvait s'y attendre, la perplexité diminue à mesure que la taille du corpus d'entrainement augmente. La perplexité de la tâche, relativement à un modèle construit sur des données proches de $I_{tâche}$ est plus faible que relativement à des données choisies aléatoirement dans la ressource pour une quantité de données inférieure à 15% des données (soit un peu plus de 20 000 phrases, ou environ $1,5$ mégaoctets de données). Passé ce cap, la situation est inversée. La conclusion à tirer est que l'usage de données d'entrainement « ciblées » sur la tâche n'est bénéfique, en perplexité, qu'avec peu de données, mais par la suite il est bénéfique d'incorporer des données plus variées. Restreindre les données d'entrainement à des données proches de $I_{tâche}$ semble donc produire des perplexités plus importantes.

Mesurer la perplexité de modèles de langues nous a permis d'observer l'influence de l'homogénéité des données sur un système « générique » fondé sur l'approche statistique. Afin d'observer cette influence sur des systèmes fondés sur les données, mais pas sur l'approche statistique, nous effectuons dans les sections suivantes des expériences similaires sur la traduction par l'exemple : tout d'abord en évaluant un système sous sa forme dégénérée de simple mémoire de traduction ; ensuite en évaluant les résultats d'un système de traduction automatique qui génère des patrons de transfert à partir de données textuelles bilingues analysées.

Influence sur la qualité des sorties de systèmes de traduction automatique par l'exemple

Cas de la mémoire de traduction Sous sa forme la plus dégénérée, un système de traduction automatique par l'exemple va simplement chercher dans sa base de données la phrase la plus proche de celle qu'il a à traduire, et propose la traduction de celle-ci comme traduction de la phrase recherchée. Si la phrase source n'est pas trouvée telle quelle dans la base d'exemples, il propose la phrase la plus proche, au sens par exemple de la distance d'édition, ou la plus « similaire ».

On reproduit donc l'expérience précédente en faisant fonctionner la mémoire de traduction sur la tâche avec une base d'exemples dont on incrémente la taille. Les résultats de traduction en langue anglaise sont ensuite évalués. Puisqu'il n'est pas

possible de réaliser un grand nombre d'évaluations humaines de la traduction automatique, nous nous servons de mesures objectives bien connues pour l'évaluation de la qualité[40] :

- BLEU[41] est comme on l'a vu dans l'annexe B.1 la moyenne géométrique des précisions en N-grammes de la phrase à juger, calculées sur un ensemble de références. Elle est comprise entre 0 et 1, et un score élevé indique une meilleure qualité de traduction ; la mesure semble être plus corrélée avec la *fluidité*[42], la naturalité de l'expression des phrases évaluées ;

- NIST[43] est comme on l'a vu dans l'annexe B.2 une variante de BLEU fondée sur la moyenne arithmétique des précisions en N-grammes pondérées de la phrase à juger, calculées sur un ensemble de références. Elle possède une borne inférieure à 0, pas de borne supérieure, et un score élevé indique là encore une meilleure qualité de traduction. La mesure semble être plus corrélée avec l'*informativité*[44], le sens des phrases évaluées ;

- mWER[45] ou *Multiple Word Error Rate*, est la distance d'édition en mots entre la phrase de sortie et la phrase la plus proche comprise dans un jeu de références. Elle est bornée entre 0 et 1, et plus le score est bas, meilleure est la traduction.

Pour plus de détails sur les mesures automatiques BLEU et NIST, on pourra se référer à l'annexe B. Les trois mesures utilisent un jeu de références en langue anglaise constitué de 16 références par traduction à juger.

La figure 2.12 montre les résultats en BLEU, NIST et mWER pour des données de taille croissante variant de $0,5\%$ à 100% du BTEC. En haut, la qualité des phrases produites par mémoire de traduction est évaluée par les méthodes BLEU, NIST et mWER, pour une quantité croissante de données sélectionnées d'une part aléatoirement, d'autre part pour leur similarité avec la tâche. La figure du dessous montre la différence de qualité entre l'utilisation de données sélectionnées aléatoirement et de données sélectionnées pour leur similarité avec la tâche. On remarque que pour les trois mesures, la qualité de traduction a tendance à augmenter à mesure que la taille de la mémoire de traduction augmente. Pour de faibles quantités de données, l'utilisation de données dont le coefficient est proche de $I_{tâche}$ produit de moins bonnes traductions que dans le cas de données sélectionnées aléatoirement. Par la suite, la différence n'est plus significative.

Qualité des sorties d'un système de traduction automatique par l'exemple
Nous nous intéressons à présent à la qualité de traduction d'un système de traduction automatique par l'exemple, utilisant des patrons de transfert obtenus à partir de données textuelles bilingues analysées.

Il s'agit du système HPATR d'Imamura[46]. C'est un système japonais-anglais, qui analyse les deux parties alignées d'un bicorpus à l'aide de grammaires spécifiques.

[40] Le détail des mesures BLEU et NIST est donné en annexe B.
[41] Papineni *et al.*, *BLEU: a method for automatic evaluation of machine translation*, 2002.
[42] *Fluency* en langue anglaise.
[43] Doddington, *Automatic evaluation of machine translation quality using N-gram co-occurrence statistics*, 2002.
[44] *Adequacy* en langue anglaise.
[45] Och, *Minimum error rate training in statistical machine translation*, 2003.
[46] Imamura, *Hierarchical phrase alignment harmonized with parsing*, 2001.

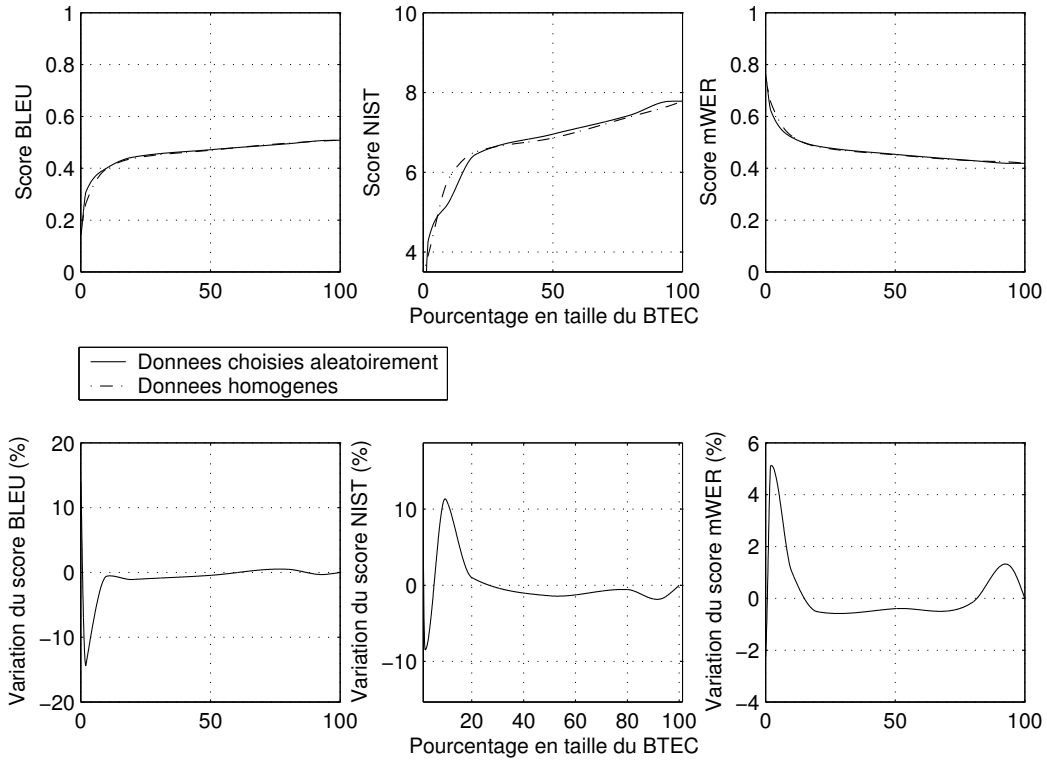

Figure 2.12: Qualité des phrases produites par mémoire de traduction, évaluée par les méthodes BLEU, NIST et mWER.

Les données analysées servent ainsi à construire des arbres bilingues. La traduction s'effectue par transfert, en générant automatiquement des patrons à partir des arbres alignés. C'est un exemple de système à base de théorie linguistique (cette connaissance étant contenue dans les règles des analyseurs). Un tel système n'utilisant pas de modèle statistique de langue, nous l'utilisons ici comme complémentaire de l'expérience menée précédemment sur la perplexité.

Nous construisons là encore plusieurs systèmes sur des quantités de données croissantes, et faisons traduire la tâche par chacun d'entre eux. Les traductions sont évaluées comme précédemment avec BLEU, NIST et mWER avec 16 références par phrase à juger. La figure 2.13 montre les résultats en BLEU, NIST et mWER pour des tailles croissantes de données variant de $0,5\%$ à 100% du BTEC.

La qualité de traduction augmente à mesure que le système dispose de données plus importantes. La courbe, d'abord exponentielle, tend vers une asymptote à mesure que les données vues par le système augmentent. Là encore, la qualité de traduction est meilleure dans le cas ou des données choisies de façon aléatoire sont choisies, passé un certain seuil de quantité de données (à partir de 3% du BTEC pour BLEU, 18% pour NIST, et 2% pour mWER), par rapport au cas où sont des données dont le coefficient est proche de $I_{tâche}$. Ce résultat est significatif : si on compare les scores produits par les 510 phrases de la tâche, pour des sytèmes construits sur 50% des données du BTEC, on trouve une différence significative pour les scores BLEU, NIST et mWER avec des valeurs de confiance respectives de $88,49\%$, $99,9\%$ et $73,24\%$.

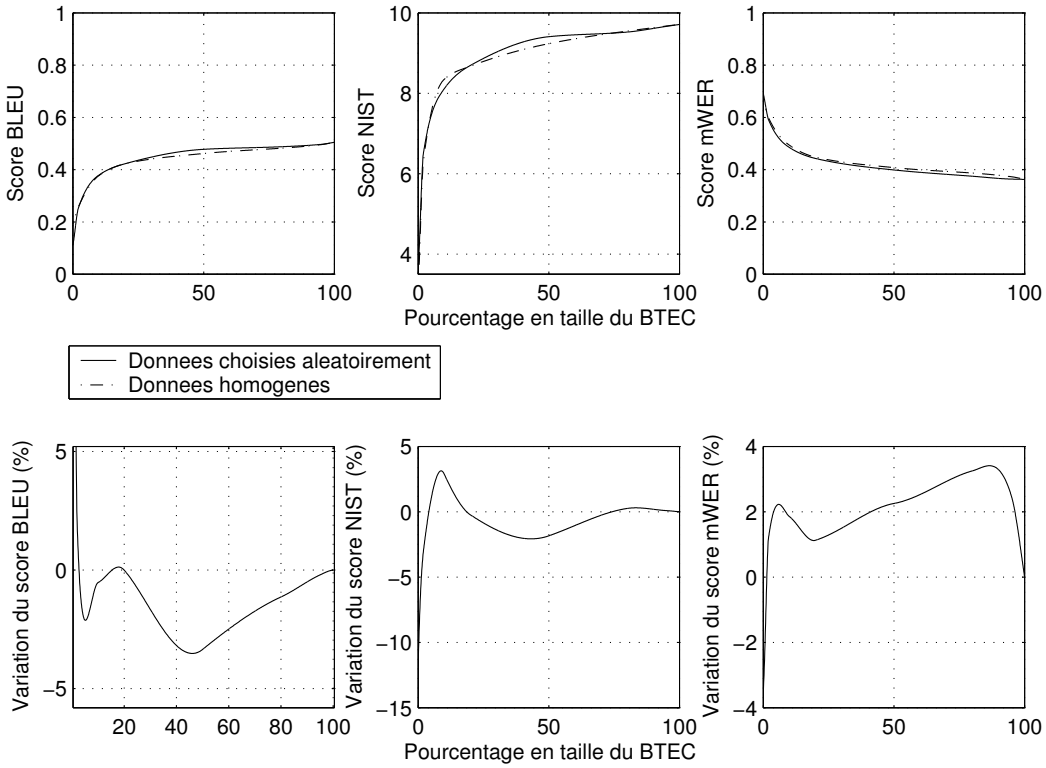

Figure 2.13: Même chose que dans la figure 2.12, mais pour un système de traduction.

2.3.4 Discussion

Dans cette section nous avons proposé une approche de l'homogénéité des ressources textuelles. Nous avons défini l'homogénéité d'une ressource linguistique comme l'ensemble des variations internes et des régularités qui la caractérisent, et proposé de quantifier ces variations à l'aide du coefficient de similarité défini dans la section 2.2.2. Ce coefficient est lui-même défini par rapport à deux sous-langages de référence, et permet de visualiser des variations internes en termes, par exemple, de style ou de registre de langue.

La visualisation de ces variations sous la forme de la distribution des coefficients attribués à des sous-parties permet d'en extraire plusieurs informations : la moyenne montre où se situe la ressource prise dans sa totalité sur l'échelle de similarité définie relativement aux références, et l'écart-type montre la quantité de variations présente dans la ressource. Il devient dès lors possible de garder ou de supprimer certaines sous-parties afin de modifier une telle distribution.

Nous nous sommes proposé de tester l'hypothèse selon laquelle, plus les données d'entrainement sont similaires à la tâche, meilleure est la performance d'un système générique de traitement automatique des langues. Pour cela, nous avons mené une expérience d'adaptation des données en fonction de la tâche à traiter. Celle-ci, menée d'une part sur des modèles de langue (systèmes orientés vers les statistiques) ainsi que sur des systèmes de traduction par l'exemple (mémoire de traduction, et système de traduction automatique par l'exemple), permettent de remettre en cause l'hypothèse énoncée précédemment.

Basés sur une même ressource linguistique, nos résultats montrent que du point

de vue de la perplexité de modèles de langue comme de la qualité des sorties d'un système de traduction automatique par l'exemple, l'utilisation exclusive de données proches de la tâche en terme de similarité produit de moins bonnes performances que l'utilisation de données équilibrées, choisies de façon aléatoire. Passés des seuils minimaux jusqu'auxquels l'utilisation de données similaires à la tâche donne de meilleurs résultats, l'utilisation de données plus équilibrées permet plus de variations. Les systèmes affichent alors de meilleures performances, statistiquement significatives, à quantité de données égale.

Nous avons donc réfuté de façon empirique l'hypothèse selon laquelle des données d'entrainement proches de la tâche en terme de similarité produisent toujours de meilleurs résultats. Nous avons réfuté cette hypothèse sur plusieurs systèmes génériques. Bien que l'hypothèse reste vérifiée sur très peu de données, elle ne tient plus à partir d'une certaine masse de données d'apprentissage. Pour expliquer cela, on peut supposer que des données éloignées de la tâche en terme de similarité permettent aux systèmes de meilleures variations, conduisant à de meilleures performances.

Conclusion

Dans ce chapitre, nous avons proposé une approche de la caractérisation automatique de données utilisant un critère entropique en unité de caractères. L'utilisation du caractère comme unité permet de proposer une méthode qui s'applique sur tout type de donnée linguistique informatisée, sans restriction sur les langues a priori. Elle apporte ainsi une alternative aux méthodes de caractérisation fondées sur le comptage de mots ou de lexèmes. L'utilisation de critères entropiques permet de profiler rapidement et automatiquement de grandes quantités de données textuelles, selon des références choisies par l'utilisateur.

Nous avons tout d'abord exposé une approche du problème de la similarité entre ressources linguistiques textuelles en proposant la construction d'une échelle de similarité relative à deux corpus de référence. Cette échelle utilise l'entropie croisée d'une ressource donnée relativement à des modèles de langue en N-grammes de caractères afin de lui attribuer un score. Une ressource se voit ainsi accorder un score compris entre 0 et 1 selon qu'elle est proche en terme de similarité de l'une ou l'autre des références.

En comparant le coefficient de similarité ainsi défini à plusieurs méthodes issues de travaux précédents, nous avons montré qu'il produisait des résultats comparables, avec un domaine d'application plus grand : on peut l'appliquer sans prétraitement à des textes écrits dans des langues ne comportant pas de découpe graphique en mots. Muni de ce coefficient, nous avons ensuite mené une expérience de classification du caractère oral ou écrit de plusieurs ressources, en langue anglaise et japonaise[47].

Nous avons ensuite proposé une approche de l'homogénéité des ressources textuelles. Ayant défini l'homogénéité d'une ressource linguistique comme l'ensemble des variations internes et des régularités la caractérisant, nous avons étudié comment quantifier ces variations à l'aide du coefficient de similarité défini précédemment. L'application d'un tel coefficient permet en effet de visualiser des variations internes, en termes par exemple de style ou de registre de langue, et de les représenter sous la forme d'une distribution de coefficients attribués à des sous-parties de la ressource

[47] Denoual, *A method to quantify corpus similarity and its application to quantifying the degree of literality in a document*, 2006.

globale. Éventuellement, on a alors la possibilité de modifier une telle distribution en gardant ou en supprimant des données.

Afin de mettre en pratique cette possibilité d'adaptation des données, nous nous sommes proposé de tester l'hypothèse selon laquelle plus les données d'entrainement sont similaires à la tâche, meilleure est la performance d'un système générique de traitement automatique des langues[48]. Une expérience menée sur des modèles de langue et sur des systèmes de traduction par l'exemple a permis de remettre en cause cette hypothèse. Les résultats montrent en effet que l'utilisation exclusive de données proches de la tâche en terme de similarité produit de moins bonnes performances que l'utilisation de données équilibrées, choisies de façon aléatoire.

[48] Denoual, *The influence of data homogeneity on NLP system performance*, 2005.

Partie III

Production de données

Introduction

L'objet de la production, ou génération automatique de données linguistiques est de produire, à partir d'une représentation quelconque du sens, un énoncé de la langue. Sans préciser plus finement de tâche particulière, on rencontre immédiatement un problème de représentation : quelle forme doit prendre la représentation initiale du sens, et quelles opérations sur cette représentation sont les plus appropriées en vue de produire des données linguistiques ?

Similairement à la schématisation des niveaux de traduction par le triangle de Vauquois, on peut considérer plusieurs formes pour la représentation du sens servant à la génération :

Le niveau le plus abstrait, et nécessitant le plus d'efforts d'analyse, est celui d'une représentation intermédiaire pivot. L'usage d'une interlangue, telle qu'IF[49] ou la représentation UNL[50] permet de représenter l'aspect sémantique de l'énoncé indépendamment de la langue dont il est issu. À un niveau moins élevé d'analyse, l'extraction de patrons intermédiaires, appris automatiquement ou décrits à partir de connaissances linguistiques, peut aussi permettre la génération de données linguistiques. Enfin au niveau le plus bas, le sens est contenu dans l'exemple brut, l'énoncé exempt d'analyse présenté dans sa langue d'origine.

A priori l'analyse peut donc s'appliquer sur des chaines de mots, de caractères, ou de syntagmes. Nous allons montrer un résultat étonant : son application en caractères permet de capturer des variations syntagmatiques et paradigmatiques.

L'usage d'une relation spécifique en caractères, l'analogie, permet en effet de traiter la divergence entre les langues, et les commutations et variations syntagmatiques et paradigmatiques présentes dans une langue. Nous effectuons en annexe D une présentation de l'analogie sur les chaines de symboles telle qu'elle est formalisée dans les travaux d'Yves Lepage, puis de son application pratique en caractères sur des données de la langue.

Nous nous intéressons dans cette partie à deux tâches de production de données linguistiques, et montrons que l'analogie sur les chaines de caractères permet de traiter élégamment plusieurs tâches de traitement automatique des langues. Nous montrons ainsi son application successivement en génération automatique de paraphrases, et en traduction automatique. Les deux tâches sont liées : tout comme la traduction automatique vise à traduire d'une langue à une autre, la production de paraphrases peut être vue comme la traduction d'une langue à la même langue[51].

Dans un premier chapitre, nous montrons que l'analogie sur les chaines de caractères, appliquée sur une ressource bilingue, peut permettre de paraphraser les énoncés de chacune des langues de la ressource, tout en conservant un niveau de qualité équivalent à celui de la ressource d'origine.

Dans un deuxième chapitre, nous montrons que l'analogie, appliquée sur une ressource bilingue, peut permettre de traduire d'une langue à l'autre un énoncé non attesté dans la ressource d'origine.

[49] *Interchange Format*, voir Levin *et al.*, *An interlingua based on domain actions for machine translation of task oriented dialogues*, 1998.

[50] Voir http://www.undl.org/ .

[51] Traduire, c'est conserver le sens d'une langue à l'autre ; paraphraser, c'est proposer un énoncé de même sens dans la même langue, mais avec une autre forme.

Chapitre 1

Production de paraphrases

1.1 Introduction au problème de la production de paraphrases

La production de paraphrases représente un cas particulier de génération de données linguistiques. La paraphrase est une opération de reformulation aboutissant à un énoncé contenant le même signifié, mais dont le signifiant est différent. On dit de deux énoncés qu'ils sont paraphrases l'un de l'autre si on peut en faire la même interprétation dans un domaine. Ils ont alors le même signifié[1], mais une forme différente. Il s'agit donc, à partir d'un énoncé, d'en obtenir un autre de même signifié, mais qui présentera des variations lexicales ou syntaxiques par rapport à l'original.

La production de paraphrases prend depuis peu une importance nouvelle, du fait de leur utilisation fréquente dans les méthodes d'évaluation automatiques, comme par exemple BLEU ou NIST en traduction automatique, ou ROUGE[2] en résumé automatique. Pour être mises en œuvre, de telles méthodes ont en effet besoin d'énoncés de référence synonymes, c'est-à-dire d'ensembles d'énoncés ayant le même sens mais différant en forme. Les paraphrases peuvent aussi être utilisées pour enrichir une ressource linguistique, et compenser la raréfaction des données qu'engendre leur utilisation publique : lorsqu'une campagne d'évaluation d'une tâche de traitement automatique des langues a lieu, de grandes quantités de données sont mises à la disposition des participants. De telles données ne peuvent plus être utilisées par la suite dans les campagnes suivantes, puisqu'elles ont déjà été vues, et sont pour ainsi dire déjà « connues ». Produire des ressources linguistiques supplémentaires coute cher en termes de travail humain et de temps, il peut donc être particulièrement intéressant de les produire automatiquement, ou semi-automatiquement.

Plusieurs études ont été menées sur la possibilité de produire des paraphrases. Par exemple, Barzilay[3] propose de les extraire à partir d'un corpus de phrases pa-

[1] Rigoureusement, le sens est une forme abstraite, correspondant à des énoncés qui auraient le même signifié dans toute interprétation. Des énoncés qui sont paraphrases l'un de l'autre n'ont donc pas forcément le même sens, mais ont le même signifié.

[2] Lin & Hovy, *Automatic evaluation of summaries using N-gram co-occurence statistics*, 2003.

[3] Barzilay & McKeown, *Extracting paraphrases from a parallel corpus*, 2001.

rallèles, et Dolan[4] à partir d'un corpus monolingue. Zhang et Yamamoto[5] proposent une méthode de génération par patrons, Langkilde et Knight[6] utilisent des méthodes statistiques, et Dras[7] des gramaires d'adjonction d'arbres. Power et Scott[8] étudient la paraphrase à plus grande échelle, les énoncés constitués de plusieurs phrases, de paragraphes ou de documents entiers. Enfin, certains groupes de recherche comme celui dirigé par Bill Dolan chez Microsoft[9], mettent à disposition les corpus de paraphrases qu'ils ont constitués à des fins de recherche.

Dans cette partie, nous présentons une méthode pour produire des paraphrases à partir d'une phrase de référence contenue dans une grande ressource multilingue. Si l'on a dans l'idée de produire des ensembles de références destinées par exemple à l'évaluation de la traduction automatique, alors on veut couvrir un maximum de variations au niveau des termes employés, et de la structure[10]. Il peut donc être intéressant de quantifier et de maximiser les variations lexicales et syntaxiques au sein d'un même ensemble. Nous utilisons l'analogie proportionnelle sur les chaines de caractères afin de saisir les commutations présentes dans une grande ressource linguistique, ainsi qu'une méthode de filtrage fondée sur l'attestation de chaines de caractères de longueur N, afin de contrôler la qualité en forme et en sens. Cette méthode est présentée en partie II, au chapitre 1. Notre ambition ici est de montrer une méthode complète de production de paraphrases, entièrement fondée sur les chaines de caractères, ne nécessitant aucune division en unité supérieure telle que celle de mots.

1.2 Méthode proposée

1.2.1 Algorithme global

La méthode que nous proposons[11] avec Yves Lepage peut se décomposer en deux phases successives, que nous décrivons tour à tour plus bas :

- la phase de **détection** : les phrases ayant une même traduction dans la ressource multilingue, sont recensées. Les paraphrases sont donc détectées par égalité de traduction ;
- la phase de **génération** : ensuite, nous générons de nouveaux énoncés à partir de ces données, en exploitant par analogie leurs commutations linguistiques internes. De nouvelles phrases sont produites, dont nous pouvons contrôler la combinatoire par des contraintes liées à la contigüité.

Nous expliquons ces phases en détail dans les sections suivantes.

[4] Dolan *et al.*, *Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources*, 2004.

[5] Zhang & Yamamoto, *Paraphrasing of Chinese utterances*, 2002.

[6] Langkilde & Knight, *Generation that exploits corpus-based statistical knowledge*, 1998.

[7] Dras, *Tree adjoining grammar and the reluctant paraphrasing of text*, 1999.

[8] Power & Scott, *Automatic generation of large-scale paraphrases*, 2005.

[9] Voir les travaux de Bill Dolan, Chris Brockett et Chris Quirk : Microsoft Research Paraphrase Corpus, `http://research.microsoft.com/research/nlp/msr_paraphrase.htm`.

[10] Babych & Hartley, *Modelling legitimate translation variation for automatic evaluation of MT quality*, 2004.

[11] Lepage & Denoual, *Automatic generation of paraphrases to be used as translation references in objective evaluation measures of machine translation*, 2005.

1.2.2 Initialisation : détection des paraphrases dans la ressource d'origine

Dans cette première phase, nous commençons par initialiser les données en détectant les paraphrases dans la ressource multilingue utilisée. Une paraphrase étant une équivalence en sens dans une interprétation donnée, des phrases traductions d'une même phrase d'origine peuvent être vues comme équivalentes en sens, c'est-à-dire qu'elles sont paraphrases les unes des autres. On peut appliquer ce principe[12] à la ressource dont nous disposons : dans un corpus multilingue, on trouve pour toutes les phrases de la ressource les traductions d'une même phrase en plusieurs langues. Par exemple, les phrases suivantes en langue anglaise partagent une même traduction japonaise (en gras ci-dessous). Par conséquent, elles sont paraphrases l'une de l'autre :

A beer, please.	**ビールをください。**
	ビールを一本。
	ビールを一本ください。
Beer, please.	ビール。
	ビールをお願いします。
	ビールをください。
	ビールを下さい。
	ビール一杯ください。
Can I have a beer?	**ビールをください。**
Give me a beer, please.	**ビールをください。**
I would like beer.	**ビールをください。**
I'd like a beer, please.	**ビールをください。**
I'd like some beer, please.	**ビールをください。**

Une telle démarche est bien entendu criticable : tirées de leur contexte et examinées en dehors de toute contrainte pragmatique, il y a un risque que ces énoncés, utilisés dans des interprétations différentes, ne soient pas réellement paraphrases. Cette méthode nous amène à considérer un problème supplémentaire, que nous approfondirons dans la section 1.3.3, p. 114 : il se peut qu'entre deux énoncés paraphrases on n'ait pas égalité stricte en sens mais une simple implication. Une véritable taxonomie des paraphrases n'ayant pas encore été établie à ce jour, nous essaierons de distinguer quantitativement la proportion des paraphrases produites qui entrent dans la catégorie des paraphrases strictes, de celles qui réalisent une simple implication de sens.

1.2.3 Utilisation des commutations dans les analogies pour la génération de paraphrases

Dans cette deuxième phase, nous implémentons la production de paraphrases proprement dite : toute phrase du corpus peut éventuellement entrer en commutation

[12] On peut retrouver globalement le même principe chez Ohtake & Yamamoto, *Applicability analysis of corpus-derived paraphrases toward example-based paraphrasing*, 2003.

avec d'autres phrases du même corpus. De telles commutations, vues dans des relations d'analogie entre chaines de caractères, mettent en évidence des variations syntagmatiques et paradigmatiques (voir les exemples en annexe D, p. 171). Par exemple, la phrase :

A slice of pizza, please.

entre dans les analogies explicitées dans le tableau 1.1. Si on remplace dans ces analogies certaines de ces phrases par leurs paraphrases déjà connues, alors ces nouvelles analogies peuvent produire de nouvelles phrases. On se sert de cette manière des paraphrases détectées dans la première phase. Par exemple, si l'on remplace la phrase :

A beer, please.

par la phrase :

Can I have a beer?

dans la première analogie dans le tableau 1.1, alors on peut former l'équation analogique qui suit, et la résoudre :

$$\textit{I'd like a beer, please.} : \textit{Can I have a beer?} :: \textit{I'd like a slice of pizza, please.} : x$$
$$\Rightarrow \quad x = \textit{Can I have a slice of pizza?}$$

Il est ainsi légitime de dire que la phrase nouvellement produite :

Can I have a slice of pizza?

est paraphrase de la phrase de départ *A slice of pizza, please.* (voir le tableau 1.2 pour une explication détaillée). Une telle méthode permet de se passer de l'utilisation de patrons tels qu'ils sont utilisés pour la génération[13]. Tous les exemples présents dans le corpus sont dans leur forme brute des patrons potentiels, le choix des commutations revenant à l'analogie entre chaines de caractères.

Tableau 1.1: Exemple d'analogies formées avec des phrases de la ressource linguistique, commutant avec la phrase *A slice of pizza, please.*

I'd like a beer, please.	:	*A beer, please.*	::	*I'd like a slice of pizza, please.*	:	*A slice of pizza, please.*
I'd like a twin, please.	:	*A twin, please.*	::	*I'd like a slice of pizza, please.*	:	*A slice of pizza, please.*
I'd like a bottle of red wine, please.	:	*A bottle of red wine, please.*	::	*I'd like a slice of pizza, please.*	:	*A slice of pizza, please.*

[13] Barzilay & Lee, *Learning to paraphrase: an unsupervised approach using multiple-sequence alignment*, 2003.

Tableau 1.2: Production d'une paraphrase de la phrase d'origine *A slice of pizza, please.* par utilisation de l'analogie sur les chaines de caractères.

I'd like a beer, please.	:	*A beer, please.*	::	*I'd like a slice of pizza, please.*	:	*A slice of pizza, please.*
I'd like a beer, please.	:	*Can I have a beer?*	::	*I'd like a slice of pizza, please.*	:	x
I'd like a beer, please.	:	*Can I have a beer?*	::	*I'd like a slice of pizza, please.*	:	*Can I have a slice of pizza?*

1.2.4 Limitation de la combinatoire par des contraintes sur la contigüité des chaines de caractères

Comme nous l'exposons dans l'annexe D, l'analogie est aveugle. Il est ainsi possible que des phrases parasites soient produites lors de la phase de génération. Par exemple, dans l'analogie citée précédemment dans le tableau 1.2 le remplacement de la phrase :

A beer, please.

par la paraphrase détectée dans la première phase :

A bottle of beer, please.

produira le résultat malencontreux :

A bottle of pizza, please.

Dans cet exemple précis, et comme on le verra dans la section 1.3.3, cela vient du fait que la paraphrase détectée entretient une relation de simple implication avec la phrase d'origine, et non pas d'équivalence. D'autre part, comme exposé en annexe D, étant donné qu'à ce jour la formalisation complète et valide de l'analogie entre chaines de caractères n'est pas complète, l'algorithme utilisé[14] produit parfois des chaines inacceptables telles que :

A slice of pizzthe, pleaset for tha, please.

Afin d'améliorer la qualité globale des productions, il est essentiel de les filtrer pour ne laisser passer que celles qui sont correctes. Nous utilisons ici la méthode de filtrage de la grammaticalité par attestation de chaines de N caractères, étudiée dans la section 1.2.2, car on a vu qu'elle permettait d'atteindre une très haute précision : on élimine d'office les données contenant des séquences de caractères d'une certaine longueur n'apparaissant pas dans les données d'origine. Bien qu'une telle méthode semble triviale, son application donne de bons résultats et est appropriée lorsqu'on est préoccupé par la précision, donc la qualité des phrases retenues, et non par le rappel, le nombre de phrases retenues. Ce procédé va, on l'a vu, dans le sens de

[14] Lepage, *Solving analogies on words: an algorithm*, 1998.

la tendance actuelle en traitement automatique des langues d'utiliser des séquences de N éléments contigus pour évaluer la qualité de différents systèmes[15]. En jouant sur la longueur N, la méthode permet de retenir un nombre satisfaisant de phrases qui sont indubitablement correctes, au moins au sens de la ressource linguistique elle-même.

1.3 Expériences

1.3.1 La ressource utilisée

Nous utilisons pour cette expérience la partie anglaise C-STAR du Basic Traveler's Expression Corpus, ensemble de 162 318 phrases alignées dans plusieurs langues dont l'anglais. Nous décrivons la ressource BTEC en annexe A.1. Nous nous contentons de rappeler ici que les phrases qu'on y trouve sont particulièrement courtes. Le tableau 1.3 donne quelques chiffres caractéristiques des phrases de la partie anglaise du BTEC.

Tableau 1.3: Caractéristiques de la ressource utilisée

Nombre de ≠ phrases	Taille moyenne ± Écart-type	
	en caractères	en mots
97 769	$35,14 \pm 18,81$	$6,86 \pm 3,57$

On estime sur un échantillon de 400 phrases que la qualité de la ressource BTEC est au moins de 99% de phrases correctes, avec un taux de confiance de $98,08\%$. Les phrases incorrectes contiennent uniquement des fautes de frappe, ou de légères erreurs de syntaxe.

1.3.2 Détection et génération des paraphrases

Au cours de la phase de détection, nous dénombrons 26 079 phrases (sur 97 769 phrases) possédant au moins une paraphrase, soit une moyenne de $5,35$ paraphrases par phrase d'origine. Comme en atteste la figure 1.1, la distribution des paraphrases détectées n'est pas uniforme : 60 phrases d'origine donnent plus de 100 paraphrases. Le maximum est atteint pour la très courte phrase *Sure.* Le sens d'une telle phrase dépendant en grande partie du nombre a priori important de contextes différents dans lesquels elle peut être employée, cela explique qu'on lui trouve un grand nombre de paraphrases. En voici une liste partielle :

[15] ROUGE, voir Lin & Hovy, *Automatic evaluation of summaries using N-gram co-occurence statistics*, 2003 pour le résumé automatique ou encore BLEU, voir Papineni *et al.*, *BLEU: a method for automatic evaluation of machine translation*, 2002, et NIST, voir Doddington, *Automatic evaluation of machine translation quality using N-gram co-occurrence statistics*, 2002 pour la traduction automatique, etc.

Sure. Here you are.
Sure. This way, please.
Certainly, go ahead, please.
I'm sure I will.
No, I don't mind a bit.
Okay. I understand quite well, thank you.
Sounds fine to me.
Yes, I do.
...

Un tel exemple montre en substance que plus on obtient de paraphrases par une telle méthode, moins bonne semble leur qualité.

Au cours de la phase de génération, la méthode produit 4 495 266 phrases anglaises à partir de la ressource BTEC. L'inspection manuelle d'un échantillon de 400 phrases révèle que $23,6\%$ des phrases sont correctes en forme et en sens, avec un taux de confiance de 98.81%. Le tableau 1.4 montre un exemple d'ensemble de paraphrases candidates obtenues à partir d'une même phrase d'origine, *Can we have a table in the corner?* Les candidates ont été filtrées par les chaines de caractères de longueur $N = 20$ (voir plus bas). On remarque que la phrase d'origine a été produite à nouveau par la méthode (quatrième phrase à partir du haut). La colonne de chiffres sur la gauche montre la fréquence avec laquelle chaque paraphrase a été produite.

Tableau 1.4: Paraphrases candidates pour la phrase d'origine *Can we have a table in the corner?*

43	*Could we have a table in the corner?*
43	*I'd like a table in the corner.*
43	*We would like a table in the corner.*
28	*Can we have a table in the corner?*
5	*Can I get a table in the corner?*
5	~~*In the corner, please.*~~
4	~~*We'd like to sit in the corner.*~~
2	~~*I'd like to sit in the corner.*~~
2	~~*I would like a table in the corner.*~~
2	*We'd like a table in the corner.*
1	~~*I'd prefer a table in the corner.*~~
1	*I prefer a table in the corner.*

Dans l'expérience de la section 1.2.2, p. 69, et de façon à obtenir une qualité comparable à celle de la ressource d'origine, nous avions évalué la qualité de la ressource d'origine par échantillonnage (voir section 1.3.1). Après tests, et pour notre expérience présente de paraphrasage, c'est la valeur $N = 20$ qui nous permet d'obtenir une qualité comparable.

Parmi les phrases d'origine, 16 153 phrases produisent au moins une paraphrase. Nous avons généré 147 708 paraphrases[16] avec une moyenne de $8,65$ paraphrases par phrase d'origine et un écart-type de $16,98$. La distribution est déséquilibrée, comme

[16]Une même phrase peut avoir été produite plusieurs fois à partir de plusieurs phrases d'origine. En tout, nous avons produit 42 249 phrases différentes, dont la longueur en caractères et en mots est donnée dans le tableau 1.5.

en atteste le graphe donné en figure 1.2, p. 120, qui montre le nombre de phrases d'origine produisant un même nombre de paraphrases. Le graphe de la figure 1.3, p. 121, montre le nombre de phrases d'origine produisant un même nombre de paraphrases en fonction de la longueur en caractères de la phrase d'origine. Le graphe de la figure 1.4, p. 121, montre lui le nombre de phrases d'origine produisant un même nombre de paraphrases en fonction de la longueur en mots de la phrase d'origine.

Tableau 1.5: Nombre, taille moyenne et écart-type des paraphrases produites.

Nombre de phrases différentes	Taille moyenne ± Écart-type en caractères	en mots
42 249	33.15 ± 9.31	6.44 ± 1.90

Le tableau 1.5 montre le nombre, la taille moyenne et l'écart-type des paraphrases produites, des chiffres qui sont directement comparables à ceux de la ressource originale (voir le tableau 1.3).

1.3.3 Qualité des paraphrases produites

Évaluation en forme : grammaticalité

De façon similaire à la démarche adoptée dans la section 1.3.1, nous estimons la qualité grammaticale des paraphrases produites sur un échantillon de 400 phrases : au moins 99% des paraphrases sont correctes, avec un taux de confiance de 98%. Cette qualité est similaire à celle de la ressource de départ BTEC, dont la qualité était elle aussi supérieure à 99% avec un taux de confiance de 98%. Une rapide analyse des erreurs présentes dans les paraphrases produites automatiquement montre que les erreurs sont du même type que celles présentes dans les données de la ressource linguistique de départ. Par exemple, on remarque dans la phrase produite qui suit l'absence d'article devant le groupe nominal *tourist area* :

Where is tourist area?

or, on trouve dans les données présentes dans le BTEC des phrases comportant le même type d'erreur, par exemple :

Where is information office?

De telles erreurs dans les données d'origine rendent possible des commutations malencontreuses et expliquent ainsi une grande partie des erreurs présentes dans les paraphrases produites. C'est pourquoi il nous est impossible d'exiger une qualité absolue et que nous visons donc seulement une qualité au moins égale à celle de la ressource d'origine.

Évaluation en sens : équivalence ou implication

Au delà d'une évaluation en forme des productions, l'évaluation de la prodution de paraphrases implique que l'on vérifie l'équivalence sémantique des énoncés. Il nous parait dès lors utile d'aborder le problème de l'équivalence en sens d'une manière plus nuancée. L'équivalence stricte entre deux énoncés est souvent discutable : on

peut citer par exemple Fujita[17], qui dans sa thèse différencie bien connotation et dénotation dans son approche de la paraphrase. Par exemple, entre les deux phrases :

C'est un homme mince.
C'est un homme maigre.

on a une différence de connotation. Le terme *mince* a une connotation plutôt positive, alors que *maigre* a une connotation négative. Pourtant, les deux énoncés décrivent la même information factuelle. En revanche, entre les deux phrases appliquées à la même situation :

Cet animal est très maigre.
Ce chat est très maigre.

on a une différence de dénotation. La première phrase décrit la condition d'un *animal*, la deuxième précise qu'il s'agit d'un *chat*. Si un chat est bien un animal, on ne peut dire en revanche que tous les animaux sont des chats.

On a donc une simple implication en sens : la deuxième phrase comporte une information en plus par rapport à la première.

Dans le cadre notre étude, et ainsi que le fait Bill Dolan dans son introduction au corpus de paraphrases Microsoft[18], nous ne considérons pas que les différences de connotation remettent en question l'équivalence en sens : de tels énoncés sont donc vus par nous comme strictement équivalents.

Nous ne considérons pas non plus que des différences emphatiques telles que celles relevées par Dras[19] remettent en question l'équivalence en sens :

Il a recouvert de peinture ce mur.
C'est de peinture qu'il a recouvert ce mur.

Différences en connotation et différences emphatiques ne remettent donc pas en question la relation d'équivalence de sens dans notre étude. En revanche, nous considérons que la différence de dénotation indique une simple implication en sens entre deux énoncés.

Nous différencions donc bien dans notre évaluation les relations d'équivalence, ou de simple implication, qu'entretiennent les paraphrases entre elles. Ainsi les paraphrases suivantes, en colonne de gauche, seront vues comme entretenant une relation d'équivalence avec les phrases d'origine données à droite :

Can I see some ID?	$\Longleftrightarrow$	*Could you show me some ID?*
Please exchange this.	$\Longleftrightarrow$	*Could you exchange this, please.*
Please send it to Japan.	$\Longleftrightarrow$	*Send it to Japan, please.*

En revanche, les paraphrases produites suivantes, en colonne de gauche, seront vues comme entretenant une relation de simple implication de sens avec les phrases d'origine données à droite :

[17] Fujita, *Automatic generation of syntactically well-formed and semantically appropriate paraphrases*, 2005, p.4 .

[18] Voir `http://research.microsoft.com/research/nlp/msr_paraphrase.htm`.

[19] Dras, *Tree adjoining grammar and the reluctant paraphrasing of text*, 1999.

Coke, please.	$\Longleftarrow$	*Miss, could I have a coke?*
I want to change money.	$\Longrightarrow$	*Please exchange this.*
Sunny-side up, please.	$\Longleftarrow$	*Fried eggs, sunny-side up, please.*

La qualité en sens des paraphrases produites est jugée manuellement sur un échantillon de 470 paraphrases, qui sont comparées directement avec leurs phrases d'origine. Les résultats de cet échantillonnage montrent que les paraphrases candidates peuvent être considérées comme des paraphrases valides dans au moins 94% des cas, avec un taux de confiance de 97%. Les phrases suivantes sont données en exemple de ce qui n'a pas été jugé comme des paraphrases valides :

Do you charge extra if I drop it off?	$\Longleftrightarrow\!\!\!\!\!\!/\;\;$	*There will be a drop off charge.*
Here's one for you, sir.	$\Longleftrightarrow\!\!\!\!\!\!/\;\;$	*You can get one here.*
There it is.	$\Longleftrightarrow\!\!\!\!\!\!/\;\;$	*Yes, please sit down.*

Les résultats de l'évaluation sont détaillés ci-dessous dans le tableau 1.6.

Tableau 1.6: Évaluation d'un échantillon de 470 paraphrases produites à partir de plusieurs phrases d'origine, en termes d'équivalence ou d'implication en sens.

Paraphrase		Non paraphrase
Equivalence	Implication	
346	104	20

1.3.4 Mesure de la variation lexico-syntaxique des paraphrases

Mesures objectives

La raison première de notre étude sur la production automatique de paraphrases était de produire des ensembles de phrases références utilisables dans l'évaluation automatique de la traduction automatique, par des méthodes fondées uniquement sur l'unité de caractère. On désire donc que les ensembles de phrases références produits présentent le plus de variations lexico-syntaxiques possibles.

Nous avons évalué la variation lexico-syntaxique des paraphrases produites sur un échantillon de 400 phrases d'origine, à l'aide des mesures BLEU et NIST. Dans le cas de la traduction automatique, on évalue une phrase candidate afin de juger de la qualité de sa traduction par un système : le but est d'obtenir une grande similarité avec les phrases de référence. Des scores élevés en BLEU ou en NIST reflètent pour une phrase une bonne corrélation avec ses références. À l'inverse du cas de l'évaluation de la traduction automatique, où l'on cherche à maximiser BLEU et NIST, nous cherchons ici à minimiser ces scores.

En effet, dans le cas de la production de paraphrases, la situation est différente : on a déjà vérifié, via une évaluation manuelle, que l'équivalence en sens était conservée. Les mesures d'évaluation objective ne vont donc pas être utilisées pour juger la qualité en sens puisque cela a déjà été fait. Nous désirons en revanche qu'au sein d'un ensemble produit, les phrases présentent le plus possible de variations lexico-syntaxiques entre elles. Nous préférons donc qu'elles soient le moins similaires possibles entre elles afin de refléter l'ensemble des expressions possibles d'un même sens. C'est pourquoi nous recherchons les scores en BLEU et en NIST les plus faibles.

Cette démarche n'est valable que dans le cas où les phrases jugées sont grammaticalement correctes, et sont réellement paraphrases les unes des autres. Nous avons vérifié que c'était le cas, les évaluations précédentes ayant montré d'une part que les phrases produites par notre méthode sont correctes grammaticalement dans 99% des cas, et d'autre part qu'elles sont des paraphrases valides de leurs phrases d'origine dans 94% des cas.

Les mesures BLEU et NIST sont supposées mesurer deux aspects complémentaires d'une traduction : respectivement la *fluidité*, et la *fidélité* (ou l'informativité). BLEU refléterait plutôt la qualité dans la forme de l'expression[20] alors que NIST a tendance à mesurer la qualité informationnelle de la traduction. En pratique, un score BLEU est compris entre 0 et 1, alors qu'un score NIST n'est pas de borné : il est donc difficile de comparer des scores pour des phrases d'origine différentes. Afin de pouvoir établir une telle comparaison, nous normalisons donc tout score NIST obtenu ici par le score NIST de la phrase d'origine obtenu par calcul sur elle-même.

Résultats

Les scores en BLEU et NIST présentés sur les figures 1.5 et 1.6, p. 122 peuvent être interprétés comme des mesures de la variation lexico-syntaxiques entre paraphrases, comme on l'a montré dans le paragraphe précédent. Sur ces figures, chaque point représente le score d'un ensemble de paraphrases calculé sur la phrase d'origine correspondante (la moyenne est tracée en pointillés). Plus un score est bas, et plus grandes sont les variations dans l'ensemble de paraphrases. Les graphes de la figure 1.5 montrent que cette variation dépend beaucoup de la longueur des phrases d'origine. Plus les phrases d'origine sont courtes et plus la variation produite est importante. Comme on l'a vu plus haut avec l'exemple de la phrase *Sure.*, la phase de détection introduit un certain biais dans la méthode.

Les graphes de la figure 1.6 montrent que la variation ne dépend pas du nombre de paraphrases produites par phrase d'origine. À l'opposé d'une méthode qui produirait plus de variations lorsque plus de paraphrases sont produites, dans notre méthode la variation ne semble pas changer significativement lorsqu'on produit des paraphrases supplémentaires (mais la qualité grammaticale ou en équivalence de sens peut varier).

La méthode est donc paramétrable : il est envisageable de choisir le nombre de paraphrases à produire à l'avance, sans influencer les variations lexico-syntaxiques.

Comparaison avec des ensembles de paraphrases produits à la main

La méthode présentée ici permet de produire des ensembles de paraphrases pouvant être utilisés par exemple comme phrases références dans une tâche d'évaluation de la

[20] Akiba *et al.*, *Overview of the IWSLT04 evaluation campaign*, 2004.

traduction automatique. Nous avons de plus montré comment quantifier les variation lexico-syntaxiques de ces paraphrases. Afin d'être complet, il ne reste donc plus qu'à comparer les variations lexico-syntaxiques d'ensembles produits automatiquement par notre méthode, avec celles d'ensembles constitués à la main pour une campagne d'évaluation de la traduction automatique ayant eu lieu dans le passé[21] pour deux paires de langues : japonais-anglais, et chinois-anglais.

Pour tous les ensembles de référence, nous évaluons le score de chaque phrase sur une phrase choisie au hasard et laissée de côté[22]. La moyenne de ces scores donne une indication de la variation lexico-syntaxique globale dans les ensembles de références. Plus les scores sont faibles, et plus cette variation est importante. On applique cette méthode aux ensembles constitués à la main d'une part, et produits automatiquement d'autre part. Les scores sont rassemblés ci-dessous dans le tableau 1.7.

Tableau 1.7: Mesure des variations lexico-syntaxiques d'ensembles de références produits à la main, et produits automatiquement.

Ensemble de références	Moyenne BLEU	Moyenne NIST
Produit automatiquement	0,11	**0,39**
Produit à la main 1	**0,10**	0,49
Produit à la main 2	0,11	0,49

Les scores BLEU sont comparables pour tous les ensembles de références. Il n'y a donc pas de différence flagrante en terme de fluidité de l'expression. En revanche, la moyenne des scores NIST est plus faible : les ensembles produits automatiquement par notre méthode semblent donc présenter plus de variations lexico-syntaxiques que les ensembles constitués à la main.

Conclusion

Nous avons proposé dans ce chapitre une méthode de production de paraphrases, en particulier en vue de leur utilisation dans une tâche d'évaluation automatique de la qualité de la traduction automatique avec des mesures telles que BLEU ou NIST. La méthode opère intégralement en caractères : après une première détection de paraphrases dans la ressource de départ, l'analogie proportionnelle appliquée sur les chaines de caractères permet de produire une grande quantité de phrases candidates. De telles phrases sont ensuite elles mêmes filtrées par une méthode fondée sur l'attestation de chaines de N caractères[23]. Dans une expérience, et en partant d'une ressource de départ de 97 769 phrases uniques, le BTEC, nous avons été en mesure de produire en moyenne $8,65$ paraphrases pour 16 153 phrases d'origine.

Nous avons évalué par échantillonnage la qualité grammaticale des phrases produites. Elle est correcte dans au moins 99% des cas, avec un taux de confiance de 98%, et a donc une qualité comparable à celle de la ressource BTEC. De plus, au

[21] Il s'agit de la campagne d'évaluation IWSLT 2004 (*International Workshop for Spoken Language Translation*), qui a pris place à Keihanna, au Japon en septembre 2004. Voir Akiba *et al.*, *Overview of the IWSLT04 evaluation campaign*, 2004 pour plus de détails.

[22] Principe plus connu sous l'appelation *leaving one out*.

[23] Méthode qui est étudiée en partie 1.2.2, p.69

moins 96% des ces phrases sont véritablement paraphrases et elles entretiennent avec la phrase d'origine une relation d'équivalence en sens ou d'implication, avec un taux de confiance de 97%.

Enfin, nous avons proposé une méthode permettant de quantifier les variations lexico-syntaxiques d'un ensemble de paraphrases à l'aide des mesures BLEU et NIST calculées sur la phrase d'origine. Nous avons montré que la quantité de variations lexico-syntaxiques ne dépendait pas du nombre de paraphrases produites, mais de la longueur de la phrase de départ : plus cette phrase est courte, plus elle peut être interprétée de façons différentes en fonction du contexte où elle est employée, ce qui explique le nombre important de paraphrases pouvant être produites dans ce cas.

L'avantage de cette méthode de production de paraphrases, dans le cas où l'on veut produire des phrases références pour l'évaluation de la traduction automatique, est le suivant : non seulement elle produit des phrases qui sont correctes en forme, et qui sont véritablement des paraphrases, mais en outre les ensembles de paraphrases produits présentent des variations lexico-syntaxiques internes légèrement supérieures à celles observées dans des ensembles constitués manuellement.

La méthode fonctionne sur les caractères, mais les bons résultats obtenus le sont dans des dimensions bien supérieures. Nous obtenons en effet de bons résultats en termes de variation lexico-syntaxique. On attribue d'habitude de telles variations au niveau des mots (lexique) ou des groupes de mots (syntaxe). Nous parvenons de plus à obtenir de vraies paraphrases, c'est-à-dire à agir au niveau du sens, et de la signification (à l'échelle des mots et des phrases).

Une méthode en caractères a été capable de toucher à ces niveaux, que l'on a l'habitude de traiter par les mots et les groupes de mots, voire par des représentations sémantiques.

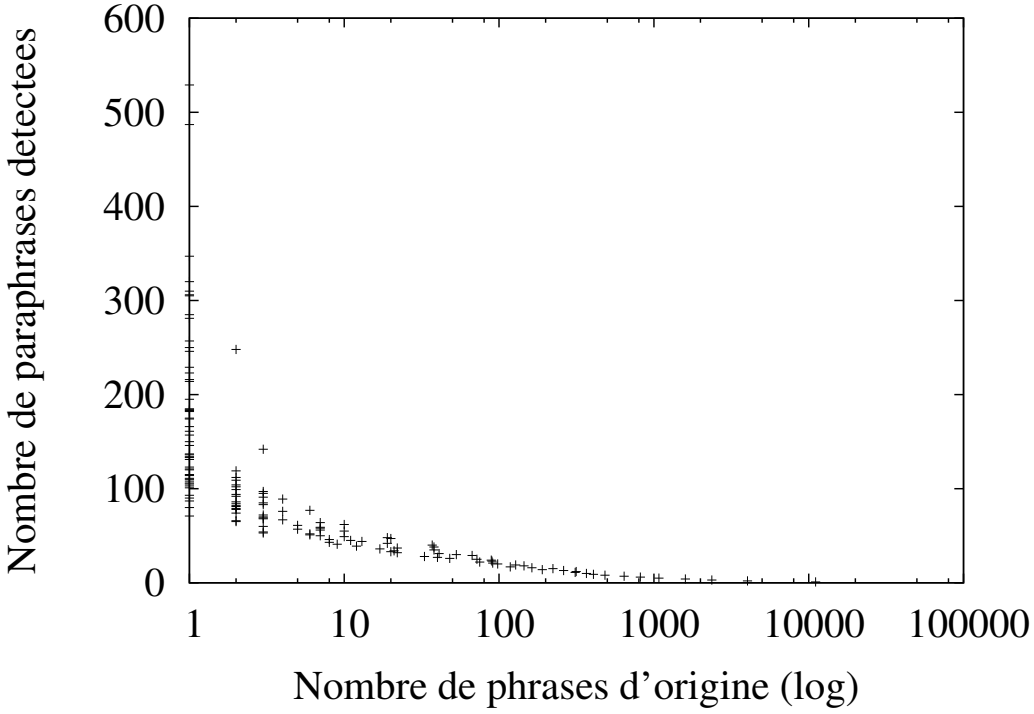

Figure 1.1: Nombre de paraphrases détectées, par phrase de la ressource originale.

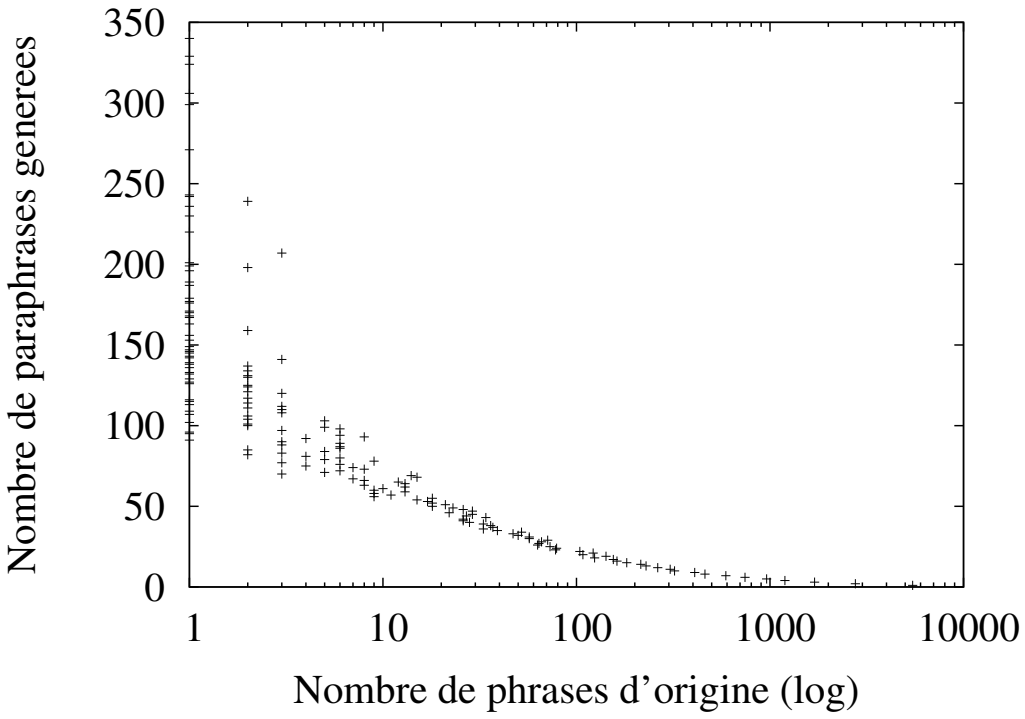

Figure 1.2: Nombre de phrases d'origine produisant un même nombre de paraphrases.

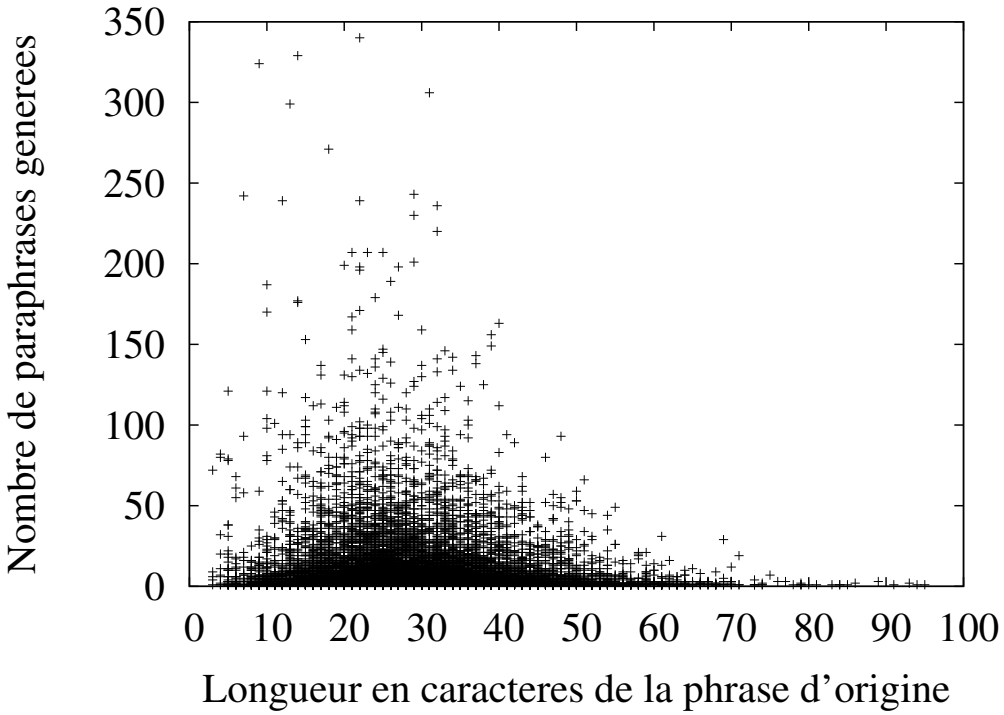

Figure 1.3: Nombre de paraphrases produites en fonction de la longueur de la phrase d'origine en caractères.

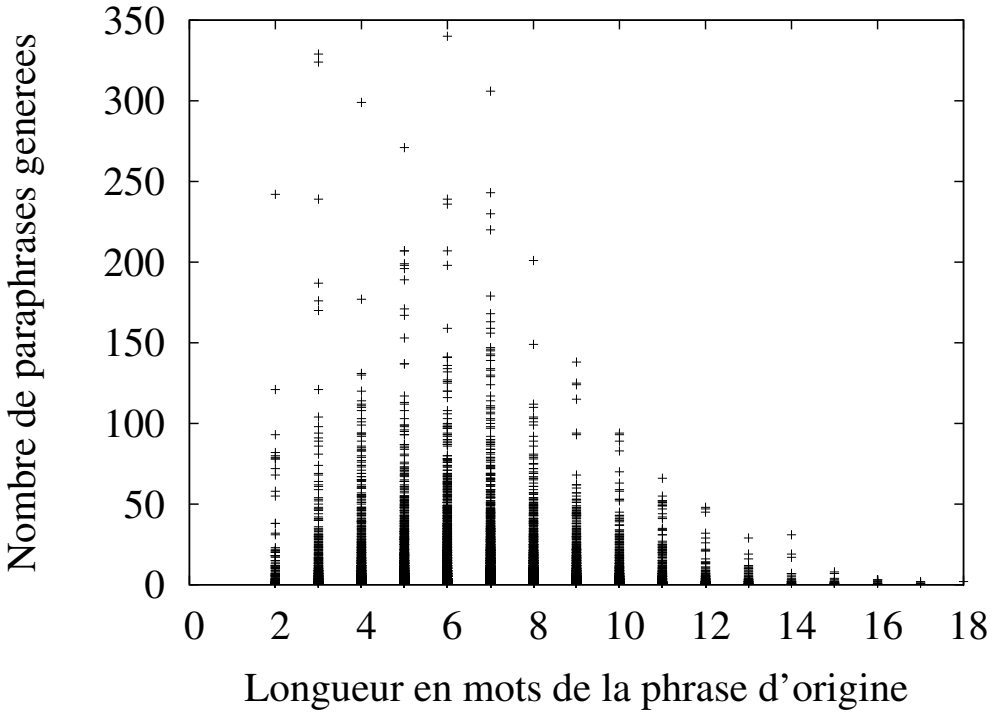

Figure 1.4: Nombre de paraphrases produites en fonction de la longueur de la phrase d'origine en mots.

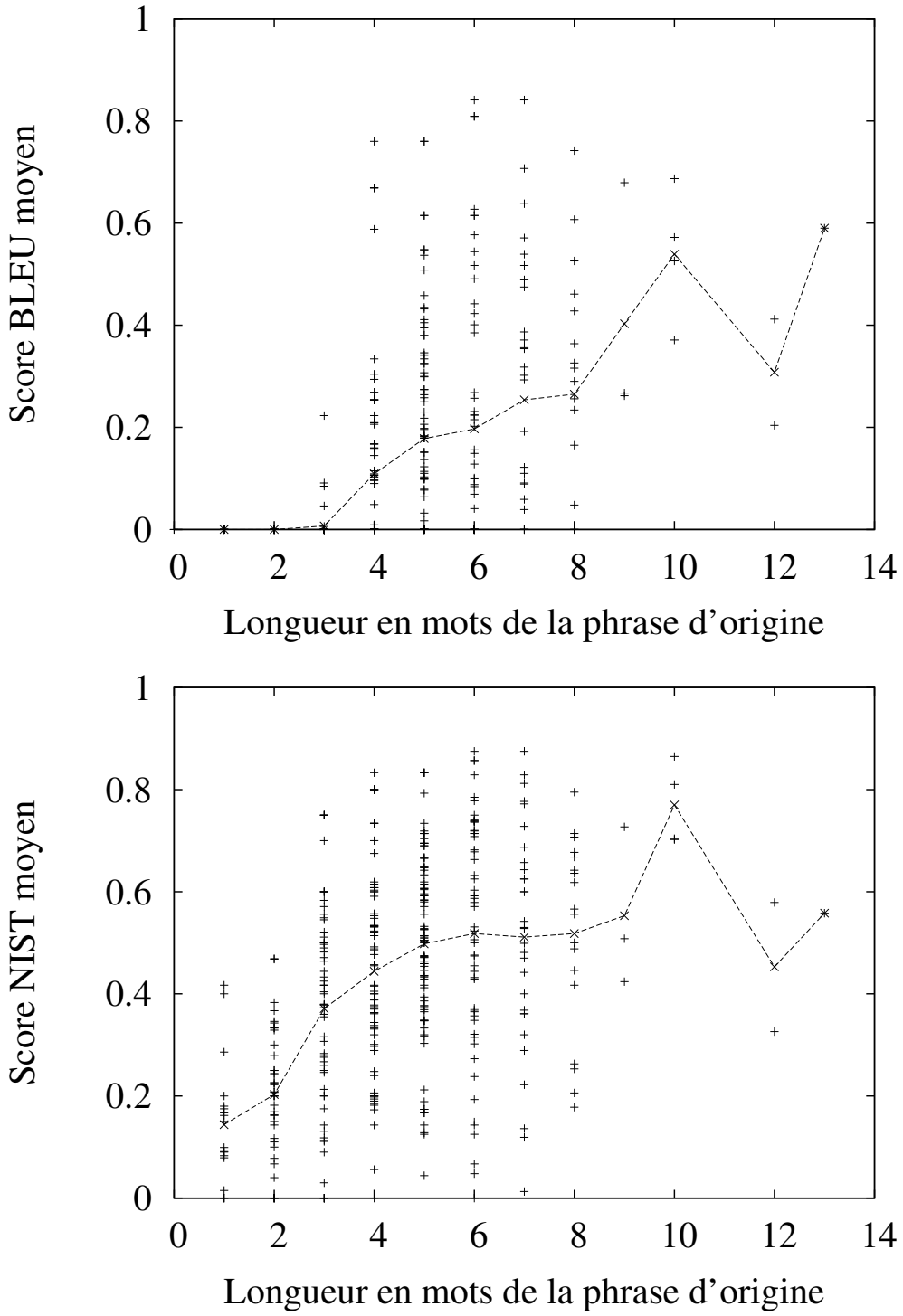

Figure 1.5: Scores BLEU et NIST en fonction de la longueur en mots de la phrase d'origine.

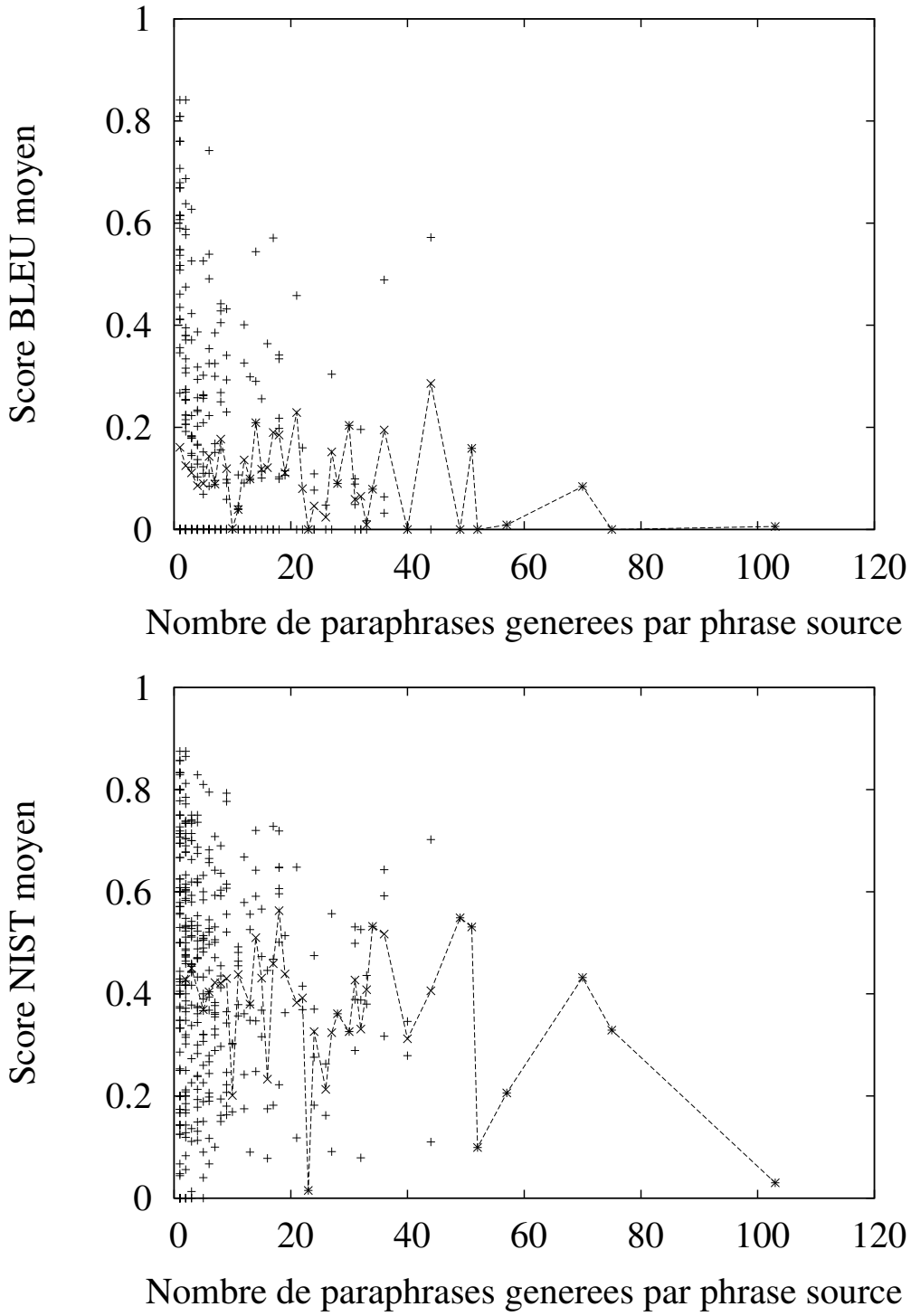

Figure 1.6: Scores BLEU et NIST en fonction du nombre de paraphrases produites par phrase d'origine.

Chapitre 2

Traduction automatique

2.1 Introduction aux problèmes généraux de la traduction automatique

En recherche en traduction automatique, la tendance dominante actuellement est l'approche fondée sur les données. L'approche fondée sur les données se partage elle même entre deux sous-approches aux conceptions de départ très différentes[1] : l'approche par l'exemple, et l'approche statistique. Bien qu'on assiste dans les faits à une tendance à la convergence des systèmes par l'exemple et statistiques[2], les façons de concevoir et d'approcher le problème de la traduction automatique y sont radicalement différentes.

À l'opposé de l'attitude adoptée en traduction automatique par l'exemple, les différentes approches actuelles fondées sur les statistiques traitent les données linguistiques comme des données quelconques, probablement parce qu'historiquement elles utilisent des techniques provenant du domaine de la reconnaissance automatique de parole. Paradoxalement, alors qu'en reconnaissance automatique de parole on tente de reconnaitre des unités cohérentes (phonèmes), qui sont en quelque sorte la base du travail en linguistique générale, et auxquelles on applique avec plus ou moins de bonheur des techniques de traitement du signal, en traduction automatique par approche statistique, on traite traditionnellement les données en unité de mot. La phrase est toujours vue comme une suite de mots, séparés par des blancs ou de la ponctuation. Dans le cas où ces séparateurs n'existent pas dans le système d'écriture de la langue considérée, les données sont segmentées préalablement par des outils spécifiques.

En traduction automatique par approche statistique, la tendance la plus récente de ces dernières années est de travailler sur une unité encore plus grande que celle des mots : après une segmentation en mots, les atomes sont regroupés statistiquement en groupes de mots, intuitivement proches du syntagme. Ces groupes de mots sont couramment appelés en anglais *phrases*, bien que ce terme ne recouvre pas le sens linguistique qu'il a en anglais. Une *phrase* statistique est une simple suite de mots. Cette approche cherche par là à gérer des dépendances à plus longue distance, ainsi qu'à traduire des expressions complexes autrement que mot à mot.

Le domaine de la traduction automatique statistique se heurte entre autres au

[1] Pour ne pas dire antagonistes.

[2] On relève de plus en plus d'usages d'appellations du type *Example-based SMT* et même *Statistical EBMT*, voir par exemple Wu, *MT model space : statistical vs. compositional vs. example-based machine translation (EBMT-II panel on future directions of EBMT)*, 2005.

fait qu'elle considère le mot comme atome de base. Comme nous l'avons vu, cette unité n'a d'une part pas de réalité en linguistique générale, et d'autre part nécessite invariablement des prétraitements pour être dégagée. Ce problème se répercute comme on l'a vu dans la section 2.3 jusqu'au domaine de l'évaluation de la traduction automatique.

Après avoir appliqué l'unité de caractère à diverses tâches du traitement automatique des langues, à l'évaluation automatique, à la détection de grammaticalité, à la caractérisation de données et à la génération de paraphrases, on peut légitimement imaginer que l'utilisation d'une unité plus élémentaire, incontestable et donc plus universelle pourra de la même façon apporter une plus grande simplicité au domaine de la traduction automatique. L'intérêt de l'étude décrite ci-dessous est de proposer une approche pour la traduction automatique entièrement fondée sur le traitement de chaines de caractères, et qui ne nécessite aucun prétraitement des données.

À la suite des travaux sur la sur la traduction automatique par analogie effectués par Yves Lepage[3], nous allons présenter des expériences qui mettent en lumière les performances d'un système de traduction automatique opérant sur les chaines caractères.

2.1.1 Spécificité des données linguistiques

Nous sommes convaincu que les tâches de traitement automatique des langues sont spécifiques, car les données qu'on y manipule sont spécifiques. Nous allons montrer qu'une opération spécifique réalisée uniquement entre chaines de caractères, l'analogie, permet de traiter élégamment la traduction automatique et en plus d'arriver à un bon compromis entre le temps de calcul et la qualité des résultats. Contrairement à l'approche statistique qui nécessite l'application de traitements préalables intensifs, cette méthode a l'avantage de ne nécessiter aucun prétraitement particulier.

Toute donnée linguistique appartient de fait à une langue particulière, qui constitue un système au sens de Saussure. Il serait plus logique de traiter des données linguistiques par une opération qui saisisse la systématicité de la langue. Une telle systématicité apparait de façon explicite dans les commutations que présentent des analogies comme :

$$\begin{array}{ccccccc} \textit{Je voudrais ouvrir cette fenêtre.} & : & \textit{Pourriez vous ouvrir la fenêtre ?} & :: & \textit{Je voudrais encaisser ces chèques de voyage.} & : & \textit{Pourriez vous encaisser les chèques de voyage ?} \end{array}$$

De telles commutations font apparaitre des variations paradigmatiques et syntagmatiques, et rendent possibles des variations lexicales et syntaxiques qui peuvent être utilisées dans un système de traduction automatique.

2.1.2 Problème des divergences entre langues

La traduction automatique pose des problèmes spécifiques au traitement automatique des langues. L'un d'eux constitue le cœur de l'activité de traduction : le traite-

[3]Voir Lepage, *Translation of sentences by analogy principle*, 2005 et Lepage & Denoual, *The 'purest' ever built EBMT system: no variable, no template, no training, examples, just examples, only examples*, 2005.

ment des divergences entre les langues.

Un exemple classique de divergence entre les langues est l'échange des arguments d'un prédicat, illustré par le professeur Vauquois, entre le français et l'anglais :

Elle$_1$ *lui*$_2$ *plaît.* $\leftrightarrow$ *He*$_2$ *likes her*$_1$.

Afin de confirmer l'importance du phénomène, une étude réalisée par Habash[4] a été effectuée sur un échantillon de 19 000 traductions entre l'espagnol et l'anglais. Il montre que pas moins d'une phrase sur trois présente des divergences avec sa traduction. Ces divergences peuvent être classifiées en 5 types : elles peuvent être catégorielles, conflationnelles, structurales, réaliser un échange de la tête de phrase, ou thématiques. Voici un exemple du 4ème type de divergences recensées dans la traduction d'un verbe espagnol en une préposition anglaise :

1: *Atravesó*$_{\text{V}}$
2: *el río*$_{\text{N}}$
3: *flotando*$_{\text{participe.}}$

$\leftrightarrow$

0: *It*
3: *floated*$_{\text{V}}$
1: *across*$_{\text{prep.}}$
2: *the river*$_{\text{N}}$

où le verbe espagnol *atravesó* correspond à la préposition anglaise *across*. Cet exemple précis montre que des approches fondées sur l'unité de mot négligent le fait que dans des langues différentes, l'information correspondante est distribuée sur toute la chaine, et ne correspond pas nécessairement à des mots entiers. Ainsi, la correspondance qu'on fait apparaitre entre les mots de l'exemple ci-dessus n'est pas assez détaillée. La terminaison *-ó* du premier mot correspond au passé de la troisième personne du singulier. Non seulement *atravesó* correspond à la préposition anglaise *across*, mais elle correspond aussi à un autre mot entier anglais (le pronom *it*), et à une partie d'un autre mot anglais (la terminaison *-ed* de *floated*).

Nous proposons de remédier à ce problème en utilisant une méthode fondée sur le caractère : l'analogie proportionnelle sur les chaines de caractères permet lors de la traduction de distribuer l'information sur l'intégralité de la chaine. Les correspondances entre langue source et langue cible dans les analogies sont entièrement et seules responsables de la sélection des bons lemmes, et de leur ordre dans la phrase. Reprenons l'exemple de traduction de l'espagnol vers l'anglais cité ci-dessus :

$$\begin{array}{ccccccc} \textit{They swam in the sea.} & : & \textit{They swam across the river.} & :: & \textit{It floated in the sea.} & : & \textit{It floated across the river.} \\ \updownarrow & & \updownarrow & & \updownarrow & & \updownarrow \\ \textit{Nadaron en el mar.} & : & \textit{Atravesaron el río nadando.} & :: & \textit{Flotó en el mar.} & : & x \end{array}$$

Utiliser le caractère dans la résolution de l'équation analogique permet de produire la traduction exacte de *It floated across the river*, si les trois couples de phrases sur la gauche forment des couples de traductions valides. La phrase correcte obtenue en espagnol est $x =$ *Atravesó el río flotando.*

Dans la suite, nous allons montrer qu'un système de traduction automatique par l'exemple fondé sur l'usage exclusif de l'analogie permet de traiter élégamment le problème de divergences entre les langues. Avant cela, nous présentons le système et la méthode utilisée.

[4] Habash, *Generation-heavy hybrid machine translation*, 2002.

2.2 Traduction automatique par l'exemple fondée sur l'analogie

2.2.1 Exposé de la méthode

Supposons que l'on dispose d'un corpus bilingue de phrases alignées, qu'on appellera un bicorpus. L'algorithme pour la traduction d'une phrase unique peut s'exprimer de la façon suivante[5] :

- Former toutes les équations analogiques mettant en relation la phrase d'entrée D, avec tous les couples de phrases (A_i, B_i) de la partie source du bicorpus[6];

$$A_i : B_i :: x : D$$

- Résoudre toutes les équations analogiques précédentes, et traduire récursivement par analogie toutes les solutions qui n'appartiennent pas au bicorpus. Puis, les ajouter elles-mêmes au bicorpus accompagnées de leurs traductions ;

- Faire de même pour toutes les phrases $x = C_{i,j}$ qui sont solutions[7] des équations analogiques précédentes, et appartiennent au bicorpus ;

- Former toutes les équations analogiques mettant en relation les phrases en langue cible correspondant aux phrases en langue source[8]. Si on note $\widehat{A_i}^k$ les traductions de A_i :

$$\widehat{A_i}^k : \widehat{B_i}^k :: \widehat{C_{i,j}}^k : y$$

- Énumérer toutes les solutions $y = \widehat{D_{i,j}}^k$ des équations analogiques, traductions de D, classées par fréquences[9].

Bien que l'implémentation du système de traduction automatique soit réalisée en langage C, nous pouvons en exprimer l'algorithme de façon triviale en langage Prolog (voir la figure 2.1, p. 129, les variables sont en majuscules, les constantes en minuscules). Seuls deux prédicats sont utilisés : `traduction`, pour les couples de phrases traduction, et `analogie`, pour résoudre une équation analogique (les inconnues sont respectivement C et $\widehat{D}$ sur les deux lignes). La dernière ligne permet d'ajouter le nouveau couple de traduction $(D,\widehat{D})$ à la base de connaissances : le système « apprend » au fur et à mesure qu'il traduit. On notera enfin, que l'algorithme a une complexité quadratique en fonction du nombre d'exemples qu'il utilise.

[5] Voir aussi Lepage, *De l'analogie rendant compte de la commutation en linguistique*, 2003.

[6] Les couples (A_i, B_i) sont sélectionnés à la volée à l'aide d'un critère de similarité.

[7] Une équation analogique peut avoir plusieurs solutions.

[8] Des phrases en langue cible différentes peuvent correspondre à une même phrase en langue source.

[9] Des équations analogiques différentes peuvent produire des solutions identiques.

% base de connaissance pour la traduction

`traduction`(s_1,$\widehat{s_1}$) .
`traduction`(s_2,$\widehat{s_2}$) .
⋮
`traduction`(s_n,$\widehat{s_n}$) .

% prédicat de traduction

`traduction`(D,$\widehat{D}$) :–
    `traduction`(A,$\widehat{A}$),
    `traduction`(B,$\widehat{B}$),
    `analogie`(A,B,C,D),
    `traduction`(C,$\widehat{C}$),
    `analogie`($\widehat{A}$,$\widehat{B}$,$\widehat{C}$,$\widehat{D}$),
    `assert`(`traduction`(D,$\widehat{D}$)).

Figure 2.1: Programme en Prolog pour la traduction automatique fondée sur l'analogie.

2.2.2 Exemple

Supposons que l'on veuille traduire la phrase japonaise suivante :

濃いコーヒーが飲みたい。
/koi kōhī ga nomitai/

Parmi les phrases du bicorpus, on trouve les deux couples suivants :

紅茶をください。
/kōcha wo kudasai/ ↔ *May I have some tea, please?*

コーヒーをください。
/kōhī wo kudasai/ ↔ *May I have a cup of coffee?*

qui nous permettent de former l'équation analogique suivante :

紅茶をください。 : コーヒーをください。:: x : 濃いコーヒーが飲みたい。

Cette équation a pour solution $x =$ 濃い紅茶が飲みたい。. Si cette phrase appartient déjà au bicorpus, c'est-à-dire si l'on est en mesure de trouver le couple de phrases :

濃い紅茶が飲みたい。
/koi kōcha ga nomitai/ ↔ *I'd like some strong tea, please.*

on peut former l'équation analogique suivante avec les traductions correspondantes en langue anglaise :

May I have some tea, please?	:	*May I have a cup of coffee?*	::	*I'd like some strong tea, please.*	:	x

Par construction, la solution $x =$ *I'd like a cup of strong coffee.* est donc une traduction candidate de la phrase source : 濃いコーヒーが飲みたい。

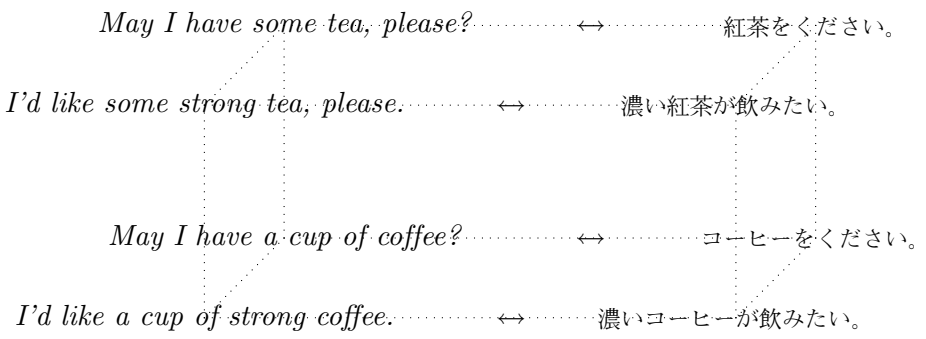

Figure 2.2: Vue géométrique du parallélépipède : dans chaque langue, quatre phrases forment une analogie. Il y a quatre relations de traduction entre les phrases.

2.2.3 Illustration géométrique de la méthode

L'exemple ci-dessus peut être schématisé sous la forme d'un parallélépipède donné dans la figure 2.2. Sur le plan vertical de gauche figure l'analogie en langue anglaise, et sur le plan vertical de droite l'analogie en langue japonaise. Chacun de ces deux plans contient une analogie dans une et une seule langue : les termes d'une équation analogique mettent donc en relation des données uniquement monolingues, de telle sorte qu'elle puisse être résolue par un algorithme tel que celui donné en annexe D, p. 171.

2.3 Caractéristiques de la méthode

2.2.1 Pas de transfert, pas d'extraction explicite de connaissances

La méthode proposée ici est originale sur plusieurs points : tout d'abord au niveau de son application, elle ne réalise pas de transfert, et n'extrait pas de connaissance de façon explicite, par exemple sous forme de règles ou de patrons. Le choix de la traduction adaptée est entièrement laissé à la structure de la langue, qui est saisie par l'opération d'analogie appliquée sur une ressource linguistique bilingue. Si on reprend l'exemple de la traduction de l'espagnol vers l'anglais de la section 2.1.2, on remarque que les correspondances présentes dans une analogie mettant en relation des phrases en langues source et cible seront entièrement responsables de la sélection des bons lemmes, mais aussi de leur ordre. On peut rapprocher dans une certaine mesure cette méthode de celle utilisée par Sumita[10] lorsqu'il traduit la forme N_1 *no* N_2 du japonais vers l'anglais. Dans sa méthode, le choix des prépositions correctes, ainsi que leur ordre, est entièrement laissé à la liste d'exemples.

They swam in the sea.	:	*They swam across the river.*	::	*It floated in the sea.*	:	*It floated across the river.*
↕		↕		↕		↕
Nadaron en el mar.	:	*Atravesaron el río nadando.*	::	*Flotó en el mar.*	:	*x*

[10] Sumita & Iida, *Experiments and prospects of example-based machine translation*, 1991.

Dans de telles analogies, il n'est pas précisé quel mot correspond à quel mot, ni quelle structure syntaxique correspond à quelle structure syntaxique. Seule l'application de l'analogie sur les chaines de caractères produit la traduction exacte de *It floated across the river.*, c'est-à-dire x = *Atravesó el río flotando.*

Ensuite, la méthode ne réalise pas dans son application d'extraction explicite de connaissance. Dans un système de traduction automatique de 2ème génération, la connaissance relative aux divergences entre les langues est rendue explicite sous la forme de règles de transfert lexical, et structural. Dans l'approche par l'exemple usuelle, la connaissance est aussi rendue explicite par l'acquisition automatique de patrons qui englobent ces divergences. Dans les deux cas, la connaissance relative aux divergences doit être explicitée. Nous pensons pour notre part que le choix de l'expression devrait être laissé implicite, puisqu'il est lié à la structure de la langue cible. En effet, les commutations syntagmatiques et paradigmatiques qui sont la composante implicite des analogies parviennent à neutraliser ces divergences.

La méthode que nous exposons ici se positionne donc fermement du côté de l'approche par l'exemple. Elle s'en écarte cependant sur un point essentiel : nous n'utilisons pas de représentation explicite de la connaissance sous la forme de patrons munis de variables. Il nous semble en effet que l'usage de patrons ne suffit pas à représenter toute la connaissance implicite contenue dans les exemples bruts : les variables y permettent des variations paradigmatiques, mais uniquement à des positions prédéfinies[11]. Remplacer les exemples par des patrons représente en soi un risque important de perte d'information, et d'ajout de bruit.

2.3.2 Pas d'entrainement, pas de préparation des données

La conséquence des caractéristiques précédentes est que cette méthode ne comporte pas de phase d'apprentissage ni d'entrainement : le bicorpus des exemples est tout simplement chargé en mémoire. On ne calcule pas de modèle de langue, on ne réalise pas d'autre alignement que celui initialement fourni dans la ressource bilingue, et enfin on ne réalise aucun prétraitement tel qu'un étiquetage, ou une segmentation.

Le bicorpus n'a pas besoin d'être segmenté en mots pour que le système puisse fonctionner, l'analogie agissant directement au niveau des caractères. Lors d'une campagne d'évaluation comme IWSLT (voir section 2.4.5), où les données japonaises sont fournies segmentées, nous enlevons cette segmentation afin d'obtenir des textes naturels en sortie du système.

On a la possibilité de rajouter à la ressource linguistique utilisée des informations supplémentaires : nous verrons par la suite dans la section 2.4.4 que l'ajout de paraphrases ou de dictionnaires est possible, et peut conduire à de meilleures performances, mais laisse la structure du système inchangée.

2.4 Évaluation de la traduction automatique

2.4.1 Ressources utilisées pour l'évaluation

Afin d'évaluer la méthode proposée, nous utilisons la partie C-STAR[12] du Basic Traveler's Expression Corpus (BTEC), ensemble de 162 318 phrases du domaine touristique, alignées en japonais et en anglais. Nous en rappelons ici les principales

[11] Dans Sato, *Example-based machine translation*, 1991, Sato fournit successivement à son système des phrases différant d'un mot seulement afin d'acquérir une grammaire.

[12] `http://www.c-star.org/`.

caractéristiques (il est décrit plus précisément en annexe A.1). Les phrases qui s'y trouvent sont particulièrement courtes, comme le montrent les chiffres ci-dessous. Le nombre de phrases uniques n'est pas égal en anglais et en japonais, une même phrase pouvant potentiellement apparaitre plusieurs fois avec des traductions différentes.

	Nombre de phrases $\neq$	Taille en caractères moyenne. $\pm$ écart-type
anglais	97 395	$35,17 \pm 18,83$
japonais	103 051	$16,22 \pm 7,84$

La méthode repose sur l'hypothèse que des analogies de forme sont presque toujours des analogies de sens. Dans des travaux précédents, la proportion d'analogies de forme qui sont des analogies de sens sur cette même ressource a été estimée à 96% (par échantillonage de 666 analogies, avec un taux de confiance de $99,90\%$[13]). On peut raisonnablement penser que cette proportion est en pratique trop faible pour nuire vraiment à la qualité des résultats de traduction.

2.4.2 Système de référence et système absolu

Afin d'évaluer les performances de la méthode, nous utilisons un ensemble de test de 510 phrases, provenant du même domaine que le corpus BTEC. Nous disposons pour chacune de ces phrases de 16 traductions références dans la langue cible, ce qui nous permet d'effectuer une évaluation des sorties de traduction à l'aide de mesures telles que BLEU, NIST ou mWER (Voir sections 2.3, p. 41 et 2.3.3, p. 97.).

Tout d'abord, nous déterminons ce que seraient les scores d'un système absolu (par "absolu" nous entendons un système dont les performances sont les plus hautes possibles en terme de la mesure d'évaluation considérée) : pour cela, pour chaque phrase de l'ensemble de test, nous évaluons la première traduction référence, comme si elle était une sortie de système de traduction automatique. De cette façon, on obtient pour chacune des mesures considérées les valeurs qui peuvent être considérées comme les meilleures possibles.

Ensuite, nous déterminons des scores de référence (*baseline*) en simulant une mémoire de traduction : pour cela, pour chaque phrase de l'ensemble de test, nous recherchons la phrase du corpus BTEC la plus proche au sens de la distance d'édition, et l'évaluons comme si elle était une sortie de système de traduction automatique. On obtient ainsi les scores planchers pour chacune des mesures considérées. Tout score inférieur aux scores planchers indique donc un système très mauvais. Les résultats sont consignés dans le tableau 2.1, p. 133. La comparaison est effectuée avec deux autres systèmes de traduction automatique par l'exemple, nécessitant des prétraitements importants du bicorpus afin d'extraire des patrons automatiquement (Système A) ou à la main (Système B).

2.4.3 Résultats avec la ressource seule

Le système décrit plus haut produit plusieurs traductions candidates : les figures 2.3 et 2.4 de la page 134 montrent des exemples de traduction, avec pour chaque traduction candidate sa fréquence de sortie sur la colonne de gauche[14]. Nous faisons

[13] Lepage & Peralta, *Using paradigm tables to generate new utterances similar to those existing in linguistic resources*, 2004.

[14] Nous avons vu dans la section 2.2, p. 128, que plusieurs équations analogiques pouvaient aboutir à une même solution.

Tableau 2.1: Scores du sytème absolu et de référence, et scores du système avec plusieurs types de données.

Système:	Nombre de couples d'exemples	BLEU	NIST	mWER	mPER	GTM
Absolu	n.r.	1,00	14,95	0,00	0,00	0,91
Système A	inconnu	0,66	10,36			
+ Paraphrases en source et cible	438 817	0,50	**8,98**	0,46	0,42	0,67
+ Paraphrases en cible	158 409	0,49	8,91	0,47	0,43	0,67
+ Paraphrases en source	158 409	0,53	8,53	**0,38**	**0,35**	**0,68**
+ Dictionnaire	206 382	**0,54**	8,54	0,39	0,36	**0,68**
Ressource entière	158 409	0,53	8,53	0,39	0,36	**0,68**
1/2 ressource	81 058	0,45	7,78	0,50	0,45	0,63
1/4 ressource	40 580	0,42	7,18	0,53	0,49	0,60
Système B	inconnu	0,41	9,00			
Référence: mémoire de traduction	n.r.	0,38	7,54	0,58	0,53	0,61

l'hypothèse que le candidat le plus fréquent est le plus fiable, et évaluons donc les traductions sur les premiers candidats pour chaque phrase de l'ensemble de test. Le système est évalué sur les traductions produites uniquement à l'aide des exemples présents dans la ressource BTEC (Voir tableau 2.1, ligne : Ressource entière).

2.4.4 Influence des ressources linguistiques utilisées

Influence du nombre d'exemples

On peut s'attendre, pour un système de traduction automatique par l'exemple, à ce que la qualité de traduction dépende beaucoup de la quantité et de la qualité des exemples présents dans la ressource utilisée. En effet, ce sont eux qui contiennent implicitement l'intégralité de la connaissance linguistique qui sert au système à traduire. On peut s'attendre intuitivement à ce qu'un système disposant de trop peu de données voie sa performance bridée, et à l'opposé, à ce qu'un système disposant de beaucoup de données ait une meilleure couverture des variations lexico-syntaxiques de la langue, pour une meilleure performance finale. Les chiffres donnés dans le tableau 2.1, pour la moitié et un quart de la ressource BTEC, confirment cette intuition : dans ce cas, plus on a de données à disposition, meilleurs sont les résultats.

Utilisation de dictionnaires comme liste d'exemples particuliers

Les phrases entières contenues dans la ressource BTEC (par opposition à des mots isolés) peuvent ne pas permettre d'aboutir à la traduction de certaines expressions, si on ne peut trouver de commutations entre elles. Ce cas est particulièrement fréquent si on essaie de traduire des phrases contenant des nombres, ou des expressions idiomatiques invariables.

コーヒーのおかわりをいただけますか。

2 318	*I'd like another cup of coffee.*
2 296	*May I have another cup of coffee?*
1 993	*Another coffee, please.*
1 982	*May I trouble you for another cup of coffee?*
1 982	*Can I get some more coffee?*
530	*Another cup of coffee, please.*
516	*Another cup of coffee.*
466	*Can I have another cup of coffee?*
337	*May I get some more coffee?*
205	*May I trouble you for another cup of coffee, please?*

Figure 2.3: Exemples de traduction en anglais de la phrase japonaise コーヒーのおかわりをいただけますか。.

小銭をまぜてください。

924	*Can you include some small change?*
922	*Can you include some small change, please?*
899	*Would you include some small change?*
896	*Include some small change, please.*
895	*I'd like to have smaller bills mixed in.*
895	*Please change this into small money.*
895	*Will you include some small change?*
885	*Could you include some small change, please?*
880	*May I have some small change, too?*

Figure 2.4: Exemples de traduction en anglais de la phrase japonaise 小銭をまぜてください。.

Une solution possible est d'ajouter des entrées de dictionnaire à la ressource originale afin qu'ils soient vus comme des exemples supplémentaires. En réalité, dans la méthode proposée, le système ne fait pas de différence entre un bicorpus de phrases alignées et un dictionnaire, puisque les deux sont vus indifféremment comme des chaines de caractères alignées. Les exemples suivants illustrent le fait que le système traite ces exemples de la même manière :

フィルムを買いたいのですが。	↔	*I'd like a film, please.*
三十六枚撮りを二本ください。	↔	*Two rolls of thirty-six exposure film, please.*

このカメラの電池がほしいのです。	↔	*I'd like a battery for this camera, please.*
フィルム	↔	*film*
映画	↔	*film*
電池	↔	*battery*
砲台	↔	*battery*

Comme on le voit sur les résultats du tableau 2.1, p. 133, on n'obtient pas de scores significativement meilleurs en ajoutant un dictionnaire à la ressource, à l'exception d'une légère amélioration du score BLEU.

Utilisation de paraphrases générées à partir de la ressource

Plusieurs études passées ont montré que l'ajout de paraphrases pouvait améliorer la qualité des résultats de traduction automatique : puisqu'en traduction automatique on travaille sur deux langues, source et cible, on peut choisir d'ajouter des paraphrases en langue source[15], en langue cible[16], ou dans les deux à la fois.

Afin d'augmenter les chances d'une phrase d'entrer en analogie, nous avons regroupé les phrases en langue source d'une part, et cible d'autre part, en ensembles de paraphrases. Pour ce faire, nous avons employé la méthode exposée dans la section 1.2.2 (p. 109) : nous avons regroupé les phrases partageant au moins une traduction commune, en faisant l'hypothèse qu'elles sont dans ce cas équivalentes en sens, c'est-à-dire paraphrases[17]. On obtient pour la ressource utilisée une moyenne de $3,03$ paraphrases mutuelles par phrase source, ce qui permet au système de traduction d'avoir plus d'analogies à former pour une même phrase à traduire. Lorsqu'un couple (A, B) est considéré pour une phrase d'entrée D dans une équation analogique, alors le système essaie de résoudre non seulement l'équation $A : B :: x : D$ mais aussi toutes les équations possibles $A' : B' :: x : D$ où A' et B' sont paraphrases de A et B respectivement.

Les résultats obtenus en rajoutant les paraphrases en langue source sont rassemblés dans le tableau 2.1, et montrent une légère amélioration en mWER (distance d'édition en unité de mots). Si on rajoute des paraphrases en langue cible, on constate une détérioration des scores BLEU, mais une vraie amélioration des scores NIST. Si on ajoute des paraphrases des deux côtés, en source et en cible, on obtient des scores qui ne sont pas meilleurs que ceux obtenus à l'aide de la ressource BTEC seule, à l'exception des scores NIST : il est en effet possible que les paraphrases aient introduit des variations lexico-syntaxiques pour l'expression d'un même sens. On peut aussi expliquer cette baisse par le fait que l'ajout de nombreux exemples supplémentaires a pu surcharger le système au niveau calculatoire, un seuil constant en temps étant imposé au système pour toutes les expériences. La complexité de l'algorithme étant quadratique en fonction du nombre d'exemple que le système utilise, l'ajout de paraphrases augmente la charge de calcul : l'imposition d'un seuil pour limiter le temps de calcul implique que l'algorithme ne peut pas former en pratique toutes les analogies possibles avec les exemples dont il dispose.

[15] Yamamoto, *Interaction between paraphraser and transfer for spoken language translation*, 2004.

[16] Habash, *Generation-heavy hybrid machine translation*, 2002.

[17] En réalité la relation peut être moins forte, puisqu'il peut y avoir une simple relation d'implication, comme on en a discuté dans la section 1.3.3, p. 114.

2.4.5 Campagnes d'évaluation IWSLT

Les campagnes IWSLT (*International Workshop on Spoken Language Translation*) d'évaluation de la traduction automatique sont organisées par le consortium C-STAR afin d'évaluer les systèmes de traduction automatique développés par ses différents membres.

Les données à traduire proviennent du corpus multilingue C-STAR BTEC (voir annexe A.1, p. 145), développé par les différents membres. Bien que seuls les membres du consortium ou les organisations qui en ont réglé les droits puissent utiliser l'intégralité du BTEC (environ 160 000 lignes), toute organisation extérieure désireuse de présenter son système a la possibilité de participer à une catégorie intitulée *données fournies* (*Supplied data track*) : dans celle-ci, 20 000 lignes du BTEC sont mises à disposition de toutes les organisations. Aucune ressource supplémentaire ne peut être utilisée dans cette catégorie.

Le thème de ces campagnes étant de traiter la langue sous sa forme orale, les phrases à traduire sont plus courtes que celles qui seraient tirées d'articles de journaux par exemple.

Afin d'évaluer le système de traduction automatique fondé sur l'analogie sur les chaines de caractères, et auquel il a été donné le nom ALEPH, nous avons d'une part reproduit les conditions expérimentales de la campagne IWSLT-2004[18], et d'autre part réellement participé à la campagne IWSLT-2005[19]. Nous décrivons ci-dessous les conditions expérimentales de ces campagnes, et présentons les résultats obtenus.

IWSLT-2004

Pour la campagne d'évaluation de la traduction automatique IWSLT-2004, nous avons reproduit les conditions de la catégorie dite *sans restriction* (*Unrestricted data track*), qui permettait d'utiliser toutes les ressources linguistiques possibles, ainsi que tous les outils disponibles actuellement.

En ce qui concerne la traduction, nous avons participé aux deux catégories possibles : CE (du chinois vers l'anglais) et JE (du japonais vers l'anglais). Afin de nous replacer dans des conditions naturelles, et de montrer l'intérêt et la faisabilité d'opérer sur des données non segmentées, nous avons éliminé la segmentation présente dans les données fournies en chinois et en japonais. Nous montrons par là que l'absence de segmentation n'est pas génante pour un système fondé sur le traitement des chaines de caractères, par opposition à la majorité des systèmes qui travaillent en mots.

Nous avons utilisé uniquement les données du C-STAR BTEC, soit environ 160 000 lignes alignées en anglais, chinois, et japonais. Par ailleurs, nous n'avons pas utilisé de dictionnaire additionnel.

Certaines phrases à traduire étant présentes telles quelles dans le corpus BTEC, afin de différencier la mémoire de traduction d'une réelle traduction, nous avons évalué le système dans deux configurations : *ouvert* et *standard*. La seule différence entre ces deux configurations est que, dans le cas du système ouvert, si une phrase à traduire est trouvée dans la ressource BTEC, elle est retirée de la ressource afin de forcer le système à traduire, et l'empêcher de fonctionner comme une mémoire de traduction (une phrase traduite par mémoire de traduction ayant de fortes chances d'obtenir un très bon score à l'évaluation, on s'attend donc à ce que les scores en

[18] Akiba *et al.*, *Overview of the IWSLT04 evaluation campaign*, 2004.
[19] Eck & Hori, *Overview of the IWSLT 2005 evaluation campaign*, 2005.

configuration ouverte soient inférieurs ou égaux à ceux obtenus en configuration standard).

Les résultats de l'évaluation des traductions produites sont consignés dans le tableau 2.2 pour le couple chinois-anglais et dans le tableau 2.3 pour le couple japonais-anglais. Les ^{s}, ^{e}, ^{r}, et ^{h} en exposant indiquent respectivement des systèmes de traduction automatique statistiques, par l'exemple, par règles et hybrides.

Même s'il est aisé, nous en convenons, d'obtenir des scores honorables lorsqu'on participe après coup, et pas sur le moment même à une évaluation, nous obtenons des résultats satisfaisants : le système ALEPH atteint la deuxième place pour le couple chinois-anglais, et la troisième pour le couple japonais-anglais. Si on regarde en particulier les scores BLEU, le système passe près de la première place en chinois-anglais (0, 522 pour 0, 524 au meilleur système, différence non significative), et la meilleure en japonais-anglais (0, 634 pour un second à 0, 630, là encore la différence n'est pas significative).

Tableau 2.2: Scores obtenus dans les conditions de la campagne IWSLT-2004, pour le couple chinois-anglais, en catégorie *sans restriction*.

	mWER	mPER	BLEU	NIST	GTM
^{s}ISL-S	**0,379**	**0,319**	**0,524**	**9,56**	**0.748**
^{e}ALEPH *standard*	0,434	0,400	0,522	8,42	0,687
^{e}ALEPH *open*	0,437	0,404	0,512	8,24	0,682
^{s}IRST	0,457	0,393	0,440	7,24	0,671
^{s}IBM	0,525	0,442	0,350	7,36	0,684
^{h}ISL-E	0,531	0,427	0,275	7,50	0,666
^{s}ISI	0,573	0,499	0,243	5,42	0,602
^{h}NLPR	0,578	0,531	0,311	5,92	0,563
^{e}HIT	0,594	0,487	0,243	6,13	0,611
^{r}Systran (CLIPS)	0,658	0,542	0,162	6,00	0,584
^{e}ICT	0,846	0,765	0,079	3,64	0,386

IWSLT-2005

Lors de la campagne IWSLT-2005, cinq couples de langues étaient proposés aux participants : arabe-anglais, chinois-anglais, coréen-anglais, japonais-anglais, et enfin, anglais-chinois. La facilité d'utilisation du système ALEPH nous a permis de participer pour tous ces couples. Le système étant bidirectionnel, il suffit en effet de changer les données pour qu'il soit utilisable.

Nous avons uniquement proposé le système dans sa configuration *ouverte*, et nous sommes encore une fois limité aux 160 000 lignes du C-STAR BTEC, à l'exception du couple arabe-anglais, où seules 20 000 lignes sont disponibles. Les résultats sont consignés dans le tableau 2.4, p. 138. Sauf mentionné en remarque dans le tableau 2.4, le système est en configuration *ouverte*, utilise 160 000 lignes de données du C-STAR BTEC, et l'évaluation est réalisée avec 16 références.

Tableau 2.3: Scores obtenus dans les conditions de la campagne IWSLT-2004, pour le couple japonais-anglais, en catégorie *sans restriction*.

	mWER	mPER	BLEU	NIST	GTM
[h]ATR-H	**0,263**	**0,233**	0,630	10,72	0,796
[s]RWTH	0,305	0,249	0,619	**11,25**	**0,824**
[e]ALEPH *standard*	0,324	0,300	**0,634**	9,19	0,731
[e]ALEPH *open*	0,437	0,403	0,534	8,97	0,697
[e]UTokyo	0,485	0,420	0,397	7,88	0,672
[r]Systran (CLIPS)	0,730	0,597	0,132	5,64	0,568

Dans les trois couples de langue réellement comparables du fait des conditions de données et d'évaluation rigoureusement semblables (chinois-anglais, coréen-anglais et japonais-anglais), il est intéressant de remarquer que pour toutes les mesures, le système obtient les meilleures performances en japonais-anglais, et les moins bonnes en coréen-anglais. On peut avancer une explication pour le cas du coréen anglais : le traitement de l'écriture hangul est problématique en traitement automatique des langues, un caractère syllabique étant lui même composé de plus petites parties, consonnes ou voyelles. Le hangul est une écriture syllabaire semi-idéographique, pour laquelle un caractère ne peut donc être considéré comme la plus petite unité indécomposable, comme un atome de base. Ceci peut expliquer le fait que l'application de l'analogie sur les chaines de caractères donne de moins bonnes performances dans le cas de cette écriture.

Tableau 2.4: Scores obtenus lors de la campagne IWSLT-2005, pour tous les couples de langues.

	mWER	mPER	BLEU	NIST	GTM	*Remarques*
arabe-anglais	0,527	0,497	0,382	6,22	0,481	20 000 lignes
coréen-anglais	0,530	0,486	0,412	7,12	0,446	
chinois-anglais	0,454	0,418	0,477	7,85	0,553	
japonais-anglais	0,361	0,323	0,593	9,82	0,607	
anglais-chinois	0,798	0,746	0,098	3,029	0,363	1 référence

Conclusion

Nous avons exposé dans ce chapitre une approche de la traduction automatique par l'exemple uniquement fondée sur une relation élémentaire, l'analogie, et l'avons appliqué entre les chaines de caractères : dans une première expérience sur la ressource

BTEC, totalisant 160 000 phrases alignées, nous avons traduit 510 phrases test du japonais vers l'anglais. Les résultats obtenus sont bien meilleurs que ceux d'un système référence simulant une mémoire de traduction.

Nous avons vu que de petites améliorations pouvaient être obtenues en ajoutant des paraphrases, bien qu'elles ne soient pas significatives. En reproduisant les conditions expérimentales de la campagne IWSLT-2004 d'évaluation de la traduction automatique, puis en participant à la campagne IWSLT-2005, nous avons évalué les performances de notre système ALEPH face à d'autres sytèmes. Le système obtient des performances comparables à celles de systèmes fondés sur les statistiques[20], sur l'utilisation d'exemples, sur les règles, et enfin utilisant des approches hybrides.

L'analogie sur les chaines de caractères permet donc d'atteindre des résultats satisfaisants, sans qu'on ait besoin d'appliquer de prétraitement sur les données, sans avoir à accomplir de phase d'apprentissage caractéristique des méthodes à entrainement intensif, très lourdes à mettre en œuvre.

Dans notre système, l'absence de prétraitement tel qu'une segmentation des données, permet de s'affranchir des contraintes liées aux méthodes faisant usage d'une unité plus élevée telle que le mot, et ainsi de traiter n'importe quelle langue sans distinction, sous la forme de chaines de caractères. C'est évidemment intéressant dans le cas des langues dont le système d'écriture n'admet pas de séparateur graphique, c'est indispensable dans le cas de langues pour lesquelles on ne dispose pas de segmenteur ou d'analyseur qui permettent d'effectuer une découpe des données.

Le système que nous avons présenté se caractérise donc par sa facilité d'utilisation, de mise en place, et par son universalité : il peut fonctionner immédiatement, quel que soit le couple de langue considéré, quel que soit le type de données informatisées fournit.

[20]Ses performances sont comparables en termes de mesures BLEU et NIST, à celles d'un système statistique utilisant un modèle IBM4 fonctionnant sur les mots (sans les *phrases*).

Conclusion et perspectives

Notre travail a été dirigé par un impératif d'universalité des méthodes, de multilinguisme. Le problème a son origine dans l'atome utilisé dans la transcription des langues, de l'oral à l'écrit : le mot n'est pas noté dans toutes les langues. C'est pourquoi dans cette thèse, nous avons cherché à promouvoir un traitement automatique des langues en caractères. C'est en effet un atome universel, immédiatement accessible, et plus petit que le mot, unité actuellement la plus utilisée.

L'utilisation du caractère comme atome du traitement automatique des langues permet de se passer de prétraitement sur les données tel qu'une segmentation en mots, actuellement incontournable dans des langues telles que le chinois ou le japonais. D'autre part, il est possible de transposer et d'appliquer de façon immédiate des méthodes de traitement déjà éprouvées en mots. Nous avons montré que par un tel changement de résolution du mot au caractère, on pouvait arriver à traiter élégamment plusieurs tâches du traitement automatique des langues, en traitement et en production de données linguistiques, avec des performances satisfaisantes. En effet, passer d'un travail en mots à un travail en caractères ne revient pas à effectuer un simple changement d'échelle, comme si on passait du mètre au centimètre. Un mot n'ayant pas de longueur fixe en nombre de caractères, un tel changement est plus complexe, comme en atteste l'expérience préliminaire menée sur l'évaluation automatique de la traduction automatique.

Notre intérêt pour des techniques « universelles » en traitement automatique des langues nous a motivé à utiliser dans cette étude des méthodes aveugles aux données, comme l'approche statistique, ou des méthodes reposant sur une opération cognitive universelle, comme l'analogie.

Nous avons tout d'abord mené des expériences sur des procédures d'évaluation des données : nous avons vu dans une étude préliminaire (partie I, chapitre 2) que l'évaluation automatique de la traduction automatique pouvait être traitée élégamment en transposant une méthode bien connue utilisant l'unité de mot, BLEU, vers l'unité de caractère. Puis dans la partie II, nous nous sommes intéressé au filtrage de la grammaticalité en langue écrite (chapitre 1), et avons proposé deux méthodes différentes de filtrage utilisant le caractère pour unité. Ensuite, nous avons proposé une approche permettant de caractériser automatiquement des ressources linguistiques servant à un système de traduction automatique en particulier, en terme de similarité et d'homogénéité (chapitre 2).

Pour ces procédures d'« évaluation », nous avons montré que l'unité de caractère permettait d'obtenir des résultats acceptables, et comparables à ceux obtenus avec des méthodes similaires en mots.

Ensuite, nous sommes passés en partie III à des expériences de production des données en utilisant des résultats de l'analogie sur les chaines de caractères : nous avons présenté une méthode pour produire automatiquement des paraphrases, qui

utilise l'analogie pour générer des énoncés, et une des méthodes de filtrage vues en partie II, chapitre 1, afin de contrôler la qualité de cette génération. Nous avons enfin proposé l'utilisation en traduction automatique de l'analogie sur les chaines de caractères.

La production de paraphrases obtient de bons résultats, et peut servir à l'évaluation de la traduction automatique. La traduction automatique obtient des résultats acceptables en terme d'efficacité : le système ne requiert pas de phase d'apprentissage, ni de prétraitement sur les données, tout en obtenant des performances en BLEU en retard de deux ans seulement sur les meilleurs systèmes (voir les évaluations IWSLT-2004 et 2005 en partie III, chapitre 2, section 2.4).

Nous pouvons toutefois émettre des réserves quant à la possibilité de transposer n'importe quelle méthode en mots vers des méthodes en caractères. On pourrait effectuer un parallèle entre, d'une part, le mot et le morphème (ce qui est presque vrai sur une langue comme l'anglais par exemple), et, d'autre part, le caractère et le phonème. Si l'on considère que le mot est proche d'une unité de 2^{e} articulation et le caractère proche d'une unité de 1re articulation au sens de Martinet, alors on peut penser que le mot est une unité signifiante, alors que le caractère ne l'est pas. Cette réflexion parait particulièrement fondée dans le cas des systèmes d'écriture utilisant l'alphabet latin. Par contre, elle l'est moins dans le cas des systèmes d'écriture idéographiques, comme en chinois, ou en japonais, langues lesquelles l'idéogramme est porteur de sens.

Les résultats de cette étude permettent d'envisager avec plus de simplicité des travaux futurs portant sur les langues dont le système d'écriture n'admet pas de séparateur : nous nous proposons par exemple d'utiliser une modélisation statistique de langue en caractères pour segmenter des documents entiers écrits en chinois en énoncés significatifs plus courts. En outre, il serait intéressant dans le cas des langues dont le système d'écriture admet des séparateurs graphiques clairs, d'étudier des approches multiniveaux : par exemple en évaluation de traduction automatique, où nous avons vu que le niveau du mot et le niveau du caractère donnaient lieu dans le cas de la langue anglaise à des différences importantes en terme de granularité, nous nous proposons d'étudier des méthodes utilisant à la fois ces deux niveaux.

Annexes

Annexe A

Présentation des corpus utilisés

Cette annexe présente les caractéristiques des principaux corpus utilisés dans cette étude, des extraits, ainsi que les abréviations utilisées.

A.1 Le corpus BTEC

Le corpus BTEC, pour *Basic Traveler's Expression Corpus*, est à l'origine[1] un corpus bilingue anglais-japonais appelé BE, pour *Bilingual basic Expressions corpus*, et constitué d'expressions usuelles tirées en grande partie de guides de conversations touristiques. Le BTEC occupe une place centrale dans cette thèse, c'est pourquoi nous y consacrons cette annexe.

Dans le cadre du consortium C-STAR[2], le corpus BE a servi de base à l'élaboration d'un grand corpus multilingue, le BTEC. Destiné à l'évaluation des sytèmes de traduction de l'oral, le CSTAR-BTEC est une ressource comprenant environ 162 000 phrases alignées dans 6 langues[3] : japonais, anglais, coréen, italien, chinois, et espagnol. Nous faisons figurer dans le tableau A.1 quelques caractéristiques des parties chinoises, anglaises, japonaises et coréennes du BTEC.

Nous donnons ci-dessous quelques exemples de phrases tirées de la version anglaise du BTEC. Les chiffres donnés dans le tableau montrent que les phrases sont généralement très courtes :

Tableau A.1: Caractéristiques du corpus multilingue BTEC.

	Nombre de phrases uniques	Taille des phrases en caractères moy.	±	écart-type
chinois	96 234	11, 00	±	5, 77
anglais	97 769	35, 14	±	18, 81
japonais	103 274	16, 21	±	7, 84
coréen	92 628	30, 07	±	15, 54

[1] Takezawa *et al.*, *Towards a broad-coverage bilingual corpus for speech translation of travel conversations in the real world*, 2002

[2] Voir http://www.c-star.org/ .

[3] Il faut noter cependant que les parties en italien, allemand et français ne sont pas encore achevées. D'autre part, les 162 000 phrases en arabe devraient être disponibles en septembre 2006.

Thank you so much. Keep the change.
Bring plenty of lemon, please.
Please tell me about some interesting places near here.
Thank you. Please sign here.
How do you spell your name?

Ensemble de test `jpnset01` L'ensemble de test `jpnset01` est un ensemble de 510 phrases en langue japonaise, provenant de sources similaires au BTEC, et servant dans notre étude de tâche générique afin d'évaluer la qualité de la traduction automatique. Cet ensemble sert de référence pour tous les systèmes de traduction automatique développés à ATR. On notera que sur les 510 phrases qu'il comporte, seules 279 sont réellement nouvelles, les 231 restantes étant déjà présentes dans le BTEC.

Les phrases ont une longueur moyenne de $14,89$ caractères (contre $16,21$ pour le BTEC), avec un écart-type de $6,60$ caractères (contre $7,84$ pour le BTEC).

A.2 Corpus divers

Nous présentons brièvement dans cette section plusieurs corpus, utilisés principalement au chapitre 2 qui traite de la caractérisation automatique en terme de similarité et d'homogénéité des ressources linguistiques.

Certains d'entre eux sont des corpus monolingues (en anglais ou en japonais), d'autres sont multilingues (et comportent des parties en anglais et en japonais).

Nous citons pour chacun un exemple de production type, et récapitulons leurs principales caractéristiques en langue anglaise et japonaise, respectivement dans les tableaux A.2 et A.3. Pour le japonais, le tableau A.3 ne fait évidemment pas figurer de chiffres de mesures en mots.

Tableau A.2: Caractéristiques numériques de plusieurs corpus en langue anglaise.

Anglais	SLDB	MAD	SPAE	TIME	Calgary
Mots/phrase (moy.)	11,27	9,29	23,34	23,17	20,21
Mots/phrase (écart-type)	±6,85	±5,83	±26,43	±15,32	±15,18
Car./phrase (moy.)	64,51	44,86	126,11	131,74	107,70
Car./phrase (écart-type)	±35,95	±27,57	±140,71	±92,38	±84,69
Car./mots	5,72	4,83	5,40	5,68	5,33
Nbre total de car.	1 037K	475K	223K	1 515K	757K
Nbre total de mots	181,2K	98,5K	41K	264,5K	142,2K
Nbre total de phrases	16 078	10 601	1 759	11 416	7 035

Tableau A.3: Caractéristiques numériques de plusieurs corpus en langue japonaise.

Japonais	SLDB	MAD	NHK	Mainichi	Nikkei
Car./phrase (moy.)	32,61	26,87	65,39	37,73	44,21
Car./phrase (écart-type)	±22,22	±14,07	±39,16	±31,88	±28,34
Nbre total de car.	20 806K	290K	2 772K	2 740K	2 772K
Nbre total de phrases	84 751	10 612	66 512	71 647	253 016

SLDB : *Spontaneous Speech Database*

C'est une ressource multilingue de transcriptions de dialogues, en japonais et en anglais[4].

Exemples :

Okay, I go four blocks down Mason Street and then I take a left there, is that right?
えーっすぐ分かるんでしょうか、場所は。

MAD : *Machine-translation-Aided bilingual spoken Dialogue corpus*

C'est un recueil de phrases de dialogues du domaine du voyage et du tourisme en langue japonaise et anglaise, transcrites de façon réaliste, mais toutefois purgées de tout phénomène oral tel que répétition, redite, ou faux-départ.

Exemples :

Walk two blocks down this street and turn left and you'll see the bank on the right.
すいませんが、写真撮っていただけますか。

SPAE : *The Corpus of Spoken Professional American-English*

C'est un recueil de transcriptions de réunions d'affaires en langue anglaise dans un contexte professionel[5].

Exemple :

I have carefully read and heard about all of the things that the group has discussed up until now.

TIME : *Corpus du magazine TIME*

C'est un recueil d'articles de journaux du magazine TIME[6] datant de 1963.

Exemple :

The French, who got no help from the US in developing their force de frappe, were quick to crow that Britain's vaunted ties with the US had brought it nothing but humiliation.

[4] Nakamura *et al.*, *Japanese speech databases for robust speech recognition*, 1996
[5] Il est disponible à l'adresse suivante : http://www.athel.com/cspa.html .
[6] Il est disponible à l'adresse suivante : ftp://ftp.cs.cornell.edu/pub/smart/time/ .

CALGARY : *Corpus de Calgary (sous-ensembles « book1 » et « book2 »)*

Il est couramment utilisé dans le domaine de la compression de données et contient plusieurs œuvres de littérature anglaise contemporaine[7].

Exemple :

She turned her head to learn if the waggoner were coming.

NHK : *Corpus de nouvelles NHK*

Il s'agit d'un recueil de transcriptions de journaux télévisés en langue japonaise, diffusées sur le réseau NHK (télévision publique).

Exemple :

各支店では、行員が新しい仕事の進め方を学ぶ勉強会を開いてきました。

MAINICHI : *Corpus du journal Mainichi*

C'est un recueil d'articles tirés du quotidien japonais à grand tirage Mainichi Shinbun.

Exemple :

ほとんどの企業がその後に五輪競技施設や土木工事を受注していた。

NIKKEI : *Corpus du journal Nikkei*

C'est un recueil d'articles en langue japonaise, tirés du quotidien économique japonais Nikkei Shinbun.

Exemple :

当時、店内には閉店の準備をしていた従業員約二十人がいたが、ほかにけが人はなかった。

[7]Le corpus Calgary est disponible librement sur le réseau Internet par ftp à ftp.cpcs.ucalgary.ca/pub/projects/text.compression.corpus .

Annexe B

Mesures d'évaluation objective de la traduction automatique

Les méthodes d'évaluation automatiques de la traduction automatique les plus utilisées actuellement sont BLEU et NIST, cette dernière étant essentiellement une modification de l'idée originale de BLEU. Nous précisons dans la section suivante les méthodes de calcul des scores BLEU et NIST telles qu'exposées dans leur formulation originale.

B.1 BLEU

B.1.1 Calcul du score BLEU

La mesure BLEU a été proposée en 2001 par IBM[1] et se fonde sur la mesure de co-occurrences de chaines de n atomes entre une phrase candidate à juger et un ensemble de phrases de référence.

Nous adoptons la notation suivante, inspirée de celle proposée par Babych[2] : BLEU_{wN} désigne un score BLEU calculé sur les mots (orthographiques), avec un ordre maximal N. Pour un ordre maximal N, un score BLEU_{wN} est le produit de deux termes : une pénalité BP, fonction de la brièveté de la phrase jugée, et la moyenne géométrique des "précisions modifiées" à l'ordre n notées p_n, calculées pour toutes les longueurs de chaines jusqu'à N. L'expression de BLEU_{wN} est alors la suivante :

$$\text{score BLEU}_{wN} = BP \times \sqrt[N]{\prod_{n=1}^{N} p_n}$$

En pratique, on choisit habituellement la valeur de $N = 4$ pour l'évaluation en langue anglaise, car c'est cette valeur qui a donné dans l'étude originale les meilleures corrélations avec un jugement humain. On omet généralement de préciser cette valeur, à tel point que dans la littérature on se réfère souvent à BLEU_{w4} par le terme générique BLEU.

La pénalité, notée BP pour *Brevity Penalty* dans la notation originale, est l'exponentielle de la variation relative de longueur entre la phrase à juger (appelée

[1] Papineni *et al.*, *BLEU: a method for automatic evaluation of machine translation*, 2001

[2] Babych & Hartley, *Modelling legitimate translation variation for automatic evaluation of MT quality*, 2004.

par la suite phrase candidate) et la référence de longueur la plus proche de celle-ci. Notons respectivement $\mathcal{C}$ et $\mathcal{R}$, la phrase candidate et la phrase de référence la plus proche en terme de longueur. Si on note $|\mathcal{P}|$ la longueur en mots de la phrase $\mathcal{P}$, alors on peut noter respectivement $|\mathcal{C}|$ et $|\mathcal{R}|$ la longueur de la phrase candidate et la longueur de la phrase référence la plus proche. Dans ce cas, la pénalité peut s'écrire de la manière suivante :

$$BP = \begin{cases} 1 & \text{si } |\mathcal{C}| > |\mathcal{R}| \\ e^{1-r/c} & \text{si } |\mathcal{C}| \leq |\mathcal{R}| \end{cases}$$

Notons $|\mathcal{P}|_{\mathcal{W}}$ le nombre d'occurrences de la (sous-)chaine $\mathcal{W}$ dans la chaine $\mathcal{P}$, de telle façon que $|\mathcal{P}|_{w_1 \dots w_n}$ soit le nombre d'occurrences de la chaine de n mots $w_1 \dots w_n$ dans la phrase $\mathcal{P}$. Avec cette notation, la précision modifiée à l'ordre n est le quotient de deux sommes[3] :

$$p_n = \frac{\sum\limits_{w_1 \dots w_n \in \mathcal{C}} \min\left(|\mathcal{C}|_{w_1 \dots w_n}, \max\limits_{\mathcal{R}} \left(|\mathcal{R}|_{w_1 \dots w_n} \right) \right)}{\sum\limits_{w_1 \dots w_n \in \mathcal{C}} |\mathcal{C}|_{w_1 \dots w_n}}$$

- le numérateur donne le nombre de chaines de longueur n du candidat apparaissant dans les références, limité au nombre maximal d'occurrences de la chaine de longueur n vu dans les références.

- le dénominateur donne le nombre total de chaines de longueur n apparaissant dans le candidat.

La mesure BLEU est bornée. Tout score est compris entre 0 et 1 au sens large ($0 \leq$ score BLEU ≤ 1). Un score de 0 est censé indiquer une mauvaise traduction, alors qu'un score de 1 est censé indiquer une bonne traduction.

B.1.2 Exemples et limitations

Application classique

Considérons la phrase candidate suivante :

I'd like to have some strong coffee.

Afin de juger de la qualité de cette sortie de système de traduction automatique avec la méthode BLEU, nous devons avoir préparé au préalable plusieurs traductions de référence, par exemple :

I'd like some strong coffee.
I want some strong coffee.
I want to drink some strong coffee.

[3] Nous nous limitons au cas où le candidat et les références sont des phrases. La méthode BLEU permet de juger des candidats composés de plusieurs phrases, donc des paragraphes.

Afin de simplifier les calculs dans cet exemple, nous utilisons ici un cas simple, celui où $N = 1$, donc BLEU_{w1}. Dans ce cas, la formulation se simplifie :

$$\text{BLEU}_{w1} = BP \times p_1$$

Tous les mots du candidat sont présents au maximum une fois dans l'une des références, à l'exception du mot *have*. Le candidat est constitué de 8 mots différents, sa précision à l'ordre 1 vaut donc $p_1 = \frac{7}{8}$. Les références ayant respectivement des longueurs de 6, 5, et 7 mots, le candidat de longueur 8 ne subit pas de pénalité, on a donc $BP = 1$. On a donc :

$$\text{BLEU}_{w1} = 1 \times \frac{7}{8} = 0,875$$

En revanche, si la phrase candidate avait été :

I want strong coffee.

dans ce cas, et bien que tous les mots présents dans le candidat puissent être trouvés exactement une fois dans les références (d'où $p_1 = 1$), la phrase candidate devrait subir une pénalité pour sa longueur trop faible de 4 mots. La phrase référence la plus courte faisant 5 mots, on aurait :

$$BP = e^{1-\frac{5}{4}} = e^{-\frac{1}{4}} \simeq 0.779$$

soit $BLEU_{w1} = 0,779$. La phrase candidate, bien que paraphrase des références et contenant uniquement des mots présents dans celles-ci, a donc été uniquement pénalisée pour sa brièveté. Ce choix dans la méthode est une des caractéristiques essentielles de la mesure BLEU.

Limitations de BLEU

Il est intéressant d'illustrer par l'exemple les limites d'une méthode telle que BLEU. Si l'on reprend les mêmes références que précédemment et si l'on juge la phrase candidate suivante :

I'd like to have some strong tea.

on obtient un score BLEU_{w1} de $0,750$. C'est un score relativement élevé, pourtant la traduction est incorrecte (différence entre *tea* et *coffee*). Cet exemple illustre une limitation de BLEU : la mesure étant fondée sur le comptage d'occurrences de mots, et la méthode étant en quelque sorte « aveugle », il est donné autant d'importance à un mot qu'à un autre. Une simple permutation, ou un simple ajout, ne pénalisent pas une phrase candidate longue. Cette limitation fait l'objet de l'un des reproches les plus courants adressés à la méthode BLEU, mais elle n'est pas la seule.

Si l'on juge à présent la phrase candidate suivante :

Pour me a cup of strong coffee.

la phrase ne subit pas de pénalité de longueur, par contre seuls deux des mots qui la composent sont attestés dans les références : *strong* et *coffee*. Le candidat, pourtant paraphrase valide des références, se voit attribuer un score BLEU_{w1} de $0,286$. Cet exemple illustre une deuxième limitation de BLEU : la méthode requiert

l'utilisation de références afin de juger les phrases candidates. Ces références doivent être des paraphrases, pourtant rien ne permet de contrôler la quantité de variation lexico-syntaxique présente dans un tel ensemble. Même si l'on dispose d'un grand nombre de références, rien ne peut donc nous assurer qu'elles couvriront de façon satisfaisante toutes les phrases candidates possibles. La méthode peut donc aisément attribuer un score médiocre à une traduction en réalité valide.

Les deux limitations exposées ci-dessus constituent les critiques les plus fréquemment formulées à l'égard de la méthode BLEU.

B.2 NIST

B.2.1 Calcul du score NIST

L'organisme NIST (National Industry Standards and Technology) a proposé en 2002 une modification[4] de la méthode BLEU, censée en améliorer la fiabilité. Basée elle aussi sur des mesures de co-occurrences de chaines de longueur n, la mesure NIST se distingue de BLEU par trois points :

- la précision modifiée à l'ordre n est remplacée par une mesure d'informativité des chaines de longueur n censée accorder plus d'importance aux chaines *informatives*, c'est-à-dire attestées moins souvent dans les références ;
- la pénalité de brièveté est modifiée de façon à être plus tolérante pour des petites variations de taille, et plus sévère pour de grandes variations de taille ;
- la moyenne géométrique présente dans la formule globale est remplacée par une moyenne arithmétique.

L'informativité d'une chaine de longueur n candidate est définie comme le logarithme en base deux du rapport entre le nombre d'occurrences de la chaine de longueur $(n-1)$ associée dans l'ensemble des références, et le nombre d'occurrences de la chaine de longueur n dans l'ensemble des références. En suivant les mêmes notations que précédemment, on peut exprimer l'informativité d'une chaine de longueur n sous la forme suivante :

$$\mathrm{Info}(w_1 \dots w_n) = \log_2 \frac{|\mathcal{R}|_{w_1 \dots w_{n-1}}}{|\mathcal{R}|_{w_1 \dots w_n}}$$

On notera que d'après une telle formule, l'informativité n'est alors définie que pour $n > 1$. Pourtant, l'implémentation officielle de la méthode[5] fournie par l'organisme NIST inclut un calcul de l'informativité à l'ordre $N = 1$ comme suit :

$$\mathrm{Info}(w_1) = \log_2 \frac{r}{|\mathcal{R}|_{w_1}}$$

r étant le nombre de mots dans les références. Tout comme pour BLEU, on omet souvent de préciser en pratique la valeur usuelle de $N = 5$ qui a produit la meilleure corrélation avec les jugements humains dans le cas de la langue anglaise.

[4] Doddington, *Automatic evaluation of machine translation quality using N-gram co-occurrence statistics*, 2002

[5] Voir http://www.nist.gov/speech/tests/mt/resources/scoring.htm .

La pénalité de brièveté est elle aussi modifiée de manière à réduire l'impact de petites variations dans la longueur de la phrase candidate. On peut l'exprimer sous la forme suivante :

$$BP_{NIST} = \exp \beta \log_2[min(\frac{c}{\bar{r}}, 1)]$$

c étant le nombre de mots dans la phrase candidate, $\bar{r}$ le nombre moyen de mots dans une phrase référence, et β une valeur réelle telle que $BP_{NIST} = 0,5$ lorsque $c = \frac{2}{3}\bar{r}$ (soit $\beta \simeq -4,22$).

La formule globale de NIST est le produit de la pénalité de brièveté par la moyenne arithmétique de l'informativité de toutes les chaines de longueur n attestées dans la phrase candidate. Si on note $\mathcal{C}^*_{w_1 \ldots w_n}$ le nombre des chaines de longueur n différentes présentes dans la phrase candidate, alors l'expression du score NIST à l'ordre N en unité de mots, que nous notons NIST_{wN} par analogie avec la notation de BLEU, est :

$$\text{score NIST} = BP_{NIST} \times \sum_{n=2}^{N} [\frac{\sum_{w_1 \ldots w_n \, co-occurrents \, de \, \mathcal{C} \, et \, \mathcal{R}} \text{Info}(w_1 \ldots w_n)}{\mathcal{C}^*_{w_1 \ldots w_n}}]$$

Contrairement à la mesure BLEU, NIST n'a pas de borne supérieure théorique. De même, un score de 0 est censé caractériser une mauvaise traduction.

B.2.2 Exemples et limitations

Application classique

Reprenons la phrase candidate utilisée précédemment :

I'd like to have some strong coffee.

ainsi que l'ensemble de trois traductions de référence :

I'd like some strong coffee.
I want some strong coffee.
I want to drink some strong coffee.

Là encore, pour alléger les calculs dans cet exemple, nous utiliserons l'ordre N minimal : nous avons vu que NIST permet via un cas particulier le calcul de l'informativité à partir de l'ordre $N = 1$.

Comme lors du calcul du score BLEU de la même phrase, tous les mots du candidat sont présents au maximum une fois dans l'une des références, à l'exception du mot *have*. Les mots *I*, *some*, *strong* et *coffee* sont présents 3 fois dans les références, les mots *'d*, *like* et *to* ne sont présents qu'une fois. Les références ayant respectivement des longueurs de 6, 5, et 7 mots, le candidat ne subit pas de pénalité, on a donc $BP = 1$. Au total, les références représentent 18 mots. On aboutit donc à :

$$\text{Score NIST} = 1 \times (4 \log_2 \frac{18}{3} + 3 \log_2 \frac{18}{1}) = 0,875$$

En revanche, si la phrase candidate avait été :

I want strong coffee.

bien que tous les mots présents dans le candidat se trouvent dans les références, la phrase candidate subirait une pénalité pour sa longueur trop faible de 4 mots :

$$\text{Score NIST} = \exp -4,22(\log_2 \frac{4}{6})^2 \times (3 \log_2 \frac{18}{3} + \log_2 \frac{18}{2}) = 5,459$$

Cette phrase est donc plus lourdement pénalisée qu'avec la méthode BLEU.

Limitations de NIST

Tout comme BLEU, la méthode NIST possède des limitations dues à l'attestation de chaines de longueur n. La méthode étant tout aussi aveugle, la simple permutation de deux mots ne pénalise pas lourdement une phrase, alors que la quantité de variations lexico-syntaxiques présente dans les références pose toujours un problème de couverture.

À titre d'exemple, la phrase :

I'd like to have some strong tea.

obtient un score NIST de 20, 26 alors qu'elle est visiblement incorrecte, et la phrase :

Pour me a cup of strong coffee.

obtient un score NIST de 5, 17 très inférieur alors qu'elle est correcte. Bien que les calculs aient été effectués à l'ordre minimal $N = 1$, et que la corrélation d'une telle mesure avec le jugement humain doive en réalité être évaluée sur un grand nombre de phrases choisies au hasard pour que les résultats soient probants, on remarque qu'il est extrêmement aisé de trouver des exemples sur lesquels la mesure ne donne pas de résultats satisfaisants.

B.3 Conclusion

Nous avons rappelé dans cette annexe les méthodes de calcul des méthodes BLEU et NIST, et donné des exemples de leur application. À travers des exemples nous avons exposé l'utilité, mais aussi les limites de ces mesures dites ≪ objectives ≫.

Tout d'abord, ces mesures sont fondées sur le comptage d'occurrences de mots, et une simple permutation, ou un simple ajout ne pénalisent pas beaucoup une phrase candidate longue. La méthode NIST essaie toutefois de répondre à ce problème en affectant une pondération aux mots.

Enfin, BLEU et NIST doivent disposer de paraphrases références pour chaque phrase à juger afin d'être mises en œuvre. La quantité de variation lexico-syntaxique présente dans ces références est difficilement quantifiable, et rien ne nous assure que les paraphrases références apportent une couverture suffisante.

La méthode BLEU est utilisée dans une expérience préliminaire en partie I, chapitre 2, afin de voir si une méthode utilisant une découpe en mots peut être transposée eficacement à une découpe en caractères. Bien que les résultats soient probants en ce qui concerne BLEU, nous aurions aimé effectuer une expérience similaire sur la méthode NIST : on peut légitimement penser[6] que celà marcherait moins bien. En effet, NIST attribue des pondérations aux mots en calculant une

[6]Remarque attribuée à Kenji Imamura.

valeur d'informativité pour chacun d'entre eux. Il parait risqué d'attribuer une informativité à des caractères, au moins en anglais où le caractère n'est pas une unité signifiante. Nous développons ces considérations dans les conclusions de notre étude.

Annexe C

Modélisation stochastique de langue

Nous présentons dans cette annexe plusieurs notions théoriques liées à la théorie de l'information et aux modèles statistiques des langues. Nous faisons une brève introduction sur le traitement statistique des langues, puis nous rappelons diverses notions de modélisation du langage et décrivons plusieurs méthodes de lissage probabiliste ; enfin nous proposons l'usage d'outils issus de la théorie de l'information. Nous montrons comment leur application en traitement automatique des langues peut nous aider à mettre au point des mesures qualitatives.

C.1 Introduction au traitement statistique des langues

Qu'est ce que la statistique ? Si l'on se réfère à la définition proposée par le Trésor de la langue française[1], la statistique est la *branche des mathématiques ayant pour objet l'analyse (généralement non exhaustive) et l'interprétation de données quantifiables*. La statistique définie par Efron[2] est *la science de l'apprentissage par l'observation et l'expérience*. Elle permet, avec une expérience forcément limitée d'un phénomène, de produire une prédiction sur ce qui va survenir. L'approche statistique est, dans ces conditions, indispensable et partie intégrante de tout système fondé sur l'apprentissage.

La statistique est souvent utilisée dans le cadre du traitement des langues avec l'interprétation bayésienne, c'est-à-dire que l'on s'en sert pour mettre en évidence des relations causales sous-jacentes entre les événements de la langue. Ces événements sont vus comme étant dépendants les uns des autres, et la mise en évidence des relations de causalité permet de réaliser des prédictions sur les événements futurs : la statistique est ici envisagée sous l'angle de la théorie probabiliste, utilisée pour son caractère prédictif, et par « statistique » on entend alors implicitement « inférence statistique ». Le terme désigne un ensemble de méthodes permettant d'extraire de la connaissance et de prendre des décisions à partir de données tirées d'échantillons.

Le traitement statistique des langues a pour particularité de se baser sur des quantités importantes de données : la connaissance est vue comme étant contenue implicitement dans ces données observables. Ainsi, le traitement statistique des

[1] Plus précisément dans sa version informatisée par l'ATILF (CNRS), voir http://atilf.inalf.fr/tlfv3.htm .

[2] Efron & Tibshirani, *An introduction to the bootstrap*, 1993, p. 1.

langues fait le postulat que la langue est régie par des modèles cachés. L'observation de grandes quantités de données permettrait d'approcher le comportement de ces modèles. Une fois les modèles obtenus, ils peuvent être utilisés pour réaliser des prédictions.

Nous préférons au terme *statistique* le terme *stochastique* qui, dans sa concision, ne laisse pas d'ambigüité : un processus stochastique est de manière générale un processus *qui utilise la théorie des probabilités*. Par la suite, on se réfèrera donc aux modèles statistiques de langue par « modèles stochastiques de langue », ou « modèles de langue » pour abréger.

C.2 Les modèles stochastiques de langue

C.2.1 Définitions

Le traitement statistique de la langue a pour objet, dans le cadre de l'interprétation bayésienne de la statistique, de déterminer des relations de causalité entre les différents événements qui la composent, afin d'en inférer des prédictions. Les relations de causalité sont déduites sous forme de modèles créés uniquement à partir de données linguistiques observées. L'interprétation bayésienne, qui interprète une probabilité comme la confiance qu'on accorde à une prédiction, s'oppose à l'interprétation fréquentielle dans laquelle une probabilité est vue uniquement comme une proportion.

Les modèles stochastiques de langue sont des outils largement utilisés actuellement dans des domaines très divers, allant de la reconnaissance de parole[3] ou de l'écriture graphique imprimée ou manuscrite[4], à la correction orthographique, en passant par la catégorisation automatique[5], la compression de données[6], la traduction automatique statistique[7] (SMT) ou l'évaluation de la traduction automatique[8]. Si on considère une langue comme une source émettrice de symboles (par exemple, des caractères informatiques), la fonction d'un modèle stochastique de langue est de prédire le symbole qui va suivre, ou plus précisément de réduire au maximum la surprise que va provoquer l'apparition d'un symbole, la connaissance des précédents étant acquise. Il fournit une approximation de la probabilité d'apparition d'un symbole.

On fait dans l'approche statistique l'hypothèse que la langue est une source de symboles, symboles qui possèdent chacun une probabilité d'apparition. La somme des probabilités affectées à tous les événements possibles est alors égale à 1. Dans notre étude, un symbole sera toujours un caractère. Si l'on note $C = c_1...c_n$ une chaine de n caractères, un modèle de langue spécifie une distribution $P(C)$ sur l'ensemble V^* de toutes les chaines C possibles sur l'alphabet V. On peut donc écrire :

$$\sum_{C \in V^*} P(C) = 1 \tag{C.1}$$

[3] Rosenfeld, *Two decades of statistical language modeling: Where do we go from here?*, 2000.

[4] Srihari & Baltus, *Combining statistical and syntactic methods in recognizing handwritten sentences*, 1992, p. 121.

[5] Cavnar & Trenkle, *N-gram-based text categorization*, 1994, p. 161.

[6] Cleary & Teahan, *Unbounded length contexts for PPM*, 1997, p. 67.

[7] Brown *et al.*, *A statistical approach to machine translation*, 1990, p. 79.

[8] Papineni *et al.*, *BLEU: a method for automatic evaluation of machine translation*, 2001, p. 2.

Si l'on suppose que seules les chaines d'un langage $\mathcal{L}$ sont produites, on a $P(C) = 0$ pour toute chaine $C \notin \mathcal{L}$. Si l'on considère que toute apparition d'un caractère c_i dépend uniquement de l'historique des $i-1$ caractères qui le précèdent, alors la probabilité d'occurrence $P(C)$ d'une chaine C s'écrit sous la forme suivante :

$$P(C) = \prod_{i=1}^{n} P(c_i|c_1...c_{i-1}) \tag{C.2}$$

Toutefois, si l'on dispose de longues observations de la source, $c_1...c_{i-1}$ est trop grand pour être traité. Afin de réduire la taille maximale à traiter, on recourt en pratique à une approximation sur cet historique, qu'on appelle approximation *N-gramme*. Nous introduisons dans la section suivante l'approximation N-gramme, et nous appelons par la suite *modèles de langue N-grammes* les modèles subissant cette approximation.

C.2.2 L'approximation N-gramme et ses variantes

Approche classique

On peut, comme on l'a vu ci-dessus, décomposer la probabilité $P(C) = P(c_1...c_n)$ sous la forme du chainage des probabilités conditionnelles :

$$P(C) = P(c_1) \times P(c_2|c_1) \times P(c_3|c_1c_2) \times ... \times P(c_n|c_1...c_{n-1}) \tag{C.3}$$

Soit, si on note $c_i^j = c_i...c_j$:

$$P(C) = P(c_1) \times \prod_{i=2}^{n} P(c_i|c_1^{i-1}) \tag{C.4}$$

Soit C une chaine observable de la langue : on ne peut aisément recenser toutes les probabilités d'occurrence des séquences partielles de caractères composant la chaine C de longueur n. On a donc recours à l'*approximation N-gramme* : on suppose dans cette approximation que le caractère observé dépend uniquement des $(N-1)$ caractères qui l'ont précédé. Si on note $c_i^j = c_i...c_j$, alors on peut écrire la probabilité d'occurrence d'une chaine C dans l'approximation N-gramme de la façon suivante :

$$P(C) \approx P(c_1) \times \prod_{i=2}^{N-1} P(c_i|c_1^{i-1}) \times \prod_{i=N}^{n} p(c_i|c_{i-N+1}^{i-1}) \tag{C.5}$$

Cette probabilité $P(C)$ est donc décomposée en le produit des probabilités d'occurrence des caractères c_i, cette probabilité ne dépendant plus, au maximum, que des $N-1$ caractères précédant c_i dans leur ordre d'émission.

En pratique, ces probabilités en N-grammes sont estimées sur un corpus, lors de ce qu'on appelle la phase d'apprentissage du modèle de langue. La distribution de probabilité est déduite directement et uniquement des données observées, sans impliquer par ailleurs de règle linguistique explicite, ni de connaissance extérieure.

Un avantage évident de tels modèles est que leur construction dépend uniquement des données à disposition, sans nécessiter de travail humain supplémentaire. La connaissance est extraite entièrement des données, et un modèle permet donc de rendre compte des phénomènes linguistiques qu'il a observé lors de sa phase d'apprentissage.

Le désavantage majeur de tels modèles est directement lié à cet état de fait : puisqu'aucune connaissance extérieure n'entre en jeu et que la connaissance est uniquement extraite des données, la couverture de tels modèles se limite du même coup aux données observées lors de la phase d'apprentissage. En pratique, aucune quantité de données ne suffit à couvrir la langue de façon adéquate, et il est souvent plus efficace de décrire un phénomène linguistique particulier par un certain nombre fini de règles que par une grande quantité de données. En effet, si l'ajout de données a l'avantage d'augmenter la couverture d'un modèle de langue, la redondance conduit à une augmentation des ambigüités.

Un modèle N-gramme permet donc de saisir dans une certaine mesure des caractéristiques lexicales et syntaxiques de la langue, ainsi que des phénomènes linguistiques et des dépendances locales, du moment qu'elles apparaissent de façon implicite dans les données. En terme d'appellation, dans la littérature portant sur la modélisation de langage, on appelle *N-gramme* une chaine de N unités (caractères ou mots le plus souvent), et *trigramme* une chaine de 3 unités, *bigramme* une chaine de 2 unités et enfin *unigramme* une chaine d'une seule unité.

En principe, on estime les probabilités N-grammes de type $p(c_k|c_{k-N+1}^{k-1})$ en dénombrant les occurrences de chaque chaine de caractères de longueur N dans les données d'apprentissage, ainsi que les occurrences de toutes les chaines de longueur $N-1$ précédant c_k. Si on note $Occ(c_i^j)$ le nombre d'occurrences de la chaine c_i^j dans l'ensemble d'apprentissage, on peut donc écrire :

$$P(c_k|c_{k-N+1}^{k-1}) \approx \frac{Occ(c_{k-N+1}^{k})}{Occ(c_{k-N+1}^{k-1})} \tag{C.6}$$

On se réfère habituellement à cette méthode sous le nom d'*estimateur N-gramme*, ou encore sous le nom d'estimation du maximum de vraisemblance. Cette estimation se révèle très vite limitée. En pratique, un modèle de langage ne peut pas dénombrer toutes les chaines de longueur N possibles lors de la phase d'apprentissage : aussi grand qu'un corpus puisse être, il n'est jamais exhaustif en termes de productions linguistiques. Par conséquent, on devrait attribuer une probabilité nulle à un caractère dans le cas où la séquence le précédant n'aurait pas été attestée dans les données d'apprentissage. Un modèle de langue n'a donc pas la capacité de rendre compte de la productivité de la langue, et de son pouvoir créateur tel qu'il est vu par Chomsky[9] : un locuteur d'une langue donnée est capable de créer un nombre infini de phrases de la langue, que pour la plupart il n'a jamais entendues ni prononcées auparavant.

Les données d'apprentissages seront donc toujours insuffisantes pour modéliser parfaitement la langue. À titre d'exemple, pour donner un ordre de grandeur de la difficulté, nous rapportons les résultats d'une expérience menée par Jelinek[10] dans les années 1970 chez IBM: on conduit une expérience dans laquelle on divise un corpus en un ensemble d'apprentissage et un ensemble de test de 1 500 000 et 300 000 mots respectivement. On trouve dans le corpus de test 1 000 mots différents. Il s'avère alors que 23% des trigrammes (donc des chaines de 3 mots) de l'ensemble des données de test n'apparaissent pas dans l'ensemble des données d'apprentissage. Il s'ensuit qu'un système fondé sur un tel modèle commettrait au moins 23% d'erreurs sur sa tâche de prédiction.

[9] Chomsky, *Syntactic structures*, 1957.
[10] Jelinek, *Statistical methods for speech recognition*, 1997, p. 61.

L'augmentation de la taille des données d'apprentissage ne résout jamais le problème de *rareté des données* (dans le sens de *sous-représentation des données*), qui est le problème fondamental en modélisation de la langue. Ce problème croît évidemment lorsque l'ordre du modèle N-gramme augmente. Dès lors, il est nécessaire de recourir à un *lissage* des occurrences attestées, qui affectera automatiquement aux N-grammes non vus des probabilités plus proches de la réalité. Nous détaillons plus loin plusieurs méthodes de lissage que nous mettons en œuvre dans notre étude par la suite.

Variantes

Il existe un certain nombre de variantes et déclinaisons de l'approche classique en N-gramme telle qu'exposée ci-dessus. Nous mentionnons ici les plus usitées :

- modèles N-grammes distants, ou modèles à décalage : l'approche conventionnelle en N-grammes a pour but d'estimer la probabilité d'occurrence d'un caractère, en tenant compte d'un historique de longueur $N-1$ immédiatement contigu au caractère courant. On rend ainsi compte dans une certaine mesure des contraintes linguistiques locales. Il peut cependant être intéressant dans le cas de certaines langues, ou plus généralement dans le cas de certaines constructions, d'essayer de prendre en compte des unités plus éloignées en s'intéressant à un historique qui n'est pas situé immédiatement avant le caractère à prédire. Rosenfeld[11], Huang[12] et Ney[13] montrent la relative efficacité d'une telle procédure en conjonction avec un modèle standard, sur l'anglais et une découpe en mots ;

- modèles de classes N-gramme : de tels modèles regroupent les unités en classes : on peut par exemple tenter de regrouper le vocabulaire par similarité. Les recherches visent à étudier les meilleures façons de regrouper les unités en classes significatives. Alors que de tels modèles ont été étudiés en mots[14], aucune étude n'a encore été menée à notre connaissance sur les caractères des langues à écriture idéographique ;

- modèles N-grammes tampon : Kuhn[15] considère que si, lors de l'apprentissage, on atteste une unité donnée, alors on a en pratique de grandes chances d'attester de nouveau cette même unité par la suite. L'historique des données de test peut être combiné avec un modèle N-gramme conventionnel. Cette méthode présente le désavantage d'inclure un modèle entrainé sur des données où des erreurs peuvent être présentes.

On trouve bien d'autres approches, et bien entendu leurs combinaisons ; l'étude menée par Goodman[16], si elle ne prétend pas être exhaustive, donne un aperçu com-

[11] Rosenfeld, *Adaptive statistical language modeling: A maximum entropy approach*, 1994.

[12] Huang *et al.*, *The Sphinx-II speech recognition system: An overview*, 1993.

[13] Ney *et al.*, *On structuring probabilistic dependencies in stochastic language modeling*, 1994, p. 1.

[14] Voir Kneyser & Ney, *Improved clustering techniques for class-based statistical language modelling*, 1993, ou Brown *et al.*, *Class-based n-gram models of natural language*, 1992.

[15] Kuhn & De Mori, *A cache-based natural language model for speech recognition*, 1990, p. 570 et Kuhn & De Mori, *Correction to a cache-based natural language model for speech recognition*, 1992, p. 691.

[16] Goodman, *A bit of progress in language modeling*, 2001.

paratif des différentes méthodes étudiées en modélisation de la langue, combinées entre elles, et associées à divers lissages probabilistes.

Comme nous l'avons suggéré plus haut, l'influence du lissage est importante à cause du problème récurrent de rareté des données en traitement statistique du langage. Nous détaillons dans la section suivante les notions de lissage probabiliste utilisées par la suite dans cette étude, et la signification du problème de la rareté des données.

C.2.3 Le lissage probabiliste

Problème de la rareté des données

Du fait de ce problème de rareté des données, il est en pratique nécessaire d'affecter une probabilité non nulle à des N-grammes non vus dans l'ensemble d'apprentissage : en effet, bien que ces N-grammes n'aient pas été observés lors de la construction du modèle, il est possible qu'ils appartiennent à la langue qu'est censé couvrir le modèle. À l'opposé, il faudra aussi minimiser l'importance de certaines occurrences qui ont été comptées, mais qui ne sont pas représentatives du langage modélisé. Une séquence, si elle est présente une seule fois sur un corpus de plusieurs millions ou milliards de caractères, aura une probabilité infime. Pourtant, aussi infime que soit cette information, elle est tout de même probablement exagérée face à d'autres phénomènes non rencontrés dans la phase d'apprentissage, et auxquels on aura par construction affecté une probabilité nulle.

Le lissage a donc pour propos de redistribuer les probabilités. Un lissage repose sur les deux méthodes fondamentales suivantes :

- Réduction du nombre d'occurrences : les occurrences des chaines de longueur N les plus fréquentes sur l'ensemble d'apprentissage sont réduites, et cette quantité de probabilité est redistribuée sur les chaines moins fréquentes ;
- Retour à un ordre inférieur : si on ne dispose pas de suffisamment de données pour déterminer une probabilité en N-grammes, on « bat en retraite » sur l'ordre inférieur $N-1$ affecté d'une pondération :

$$P(c_k|c_{k-N+1}...c_{k-1}) \approx B(c_{k-N+1}...c_{k-1}) * P(c_k|c_{k-N+2}...c_{k-1}) \qquad \text{(C.7)}$$

  ou B est une pondération fonction de l'historique. Si une chaine de longueur N n'est pas attestée dans les données, on peut espérer que la chaine préfixe de longueur $N-1$ (tronquée d'une unité, la plus « distante ») pourra être attestée. Cette technique de repli sur l'ordre inférieur est appelée en anglais *back-off* (*repli*, dans le vocabulaire militaire), ou *technique de repli*.

Méthodes de lissage

Nous détaillons dans les paragraphes suivants les méthodes de lissage qui seront considérées et utilisées dans la suite de notre étude (voir partie II, chapitre 2).

Lissage de Laplace et Lidstone L'idée simple de ce type de lissage est de considérer que toute chaine a été attestée un certain nombre α de fois de plus qu'elle ne l'a été lors du comptage réel de ses occurrences sur l'ensemble des données d'apprentissage. Si $\mathcal{V}$ est la taille de l'alphabet fini utilisé :

$$P(c_k|c_{k-N+1}^{k-1}) \approx \frac{Occ(c_{k-N+1}^{k}) + \alpha}{Occ(c_{k-N+1}^{k-1}) + \mathcal{V}\alpha} \tag{C.8}$$

Pour $\alpha \leq 1$ quelconque, on parle de loi de Lidstone ; pour le cas $\alpha = 1$, on parle de loi de Laplace. Dans le cas général d'un lissage de Lidstone ou de Laplace, on n'a donc pas recours au retour à un ordre inférieur, ni à une réduction des occurrences : les occurrences ne sont pas redistribuées, mais simplement augmentées de α.

Le lissage de Lidstone-Laplace alloue en pratique la quasi-totalité des probabilités à des chaines non vues au cours de l'apprentissage : il faudrait privilégier davantage les chaines vues une fois au cours de l'apprentissage, sur celle qui n'ont pas été vues. Le lissage Lidstone-Laplace ne fait pas cela.

Lissage de Good-Turing Good[17] propose de récupérer une partie de la masse probabiliste en réestimant les fréquences d'ordre r (chaines de longueur n apparues r fois dans le corpus d'apprentissage). Les chaines apparaissant un nombre r de fois sont regroupées en une même classe. Cette estimation est notée $r'(r)$ et appelée réestimation de Good-Turing du rang r. Pour le calcul de la réestimation $r'(r)$ de r, l'idée est de modéliser un comportement zipfien, dans le sens que le nombre de chaines apparaissant fréquemment est petit, alors que le nombre de chaines apparaissant peu fréquemment est grand. $r'(r)$ est fonction du rapport entre le nombre d'occurrences au rang r et celui au rang immédiatement supérieur $r + 1$.

La probabilité estimée par un lissage de Good-Turing prend la forme $P_{GT} = r'(r)/M(n)$, où $M(n)$ est le nombre total de chaines de longueur n attestées dans le corpus d'apprentissage et $r'(r)$ la réestimation de Good-Turing du rang r. Si on note I_r le nombre de chaines de longueur n apparaissant r fois dans le corpus d'apprentissage, alors on peut exprimer $r'(r)$ sous la forme suivante :

$$r'(r) = (r+1)\frac{I_{r+1}}{I_r} \tag{C.9}$$

Toutefois, une telle réestimation ne peut être effectuée pour les valeurs élevées de r : la chaine de longueur n la plus fréquente se verrait attribuer une probabilité nulle, puisque dans ce cas $I_{r+1} = 0$ (on ne peut observer de chaine de longueur n plus fréquente que la plus fréquente des chaines de longueur n attestée). On applique donc ce lissage uniquement sur les fréquences faibles. Good attribue la formalisation originelle de cette réduction à Turing. Ainsi, pour toute chaine $c_1...c_n$ de longueur n apparue r fois dans le corpus d'apprentissage, on obtient une nouvelle probabilité $P(c_1...c_n)$:

$$P(c_1...c_n) = \frac{r'(r)}{M(n)} = \frac{(r+1) * I_{r+1}}{M(n) * I_r} \tag{C.10}$$

Dans le cas où la chaine $c_1...c_n$ n'a pas été vue, on a :

$$P_{GT}(c_1...c_n) = \frac{I_1}{I_0 * M(n)} \tag{C.11}$$

où I_1 est le nombre d'hapax de longueur n, I_0 le nombre de chaines non-attestées de longueur n, et $M(n)$ le nombre total de chaines de longueur n dans le corpus

[17] Good, *The population frequencies of species and the estimation of population parameters*, 1953, p. 237.

d'apprentissage. Au total, la probabilité enlevée aux rangs supérieurs est redistribuée de façon uniforme sur les chaines non observées.

Le lissage de Good-Turing est intéressant dans le cas des chaines absentes ou peu fréquentes dans le corpus d'apprentissage, mais lorsqu'il s'agit des chaines aux fréquences d'apparition élevées, les fréquences estimées ont tendance à être fortement bruitées. De manière générale, on applique donc le lissage de Good-Turing aux chaines apparaissant peu fréquemment, et on conserve l'estimation du maximum de vraisemblance pour les chaines de fréquences élevées[18].

Autres méthodes de lissage Il existe d'autres méthodes de lissage, dont nous citons ici quelques unes des plus connues :

- Katz[19] reprend le principe du lissage Good-Turing, mais utilise l'idée du retour à un ordre inférieur pour effectuer la combinaison de modèles d'ordres différents. Ce lissage produit des résultats relativement bons pour un cout réduit de mise en œuvre ;

- Jelinek et Mercer[20] utilisent une idée proche, à la différence que les modèles N-grammes d'ordre inférieur sont tous utilisés par l'estimateur, et ce jusqu'à l'unigramme (donc jusqu'au modèle donnant la probabilité absolue d'apparition d'une unité, c'est-à-dire sans tenir compte d'un quelconque historique la précédant) ;

- Ney[21] utilise lui aussi une méthode d'interpolation, mais retire de chaque compte positif une quantité fixe (la réduction est constante). Ce lissage est plus connu sous le nom de *réduction absolue* (*Absolute-Discounting*) ;

- Witten et Bell[22] proposent une méthode de lissage plus adaptée aux modèles de langage pour la compression de données, dont le principe est de modéliser la probabilité d'un événement non attesté par l'estimation qu'on a de le rencontrer au fur et à mesure du traitement du corpus d'apprentissage ;

- enfin Kneyser et Ney[23] proposent une extension de la *réduction absolue* fondée sur un retour à l'ordre inférieur, mais avec une estimation plus précise de la distribution sur laquelle on se rabat. C'est en pratique le lissage qui produit les meilleurs résultats, mais sa complexité fait qu'il est difficile à mettre en place.

Ce panorama des méthodes de lissage les plus courantes a présenté les méthodes les plus utilisées actuellement en traitement statistique des langues. Pour une liste

[18] Le seuil de cette fréquence est en soi sujet à discussion, mais en pratique le lissage produit des estimées raisonnables pour des chaines de longueur N observées moins de 10 fois.

[19] Katz, *Estimation of probabilities from sparse data for the language model component of a speech recognizer*, 1987, p. 400.

[20] Jelinek & Mercer, *Interpolated estimation of Markov source parameters from sparse data*, 1980, p. 381.

[21] Ney *et al.*, *On structuring probabilistic dependencies in stochastic language modeling*, 1994, p. 1.

[22] Witten & Bell, *The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression*, 1991, p. 1085.

[23] Kneyser & Ney, *Improved backing-off for m-gram language modeling*, 1995, p. 181.

exhaustive de ces méthodes et une présentation de leurs performances expérimentales, on pourra se référer à Chen et Goodman[24].

Nous abordons dans les sections suivantes le problème de l'évaluation de cette performance.

C.3 Éléments de théorie de l'information appliqués aux modèles de langue

C.3.1 Introduction au cadre de la théorie de l'information

Si on se place dans le cadre de la théorie de l'information, on peut considérer un langage comme une source émettant des symboles, et un point de la chaine de symboles à traiter comme l'observation ponctuelle de l'émission de cette source. À chaque émission de symbole, l'incertitude qui plane sur le comportement de la source sera diminuée par la connaissance nouvelle que nous aurons de son comportement. On appelle entropie de la source la valeur moyenne de l'incertitude portant sur un symbole émis, par analogie avec la mesure de désordre thermodynamique (ces deux grandeurs étant homogènes). L'observation suivie de cette émission ne pourra dès lors que diminuer cette entropie. L'entropie est maximale :

- si le symbole émis ne dépend pas de ceux qui l'ont précédé : la source est sans mémoire ;
- si les symboles sont équiprobables : la source est uniforme.

À l'inverse, lorsqu'il s'agit de traiter la langue, on fait l'hypothèse qu'il existe des dépendances entre les symboles émis et que cette émission n'est pas uniforme.

C.3.2 Entropie d'une chaine de caractères

Dans le cadre de la théorie de l'information de Shannon, le caractère inattendu d'un événement, ou encore la quantité de surprise liée à l'apparition d'un événement, s'exprime par l'opposé du logarithme de sa probabilité : cette quantité d'information contenue dans une chaine de caractères $C = c_1c_2...c_k$ est ainsi $-log_bP(C)$ en base b, avec P la probabilité d'occurrence a priori de la chaine C. L'entropie d'une chaine de caractères C en base b est alors la somme des entropies de tous les caractères de la chaine, et s'écrit donc[25] :

$$H(C) = -\sum_{i=1}^{k} P(c_i) log_b P(c_i) \tag{C.12}$$

où $P(c_i)$ est la probabilité d'occurrence a priori du caractère c_i. La raison pour laquelle on choisit la base 2 (binaire) est qu'on a choisi l'unité fixée par l'utilisation de l'expérience la plus élémentaire en probabilités : une expérience à deux issues équiprobables, telle que l'obtention de pile ou face sur le lancer d'une pièce. Par la suite, on considèrera qu'on se place en base 2 (binaire) et toute notation H_2 sera abrégée en H.

[24] Chen & Goodman, *An empirical study of smoothing techniques for language modeling*, 1999, p. 359.

[25] Pour la justification complète, voir Jelinek, *Statistical methods for speech recognition*, 1997, p. 115-119.

Si on applique cette valeur aux chaines de caractères $c_1^k = c_1...c_k$ de longueur k, en considérant une telle chaine comme instance d'une variable aléatoire X_1^k, $p(c_1^k) = p(X_1^k = c_1^k)$:

$$H(c_1^k) = -\sum_{X_1^k} P(c_1^k) log P(c_1^k) \tag{C.13}$$

En divisant par k, on peut alors en déduire la valeur moyenne d'information d'un caractère :

$$\frac{1}{k} H(c_1^k) = -\frac{1}{k} \sum_{X_1^k} P(c_1^k) log P(c_1^k) \tag{C.14}$$

C.3.3 Entropie d'un langage

On fait l'hypothèse que l'entropie d'une source peut être évaluée grace à une seule observation suffisamment longue de ses productions : c'est l'hypothèse d'ergodicité de la source. On considère que l'observation des réalisations de la source pendant un temps infini nous donne l'entropie totale du langage $\mathcal{L}$ considéré. Si on fait tendre la longueur k de chaine vers l'infini, on peut donc écrire pour toutes les chaines c_1^k observées :

$$H(\mathcal{L}) = -\lim_{k \to \infty} \frac{1}{k} \sum_{X_1^k} P(c_1^k) log P(c_1^k) \tag{C.15}$$

Expérimentalement, la longueur k de la chaine est bornée[26] : il n'est pas possible d'observer toutes les réalisations c_1^k de la langue. L'hypothèse d'ergodicité revient à considérer que tout ce qui peut être produit par la source sera produit : appliquée à la langue, cela suppose par exemple que tout ce qui est dicible en français sera un jour prononcé ou écrit. On peut alors écrire pour cette seule observation c_1^k :

$$H(\mathcal{L}) = -\lim_{k \to \infty} \frac{1}{k} log P(c_1^k) \tag{C.16}$$

on ne fait plus la somme sur toutes les chaines c_1^k pondérées par leur probabilité d'apparition a priori, puisqu'on observe une seule chaine. Ainsi, si on calcule $H(\mathcal{L})$ à partir d'une chaine c_1^k de longueur k grande :

$$H(\mathcal{L}) \approx -\frac{1}{k} log P_M(c_1^k) \tag{C.17}$$

où P_M est la distribution de probabilité modélisée par une des méthodes citées plus haut (en général par estimation du maximum de vraisemblance et lissage probabiliste), une approximation de la distribution réelle de la source.

Cette mesure entropique représente la moyenne de l'effet de surprise que provoque l'apparition d'une chaine inconnue (c'est-à-dire non vue lors de la phase d'apprentissage). Le but de la modélisation de langue est de réduire au maximum cet effet de surprise en faisant des prédictions les plus exactes possibles, c'est pourquoi on voudra minimiser cette valeur autant que possible. On s'intéresse dans les paragraphes suivants aux mesures relatives à l'évaluation de la qualité d'un modèle stochastique de langage.

[26] Une production peut être longue, mais pas infinie.

C.3.4 Entropie croisée

Définition

Soit S une source de chaines de caractères C spécifiant une distribution $P(C)$, on a vu qu'on pouvait écrire :

$$H(S) = -\sum_{C} P(C) log P(C) \tag{C.18}$$

Si on note P_M un modèle de cette source censé rendre compte des émissions de S, on définit alors l'entropie croisée pour une source S de distribution $P(C)$, estimée par un modèle P_M de distribution $P_M(C)$:

$$H(S, P_M) = -\sum_{C} P(C) log P_M(C) \tag{C.19}$$

On dit que cette entropie est croisée, car elle met en relation une source, et un modèle de cette source[27]. Cette mesure d'entropie est en effet la somme des estimations par le modèle P_M de l'entropie des différentes chaines c produites par la source, ces chaines ayant chacune une probabilité d'occurrence bien réelle valant $P(C)$. L'entropie du modèle estimé P_M est supérieure à celle du modèle correct P : en effet, l'entropie d'une variable aléatoire est inférieure ou au mieux égale à son entropie croisée calculée dans un modèle imparfait[28].

$$H(S) \leq H(S, P_M) \tag{C.20}$$

L'égalité n'est atteinte que pour $P = P_M$. On peut en déduire une méthode de comparaison de plusieurs modèles estimés concurrents : le meilleur modèle produit l'entropie croisée la plus faible.

De la même manière qu'on a défini précédemment l'entropie d'un langage $\mathcal{L}$ en faisant tendre la longueur de chaine vers l'infini, on peut définir l'entropie croisée d'un langage $\mathcal{L}$ pour un modèle M :

$$H(\mathcal{L}, P_M) = -\lim_{k \to \infty} \frac{1}{k} \sum_{c_1^k} P(c_1^k) log P_M(c_1^k) \tag{C.21}$$

Applications à la modélisation de langue

Rigoureusement, si on voulait évaluer la qualité pratique d'un modèle stochastique de langue, on devrait l'inscrire dans une application complète de traduction automatique ou de reconnaissance automatique de parole.

En première approximation, et ce pour des raisons compréhensibles de mise en œuvre, on préfère évaluer le choix moyen auquel doit faire face le modèle lors d'une prédiction. Comme on l'a rapidement mentionné dans la section précédente, l'impératif d'une évaluation non biaisée est d'effectuer les tests sur des données non vues au cours de la phase d'apprentissage. On introduit alors le principe de *validation croisée sur N parties*, aussi appelé *leaving-one-out* : le corpus d'apprentissage est divisé en N parties, $N-1$ d'entre elles sont utilisées pour l'apprentissage du modèle et une seule pour le test. Cette méthode est répétée N fois en laissant de

[27] Il serait peut être plus logique de parler d'entropie *relative*, mais cette appellation désigne déjà une autre notion connue, la divergence de Kullback-Leibler.

[28] Pour la justification, voir Jelinek, *Statistical methods for speech recognition*, 1997, p. 132.

côté successivement chacune des N parties. (Nous utilisons systématiquement cette méthode dans la partie II, chapitre 2)

On ne peut malheureusement pas utiliser directement l'entropie croisée sous la forme exprimée dans l'équation C.21 pour comparer des modèles, car elle nécessite la connaissance de $P(c_1^k)$, c'est-à-dire la probabilité d'occurrence de telle ou telle chaine a priori. On utilise alors l'hypothèse d'ergodicité énoncée plus haut, qui suppose qu'en faisant l'observation d'une chaine suffisamment longue, on peut estimer cette probabilité en observant un ensemble de données important, et représentatif de la source. L'expression peut alors être simplifiée :

$$H(\mathcal{L}, P_M) \approx - \lim_{k \to \infty} \frac{1}{k} log P_M(c_1^k) \tag{C.22}$$

Cette hypothèse étant faite, l'entropie croisée peut alors être utilisée comme outil de comparaison (dans le cadre de la validation croisée sur N parties), et par la suite nous tâcherons de définir diverses mesures qui nous seront nécessaires dans le but de mieux comprendre, comparer et adapter différents corpus.

On peut distinguer au premier abord deux approches opposées pour l'utilisation de l'entropie croisée :

- dans un premier cas, comme décrit précédemment et illustré par la figure C.1, on recherche un modèle qui décrit, estime le mieux possible le langage $\mathcal{L}$.

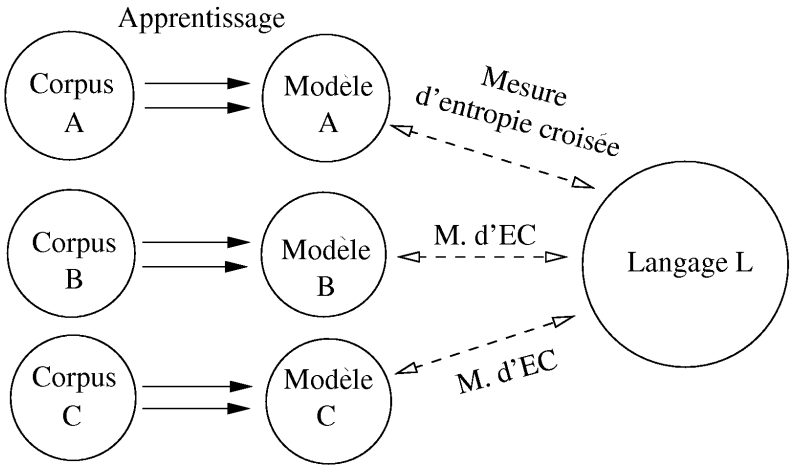

Figure C.1: Comparaison de modèles de langue pour décrire un langage $\mathcal{L}$

  On verra donc de façon pratique quel modèle est le plus approprié pour décrire $\mathcal{L}$, et on peut ici imaginer une « distance » mesurant la dissimilarité qui différencie deux ensembles donnés. Nous y trouvons une application dans la mise au point de mesures comparatives de corpus en terme de similarité, que nous exposerons en partie II, chapitre 2.

- dans un deuxième cas, illustré par la figure C.2, la mesure d'entropie croisée se rapproche de celle qu'on rencontrerait dans les domaines du codage ou de la compression de données.

  L'entropie croisée d'une chaine de caractères est alors le nombre de bits minimum dont on aura besoin pour l'encoder avec le modèle de référence. Là encore, l'utilité est de juger de la « distance » entre la chaine et le corpus utilisé

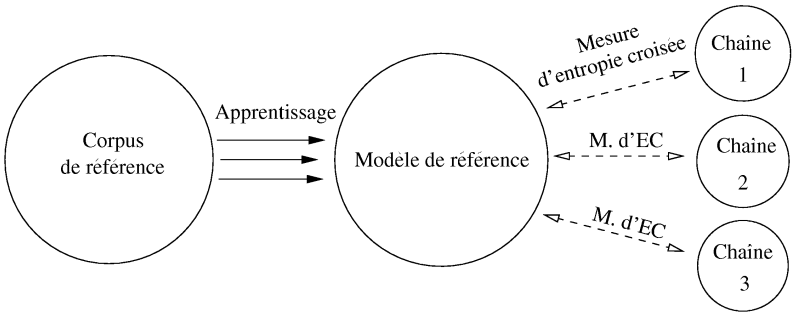

Figure C.2: Mesure de l'entropie croisée de plusieures chaines par un modèle de référence

pour obtenir le modèle d'encodage de référence. Une application pratique possible est le rejet automatique de phrases agrammaticales produites automatiquement, premier pas vers la production de corpus de données synthétiques. Nous étudions une telle application en partie II, chapitre 1.

L'entropie croisée, étant une simple valeur réelle fortement dépendante du choix des données, doit être maniée avec précaution. On peut directement déduire de ce critère la notion de perplexité, très répandue depuis la fin des années 70.

C.3.5 Perplexité

Définition

Jelinek[29] reprend la mesure entropique exprimée ci-dessus pour exprimer similairement la difficulté de la tâche vue par un éventuel reconnaisseur situé en aval du modèle de langue, en définissant la *logprob* (notée LP) :

$$LP = -\lim_{k\to\infty} \frac{1}{k} \sum_{i=1}^{k} logP(c_i|c_1^{i-1}) \tag{C.23}$$

Il en déduit une valeur qu'il nomme *perplexité* (notée PP), et qui représente le facteur de branchement moyen du modèle pour la tâche :

$$PP = 2^{LP} \tag{C.24}$$

La perplexité représente l'amplitude du choix équiprobable moyen auquel est confronté le modèle évalué dans sa tâche de prédiction.

Applications à la modélisation des langues

Il faut souligner que, bien que la perplexité donne une idée de la complexité de la tâche du « point de vue d'un reconnaisseur », et par là une idée de la qualité du modèle stochastique de langue, elle n'est pas pour autant une mesure directe de la difficulté réelle de cette tâche. Cette estimation est en effet fondée sur les hypothèses temporelles faites sur la source, et elle ne prend pas en compte les autres aspects du système. C'est pourquoi dans la partie II, chapitre 2, nous commençons par effectuer

[29] Bahl *et al.*, *Perplexity - a measure of the difficulty of speech recognition tasks*, 1977.

une expérience sur la perplexité des modèles de langue avant d'évaluer concrètement les performances de systèmes de traduction automatique.

Une des faiblesses reconnues[30] de la perplexité est sa faible corrélation avec le taux d'erreur final du système par rapport à l'entropie croisée, avec laquelle elle entretient un simple rapport logarithmique. Habituellement une réduction de la perplexité inférieure à 5% n'est pas significative, alors qu'une réduction de plus de 10% est intéressante (Rosenfeld parvient[31] à une réduction de plus de 30% en utilisant une modélisation par maximum d'entropie[32]).

C.4 Conclusion

Nous avons rappelé dans cette annexe plusieurs notions théoriques liés à la théorie de l'information, et aux modèles statistiques de langue. Notre étude se plaçe résolument dans une approche orientée par les données, nous portons donc une attention particulière aux méthodes visant à un traitement automatique ou semi-automatique de grandes quantités de données linguistiques.

La modélisation statistique en traitement automatique des langues apporte au travers des notions d'entropie croisée et de perplexité des méthodes pour décrire, prédire le mieux possible des données de la langue. La qualité de cette prédiction est évaluée grace à l'entropie croisée.

Ces outils seront utilisés principalement dans la partie II : dans le chapitre 1, nous utilisons des modèles de langue pour estimer l'entropie d'énoncés produits automatiquement, afin de juger de leur grammaticalité. Dans le chapitre 2, nous estimons la similarité de données linguistiques en terme de registre oral ou écrit grace à des modèles construits sur plusieurs types de données.

[30] Chen *et al.*, *Evaluation metrics for language models*, 1998.
[31] Rosenfeld, *Two decades of statistical language modeling: Where do we go from here?*, 2000.
[32] Rosenfeld, *Adaptive statistical language modeling: A maximum entropy approach*, 1994.

Annexe D

Introduction à l'analogie entre chaines de caractères

D.1 Introduction

Nous donnons ici une courte introduction à l'analogie et à son application sur les chaines de caractères, telle qu'elle est décrite par Yves Lepage[1]. Tout comme lui, nous pensons qu'un exemple bien choisi peut montrer en condensé les caractéristiques de l'analogie. Afin de lui rendre un hommage mérité, nous citons comme lui un grand maitre des néologismes, San Antonio[2] :

> C'est la vie, lieux-communis-je.

Dans ce premier passage, un verbe du deuxième groupe, le néologisme *lieux-communir*, formé à partir du mot composé *lieu commun*, est conjugué de la même manière que le verbe *sortir* se conjugue en *sortis-je*. On peut dire que *sortis-je* est à *sortir* ce que *lieux-communis-je* est à *lieux-communir*, et on note :

$$\textit{sortir} : \textit{sortis-je} :: \textit{lieux-communir} : \textit{lieux-communis-je}$$

Cet exemple illustre plusieurs caractéristiques de l'analogie :

- premièrement, l'analogie est universelle. Son fonctionnement est d'ailleurs supposé connu de tous dans un tel jeu de mots, et tout locuteur de la langue française le comprendra sans mal.
- deuxièmement, l'analogie est créatrice. Elle autorise en effet la conjugaison du néologisme *lieux-communir*, et est à l'origine du comique de la tirade : le jeu de mots prend forme grace à une utilisation surintensive de la langue, dont le caractère générateur, créateur, est poussé à l'extrême.
- enfin, l'analogie est aveugle. Appliquée sur les chaines de caractères, elle met en relation des éléments selon une découpe qui peut être inexacte. L'absurdité du résultat contraste ici avec l'application créatrice de l'analogie, ce qui donne l'effet comique. On a en effet conjugué un verbe qui n'existe pas, mais selon un modèle de conjugaison tout à fait valide.

[1] Lepage, *De l'analogie rendant compte de la commutation en linguistique*, 2003.
[2] San-Antonio, *Bérurier au sérail*, 1964.

Dans cette étude, nous nous cantonnons uniquement à l'analogie appliquée sur les chaines de caractères. Nous développons ces trois caractéristiques majeures de l'analogie dans les sections suivantes.

D.2 Premier avantage : une opération universelle

L'analogie est indépendante des symboles utilisés. On peut remplacer un symbole par un autre sans modifier l'opération analogique sous jacente Cette propriété remarquable peut être illustrée de la façon suivante. On peut écrire :

$$ab : aabb :: aaabbb : aaaabbbb$$

ou encore :

$$cd : ccdd :: cccddd : ccccdddd$$

C'est la même analogie qui entre en jeu : la relation analogique est indifférente aux représentations des symboles. La structure qu'elle engendre est présumée universelle chez l'homme, dans le sens que tout homme est capable de comprendre et de former des analogies sur un ensemble de symboles (là encore, l'exemple de San Antonio est parfaitement illustratif). Ce faisant, l'analogie permet comme on l'a vu de créer des chaines de symboles inédites.

D.3 Deuxième avantage : une opération créatrice

On vient de le dire : l'analogie peut créer des chaines de symboles inédites. L'opération permet en effet de générer des données à partir de données existantes : à partir de trois chaines de caractères existantes, il est possible de générer une quatrième. Par exemple, si l'on choisit l'analogie suivante :

$$feins : feindre :: teins : x$$

et que l'on cherche une solution pour x, alors on peut écrire $x = teindre$. À partir de trois éléments, les formes impératives et infinitives du verbe *feindre* plus la forme impérative du verbe *teindre*, on a produit la forme infinitive du verbe *teindre*. Dans cet exemple précis, l'analogie a réalisé uniquement une seule commutation entre suffixes. Afin que l'on ne croie pas qu'une analogie se limite aux cas d'une telle simplicité, nous donnons un autre exemple.

Il va de soi que l'analogie peut s'appliquer sur toute chaine de caractères, quelle qu'elle soit. Puisque l'espace entre mots est un caractère informatique comme un autre, rien n'empêche qu'il soit traité comme un autre dans l'analogie suivante :

$$Il\ ouvre\ la\ fenêtre. : Tu\ ouvres. :: Il\ ferme\ la\ fenêtre. : x$$

d'où l'on déduit une solution $x = Tu\ fermes.$ Cet exemple montre deux choses : l'analogie agit sur toute la chaine, quelle qu'elle soit, et c'est à tort qu'on a tendance à la limiter aux mots ; l'analogie de se limite pas à un simple échange de suffixes, elle peut réaliser en une passe préfixation, suffixation et infixation multiples. Ainsi, dans notre exemple l'analogie a « propagé » du verbe *ouvrir* au verbe *fermer* la construction de la deuxième personne du l'indicatif présent : sur la racine *ferm*

s'est ainsi rajouté le préfixe *Tu*, pronom personnel, avec un espace, et le suffixe *es*, morphologiquement terminaison du verbe *fermer* à la deuxième personne de l'indicatif présent. Au lieu de voir l'analogie qui a été réalisée comme deux opérations (préfixation et suffixation) successives, ce qu'elle n'est pas, on doit la voir comme une seule opération d'infixation multiple. On pourrait choisir une infinité d'exemples du même type afin de démontrer le caractère créateur de l'analogie qui distribue l'information sur toute la chaine de caractères.

D.4 Piège : une opération aveugle

Comme on l'a vu dans le premier exemple extrait de San Antonio, l'analogie recèle un piège lorsqu'elle est appliquée à la langue : l'opération se fait de manière aveugle sur les chaines de caractères, le résultat d'une équation peut donc éventuellement être incorrect lorsqu'on traite des chaines appartenant à la langue. Le résultat peut se révéler incorrect de deux façons :

- tout d'abord du point de vue de la langue. La résolution d'une équation analogique mettant en relation 3 chaines de caractères appartenant à la langue peut produire un résultat agrammatical. Par exemple, si l'on considère l'exemple suivant :

  $$\textit{feins} : \textit{feindre} :: \textit{tiens} : x$$

  la résolution de l'analogie donne la solution[3] $x = \textit{tiendre}$, au lieu de $x = \textit{tenir}$.

  Ce problème peut être en partie jugulé par des méthodes simples telles que celles décrites dans la partie II, chapitre 1, traitant de la détection de la grammaticalité.

- ensuite au niveau du sens. En effet, la résolution d'une équation analogique mettant en relation 3 chaines de caractères appartenant à la langue peut produire un résultat absurde au niveau du sens. Par exemple, si l'on considère l'exemple suivant :

  $$\textit{Je prends la valise} : \textit{Je porte une valise} :: \textit{Je prends le train} : x$$

  alors on trouve une solution $x = \textit{Je porte un train}$, qui si elle est indéniablement grammaticale, n'est pas correcte en sens[4]. Nous nous intéressons à une quantification de ces erreurs de sens dans la partie III, chapitre 1, lors de la production automatique de paraphrases.

[3] Au passage, on notera que ce type de confusion est courant chez les enfants dans leur période d'apprentissage de la langue. On entendra ainsi souvent *viendre* au lieu de *venir* : on peut penser que le processus d'apprentissage de la langue chez l'enfant passe probablement par une acquisition des règles, de la structure à partir d'analogies formées sur les éléments déjà maitrisés de la langue. À ce sujet, voir de Saussure, *Cours de linguistique générale, édition 1916*, 1995, pp. 231.

[4] A moins que ça ne soit un train électrique...

D.5 Résolution algorithmique d'une équation analogique entre chaines de caractères

D.5.1 Algorithme

Bien que l'analogie ait été souvent mentionnée et utilisée en linguistique, peu de propositions algorithmiques ont été faites pour la résolution d'analogies entre chaines de caractères, probablement parce que l'opération peut sembler, trompeusement, très « intuitive ». À notre connaissance, et si l'on met de côté Copycat[5], qui adopte un point de vue provenant de l'intelligence artificielle, peu utile pour un traitement linguistique, Yves Lepage[6] a été le premier à proposer un algorithme pour la résolution d'équations analogiques. L'algorithme se base sur la formalisation des analogies en terme de distance d'édition, ou de façon équivalente en terme de similarité[7]. Nous nous contentons ici de résumer très brièvement la démarche[8].

On note $\mathrm{sim}(A, B, \ldots, N)$ la longueur de la plus longue sous-séquence commune aux chaines A, B, $\ldots N$, et on l'appelle *similitude*. Dans la formule suivante[9], l'inconnue D est placée systématiquement à gauche des égalités, de manière à résoudre les équations analogiques :

$A : B :: C : D \quad \Rightarrow$

$$\begin{cases} \mathrm{sim}(B,D) &= -|A| + |B| + \mathrm{sim}(A,C) \\ \mathrm{sim}(C,D) &= -|A| + |C| + \mathrm{sim}(A,B) \\ \mathrm{sim}(\mathrm{A,B,\ldots,N}) &= -|A| + \mathrm{sim}(A,B) + \mathrm{sim}(A,C) \\ |D| &= -|A| + |B| + |C| \end{cases}$$

On procède ensuite étape par étape durant la résolution, qui s'inspire d'Itkonen[10] : la chaine A sert d'axe pour comparer des couples de chaines B et C lors de la construction de la chaine D solution de l'équation analogique.

D.5.2 Exemple

Afin d'illustrer de façon plus parlante l'algorithme exposé, résolvons un exemple d'équation analogique particulière :

$$\textit{like} : \textit{unlike} :: \textit{known} : \texttt{x}.$$

On notera que la résolution se fait bien ici en caractères : bien que l'exemple cité mette en relation trois mots graphiques, le même algorithme s'applique sur trois phrases vues comme des chaines de caractères. L'espace ou tout séparateur en général, est de fait traité comme un caractère comme les autres. L'algorithme s'applique ainsi commodément sur des langues comme le japonais, le chinois ou le coréen, et que le caractère fasse un ou deux octets en taille.

[5] Voir Hofstadter & the Fluid Analogies Research Group, *Fluid concepts and creative analogies*, 1994, p.205-265.

[6] Lepage, *Solving analogies on words: an algorithm*, 1998

[7] Voir Stephen, *String searching algorithms*, 1994, chap.3

[8] Pour l'intégralité, les détails, et même plus, voir Lepage, *De l'analogie rendant compte de la commutation en linguistique*, 2003.

[9] Pour la justification, voir Lepage, *De l'analogie rendant compte de la commutation en linguistique*, 2003, pp. 151.

[10] Itkonen & Haukioja, *Grammaticalization: Abduction, analogy, and rational explanation*, 1999

Les similitudes entre A et B, puis entre A et C sont ainsi calculées à l'aide d'un algorithme rapide[11]. Seules des bandes diagonales limitées peuvent être prises en compte dans les matrices[12]. Dans les matrices suivantes, l'algorithme suit le chemin indiqué par les valeurs encerclées, de façon similaire à la démarche employée pour déterminer la trace d'une distance d'édition[13].

	e	k	i	l	n	u		k	n	o	w	n
.	.	.	.	①	⓪	⓪	l	⓪	⓪	.	.	.
.	.	②	1	0	.	.	i	.	0	⓪	.	.
.	③	2	1	.	.	.	k	.	.	1	①	.
④	3	2	.	.	.	.	e	.	.	.	1	①

Les mouvements successifs déclenchent la copie de caractères dans la solution D, d'après des « règles » qui disent à partir de quelle chaine B ou C il faut choisir le caractère à recopier en fonction des mouvements dans les deux matrices. Enfin, la solution $\mathtt{x} = \textit{unknown}$ est produite.

dir_{AB}	dir_{AC}		copie en D	depuis la chaine
diagonale	diagonale		n	C
diagonale	diagonale		w	C
diagonale	diagonale		o	C
diagonale	diagonale		n	C
horizontale	horizontale		k	C
horizontale	diagonale		n	B
horizontale	diagonale		u	B

L'exemple choisi est volontairement simpliste, puisqu'il s'agit d'ajouter un préfixe à *known*. L'algorithme permet cependant de traiter en une seule passe des infixations multiples, comme par exemple dans l'exemple suivant :

$$\textit{aslama} : \textit{muslimun} :: \textit{arsala} : x \quad \Rightarrow \quad x = \textit{mursilun}$$ [14]

Cela est indispensable, puisque la résolution d'analogies mettant en relation des chaines de caractères faisant partie de la langue nécessite dans le cas général d'effectuer plusieurs infixations en une passe. Il est aussi intéressant de noter qu'une équation analogique peut produire zéro, une, ou plusieurs solutions.

D.6 Conclusion

Nous avons brièvement présenté dans cette annexe la relation d'analogie sur les chaines de caractères. Nous avons vu, à travers des exemples, qu'elle présente deux aspects intéressants pour le traitement automatique des langues : son caractère universel, et son caractère créatif. En revanche, de par son caractère aveugle, nous avons montré que son application pouvait parfois être trompeuse. Enfin, nous avons

[11] Allison & Dix, *A bit string longest common subsequence algorithm*, 1986

[12] Cette propriété est un résultat obtenu par Ukkonen, *Algorithms for approximate string matching*, 1985

[13] Wagner & Fischer, *The string-to-string correction problem*, 1974

[14] Arabic: *arsala* (il envoya) et *aslama* (il se convertit [à l'Islam]) sont des verbes au passé de la 3ème personne du singulier ; *mursilun* (un envoyé) et *muslimun* (un converti, *i.e.*, un musulman) sont des noms.

rappelé un algorithme permettant la résolution d'équations analogiques, et montré son application à travers un exemple simple.

L'analogie nous permet de traiter des phrases par un algorithme fonctionnant sur des chaines de caractères : nous nous en servons en partie III, chapitre 1, afin de produire automatiquement des paraphrases, puis en partie III, chapitre 2, afin de réaliser un système de traduction automatique.

Bibliographie

D. W. Aha, « Lazy learning: Special issue editorial », *Artificial Intelligence Review*, vol. 11, p. 7–10, 1997.

Yasuhiro Akiba, Marcello Federico, Noriko Kando, Hiromi Nakaiwa, Michael Paul et Jun'ichi Tsujii, « Overview of the IWSLT04 Evaluation Campaign », *in* Proceedings of the International Workshop on Spoken Language Translation, p. 1–12, Kyoto, Japan, 2004.

Lloyd Allison et Trevor I. Dix, « A bit string longest common subsequence algorithm », *Information Processing Letter*, vol. 23, p. 305–310, 1986.

ALPAC, *Language and Machines. Computers in Translation and Linguistics*, Washington, D.C., Automatic Language Processing Advisory Committee. Division of Behavioral Sciences, National Academy of Sciences, National Research Council, 1966.

Alexandre Andreewsky, Fathi Debili et Christian Fluhr, « Computational Learning of Semantic Lexical Relations for the Generation and Automatical Analysis of Content », *in* Proceedings of the IFIP Congress, p. 667–672, 1977.

Bogdan Babych et Anthony Hartley, « Modelling legitimate translation variation for automatic evaluation of MT quality », *in* Proceedings of the International Language Resources and Evaluation Conference (LREC), vol. III, p. 833–836, Lisbonne, 2004.

Lalit Bahl, JK. Baker, Frederick Jelinek, et RL. Mercer, « Perplexity - a measure of the difficulty of speech recognition tasks », *in* Program of the 94th Meeting of the Acoustical Society of America, vol. Suppl. no. 1, p. 62–63, 1977.

James Baker, « Stochastic modeling for automatic speech understanding », *Speech Recognition*, p. 521–542.

Michele Banko et Eric Brill, « Scaling to Very Very Large Corpora for Natural Language Disambiguation », p. 26–33, 2001.

Yehoshua Bar-Hillel, « The present state of research on mechanical translation », *American Documentation*, vol. 2(4), p. 229–237, 1953.

Yehoshua Bar-Hillel, « The present status of automatic translation of languages », *Advances in Computers*, vol. 1, p. 91–141, 1960.

Regina Barzilay et Lillian Lee, « Learning to paraphrase: an unsupervised approach using multiple-sequence alignment », *in* Proceedings of HLT-NAACL, p. 16–23, 2003.

Regina Barzilay et Kathleen R. McKeown, « Extracting paraphrases from a parallel corpus », *in* Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, p. 50–57, 2001.

Douglas Biber, *Variation across speech and writing*, Cambridge University Press, 1988.

Douglas Biber, *Dimensions in Register Variation*, Cambridge University Press, 1995.

Hervé Blanchon, *Comment définir, mesurer, et améliorer la qualité, l'utilisabilité et l'utilité des systèmes de TAO de l'écrit et de l'oral*, Thèse d'habilitation à diriger des recherches, Université Joseph Fourier, 2004.

Christian Boitet et René Gerber, « Expert systems and other new techniques in MT systems », *in* Proceedings of the 22nd annual meeting on Association for Computational Linguistics, p. 468–471, , Association for Computational Linguistics, 1984.

Eric Brill et Radu Soricut, « A Unified Framework for Automatic Evaluation using N-gram co-occurence Statistics », *in* Proceedings of ACL 2004, p. 613–620, Barcelone, 2004.

Chris Brockett, Takako Aikawa, Anthony Aue, Arul Menezes, Chris Quirk et Hisami Suzuki, *English-Japanese Example-Based Machine Translation Using Abstract Linguistic Representations*, Rapport technique, Microsoft Research, 2002.

Peter F. Brown, John Cocke, Stephen Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer et Paul S. Roossin, « A Statistical Approach to Machine Translation », *Computational Linguistics*, vol. 16, p. 79–85, 1990.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jennifer C. Lai et Robert L. Mercer, « Class-Based n-gram Models of Natural Language », *Computational Linguistics*, vol. 18, n° 4, p. 467–479, 1992.

Etienne Brunet, *Où l'on mesure la distance entre les distances*, 2004. `http://www.deleuze.fr.st/TXT/140174.html`, page consultée le 10 novembre 2005, Revue Texto ! Rubrique Dits et inédits. mars 2004.

Michael Burrows et David J. Wheeler, *A block-sorting lossless data compression algorithm*, Rapport technique, Digital SRC technical report 124, 1994, Palo Alto, Californie, 1994.

Gabriela Cavaglià, « Measuring corpus homogeneity using a range of measures for inter-document distance », *in* Proceedings of the International Language Resources and Evaluation Conference (LREC), p. 426–431, 2002.

William Cavnar et John Trenkle, « N-Gram-Based Text Categorization », *in* Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, p. 161–175, Las Vegas, États-Unis, 1994.

John Chandioux, « METEO: un systeme opérationnel pour la traduction automatique des bulletins météorologiques », *Meta*, vol. 21, p. 127–133, 1976.

Eugene Charniak, *Statistical Language Learning*, MIT Press, 1993.

Stanley Chen, Donald Beeferman et Ronald Rosenfeld, « Evaluation Metrics For Language Models », *in* Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop, 1998.

Stanley Chen et Joshua Goodman, « An empirical study of smoothing techniques for language modeling », *Computer Speech and Language*, vol. 13, p. 359–394, 1999.

Noam Chomsky, *Syntactic Structures*, La Haye, Mouton, 1957.

John Cleary et WJ. Teahan, « Unbounded Length Contexts for PPM », *The Computer Journal*, vol. 40, 1997.

J. Cohen, « A coefficient of agreement for nominal scales », *Educational and Psychological Measurement*, vol. 20, p. 37–46, 1960.

Ferdinand de Saussure, *Cours de linguistique générale, édition 1916*, Paris, Charles Bally et Albert Séchehaye, Payot, 1995.

Etienne Denoual, « The influence of data homogeneity on NLP system performance », *in* Companion Volume to the Proceedings of the Second International Joint Conference on Natural Language Processing (IJCNLP-05), p. 228–233, Jeju, Corée, 2005.

Etienne Denoual, « A method to quantify corpus similarity and its application to quantifying the degree of literality in a document », *International Journal of Technology and Human Interaction*, vol. 2, n° 1, p. 51–66, 2006.

Etienne Denoual et Yves Lepage, « BLEU in characters: towards automatic MT evaluation in languages without word delimiters », *in* Companion Volume to the Proceedings of the Second International Joint Conference on Natural Language Processing, p. 81–86, Jeju, Corée, 2005.

George Doddington, « Automatic evaluation of machine translation quality using N-gram co-occurrence statistics », *in* Proceedings of the Human Language Technology Conference (HLT-02), p. 138–145, 2002.

Bill Dolan, Chris Quirk et Chris Brockett, « Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources », *in* Proceedings of the International Conference on Computational Linguistics (COLING), p. 350–356, Genève, Suisse, 2004.

Mark Dras, *Tree adjoining grammar and the reluctant paraphrasing of text*, Thèse de doctorat, Université Macquarie, 1999.

Huiming Duan, Xiaojing Bai, Baobao Chang et Shiwen Yu, « Chinese Word Segmentation at Peking University », *in* Q. Ma & F. Xia (sous la direction de), Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, p. 152–155, 2003.

Ted Dunning, « Accurate Methods for the Statistics of Surprise and Coincidence », *Computational Linguistics*, vol. 19, n° 1, p. 61–74, 1993.

Ted Dunning, *Statistical identification of language*, Rapport technique, New Mexico State University, 1994.

Mathias Eck et Chiori Hori, « Overview of the IWSLT 2005 Evaluation Campaign », *in* Proceedings of the International Workshop on Spoken Language Translation, p. 11–32, Pittsburgh, Pennsylvanie, 2005.

Brad Efron et Rob J. Tibshirani, *An introduction to the Bootstrap*, Londres/New York, Chapman & Hall, 1993.

Atsushi Fujita, *Automatic generation of syntactically well-formed and semantically appropriate paraphrases*, Thèse de doctorat, Institut de science et de technologie de Nara, 2005.

Gerald Gazdar et Geoffrey Pullum, *Generalized Phrase Structure Grammar: A Theoretical Synopsis*, Bloomington, Indi- ana, Indiana University Lingnistics Club, 1982.

Ulrich Germann, Mike Jahr, Kevin Knight, Daniel Marcu et Kenji Yamada, « Fast Decoding and Optimal Decoding for Machine Translation », *Artificial intelligence*, vol. 154 (1-2), p. 127–143, 2003.

IJ. Good, « The Population Frequencies of Species and the Estimation of Population Parameters », *Biometrika*, vol. 40, p. 237–264, 1953.

Joshua Goodman, *A bit of progress in language modeling*, Rapport technique MSR-TR-2001-72, Microsoft Research, 2001.

Nizar Habash, « Generation-Heavy Hybrid Machine Translation », *in* Proceedings of the International Natural Language Generation Conference (INLG'02), p. 185–191, New York, 2002.

Catalina Hallett et Donia Scott, « Structural variation in generated health reports », *in* Proceedings of the Second International Joint Conference on Natural Language Processing (IJCNLP-05) International Workshop on Paraphrasing (IWP 2005), p. 33–40, Jeju, Corée, 2005.

David Hays, « Automatic language-data processing », *Computer applications in the behavioral sciences*, p. 395–421, 1962.

Douglas Hofstadter et the Fluid Analogies Research Group, *Fluid Concepts and Creative Analogies*, New York, Basic Books, 1994.

XD. Huang, F. Alleva, MY. Hwang, KF. Lee et Ronald Rosenfeld, « The Sphinx-II speech recognition system: An overview », *Computer Speech and Language*, 1993.

Kenji Imamura, « Hierarchical Phrase Alignment Harmonized with Parsing », *in* Proceedings of the 6th Natural Language Processing Pacific Rim Symbopium (NLPRS-2001), p. 377–384, 2001.

Esa Itkonen et Jussi Haukioja, « Grammaticalization: Abduction, Analogy, and Rational Explanation », *in* Shapiro & Haley, *The Pierce seminar papers*, 1999, p. 159–175.

JEIDA, *Methodology and criteria on machine translation evaluation*, Rapport, Japan Electronic Industry Development Association, 1992.

Frederik JELINEK, « Continuous speech recognition by statistical methods », *in* Proceedings of the IEEE, vol. 64, p. 532–556, 1976.

Frederick JELINEK, *Statistical Methods for Speech Recognition*, Cambridge, Massachussets, The MIT Press, 1997.

Frederick JELINEK et Robert MERCER, « Interpolated estimation of Markov source parameters from sparse data », *Pattern Recognition in Practice*, p. 381–397, 1980.

Aravind JOSHI, Leon LEVY et Masako TAKAHASHI, « Tree adjunct grammar », *Journal of Computer and System Science*, vol. 21(2), p. 136–163, 1975.

A. KAPLAN, « An experimental study of ambiguity in context », *Mechanical Translation*, vol. 2-2, p. 39–46, 1950.

Ronald M. KAPLAN et Joan BRESNAN, « Lexical-functional grammar: a formal system for grammatical representation », *in* The mental representation of grammatical relations, MIT press series on cognitive theory and mental representation, p. 173–281, Cambridge, Massachusetts.

Andreas KATHOL, Kristin PRECODA, Dimitra VERGYRI, Wen WANG et Susanne RIEHEMANN, « Speech Translation for Low-Resource Languages: The Case of Pashto », *in* Proceedings of European Conference on Speech Communication and Technology, p. 2273–2276, Lisbonne, Portugal, 2005.

Slava KATZ, « Estimation of probabilities from sparse data for the language model component of a speech recognizer », *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-35, p. 400–401, 1987.

Martin KAY, « Functional grammar », *in* Proceedings of the Fifth Meeting of the Berkeley Linguistics Society, p. 142–158, Berkeley, Californie, 1979.

Adam KILGARRIFF, « Using word frequency lists to measure corpus homogeneity and similarity between corpora », *in* Proceedings of the fifth workshop on very large corpora, p. 231–245, Hong Kong, 1997.

Adam KILGARRIFF, « Comparing corpora », *International Journal of Corpus Linguistics*, vol. 6:1, p. 1–37, 2001.

Adam KILGARRIFF et Tony ROSE, « Measures for corpus similarity and homogeneity », *in* 3rd conference on Empirical Methods in Natural Language Processing (EMNLP), p. 46–52, Granade, Espagne, 1998.

Margaret KING, « Evaluating natural language processing systems », *in* Communication of the ACM, vol. 29(1), p. 73–79, 1996.

Margaret KING, Andrei POPESCU-BELIS et Eduard HOVY, « FEMTI: Creating and Using a Framework for MT Evaluation », *in* Proceedings of the Machine Translation Summit IX, p. 224–231, Nouvelle Orléans, 2003.

Kenji Kita, T. Kawabata et H. Saito, « HMM Continuous Speech Recognition using Predictive LR Parsing », *in* Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, p. 703–706, 1989.

Richard Kittredge et John Lehrberger, « Sublanguage. Studies of language in restricted semantic domains », Berlin.

Dennis H. Klatt, « Review of the ARPA speech understanding project », *Journal of the acoustical society of America*, vol. 62, p. 1345–1366, 1977.

Reinhard Kneyser et Hermann Ney, « Improved clustering techniques for class-based statistical language modelling », *in* Proceedings of the European Conference on Speech Communication and Technology, p. 973–976, 1993.

Reinhard Kneyser et Hermann Ney, « Improved backing-off for m-gram language modeling », *in* Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 1, p. 181–184, 1995.

R. Kuhn et R. De Mori, « A Cache-Based Natural Language Model for Speech Recognition », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, p. 570–583, 1990.

R. Kuhn et R. De Mori, « *Correction* to a Cache-Based Natural Language Model for Speech Recognition », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, p. 691–692, 1992.

S.M. Lamb, « On the mechanization of syntactic analysis », *in* Proceedings of the International Conference on Machine Translation and Applied Language Analysis, Teddington 1961, p. 673–686, Londres, 1962.

Terence D. Langendoen et H. Michael Barnet, *PLNLP: A Linguist's Introduction*, Rapport technique, IBM, 1986.

Irene Langkilde et Kevin Knight, « Generation that exploits corpus-based statistical knowledge », *in* Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL), p. 704–710, Montréal, Québec, Canada, 1998.

Yves Lepage, « Solving Analogies on Words: an Algorithm », *in* Proceedings of COLING-ACL'98, vol. I, p. 728–735, Montréal, Québec, Canada, 1998.

Yves Lepage, *De l'analogie rendant compte de la commutation en linguistique*, Mémoire d'habilitation à diriger les recherches, Université de Grenoble, 2003.

Yves Lepage, « Translation of Sentences by Analogy Principle », *Archives of Control Sciences*, , n° 4, p. 583–592, 2005.

Yves Lepage et Etienne Denoual, « Automatic generation of paraphrases to be used as translation references in objective evaluation measures of machine translation », *in* Proceedings of the Second International Joint Conference on Natural Language Processing (IJCNLP-05) International Workshop on Paraphrasing (IWP 2005), p. 57–64, Jeju, Corée, 2005.

Yves Lepage et Etienne Denoual, « The 'purest' ever built EBMT system: no variable, no template, no training, examples, just examples, only examples », *in* Proceedings of the MT Summit X Workshop on Example-Based Machine Translation, p. 81–80, Phuket, Thailande, 2005.

Yves Lepage et Guilhem Peralta, « Using paradigm tables to generate new utterances similar to those existing in linguistic resources », *in* Proceedings of the International Language Resources and Evaluation Conference (LREC), vol. 1, p. 243–246, Lisbonne, 2004.

Victor R. Lesser et Lee D. Erman, « A retrospective view of the HEARSAY-II architecture », *Blackboard Systems*, p. 87–121, 1977.

Lori Levin, Donna Gates, Alon Lavie et Alex Waibel, « An Interlingua Based on Domain Actions for Machine Translation of Task Oriented Dialogues », *in* Proceedings of the International Conference on Spoken Language Processing (ICSLP), vol. 4, p. 1155–1158, Sydney, Australie, 1998.

Robert A. Liebscher, *New corpora, new tests, and new data for frequency-based corpus comparisons*, Newsletter 15:2, Center for Research in Language, 2003.

Chin-Yew Lin et Eduard Hovy, « Automatic Evaluation of Summaries Using N-gram co-occurence Statistics », *in* Proceedings of HLT-NAACL 2003, p. 150–157, Edmonton, 2003.

Bruce T. Lowerre, *The HARPY speech recognition system*, PhD thesis, 1976.

Christopher D. Manning et Hinrich Schütze, *Foundations of Statistical Natural Language Processing*, Cambridge, Massachusetts, The MIT Press, 1999.

André Martinet, *Éléments de linguistique générale*, Collection Cursus. Paris, Armand Colin, 1970.

Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano, Hiroshi Matsuda, Kazuma Takaoka et Masayuki Asahara, *Morphological Analysis System ChaSen version 2.2.9*, Manuel, Nara Institute of Science and Technology, 2002.

Dan Melamed, « Automatic Evaluation and Uniform Filter Cascades for Inducing N-Best Translation Lexicons », *in* Proceedings of the Third Workshop on Very Large Corpora (WVLC3), p. 184–198, Boston, États-Unis, 1995.

Georges Mounin, *Dictionnaire de la linguistique*, Paris, Quadrige / Presses Universitaires de France, 1974.

Atsushi Nakamura, Shoichi Matsunaga, Tohru Shimizu, Masahiro Tonomura et Yoshinori Sagisaka, « Japanese speech databases for robust speech recognition », *in* Proceedings of the ICSLP'96, vol. 4, p. 2199–2202, Philadelphie, États-Unis, 1996.

Hermann Ney, Ute Essen et Reinhard Kneser, « On structuring probabilistic dependencies in stochastic language modeling », *Computer Speech and Language*, vol. 8, p. 1–28, 1994.

Hermann Ney, Franz Josef Och et Stephan Vogel, « Statistical Translation of Spoken Dialogues in the Verbmobil System », p. 69–74, 2000.

Franz Josef Och, « Minimum Error Rate Training in Statistical Machine Translation », *in* Proceedings of the ACL 2003, p. 160–167, 2003.

Franz Josef Och, « Statistical Machine Translation: Foundations and Recent Advances », *in* Tutorial Notes of the Tenth Machine Translation Summit.

Kiyonori Ohtake et Kazuhide Yamamoto, « Applicability Analysis of Corpus-derived Paraphrases toward Example-based Paraphrasing », *in* Language, Information and Computation, Proceedings of 17th Pacific Asia Conference, p. 380–391, 2003.

D. Pallett, W. Fisher, J. Fiscus et J. Garofolo, « DARPA ATIS test results », *in* Proceedings of the Third DARPA Speech and Natural Language Workshop, 1990.

Z. Pankowicz, *Commentary on ALPAC report*, Memorandum, RADC, Griffiss Air Force Base, 1966.

Kishore Papineni, Salim Roukos, Todd Ward et Wei-Jing Zhu, *BLEU: a method for automatic evaluation of machine translation*, Rapport technique RC22176 (W0109-022), IBM, 2001.

Kishore Papineni, Salim Roukos, Todd Ward et Wei-Jing Zhu, « BLEU: a Method for Automatic Evaluation of Machine Translation », *in* Proceedings of the ACL 2002, p. 311–318, 2002.

Fernando Pereira et David Warren, « Definite clause grammars for language analysis », *Artificial Intelligence*, vol. 13, p. 231–278, 1980.

Emmanuel Planas, *TELA : Structure et algorithmes pour la traduction fondée sur la mémoire*, Thèse de doctorat, Laboratoire CLIPS, 1998.

Richard Power et Donia Scott, « Automatic generation of large-scale paraphrases », *in* Proceedings of the Second International Joint Conference on Natural Language Processing (IJCNLP-05) International Workshop on Paraphrasing (IWP 2005), p. 73–79, Jeju, Corée, 2005.

Mark Przybocki, *The 2005 NIST Machine Translation Evaluation Plan (MT-05)*, 2004.
`http://www.itl.nist.gov/iaui/894.01/tests/mt/doc-/mt05_evalplan.v1.pdf`.

Lawrence R. Rabiner, « A tutorial on hidden Markov models and selected applications in speech recognition », *in* Proceedings of the IEEE, vol. 77(2), p. 257–286, 1989.

Martin Rajman et Tony Hartley, « Automatically predicting MT systems rankings compatible with Fluency, Adequacy or Informativeness scores », *in* Proceedings of the Workshop on Machine Translation Evaluation : *Who Did What To Whom*, p. 29–34, Santiago de Compostela, Espagne, 2001.

Ronald Rosenfeld, *Adaptive Statistical Language Modeling: A Maximum Entropy Approach*, Thèse de doctorat, Université de Carnegie Mellon, Pittsburgh, 1994.

Ronald Rosenfeld, « Two decades of Statistical Language Modeling: Where Do We Go From Here? », *in* Proceedings of the IEEE, vol. 88, p. 1270–1278, 2000.

Gérard Sabah, *L'intelligence artificielle et le langage*, Paris, Hermès, 1989.

San-Antonio, *Bérurier au sérail*, Paris, Fleuve noir, 1964.

Satoshi Sato, *Example-Based Machine Translation*, Thèse de doctorat, Université de Kyoto, 1991.

M. Shapiro & M. Haley (sous la direction de), *The Pierce Seminar Papers*, vol. IV, New York, Berghahn, 1999.

Stuart Shieber, « Criteria for Designing Computer Facilities for Linguistic Analysis », *Linguistics*, vol. 23, p. 189–211, 1985.

R. Srihari et C. Baltus, « Combining statistical and syntactic methods in recognizing handwritten sentences », *in* Proceedings of the AAAI Symposium : Probabilistic Approaches to Natural Language, p. 121–127, Las Vegas, États-Unis, 1992.

Graham A. Stephen, *String searching algorithms*, Singapore New Jersey London Hong Kong, World scientific publishing, 1994.

Eiichiro Sumita et Hitoshi Iida, « Experiments and prospects of Example-Based Machine Translation », *in* Proceedings of the 29th Conference on Association for Computational Linguistics, p. 185–192, , Association for Computational Linguistics, 1991.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto et Seiichi Yamamoto, « Towards a broad-coverage bilingual corpus for speech translation of travel conversations in the real world », *in* Proceedings of the International Language Resources and Evaluation Conference (LREC), p. 147–152, Las Palmas, Iles Canaries, Espagne, 2002.

Lucien Tesnière, *Eléments de syntaxe structurale*, Paris, Klincksieck, 1959.

James W. Thatcher et Jesse B. Wright, « Generalized Finite Automata Theory with an Application to a Decision Problem of Second-Order Logic », *Mathematical Systems Theory*, vol. 2(1), p. 57–81, 1968.

Nicholas Haddock Tony Rose et Roger Tucker, « The effects of corpus size and homogeneity on language model quality », *in* Proceedings of the ACL SIGDAT workshop on very large corpora, p. 178–191, Beijing and Hong Kong, 1997.

Esko Ukkonen, « Algorithms for Approximate String Matching », *Information and Control*, vol. 64, p. 100–118, 1985.

Bernard Vauquois, *La traduction automatique à Grenoble*, Paris, Dunod, 1975.

Robert A. Wagner et Michael J. Fischer, « The String-to-String Correction Problem », *Journal for the Association of Computing Machinery*, vol. 21, n° 1, p. 168–173, 1974.

J.S. White, Th. O'Connell et T. O'Mara, « ARPA MT evaluation methodologies : evolution, lessons and further approaches », *in* Proceedings of the Technology partnerships for crossing the lanuage barrier (the first conference of the association for machine translation in the Americas), , Association for Computational Linguistics, 1994.

Ian Witten et Timothy Bell, « The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression », *IEEE Transactions Information Theory*, vol. 34, p. 1085–1094, 1991.

William A. Woods, « Transition network grammars for natural language analysis », *Communications of the Association for Computing machinery (ACM)*, vol. 13, p. 561–602, 1970.

Dekai Wu, « MT model space : statistical vs. compositional vs. example-based machine translation (EBMT-II panel on future directions of EBMT) », *in* MT Summit X Workshop on Example-Based Machine Translation, , Technical Report HKUST-C50S-20, Department of Computer Science, HKUST, Hong Kong, 2005.

Kazuhide Yamamoto, « Interaction between paraphraser and transfer for spoken language translation », *Journal of Natural Language Processing*, vol. 11, n° 5, p. 63–86, 2004.

Victor Yngve, « Syntax and the problem of multiple meaning », *Machine Translation of Languages*, p. 208–266, 1955.

Ying Zhang, Stefan Vogel et Alex Waibel, « Interpreting BLEU/NIST scores: how much improvement do we need to have a better system? », *in* Proceedings of the International Language Resources and Evaluation Conference (LREC), vol. V, p. 2051–2054, Lisbonne, 2004.

Yujie Zhang et Kazuhide Yamamoto, « Paraphrasing of Chinese Utterances », *in* Proceedings of the International Conference on Computational Linguistics (Coling), p. 1163–1169, Taipei, Taiwan, 2002.