Prétraitement de données et création d'un segmenteur de l'arabe pour un système de traduction probabiliste vers le français

Sahnoun Ben Taamallah

Prétraitement de données et création d'un segmenteur de l'arabe pour un système de traduction probabiliste vers le français

Nom : Ben Taamallah
Prénom : Sahnoun

UFR des Sciences du langage

Mémoire de master 2 recherche - 30 **crédits** – **Mention** sciences du langage

Spécialité: Industries de la langue

Parcours: TALEP

Sous la direction de Laurent BESACIER, Hervé Blanchon et Olivier Kraif

Année universitaire 2011-2012

Approches de prétraitement de données et de création d'un segmenteur de l'arabe pour un système de traduction probabiliste vers le français

Nom : Ben Taamallah
Prénom : Sahnoun

UFR des Sciences du langage

Mémoire de master 2 recherche - 30 **crédits** – **Mention** sciences du langage

Spécialité: Industries de la langue

Parcours: TALEP

Sous la direction de Laurent BESACIER, Hervé Blanchon et Olivier Kraif

Année universitaire 2011-2012

Remerciements

C’est une habitude saine que de remercier au début d’un tel travail tous ceux qui, plus ou moins directement, ont contribué à le réaliser. C’est avec mon enthousiasme le plus vif et le plus sincère que je voudrais rendre mérite à tous ceux qui à leur manière m’ont aidé à mener à bien ce travail.

Je tiens d’abord à témoigner de ma plus profonde gratitude à mes encadreurs de recherche, monsieur Laurent Besacier, monsieur Olivier Kraif et monsieur Hérvé Blanchon. Ils ont su, par leur extrême dévouement, leur disponibilité et leur gentillesse débordante rendre mon travail fort agréable, voire même amusant. De conseils judicieux en mots d’encouragements, ils étaient toujours d’une aide précieuse et je leurs en suis très reconnaissant.

Je remercie tous les membres de GETALP qui m’ont aidé surtout Marwen Azouzi à m’intégrer dans l’équipe et à me donner ses précieux conseils. Ainsi que tous les autres membres, j'ai passé des bons moments avec eux.

Je dois et j’adresse un remerciement tout particulier à mes amis qui me soutiennent toujours et particulièrement : Wael, Moez, Wassim, Marwen, Selma et Meriam pour leur aide dans ce mémoire (je ne peux pas nommer tous mes amis!).

Finalement, une attention toute particulière est dirigée vers ma famille, et en particulier mon père et ma mère qui n’ont pas négligé les sacrifices tout au long des mes études. Merci infiniment d’avoir toujours été si attentionnés et dévoués.

Sohnoun

DECLARATION

1. Ce travail est le fruit d'un travail personnel et constitue un document original.
2. Je sais que prétendre être l'auteur d'un travail écrit par une autre personne est une pratique sévèrement sanctionnée par la loi.
3. Personne d'autre que moi n'a le droit de faire valoir ce travail, en totalité ou en partie, comme le sien.
4. Les propos repris mot à mot à d'autres auteurs figurent entre guillemets (citations).
5. Les écrits sur lesquels je m'appuie dans ce mémoire sont systématiquement référencés selon un système de renvoi bibliographique clair et précis.

NOM : Ben Taamallah PRENOM : Sahnoun

DATE : 10/09/2012

Sommaire

MOTS-CLÉS : Traduction Automatique Probabiliste, Traitement automatique de la langue arabe, Prétraitement et segmentation de données, Modèle de langage, Modèles de traduction, Maxent.

RÉSUMÉ

Le domaine du traitement automatique des langues naturelles a connu des évolutions très rapides ces dernières années, et spécialement dans la traduction automatique, c'est pourquoi les demandes en matière de traducteurs automatiques fiables augmentent sans cesse. De ce fait, nous nous sommes intéressés à ce domaine afin de concevoir un traducteur automatique de la langue arabe vers le français, basé sur un modèle probabiliste.

Les performances de traduction des systèmes probabilistes dépendent considérablement de la qualité et de la quantité des données d'apprentissage disponibles. Néanmoins la langue arabe compte encore parmi les langues dites « peu dotées », c'est pourquoi la plupart des travaux sur cette langue sont basés sur les données libre d'accès qui proviennent d'organisations internationales (ONU, etc.).

Nous présentons dans ce travail, une approche d'optimisation des performances d'un système de traduction de l'arabe. Compte tenu du manque de données et d'outils accessibles, nous avons cherché à moindre coût la meilleure combinaison de prétraitements à appliquer sur nos données en arabe pour améliorer la traduction vers le français.

KEYWORDS : Statistical Machine Translation, Preprocessing and Tokenization of data, Language Models, Translation Models, Maxent.

ABSTRACT

In recent years, Natural Language Processing has rapidly evolved, especially in the domain of Statistical Machine Translation causing the need for reliable automatic translations to skyrocket with no sign of slowing. Due to this increased need, we have taken a special interest in this domain with the goal of creating a translation machine capable of translating Arabic into French, based on statistical models.

The performance of Statistical Machine Translation relies heavily on the quality and the quantity of available training data. However, the Arabic language remains one of the languages with the fewest available resources which is why most of the available works in this language are based on open access data from international organizations, such as the U.N.

In this work, we will present our approach to optimizing the performance quality of our Arabic translator. Taking into account the lack of data and available resources, we were able to find a low-cost solution to search for the best pre-processing combinations to apply to our Arabic database in order to obtain the highest quality French translation.

Introduction

Ce mémoire de recherche est l'aboutissement d'un stage, et s'inscrit dans le cadre de la formation de deuxième année du Master sciences du langage, spécialité industrie de la langue (IDL), de l'Université Stendhal à Grenoble. L'organisme d'accueil est le GETALP (Groupe d'Étude en Traduction Automatique/Traitement Automatisé des Langues et de la Parole) qui est une équipe du laboratoire d'informatique de Grenoble (LIG). Le but des travaux du GETALP est de contribuer de façon significative à l'émergence d'une informatique ubilingue « l'intégration de la linguistique dans les processus informatique », dans le contexte du développement de l'informatique ubiquitaire « l'informatique qui s'intéresse au recueil des informations à partir des environnements perceptifs et communicants». Cet objectif nécessite de mener à bien des recherches à caractère souvent pluridisciplinaire, en informatique, en linguistique et psycholinguistique, en sémantique (lien avec les ontologies), en pragmatique (pour le dialogue), et en traitement de l'oral.

Le stage a été co-encadré par M. Laurent Besacier, professeur à l'Université Joseph Fourier, M. Hervé Blanchon, maître de conférences à l'Université Pierre Mendès-France. M. Olivier Kraif, maître de conférences à l'Université Stendhal été mon encadrant du mémoire.

Au niveau de la qualité de traduction, la sortie brute d'un système de traduction automatique probabiliste n'est en général pas suffisamment satisfaisante pour que les lecteurs, qui ne connaissent pas la langue source, comprennent le sens. Récemment, pour certaines langues les traductions sont devenus assez fiables, car des corpus pour ces langues sont généralement disponibles en grande quantité, et car elles possèdent une morphologie flexionnelle faible ou nul, comme l'anglais et le chinois. Mais ce n'est pas le cas avec la langue arabe, qui est d'une morphologie très complexe et encore peu dotée en ressources génériques pour le TAL.

Il se trouve que les performances de traduction d'un système probabiliste dépendent essentiellement de la qualité et de la quantité des données d'apprentissage disponibles. Dans ce travail nous n'allons pas aborder les solutions pour améliorer la quantité de nos données, mais nous allons nous concentrer sur leur qualité.

Comme notre objectif général est d'améliorer les performances d'un système de traduction de l'arabe vers le français, et comme l'arabe représente une langue morphologiquement

très riche, nous allons diriger nos recherches vers le perfectionnement des prétraitements morphologiques du côté arabe de nos données. Nous parlons de « qualité » de données d'apprentissage, car nous entendons de déterminer la forme la plus adapté de ces données pour un système de traduction vers le français.

Toutes les langues morphologiquement riches présentent un défi pour la traduction automatique probabiliste, et c'est le cas avec la langue arabe, car cette richesse produit un très grand nombre de formes de surface, et par conséquent une augmentation de la taille du vocabulaire.

Cette augmentation peut gravement nuire aux performances d'un traducteur probabiliste lorsqu'elle coïncide avec une disponibilité faible des corpus d'apprentissage.

Dans ce travail, nous relevons ce défi posé par la langue arabe en cherchant la meilleure combinaison de prétraitements à effectuer sur les données arabes avant la création d'un système de traduction probabiliste vers le français.

Une deuxième partie importante de notre recherche dans le cadre de ce stage, consiste à développer et à mettre en œuvre une approche d'apprentissage et de développement, afin d'obtenir une application à notre propriété qui effectuera la combinaison choisie.

Le premier chapitre de ce mémoire donne une présentation du contexte du stage et débouche sur une introduction des notions de base et des concepts indispensables à la compréhension de notre problématique. Dans une deuxième partie, nous présentons nos premières expérimentations effectuées à la recherche de la meilleure séquence de fonction de nos hypothèses de travail, nous avons procédé, dans la troisième partie, à l'évaluation et à la validation de ces choix grâce à des expérimentations plus avancées impliquant plus de données.

Nous appliquons, dans la dernière partie, une approche d'apprentissage supervisé afin de créer un outil de prétraitement de l'arabe, qui mettra en œuvre un tokéniseur spécifique à la traduction vers le français, et permettra de mettre en œuvre concrètement tous les résultats de notre recherche.

I. Partie 1

–

Présentation du stage et état de l’art

1. Chapitre 1 – Présentation de l'entreprise

Mes travaux ont été réalisés au LIG sur un projet avec le pôle Cassidian de la société EADS. Je détail, dans les sections suivantes, quelques informations sur le contexte de mon stage.

1.1 EADS

EADS pour *European Aeronautic Defence and Space Company*, est une société internationale qui mène ses activités dans le développement, la production, la commercialisation et la vente de constructions aéronautiques, civiles et militaires.

Cette société comporte **Airbus**, le premier constructeur d'avions commerciaux, **Eurocopter** comme premier fournisseur mondial de grands hélicoptères, et d'autres grands constructeurs de ce domaine.

Dans l'ensemble, EADS développe et commercialise des avions civils et militaires, ainsi que des systèmes de communications, des missiles, des fusées et des satellites.

EADS s'est formée le 10 juillet 2000, après la fusion de trois sociétés :

- DASA pour *DaimlerChrysler Aerospace AG*, une entreprise allemande.
- *Aerospatiale Matra*, entreprise française.
- CASA pour *Construcciones Aeronauticas SA*, entreprise espagnole.

EADS comme société est de droit néerlandais, elle est cotée aux bourses de Francfort, Madrid et Paris. Cette société emploie actuellement plus de 121 000 personnes. Son chiffre d'affaire en 2011 dépasse 49,1 milliards d'euro, et il est reparti sur le monde entier : 50 % en Europe, 14 % en Amérique et 36% dans le reste du monde.

Le groupe EADS est constitué de quatre divisions indépendantes, correspondant à ses activités spécifiques :

- Airbus : Aviation commerciale et militaire.
- Astrium : principal fournisseur européen de satellites.
- Eurocopter : premier constructeur mondial d'hélicoptères.

- Cassidian : le pôle d'activités de défense et de sécurité chez la société EADS.

1.2 La division Cassidian

Cassidian est le pôle d'activités de défense et de sécurité du territoire d'EADS. C'est une société par actions simplifiée, au capital social de 61 388 009 €, et ayant pour objet l'étude, le développement, la réalisation et la commercialisation de systèmes de sécurité de l'information. CASSIDIAN conçoit et met en œuvre des systèmes de renseignement, de communication, de sécurité, et des systèmes de commandement et de contrôle.

Cassidian comporte Cinq sous-composantes :

- *Defence and Communications Systems*, architecte et intégrateur de systèmes ;
- *Defence Electronics*, specialiste des capteurs, avionique et guerre électronique ;
- *Military Air Systems*, fabricant d'aéronefs de combat ;
- *MBDA*, leader mondial en conception et production de missiles ;
- *Eurofighter GMBH*, constructeur d'avions de combat.

C'est l'unité *Defence and Communications Systems* qui est chargé de la conception des systèmes innovants et de leur interopérabilité au sein de Cassidian.

Cette unité est également décomposée en plusieurs entités, où on trouve le SCDE pour *Studies Concept Development & Experimentation*, le laboratoire fournissant les prototypes expérimentaux, au sein duquel se trouve le département IPCC dans lequel les recherches se concentrent autour le traitement de l'information. C'est avec ce département que j'ai été en contact pour effectuer ce stage.

1.2.1 Le département IPCC : Information Processing, Control & Cognition

Le département IPCC est une entité R&D comportant plus de 20 personnes. Les recherches dans d'IPCC sont généralement en lien avec la fouille et la fusion de documents multimédias et de données non structurées.

Le département a participé au développement de plusieurs projets français et européens, en collaboration avec d'autres acteurs industriels et universitaires, parmi lesquels on trouve :

GREYC[1] (laboratoire de Caen), LIP6 (laboratoire de Paris VI), Sinequa, Systran, Xerox, Mondeca, Synapse, Exalead Thales, etc.

[1] Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen

2 Chapitre 2 - Présentation du stage

2.1 Présentation de l'équipe

Notre stage a été réalisé au sein du groupe GETALP (Groupe d'Étude en Traduction Automatique/Traitement Automatisé des Langues et de la Parole) qui est une équipe du laboratoire d'informatique de Grenoble (LIG).
Les finalités des recherches du GETALP sont, d'un point de vue global, de contribuer de façon significative à l'émergence d'une informatique ubilingue « l'intégration de la linguistique dans les processus informatique », tout en effectuant des recherches pluridisciplinaires, en informatique, en linguistique, en psycholinguistique, en sémantique (lien avec les ontologies), en pragmatique (pour le dialogue), et en traitement de l'oral.
Les axes de recherche principaux de notre équipe sont :

- Traduction Automatique (TA) et assistée par ordinateur (TAO)
- Traitement Automatique des Langues (TALN) et plates-formes associées
- Collecte et construction de ressources linguistiques
- Multilinguisme dans les systèmes d'information
- Reconnaissance automatique de la parole, des locuteurs, des sons et des dialectes
- Analyse sonore et interaction dans les environnements perceptifs

Notre stage s'insère dans l'axe de traduction automatique, axe dans le quel notre équipe possède une tradition de recherche qui remonte à 2007, pendant cette période notre équipe a participé, régulièrement, à diverses campagnes d'évaluation dans ce domaine comme la campagne IWSLT.

2.2 Contexte et besoins de l'entreprise

TRAD 2012 est une campagne d'évaluation dont l'objectif est de rendre compte des meilleures performances actuelles des systèmes de traduction automatique pour le couple de langue arabe-français. Cette campagne est financée par la Direction Générale de l'Armement (DGA), et organisée par l'entreprise CASSIDIAN et le Laboratoire national de métrologie et d'essais (LNE).

Lors de cette campagne, la DGA a notifié à CASSIDIAN la mise en œuvre du Programme Etude Amont, appelé TRAD, comportant la réalisation d'une étude dans le domaine de la « Traduction pour l'aide à l'Analyse Documentaire » en vue d'accroître les performances et les capacités de généralisation des systèmes de traduction automatique actuels.

Comme nous avons pu le constater lors de la présentation de Cassidian, le département IPCC concentre ses activités sur le traitement automatique des documents, mais il a des besoins en matière d'assistance technique, d'expérience et de réalisation d'études pour un projet en traduction automatique.

C'est dans ce cadre là que la société CASSIDIAN a pris contact avec le LIG, afin de réaliser ce projet en collaboration avec l'équipe GETALP.

Dans ce cadre, ma mission consiste à réaliser ce système et orienter le développement à mes finalités de recherche.

3 Chapitre 3 - Etat de l'art

3.1 La traduction automatique

3.1.1 Introduction

La traduction automatique (TA), appelée en anglais « *Machine translation* », fait référence à la traduction réalisée par une machine, et sans aucune intervention humaine. Effectuer cette tâche entre deux langues naturelles en utilisant les ordinateurs, a été un objectif de l'informatique depuis ses débuts.

3.1.2 Histoire de la traduction automatique

Pendant la guerre froide, dans la période 1958-1966, le besoin des américains de traduire la langue russe, a déclenché les premiers grands projets de développement de machines de traduction automatique.

Néanmoins, la première conception théorique d'un système de traduction automatique date de l'année 1933 où Georges Astrouni a proposé un système de traduction générique fonctionnant comme un dictionnaire mécanique.

En juin 1952, Andrew Booth et Warren Weaver ont présenté la première conférence en traduction automatique, où ils ont montré quelques tentatives d'utilisation des premiers ordinateurs pour l'automatisation de la traduction.

Le premier système de traduction fonctionnel, a été présenté le 7 janvier 1954, dans le cadre du « *IBM-Georgetown Experiment* » ; ce système utilisait un dictionnaire de 250 mots et 6 règles.

3.2 Traduction automatique statistique

En anglais « *Statistical Machine Translation* » (SMT), c'est une approche permettant de traduire automatiquement un texte d'une langue source vers une langue cible en n'utilisant que des méthodes statistiques. Cette approche a été conceptualisée en 1949 par Warren Weaver, mais elle été rapidement abandonné à l'époque, faute de machines puissantes en terme de mémoire et de capacité de calcul. Mais elle a été ré-introduite avec succès au début des années 1990 par [Brown et al. 1990], et a immédiatement intéressé de nombreux chercheurs.

Cette approche se base sur la théorie mathématique de distribution et d'estimation probabiliste de *Frederick Jelinek*, développée au *IBM T.J. Watson Research Center*, et elle bien détaillée dans [Brown et al, 1993] et [Carl, 2003].

L'idée de la traduction automatique probabiliste est de s'intéresser aux probabilités de correspondance entre les mots ou les séquences de mots, des langues source et cible. Et à partir des correspondances jugées les plus probables, on effectue la tâche de traduction.
La création des correspondance repose sur l'apprentissage d'un modèle de traduction *P(f/e)* qui représente la probabilité que la séquence cible *e* soit la traduction de la séquence source *f*, en utilisant un corpus bilingue, et par un modèle de langage *P(e)* qui représente la probabilité que la séquence *e* soit déjà observée en lange cible, à partir d'un corpus monolingue.

La qualité de la traduction dépendra du couple de langues, de la quantité de données utilisées, de la qualité de ces données et de leurs prétraitements avant l'entrainement.

En la traduction automatique statistique, aucune limitation linguistique n'est imposée par le système, ce qui représente un avantage important pour cette approche de traduction, car il augmente la robustesse du système - c'est-à-dire que le système ne tiendra plus compte des règles hors apprentissage comme le non-respect de certaines règles linguistiques, et il retournera une traduction même pour des énoncés incorrects sur le plan syntaxique.

Passons maintenant à l'approche sur laquelle la traduction automatique probabiliste est basée : soient deux textes, un texte *S* en langue source et l'autre texte *C* en langue cible. Nous faisons l'hypothèse que chaque phrase *e* dans l'ensemble *C* peut être une traduction d'une phrase *f* de l'ensemble *S*.
Pour chaque couple de phrases (e_i , f_j), nous affectons une probabilité $P(e_i \mid f_j)$. Cela signifie que le système de traduction traduit f_j en e_i avec une probabilité *P*.

L'enjeu de la traduction probabiliste est de trouver la phrase ê, qui maximise $P(e^I \mid f^J)$, étant donnée une phrase f^J. De manière plus formelle :

$$\hat{e} = argmax_e \ P(e\,I \mid f\,J)$$

Équation 1

D’après le théorème de Bayes qui dit:

$$P(e^I \mid f^J) = \frac{P(f^J|e^I) * p(e^I)}{P(f^J)}$$

Équation 2

Comme le dénominateur de la première équation est indépendant de e^I, la maximisation devient alors:

$$\hat{e} = \underset{e}{\operatorname{argmax}} P(e^I \mid f^J) = \underset{e}{\operatorname{argmax}} P(e^I) * P(f^J \mid e^I)$$

Équation 3

$P(e^I)$ est calculé selon le modèle de langue cible.
$P(f^J \mid e^I)$ est calculé le modèle de traduction.

Si on décompose la tâche de la traduction en trois sous-opérations, on aura :

1) Calculer les paramètres du modèle de langage (phase d'entrainement).
2) Calculer les paramètres du modèle de traduction (phase d'entrainement).
3) Appliquer l’algorithme de maximisation en un temps acceptable. (le décodeur)

Ici, on est dans le cas du modèle de canal bruité[2] introduit par (Shannon 1948). Grâce à ce modèle, on peut avoir une traduction plus efficace, car la qualité de la traduction subira un double contrôle, le premier effectué par le modèle de langue et le deuxième par le modèle de traduction, par conséquent les erreurs d’un des modèles pourront être compensées par l’autre.

3.2.1 Modèle de langage

Le modèle de langage est une composante essentielle du système de traduction. Cette composante s’occupe de la prise en compte des contraintes exigées par la syntaxe, la grammaire et le lexique de la langue cible, afin d’assurer le respect de ses normes et de ses usages.

[2] Terme venant du domaine de reconnaissance de la parole (Noisy Channel Model)

Son rôle est d'estimer la probabilité d'une phrase donnée. Cette estimation doit être cohérente avec la conformation de la phrase au modèle de langage : plus la phrase est conforme au modèle de langage, plus sa probabilité doit être élevée.

Comme nous avons vu précédemment, l'entrainement d'un modèle de langage s'appuie sur un corpus monolingue de la langue cible. En effet il en extrait une distribution P(e) sur les chaînes ei de la langue modélisée:

$$\sum_{i} P\ (e^{i}) = 1$$

Équation 4

Généralement on s'intéresse à estimer la probabilité d'une séquence de mots, pour cela on utilise les probabilités de « n-gramme », où un n-gramme est une séquence de *n* mots consécutifs. Par conséquent les phrases d'une probabilité élevée sont celles qui sont composés des n-grammes qui ont des probabilités élevés, et les phrases les moins probables sont celles qui comportent des n-grammes moins probables.

Le modèle est appelé unigramme si n=1.

Si n=2, le modèle est dit d'ordre 1, et il est appelé bigramme.

Un modèle trigramme est d'ordre 2, et il est calculé avec la formule suivante:

$$P(e)\ = \prod_{i=1}^{I+1} P(e_i|e_{i-1}, e_{i-2})$$

Équation 5

3.2.2 Modèle de traduction

Le modèle de traduction, est une composante principale de tout système de traduction probabiliste. C'est, en effet, la composante qui modélise le processus de génération d'une phrase source à partir une phrase cible.

Ce modèle est entrainé à partir d'un corpus bilingue aligné au niveau des phrases : à chaque phrase (ou un groupe de phrases) de côté source, lui correspond une phrase qui est sa traduction en langue cible.

Un modèle statistique de traduction mesure la probabilité $P(T|S)$ où $T= T_1T_2....T_j$ est la traduction de la phrase $S= S_1S_2....S_i$ où T_j et S_i sont les unités des phrases *T* et *S*.

Les unités T_j et S_i, peuvent êtres des mots ou des séquences de mots. La figure ci-dessous présente un alignement de séquences de mots[3] :

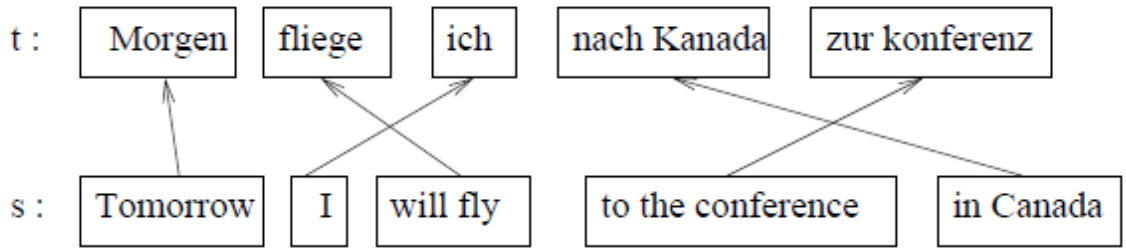

Figure 1– Exemple d'un alignement de séquences de mots

L'estimation de la probabilité *P(T|S)* est calculée par la somme des alignements possibles entre les différents éléments de *T* et de *S* :

$$P(S|T) = \sum_{a \in A} P(S, a|T)$$

Équation 6

Où A est l'ensemble des alignements possibles.

Cette approche a été appliquée par [Koehn et al.2003], où une table de traduction contient tous les alignements en séquence de mots ainsi que leurs probabilités.
Le calcul de cette table est obtenu par l'application de l'équation suivante :

$$P(T|S) = P(T) \times P(S,T) \times \Omega(S|T)$$

Équation 7

Où :

P(T) est le modèle de langage.
P(S,T) est le modèle de traduction.
Ω(S|T) est le modèle de distorsion.

[3] Exemple pris de rapport M2 « AFLI Haithem : Approche mixte pour la traduction automatique statistique. »

3.2.2.1 Le modèle de distorsion

Appelé aussi *modèle de réordonnancement*, ce modèle est intégré dans le modèle de traduction, il permet de réordonnancer les séquences de mots produites par la traduction. En effet, dans la phrase cible, les séquences de mots ne sont pas obligées d'avoir le même ordre que les séquences de la phrase source.

Lors de la construction de la phrase cible, l'hypothèse est alors de suivre un ordre plus ou moins « distordu ».

La distorsion est calculée par le nombre de déplacements de séquences qu'une phrase a subi, la modélisation de la distorsion, se fait par l'attribution des scores à chaque déplacement effectué.

Supposons que a_p indique que la position initiale de la séquence source, et elle été traduite en p-ième séquence cible. La distorsion est donnée par le calcul du modèle exponentiel :

$$d(a_p - b_{p-1}) = \alpha^{|a_p - b_{p-1}|}$$

Équation 8

Où α doit être choisi convenablement.

3.2.3 Décodage

Dans la traduction probabiliste, le décodage est la tache de transformation d'une phrase source en phrase cible. Ce terme est inspiré par le cryptographe Warren Weaver5 qui considérait une phrase en Russe comme une phrase en anglais chiffrée.

Le décodage peut être considéré comme la tâche la plus compliquée dans tout le processus de traduction, vu qu'il s'agit de sélectionner à partir d'un très grand nombre de possibilités de traduction, l'hypothèse qui assure le plus le meilleur transfert du sens tout en garantissant la correction en langue cible.

L'implémentation du décodage consiste à effectuer une recherche approfondie en parcourant toutes les chaînes *e** possibles en langue source. Elle utilise la fonction de densité fournie par le modèle pour générer la traduction la plus probable du texte source.

L'équation suivante donne la solution cette tâche:

$$\hat{e} = \underset{e}{\operatorname{argmax}}\, P(e^I|f^J)$$

Équation 9

Où f^J est le texte source et e^I est l'ensemble des textes de la langue cible.

3.3 Evaluation de la qualité des traductions

Plusieurs approches sont utilisées pour évaluer les performances d'un système de traduction automatique. Le nombre important de mesures automatiques est représentatif de la difficulté d'évaluer la qualité d'une traduction.
Déterminer la qualité d'une traduction est un problème difficile et ouvert.

3.3.1 L'évaluation humaine

L'évaluation humaine de la traduction automatique demande plusieurs participants, chacun évaluant le système en fonction de critères précis, comme la correction grammaticale et la fidélité au sens du texte.
Ce type d'évaluation donne la mesure la plus exacte des performances de système, mais elle sollicite plusieurs experts, ce qui rend la tâche coûteuse.
De plus, ce type d'évaluation pose des problèmes de non-reproductibilité et de variabilité inter-annotateur. C'est pourquoi plusieurs mesures automatiques et objectives ont été développées, dont l'objectif est d'être corrélées avec les scores que produirait une évaluation humaine, tout en étant beaucoup moins coûteuses.

3.3.2 Évaluation automatique

Les évaluations automatiques demandent une ou plusieurs traductions de référence des données sources, afin d'estimer les performances des systèmes de traduction en déterminant le degré de ressemblance entre la sortie de ces systèmes et cette référence.
La qualité de la traduction de référence est donc très importante. Les mesures présentées ci après sont parmi les plus utilisées dans la communauté de la traduction probabiliste.

3.3.2.1 Le score BLEU

BLEU pour Bilingual Evaluation Understudy, a été proposé par [Papineni et al. 2001]. L'idée principale est la comparaison de la sortie du traducteur avec une traduction

de référence. Les statistiques de cooccurrence et de *n*-grammes, basées sur les ensembles de *n*grammes pour les segments de traduction et de référence, sont calculées pour chacun de ces segments et sommées sur tous les segments.

Cette moyenne est multipliée par une pénalité de brièveté, destinée à pénaliser les systèmes qui essaieraient d'augmenter artificiellement leurs scores en produisant des phrases délibérément courtes. Le score BLEU varie de 0 à 1, ou il peut aussi être sous la forme de pourcentage, et il est d'autant meilleur qu'il est grand. BLEU a gagné le statut de mesure automatique de référence au sein de la communauté de traduction automatique.

3.3.2.2 Le score NIST

NIST, du nom de l'organisme américain *National Institue of Standards and Technology* qui a été proposé en 2002 [Doddington, 2002].
Ce score reprend le principe du score BLEU et l'adapte légèrement. La modification la plus notable est que, dans le score NIST, les *n*grammes sont pondérés par leur quantité d'information, et par leur fréquence : les *n*grammes rares contribuent plus au score final que les *n*grammes fréquents. Par ailleurs, l'expression de la pénalité de brièveté est légèrement différente de celle de BLEU, et enfin le score NIST prend en compte les précisions des 1-grammes jusqu'aux 5-grammes.

3.3.2.3 Le score TER

Le score TER pour Translation Edit Rate ou Translation Error Rate, compte aussi le nombre minimum d'opérations à effectuer sur l'hypothèse de traduction pour la transformer en phrase acceptable. Les opérations considérées sont l'insertion, la suppression, la substitution, mais aussi le déplacement d'un groupe de mots vers la gauche ou la droite. Chaque déplacement est compté comme une seule opération quels que soient le nombre de mots déplacés et l'amplitude du déplacement.

3.3.2.4 Le score METEOR

METEOR pour Metric for Evaluation of Translation with Explicit ORdering. Proposé par [Banergie et Lavie, 2005], il introduit plusieurs concepts intéressants comme l'équilibrage entre la précision et le rappel. Ce score est calculé sur la base d'un alignement entre les uni-grammes d'une hypothèse et ceux d'une référence.

3.3.2.5 Le score OOV

Ce score représente le pourcentage des mots qu'un système de traduction n'a pas réussi à traduire, le plus souvent, en traduction automatique, ces mots sont conservés sous leur forme initiale. Ce score dépend essentiellement de la quantité de données utilisées dans l'entrainement de la table de traduction.
Ce score est d'autant meilleur qu'il est petit.

Dans ce mémoire, les performances des systèmes qui seront développés vont être mesurées en termes de score BLEU, TER et OOV qui sont les métriques les plus significatives et les plus répandues dans le domaine de notre recherche.

3.4 La Langue arabe et le TALN[4]

3.4.1 Introduction

La langue arabe est la langue la plus utilisée parmi les langues sémitiques. Elle est parlée au Proche-Orient asiatique, dans tout le nord de l'Afrique, en Asie centrale, en Méditerranée et en Afrique sub-saharienne.
Son alphabet comporte vingt-huit lettres, dont trois sont des semi-consonnes ou voyelles longues. L'écriture de l'arabe est orientée de droite à gauche. Elle est curviligne et composée de consonnes liées entre elles. Quelques lettres arabes changent de forme selon leur position dans le mot.

Sibawahi (8e siècle), dans son livre *Al-Kitab* a formulé la première grammaire arabe et a proposé le premier travail de normalisation de cette langue.
Son ouvrage était une réponse aux inquiétudes des religieux, qui voulaient éviter tout risque de modification de la parole divine pouvant résulter de la mauvaise manipulation de la langue par les nouveaux convertis à l'Islam.

La structuration de la langue arabe comporte plusieurs variétés. En effet on trouve :

- L'arabe classique : l'ancienne forme linguistique, qui est apprise dans les établissements d'enseignement.

[4] Traitement automatique de langue naturel

- L’arabe littéral : c’est une variante moins formelle que l’arabe classique, qui contient quelques nouveaux mots et des procédés syntaxiques évolués, et elle a eu par conséquent, des évolutions lexicales et syntaxiques
- L’arabe dialectal : c’est la langue utilisée dans la vie quotidienne. Ses formes linguistiques peuvent varier d'une région à une autre.
- L’arabe médian : c’est une forme intermédiaire entre l'arabe moderne et dialectal, qui conserve la syntaxe et la morphologie de l’arabe dialectal et comporte un lexique mélangé de néologismes et d'arabe classique.

Les premières recherches sur le traitement automatique de l'arabe datent des années 1970. A cette époque les travaux s’intéressaient au lexique et à la morphologie.

> « A la différence des autres langues comme le français ou l’anglais, dont les étiquettes grammaticales proviennent d’une approche distributionnelle caractérisée par une volonté "d’écarter toute considération relative au sens", les étiquettes de l’arabe viennent d’une approche où le sémantique côtoie le formel lié à la morphologie du mot, sans référence à la position de ce dernier dans la phrase » [Débili F., Achour H., Souici E, 2002].

Ce phénomène se manifeste par la présence importante des notions de schèmes et de fonctions dans la langue arabe.

3.4.2 Morphologie de la langue arabe

Une caractéristique importante de la langue arabe est de langue à racines réelles, c’est-à-dire qu’à partir de ces dernières on peut déduire le reste du lexique arabe par application de différents schèmes morphologique, qui consiste en l’adjonction de voyelles et en manipulations de la racine. En effet, en arabe, les verbes et les noms sont le plus souvent issus d’une dérivation d’une racine de trois ou quatre lettres.

Une famille de mots, dénotant un même contenu sémantique, peut être générée d’une seule racine à l'aide de différents schèmes. Ce phénomène est caractéristique de la morphologie arabe.

Le tableau suivant donne quelques dérivations du mot « أكل » *akl* : *manger* par l’application des schèmes sur la racine :

Sens de 'manger'	*Le rythme du mot*	*أَكَلَ*
A mangé	فَعَلَ	أَكَلَ
Celui qui mange	فاعِلٌ	آكِلٌ
Celui qui a été mangé	مَفعولٌ	مَأكولٌ
Celui qu'on peut manger	فعْلٌ	أَكْلٌ
A été mangé	فُعِلَ	أُكِلَ
Celui qui mange beaucoup	فَعولٌ	أَكُولٌ
Celui qui est entrain de se faire manger	نُفعلُ	نُؤْكَلُ

Tableau 1- Schèmes de dérivation du mot **اكل** « akl »

L'arabe comprend environ 150 schèmes ou patrons dont certains plus complexes, tel le redoublement d'une consonne ou l'allongement d'une voyelle de la racine, l'adjonction d'un ou de plusieurs éléments ou la combinaison des deux.

Une caractéristique de la langue arabe est la structure composée des mots résultats d'une agglutination d'éléments de la grammaire. Du fait de cette structure, on peut avoir des mots qui peuvent désigner toute une phrase.

« Un mot arabe peut parfois signifier une phrase en français » [Chafik Aloulou et al. 2004].

Nous prenons comme exemple la représentation ci dessous qui schématise une structure possible d'un mot.

Rappelons que la lecture et l'écriture en langue arabe se fait de droite vers la gauche.

Post fixe (ou enclitique)	Suffixe	Corps schématique	Préfixe	Antéfixe (ou proclitique)

Tableau 2- Exemple de structure d'un mot en arabe

Les préfixes et les suffixes marquent des traits et fonctions grammaticales, alors que les post-fixes sont des pronoms personnels, et les antéfixes sont des prépositions ou des conjonctions.

Exemple :

أَتَسْتَقْبِلُونَنَا ***« Atastakbilounana »***

Ce mot exprime la phrase en français : "*Est ce que vous nous* accueillez ?"

La segmentation de ce mot donne :

Antéfixe : أَ ـَ conjonction d'interrogation.

Préfixe : تpréfixe verbal.

Corps schématique: سْتَقْبِلُ

Suffixe : ونَsuffixe verbal exprimant le pluriel.

Post fixe : ناpronom clitique complément du nom.

3.4.2.1 Catégories des mots

L'arabe compte trois catégories de mots :

- Le verbe : élément exprimant un sens dépendant du temps, c'est une unité fondamentale à laquelle se relient directement ou indirectement les divers mots qui constituent l'ensemble de la phrase.
- Le nom : élément indiquant un être ou un objet qui exprime un sens indépendamment du temps.
    - L'adjectif : sous-classe du nom se plaçant toujours après le nom qu'il qualifie. Il s'accorde en genre et en nombre avec lui ; il subit les mêmes règles de formation de féminin et de pluriel que celles des noms.
- Les particules : entités qui servent à situer les événements et les objets par rapport au temps et à l'espace, et permettent un enchaînement cohérent du texte.

3.4.2.1.1 Le verbe

Un grand nombre de mots en arabe sont des dérivations d'un verbe de trois lettres. Dans ce cas le verbe représente la racine d'une famille de mots.

Les traits flexionnels des verbes en arabe expriment les catégories suivantes :

- le temps : l'accompli (correspond au passé en français), l'inaccompli (correspond au présent en français),
- le nombre du sujet (singulier, duel, pluriel),
- le genre du sujet (masculin, féminin),
- la personne (première, deuxième et troisième),
- le mode (actif, passif).

3.4.2.1.2 Les noms

Les noms arabes sont de deux types : les noms qui sont dérivés d'une racine verbale, et les noms qui ne le sont pas comme les noms propres et les noms communs.
La déclinaison des noms suit les règles suivantes:

- Le féminin singulier : on ajoute la lettre ة à la fin du mot, par exemple كبير *grand* devient كبيرة *grande.*
- Le féminin pluriel : de la même manière, on rajoute pour le pluriel les deux lettres ات.
- Le masculin pluriel : on rajoute les deux lettres ني ou نو en fonction de la position du mot dans la phrase.
- Le pluriel irrégulier : il suit une diversité de règles complexes et dépend du nom.

3.4.2.1.3 Les particules

Les particules représentent principalement les conjonctions de coordination et de subordination. On distingue plusieurs types de particules : les introductions, les explications, et les conséquences.
En arabe le rôle des particules est important puisqu'il intervient dans l'interprétation de la phrase, car elles servent à situer des faits ou des objets en relation avec le temps ou le lieu, et de ce fait elles assurent la cohérence et l'enchaînement d'un texte.
Les particules peuvent porter des préfixes et suffixes, ce qui rend leur identification plus complexe.

3.4.3 Problèmes du traitement automatique de l'arabe

Plusieurs aspects sont particulièrement complexes à traiter automatiquement dans la langue arabe. Parmi ces aspects, on cite **la tokénisation des mots et l'absence de**

voyellation : ces deux problématiques constitueront nos axes de recherche principaux dans ce travail.

Concernant les voyelles, elles sont généralement utilisées pour faciliter la lecture ou pour rendre un texte beaucoup moins ambigu, elles permettent de distinguer des traits flexionnels tels que le genre, le nombre, la personne, etc.

Le problème que la voyelleation de nos données peut poser, est dû au manque de disponibilité de données arabes de grande taille qui sont voyellées, et aux ambiguïtés lexicales et morphologiques qui peuvent découler de son absence, vu qu'en utilisant les voyelles, un même texte peut être sous différentes formes (entièrement voyellé, semi-voyellé ou non voyellé).

D'autre part, la tokénisation de l'arabe est un processus sensible, étant donné la complexité de la morphologie de l'arabe, et les analyses locales au niveau des phrases et au niveau des mots qui doivent être effectuées afin de pouvoir tokéniser. D'où une certaine circularité dans cette problématique : « segmenter pour analyser ou analyser pour segmenter ? »

Au niveau de la traduction automatique probabiliste, nous rencontrons ces problèmes dans la phase de préparation de nos corpus. C'est pourquoi **une grande partie de notre recherche s'est concentrée sur l'influence de ces problèmes sur les résultats d'un traducteur probabiliste, afin de trouver le meilleur prétraitement des données pour le couple de langues arabe/français.**

Conclusion

Dans cette partie, nous avons commencé par présenter notre stage et son contexte. Ensuite, nous avons présenté la traduction automatique en nous intéressant à la traduction probabiliste et en présentant les principales composantes comme les modèles de langage, modèles de traduction et le décodeur**.** Ces modèles seront adoptés pour toutes nos expérimentations, en se reposant sur la boîte à outil du logiciel libre Moses.

Par la suite nous avons présenté notre langue source, la langue arabe, et nous avons abordé sa richesse morphologique et les problèmes posés par son traitement automatique. Dans notre cas ce problème consiste en la préparation de nos données arabes pour construire un système de traduction vers le français.

Nous commençons dans le chapitre suivant par l'étude de l'influence de la voyellation et de la tokénisation des données arabes sur les résultats d'un système de traduction.

II. Partie 2

–

Premières contributions : recherche d'une séquence de meilleurs prétraitements pour l'arabe

Introduction

Dans ce chapitre nous essayons de déterminer l'influence de la préparation de nos données sur l'efficacité de nos systèmes de traduction.
Un seul texte en langue arabe, comme cité dans la partie d'état de l'art, peut être sous plusieurs formes en effet il peut être voyellé ou non, comme il peut y avoir différentes représentations pour un même caractère.

Dans nos premières expérimentations nous essayons de trouver la meilleure combinaison de prétraitements à appliquer sur nos données en arabe pour avoir la meilleure traduction en français.

1 Chapitre 1 - Présentation d'outils

1.1 Corpus

Dans un premier temps, nous avons travaillé sur le corpus TRAME (figure 1), un corpus voyellé et qui est notre corpus de référence spécifique à la tâche de traduction.
Ce corpus est fourni par l'entreprise Cassidian, il consiste en une transcriptions de 82 heures des bulletins de nouvelles de certaines chaines de télévisions et radios arabes comme l'ORIENT, MTV, ALALAM, ALJAZEERA, etc.

Ce corpus est sous forme de 247 fichiers, il contient 20 660 paires de phrases alignées, 554 032 mots arabes et 769 220 mots français. Les tailles des vocabulaires français et arabes sont respectivement de 46196 et 91163 mots différents. Un tel corpus n'est pas suffisant pour créer un système de traduction probabiliste efficace, mais il est très raisonnable de l'utiliser dans des expérimentations préliminaires pour choisir le meilleur ensemble de prétraitements de l'arabe à faire.

Donc, les entrées initiales de notre système sont deux corpus monolingues, un corpus arabe et un autre français. Ces deux corpus sont structurés de telle façon que chaque ligne i dans le corpus arabe est alignée avec la ligne i, qui est sa traduction, dans le corpus français.

Nous avons décomposé le corpus TRAME en trois sous-corpus qui sont essentielles pour construire un système de traduction automatique statistique :

- un corpus d'entraînement qui constitue approximativement 90% de la totalité des données parallèles : 18557 lignes (**TRAM-TRAIN**) ;

- deux corpus pour le développement : 1035 lignes (**TRAM-DEV**) et l'évaluation : 1068 lignes (**TRAM-TEST**) des systèmes de traduction.

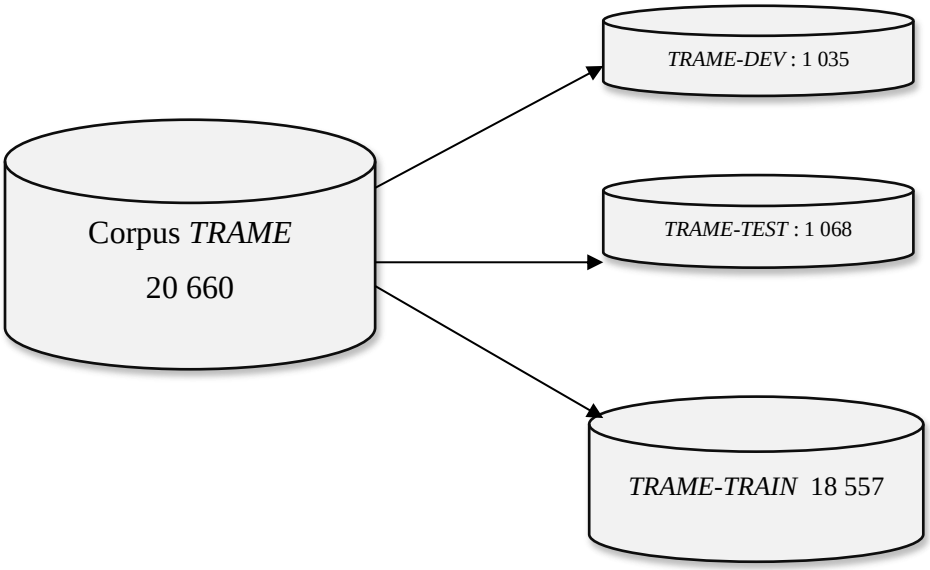

Figure 2– Répartition du Corpus TRAME

1.2 Prétraitement de l'arabe en utilisant l'outil MADA+TOKAN

MADA pour *Morphological Analysis and Disambiguation for Arabic*, un logiciel développé au sein de l'université de Columbia, représente une solution à différents problèmes de traitement automatique de la langue arabe, comme la désambiguïsation morphologique, la gestion des voyelles et la lemmatisation. Il a été utilisé par plusieurs instituts de recherches académiques dans le monde entier (l'Université de Washington, l'Université de Cambridge, la Copenhagen Business School, etc). L'approche sur laquelle ce logiciel est basée consiste à effectuer une analyse en atomes lexicaux, puis une désambiguïsation morphologique pour chaque atome.

Par la suite MADA affecte à chacun de ces atomes, une étiquette, un lemme et il le met en forme voyellé. Cette analyse morphologique est probabiliste et faite à partir d'une liste des analyses fournies par l'analyseur morphologique arabe Buckwalter (BAMA;

Buckwalter 2004). Après cette désambiguïsation, la composante TOKAN permet d'effectuer la tache de tokénisation.

MADA arrive à plus de 96% de précision sur le choix morphologique, et plus de 86% de précision sur l'accentuation.[5]

A partir de cette analyse on obtient un texte désambiguïsé. MADA+TOKAN peut tokéniser le mot d'une manière déterministe, en proposant plusieurs styles : séparation des conjonctions, propositions, et articles.

1.3 Outils de création de nos systèmes

Durant nos expérimentations, la création des systèmes de traduction automatique probabilistes s'est faite à l'aide de plusieurs outils, qui sont disponibles en licence GPL sur le Web et téléchargeables gratuitement, parmi ces outils, on cite :

- **SRILM** est une boîte à outils pour la construction de modèles de langage statistiques, souvent utilisée en reconnaissance automatique de la parole et en traduction automatique probabiliste.
- **GIZA++** est une librairie qui implémente les modèles IBM qui servent à l'alignement des corpus parallèles utilisés pour la traduction automatique par des méthodes statistiques.
- **MOSES** est une boîte à outils très performante qui implémente les algorithmes d'apprentissage et de décodage pour les systèmes de traduction automatique statistique. Cet outil est gratuitement disponible sur le web et est toujours en cours de développement.

[5] http://www1.ccls.columbia.edu/MADA/

2 Chapitre 2 - Normalisation des diacritiques

2.1 Les diacritiques en arabe

En arabe, la notion de voyelles n'existe pas sous sa forme classique : en effet elles ne sont pas des lettres de l'alphabet, mais représentées par des signes diacritiques placés facultativement au dessus ou au dessous des consonnes et qui jouent le même rôle que les voyelles dans les autres langues. Un lecteur ayant une bonne connaissance en langue arabe peut deviner les diacritiques d'un texte non voyellé au moment de la lecture.

Les voyelles en arabe sont généralement utilisées pour faciliter la lecture ou pour rendre un texte beaucoup moins ambigu, elles permettent de distinguer des traits flexionnels tels que le genre, le nombre, la personne, etc. Pour cette raison les textes religieux, les ouvrages pédagogiques ainsi que les textes juridiques sont entièrement voyellés.

2.1.1 Les diacritiques obligatoires

Les diacritiques obligatoires servent à distinguer des lettres qui ont des tracés très proches, ces diacritiques sont le point, les deux points et les trois points.

ـب		
ب	ت	ث
B	T	ṯ

Tableau 3– Exemple de diacritique obligatoire

On peut constater à partir de la transcription ci-dessus que le nombre de points est important dans certaines lettres arabes, puisqu'un point de plus ou de moins peut transformer la lettre en une autre lettre.

2.1.2 Les diacritiques de désambiguïsation

Les diacritiques de désambiguïsation sont les signes de vocalisation, utilisés pour lever des ambiguïtés morphologiques, syntaxiques et sémantiques. Par exemple, on a les

diacritiques casuels qui servent à lever l'ambiguïté syntaxique, qui s'associent à la dernière lettre d'un mot et qui précisent sa fonction syntaxique dans la phrase.

Le mot 'ذَهَبَ' est un verbe conjugué à la troisième personne du singulier qui veut dire en français 'est allé', alors que le mot 'ذَهَبٌ' est un nom qui se traduit par 'or'. La seule différence entre l'écriture de ces deux mots est le diacritique de la dernière lettre (la première lettre en lisant de gauche à droite).

Les diacritiques lexicaux servent à lever les ambiguïtés morphologiques et sémantiques.

D'après Debili (1998), 74% des mots en moyenne acceptent plus d'une accentuation lexicale, et 89,9% des noms acceptent plus d'un diacritique casuel.

La proportion des mots ambigus est de 90,5% si les comptages portent sur leurs accentuations globales 248 (lexicales et casuelles).

Dans la suite nous utilisons la convention de translittération définie par Buckwalter (2004), dans laquelle on représente entre crochets la représentation réversible avec des caractères ASCII.

On peut classer les diacritiques de désambiguïsation en trois groupes :

- Les diacritiques simples qui sont au nombre de quatre :

◌َ	[a]
◌ُ	[u]
◌ِ	[i]
◌ْ	[o]

Tableau 4– Les diacritiques simples

Tous ces diacritiques se prononcent de la même façon que leur translittération en français sauf le dernier [o] qui indique l'absence de tout son.

- Les diacritiques doubles sont des diacritiques casuels, qui produisent, respectivement le même son que les trois premières voyelles simples avec l'ajout du son « n » à la fin.

ً	[an]
ٌ ً	[on]
ٍ	[in]

Tableau 5– Les diacritiques doubles

- Le diacritique «chadda», ّ [CD] qui a pour effet le doublement de la lettre à laquelle il est associé.

Pour illustrer les ambiguïtés qui peuvent être provoquées par l'absence de diacritiques, prenons l'exemple du mot non voyellé كتب [ktb].
Ce mot peut être reconnu comme étant:

كَتَبَ	[kataba]	A écrit	troisième personne du singulier, passé, voix active
كُتِبَ	[kutiba]	A été écrit	troisième personne du singulier, passé, voix passive
كتّبَ	[kattaba]	A beaucoup écrit	troisième personne du singulier, passé, voix active, exprime l'exagération
كُتُبٌ	[kutubon]	Des livres	nom, pluriel

Tableau 6– Exemple d'ambiguïté d'un mot non voyellé

On constate que la forme كتب [ktb] peut présenter plusieurs diacritisations potentielles, pour un seul lemme et plusieurs catégories grammaticales.[6]
Cet exemple montre bien que l'ambiguïté vocalique d'un mot produit des ambiguïtés lexicales et grammaticales.

Bien que les diacritiques soient souvent destinés à lever les ambiguïtés lors d'un traitement automatique, la majorité des systèmes de traduction automatique comme Systran, Google Traduction et même les analyseurs morphosyntaxiques de l'arabe comme celui de Buckwalter, Xerox ou l'analyseur MADA ne traduisent et n'analysent que des textes non voyellés à cause du manque de ressources arabes sous cette forme.

Par conséquent, si l'entrée est voyellée ou partiellement voyellée, ces systèmes commencent par éliminer tous les diacritiques, puis ils font le traitement comme si l'entrée était non voyellée.

2.2 Expérimentations

Afin d'étudier d'une façon concrète l'influence des diacritiques sur les scores BLEU, TER et OOV d'un système de traduction probabiliste, nous avons créé neuf systèmes de traduction arabe/français, qui sont entrainés, développés et testés respectivement sur TRAME-TRAIN, TRAME-DEV et TRAME-TST avec des taux de diacritisation différentes.
Une élimination des diacritiques a fait passer le nombre de mots de vocabulaire arabe du corpus TRAME de 91 163 à 57 724, ce qui rend la partie arabe du corpus plus ambiguë.

	TRAME	
	AR	*FR*
Initiale	**91 163**	**46 196**
élimination des diacritiques	**57 724**	**46 196**

Tableau 7– Taille du vocabulaire dans un corpus voyellé

[6] GHOUL Dhaou, Olivier Kraif (2011) Outils génériques pour l'étiquetage morphosyntaxique de la langue arabe : segmentation et corpus d'entraînement

Rappelons que la partie TRAME-TRAIN présente 90% de la taille totale du corpus, et que ce corpus a été nettoyé afin de n'avoir que du texte brut, les données en français ont subi une normalisation de ponctuation, mise en minuscule et tokénisation en utilisant des scripts de Moses.

Les données en arabe ont été filtrées par la suppression de toutes les lignes de chaque côté qui se composent de plus de 100 mots.

Le modèle de langage 5-grammes est formé à partir d'un corpus français dont la taille est de 7,5 M lignes, en utilisant la boîte à outils SRILM (Stolcke, 2002).

Dans notre travaille nous évaluons les performances de nos systèmes par les scores :

- BLEU : qui mesure la qualité d'un système de traduction automatique en comparant sa sortie par une traduction humaine.
- TER : qui mesure le nombre de corrections estimées pour améliorer la sortie d'un système de traduction.
- OOV : qui représente le pourcentage des mots qu'un système de traduction n'a pas réussi à traduire.

Pour le premier et le deuxième score, nous avons utilisé des scripts Moses, alors que pour le score OOV, nous avons développé notre propre script.

Les trois scores d'évaluation des systèmes créés sont présentés dans le tableau suivant :

		Performances des systèmes		
Taux de diacritisation	*diacritiques utilisés*	*BLEU*	*TER*	*OOV*
100	[an] [un] [in] [a] [u] [i] [o] [CD]	**21,00**	**70,00**	**12,00**
87	[an] [un] [in] [a] [u] [i] [o]	**21,06**	**68,00**	**11,60**
75	[an] [un] [in] [a] [u] [i]	**21,07**	**68,00**	**11,50**
62	[an] [un] [in] [a] [u]	**21,52**	**68,00**	**11,20**
50	[an] [un] [in] [a]	**22,23**	**67,00**	**10,30**
37	[an] [un] [in]	**22,57**	**66,00**	**9,06**
25	[an] [un]	**24,12**	**63,00**	**7,77**
12	[an]	**24,95**	**63,00**	**7,77**
0		**24,96**	**63,00**	**7,52**

Tableau 8– Performances des systèmes apprentis sur des corpus voyellés

Un bon système de traduction automatique est un système ayant un score BLEU élevé et des TER et OOV faibles.

Les expérimentations ont montré que plus le corpus contient des diacritiques, plus les performances de système doivent se dégrader. En effet avec un corpus qui ne contient aucune diacritique on obtient un BLEU de 24,96% et ce score se dégrade de 4,9 points lorsque le corpus est totalement voyellé.

La figure ci-dessous illustre l'évolution des scores de système de traduction selon le taux de diacritisation du corpus.

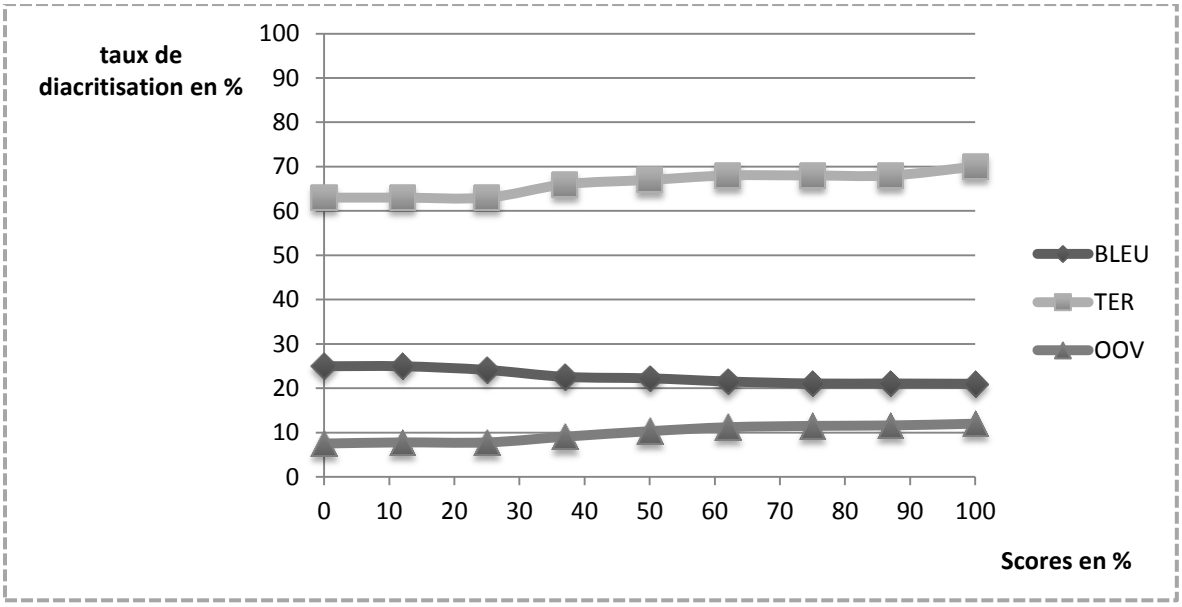

Figure 3– Evolution des scores de système selon le taux de diacritisation

Aussi la taille du vocabulaire du côté arabe peut avoir une influence sur les scores de la traduction, car en arabe on distingue entre le 'toi' féminin et le 'toi' masculin, distinction qui apparait à la prononciation et à l'écriture à l'aide des voyelles, ce qui n'est pas le cas en français.

Exemple :

Arabe	Lecteur francophone	genre	Français
هل أكلتَ ؟	[AKALTA]	Masculin	Est-ce que tu as mangé ?
هل أكلتِ ؟	[AKALTI]	Féminin	Est-ce que tu as mangé ?

Tableau 9– Exemple d'ambiguïté d'une phrase non voyellé

La traduction de ce mot en français est la même, en effaçant son dernier diacritique [A] ou [I] le système ne fera plus différence de genre et nous aurons une meilleure traduction.

Le tableau ci-dessous, il présente la taille du vocabulaire du corpus TRAME après les différentes normalisations de voyelles, et l'évolution de ses scores.

Taux de diacritisation	Taille du vocabulaire	BLEU
100%	90835	**21,00**
87%	89547	**21,06**
75%	88546	**21,07**
62%	86441	**21,52**
50%	77260	**22,23**
37%	71801	**22,57**
25%	59189	**24,12**
12%	59189	**24,95**
0%	57524	**24,96**

Tableau 10 – Evolution de la taille du vocabulaire selon le taux de diacritisation

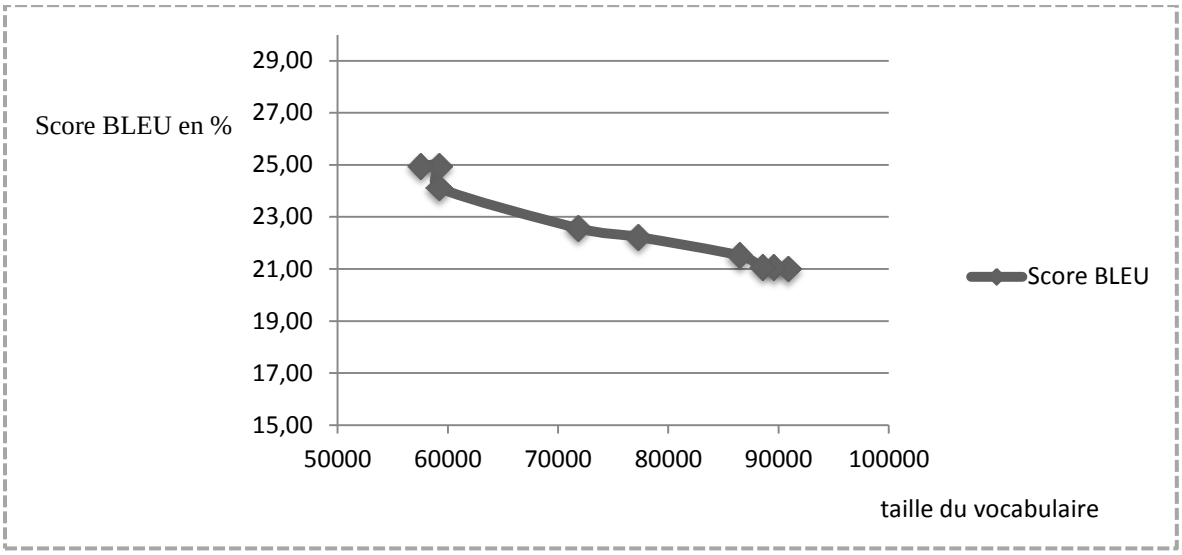

Figure 4– Evolution de score BLEU en fonction de la taille du vocabulaire

2.3 Conclusion

Dans le traitement automatique de la langue, la détermination automatique des informations lexicales sur un mot et sa désambiguïsation sont une étape importante et difficile. Pour la langue arabe, les diacritiques aident à effectuer cette tache.

Mais les expérimentations que nous avons réalisées avec le corpus TRAME ont montré que pour un système de traduction probabiliste de l'arabe entrainé sur peu de données, la désambiguïsation des mots en utilisant les diacritiques dégrade considérablement les préférences.

3 Chapitre 3 - La tokénisation

Comme nous l'avons mentionné plus tôt, l'écriture arabe est cursive et présente divers diacritiques et un mot arabe est une séquence d'entités connexes, qui sont à leurs tour formées d'un ou plusieurs caractères. En conséquence on peut avoir des mots qui peuvent désigner toute une phrase : ce mot aura une structure d'assemblage d'éléments de la grammaire (conjonctions, prépositions, article de définition ou article de possession) avec le lemme.

La tokénisation consiste à séparer le lemme de ces éléments, donc l'enjeu est d'avoir la tokénisation optimale à effectuer dans la phase d'alignement pour la traduction de l'arabe vers le français.

3.1 Les styles de tokénisation

En arabe, une forme fléchie peut être composée de la manière suivante :

Un lemme + une conjonction :

Forme fléchie en Arabe	*Forme fléchie en Français*
ويقرر	et décide

Tableau 11– Exemple lemme + conjonction

Un lemme + une préposition:

Forme fléchie en Arabe	*Forme fléchie en Français*
ليقرر	pour décider

Tableau 12– Exemple lemme + préposition

Un lemme + un article de définition:

Forme fléchie en Arabe	*Forme fléchie en Français*
القرار	la décision

Tableau 13– Exemple lemme + article de définition

Un lemme + un article de possession:

Forme fléchie en Arabe	*Forme fléchie en Français*
قراره	sa décision

Tableau 14– Exemple lemme + article de définition

Un style de tokénisation est le choix des éléments à séparer, en combinant toutes les séparations possibles on peut avoir 16 styles.

Vu qu'en français on sépare généralement les articles et les conjonctions et les prépositions du lemme, nous pensons qu'une tokénisation pareille en arabe sera avantageuse pour la création d'un système de traduction probabiliste.

Dans les expérimentations suivantes nous allons confirmer cette hypothèse en comparant l'impact de plusieurs styles de segmentation sur un système de traduction.

3.2 Expérimentations

16 systèmes différents de traduction automatique sont entrainés et développés respectivement sur **TRAME-TRAIN** et **TRAME-DEV**, en utilisant ces différents styles.

Les systèmes sont testés et comparés sur la partie **TRAME-TEST**, et sur la base de cette comparaison nous identifions le meilleur et les pires des styles de segmentation et nous observons l'effet de la tokénisation sur la traduction de couple arabe/français.

Les données en français ont subi une normalisation de ponctuation, mise en minuscule et tokénisation en utilisant des scriptes de Moses, alors que le côté arabe a été segmenté avec le toolkit Morphological Analysis and Disambiguation for Arabic (MADA) (Habache et Rambow, 2005).

Ensuite, TRAME-TRAIN a été filtré par la suppression de toutes les lignes de chaque côté qui se composent de plus de 100 mots, qui ne sont pas adaptées à l'apprentissage.

Le côté arabe du corpus d'apprentissage **TRAME-TRAIN** est utilisé pour former le modèle de langage 3-grammes en utilisant le Toolkit SRILM (Stolcke, 2002).

Le tableau ci-dessous contient l'évolution de taille du corpus en mots et la taille du vocabulaire après chaque tokénisation.

Système	*Style de Tokénisation*	*Exemple FR*	*Exemple AR*	*Taille du vocabulaire du corpus*	*Taille du corpus en mots*
S1	*ponctuation*	.	.	**49802**	578937
S2	*conjonction*	et décide	ويقرر	42980	615315
S3	*préposition*	pour décider	ليقرر	43187	611822
S4	*def.article*	la décision	القرار	42978	739634
S5	*pos.article*	sa décision	قراره	43597	611676
S6	*conjonction + préposition*	et pour décider	وليقرر	36526	648200
S7	*Conjonction + def.article*	et la décision	والقرار	36849	776012
S8	*conjonction + pos.article*	et sa décision	وقراره	37047	648054
S9	*Préposition + def.article*	pour la décision	للقرار	37168	772519
S10	*Préposition + pos.article*	pour sa décision	لقراره	37275	644561
S11	*pos.article + def.article*	la décision + sa décision	قراره القرار	36461	772373
S12	*conjonction + préposition + def.article*	et pour la décision	وللقرار	31234	808897
S13	*conjonction + préposition + pos.article*	et pour sa décision	ولقراره	30925	680939
S14	*conjonction + pos.article + def.article*	et la décision + et sa décision	وقراره والقرار	30705	808751

S15	***préposition + def.article + pos.article***	pour la décision + pour sa décision	لقراره للقرار	31010	805258
S16	***conjonction + préposition + def.article + pos.article***	et la décision + et sa décision + pour la décision + pour sa décision	وقراره والقرار لقراره للقرار	25489	**841636**

Tableau 15 - Evolution de la taille du corpus et la taille du vocabulaire selon le style de tokénisation

Nous rapportons dans le tableau 14 les résultats de l'ensemble des tests définis précédemment à l'aide des paramètres d'évaluation BLEU, TER et OOV.

Système	***Style de Tokénisation***	***Score BLEU %***	***TER %***	***OOV %***
S1	***ponctuation***	21,600	63,000	6.070
S2	***conjonction***	22,710	61,500	5,121
S3	***préposition***	22,440	62,400	5,199
S4	***def.article***	22,060	63,000	5,108
S5	***pos.article***	21,640	63,600	5,246
S6	***conjonction + préposition***	23,530	60,250	4,217
S7	***conjonction + def.article***	22,820	61,060	4,283
S8	***conjonction + pos.article***	23,040	61,890	4,316
S9	***Préposition + def.article***	22,580	61,800	4,397
S10	***Préposition + pos.article***	22,720	62,200	4,447

S11	***pos.article + def.article***	22,280	62,600	4,290
S12	***conjonction + préposition + def.article***	**23,600**	60,320	3,640
S13	***conjonction + préposition + pos.article***	23,410	60,800	3,519
S14	***conjonction + pos.article + def.article***	23,180	61,100	2,969
S15	***préposition + def.article + pos.article***	22,690	61,520	3,649
S16	***conjonction + préposition + def.article + pos.article***	23,280	**60,300**	**2,963**

Tableau 16– Les scores des systèmes de chaque style de tokénisation

3.3 Résultats

Les résultats de cette expérience ont montré que la segmentation des données a une influence directe sur les performances d'un système de traduction probabiliste. En effet on note une amélioration dans les trois scores en passant d'un système sans tokénisation (S0) à un système avec un style de séparation d'un seul élément (S2, S3, S4, S5).

Au niveau de score BLEU, les résultats ont montré qu'une tokénisation où l'on ne sépare que les conjonctions et les prépositions (S6), peut donner un système aussi bon qu'un système basé sur une tokénisation de style complexe (S16). Par contre cette dernière (S16) possède un taux de mots inconnus inferieur à celle de (S6).

En comparant les différents résultats donnés par des tests sur ces systèmes nous sommes en mesure de constater que :

- S2 est plus performant que S3, S4 et S5, donc la séparation des conjonctions est plus importante que la séparation des prépositions ou des articles.

- S7, S9 et S11 sont moins performants que les systèmes au styles à 2 séparations : S6, S8 et S10 ce qui indique que la séparation des articles de définitions nuit à la performance.
- S12 et S13 sont les meilleurs systèmes avec un style à 3 séparations, ce qui indique qu'avec une séparation des conjonctions en plus de la séparation des prépositions, la séparation des articles n'a pas d'effet significatif sur les performances du système.
- En regardant les trois mesures d'évaluation, on trouve que le meilleur système est celui dont les données ont subit la tokénisation la plus agressive, où on sépare les conjonctions, les prépositions, les articles de définition et les articles de possession de lemme.

Les figures ci-dessous illustrent l'évolution des scores des différents systèmes selon la forme de la tokénisation :

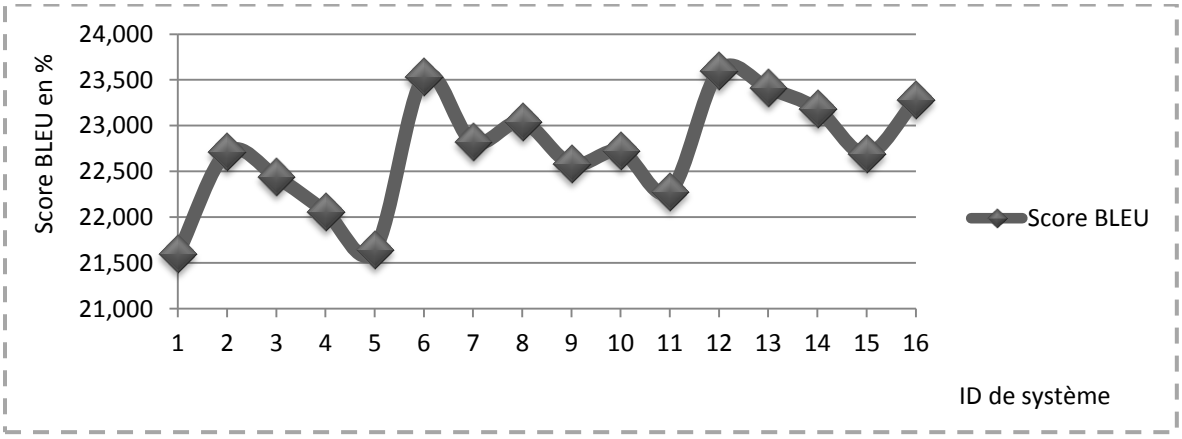

Figure 5- Evolution du score BLEU suivant les systèmes

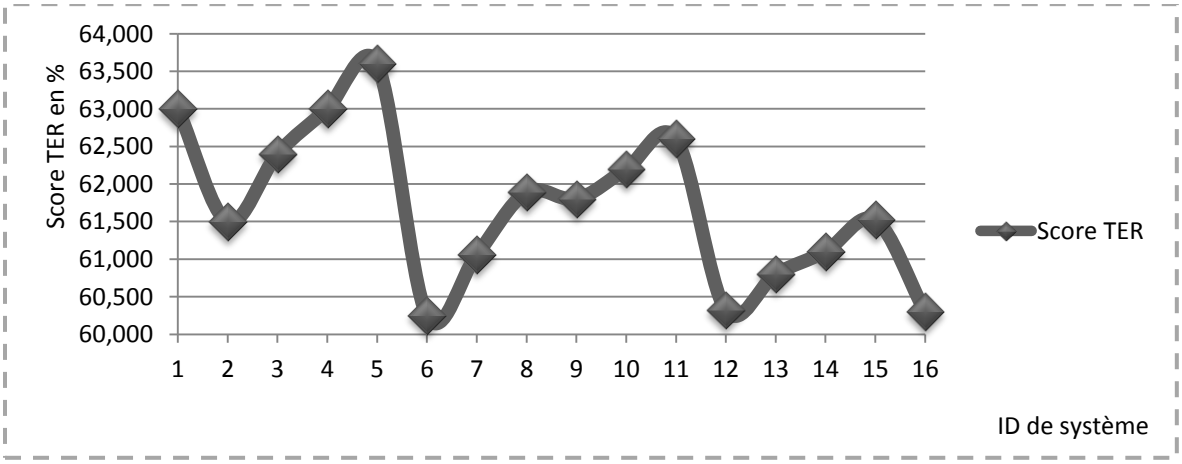

Figure 6- Evolution du score TER suivant les systèmes

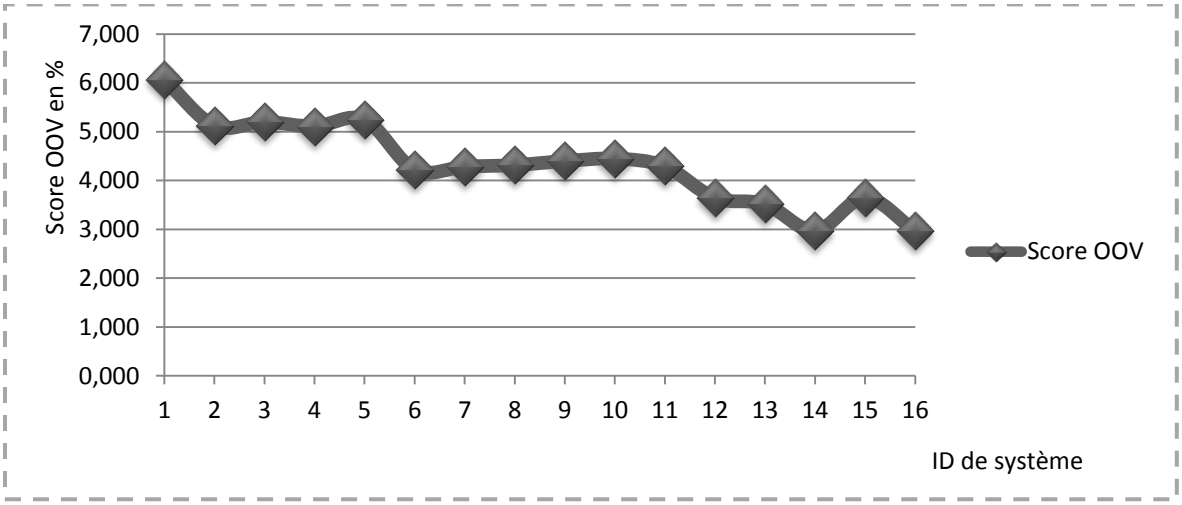

Figure 7- Evolution du score OOV suivant les systèmes

3.4 Conclusion

Dans cette partie nous avons étudié l'impact de la segmentation des textes arabe sur les performances d'un système de traduction, de l'arabe vers le Français.

Nous avons exploré exhaustivement les styles de tokénisation possibles de l'arabe, allant de la forme du mot complet à des formes totalement segmentées, et nous avons examiné les effets sur les performances du système.

Nos résultats montrent une différence de deux points BLEU entre le meilleur style de tokénisation et le pire, indiquant que le choix du style de la tokénisation a un effet significatif sur les performances d'un système de traduction de ce couple de langues, surtout avec un corpus de petite taille.

Nous avons aussi montré que le modèle de segmentation le plus complet qui divise les conjonctions, les prépositions et les articles donne le meilleur système de traduction.

4 Chapitre 4 - La Normalisation des HAMZA

4.1 Le hamza

Le "hamza" (ء) est un phonème particulier de la langue arabe. Il peut se comporter comme une lettre ou un diacritique, et ses règles d'écriture sont nombreuses et dépendent de sa nature ainsi que de sa place dans le mot.

Le hamza mis en question dans ce travail est appelé *"hamza tahri à l'initiale"*, il est classé comme un diacritique, il ne se lie jamais et ne change pas de forme, il s'écrit toujours avec la lettre *"alif"* (ا) et généralement il est suivi d'une voyelle.

Contrairement au *"alif"*, le *"hamza tahri à l'initiale"* tout seul ne fait pas partie de l'alphabet arabe tel qu'il est rendu par les dictionnaires.

Ce phonème est particulier, et s'introduit souvent dans la construction du verbe aux 4eme et 5eme personnes de singulier (il / elle) :

أَخذ	['aẖaḍa]	il a pris

Tableau 17– Exemple d'utilisation du "hamza"

Dans ce travail nous avons pensé à normaliser cette lettre (ce diacritique) car dans certains cas il peut disparaître, quand il est sous la forme dite "instable", par exemple lorsqu'il est précédé d'un autre mot :

إنتفض الشعب	'intafada ech'âbu	Le peuple a révolté
متى انتفض الشعب ؟	mata-ntafada ech'âbu ?	Quand est-ce que le peuple a révolté?

Tableau 18– Exemple d'utilisation de "hamza instable"

D'un point de vue linguistique dans le Coran on trouve des indications sur la variabilité de prononciation de "hamza" dans l'arabe, si on met en regard les textes de la lecture "hafs" (H) et ceux de la lecture "warš" (W) :

(**H**)	ولم يكن له كفواً أحد (112/4)	wa-lam yakun lahu **kufw**an 'aḥad	et il n'a nul semblable (égal)
(**W**)	ولم يكن له كفؤاً احد (112/4)	wa-lam yakun lahu **kufu**'an-ḥad	et il n'a nul semblable (égal)

Tableau 19– Exemple de prononciation de "hamza instable"

On peut dire que le rôle du "hamza" en langue arabe se limite à la prononciation, car il indique un arrêt ou une pause (aucun son) comme une ponctuation. De ce fait nous avons eu l'idée de normaliser cette lettre/diacritique dans notre corpus en rendant chaque أ ou إ ou آ en ا.

4.2 Expérimentations

Le tableau ci-dessous présente l'évolution de la taille du vocabulaire du corpus dans chaque cas de tokénisation effectué lors des expérimentations précédentes avant et après la normalisation de "hamza".

Système	***Style de Tokénisation***	***Taille du Vocabulaire du corpus avant normalisation de Hamza***	***Taille du Vocabulaire du corpus après normalisation de Hamza***
S1.N	***ponctuation***	49802	49513
S2.N	***conjonction***	42980	42697
S3.N	***préposition***	43187	42887
S4.N	***def.article***	42978	42672
S5.N	***pos.article***	43597	46329
S6.N	***conjonction + préposition***	36526	36241

S7.N	***conjonction + def.article***	36849	36565
S8.N	***conjonction + pos.article***	37047	36742
S9.N	***Préposition + def.article***	37168	36862
S10.N	***Préposition + pos.article***	37275	36948
S11.N	***pos.article + def.article***	36461	36123
S12.N	***conjonction + préposition + def.article***	31234	30961
S13.N	***conjonction + préposition + pos.article***	30925	30630
S14.N	***conjonction + pos.article + def.article***	30705	30401
S15.N	***préposition + def.article + pos.article***	31010	30681
S16.N	***conjonction + préposition + def.article + pos.article***	25489	25206

Tableau 20 – L'évolution de taille de vocabulaire dans chaque cas de tokénisation

Afin d'avoir une idée précise sur l'effet de la normalisation du "hamza" sur un système de traduction probabiliste, on a appliqué cette normalisation sur toutes les données sur lesquelles on a créé les systèmes expérimentés dans la partie précédente (S1..S16).

Les résultats obtenus sont présentés dans le tableau 15.

Système	*Style de Tokénisation*	*Score BLEU*	*TER*	*OOV*
S1.N	***ponctuation***	21,770	63,300	6,055
S2.N	***conjonction***	22,950	61,800	5,028
S3.N	***préposition***	22,410	62,570	5,117
S4.N	***def.article***	21,880	63,130	5.086
S5.N	***pos.article***	21,970	72,810	8,868
S6.N	***conjonction + préposition***	23,240	60,720	4,192
S7.N	***Conjonction + def.article***	23,090	61,090	4,302
S8.N	***conjonction + pos.article***	22,740	61,920	4,269
S9.N	***préposition + def.article***	22,810	61,940	4,390
S10.N	***préposition + pos.article***	22,530	62,280	4,350
S11.N	***pos.article + def.article***	21,970	63,220	4,239
S12.N	***conjonction + préposition + def.article***	23,340	60,200	3,626
S13.N	***conjonction + préposition + pos.article***	**23,730**	60,330	3,488

S14.N	***conjonction + pos.article + def.article***	23,240	60,790	3,454
S15.N	***préposition + def.article + pos.article***	23,030	61,17	3,624
S16.N	***conjonction + préposition + def.article + pos.article***	23,540	**60,100**	**2,937**

Tableau 21– Les scores de système dans chaque cas de tokénisation

4.3 Résultats

La normalisation du *"hamza"* a amélioré, même légèrement, le score OOV des systèmes, car on remarque une baisse de ce score dans la majorité de systèmes, et le schéma ci-dessous illustre cette baisse.

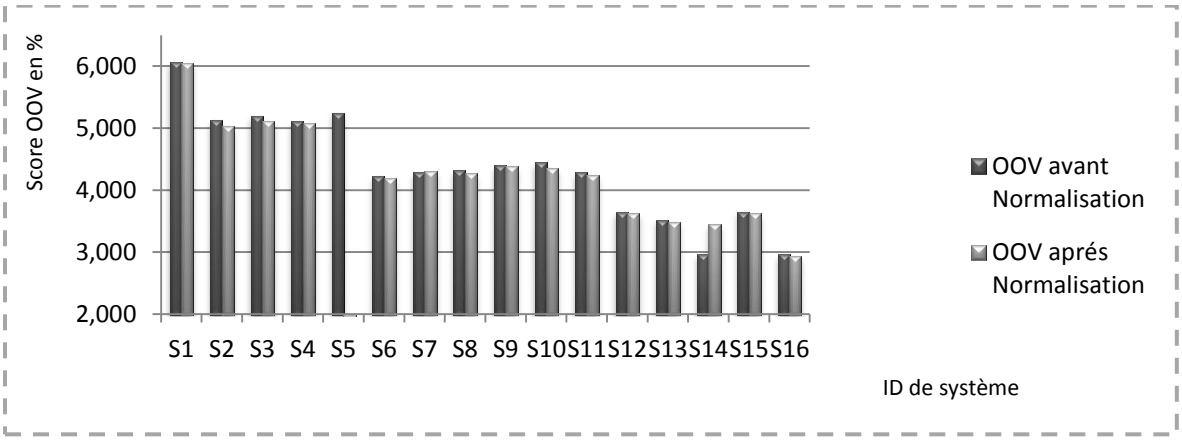

Figure 8– Comparaison d'OOV avant et après chaque tokénisation

Au niveau du score BLEU, on remarque une amélioration dans la plupart de systèmes, on peut voir cette amélioration dans la figure 9.

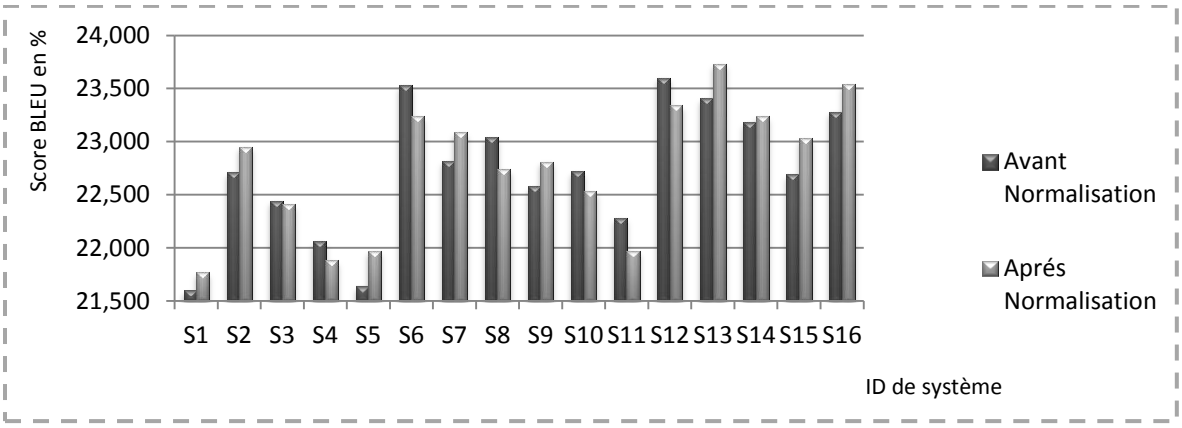

Figure 9– Comparaison de BLEU avant et après chaque tokénisation

4.4 Conclusion

Avec ces expérimentations nous avons montré que la normalisation de la lettre *"hamza"* réduit la taille du vocabulaire du côté arabe dans le corpus et améliore par la suite les performances du système de traduction automatique.

Nous avons testé l'effet de cette normalisation avec tous les styles de tokénisation possibles de l'arabe, et nous en avons examiné les effets sur les performances du système. Nos résultats montrent une amélioration dans ces systèmes développés sur le corpus TRAME qui fait 20 660 lignes de taille.

Conclusion

Avec les expérimentations effectuées dans ce chapitre, nous nous sommes basé sur le Corpus TRAME pour créer une cinquantaine de systèmes de traduction probabilistes avec différents prétraitements des données en arabe.

Ces expérimentations ont montré qu'avec une suppression des voyelles, une normalisation du *« hamza »* et une tokénisation complète des données en arabe on obtient le meilleur système de traduction du couple de langues arabe/français.

Nous confirmons ces résultats dans le chapitre 2, où nous tenterons de montrer que cette préparation a un effet positif sur les systèmes de traduction probabilistes créés avec des corpus de grande taille.

III. Partie 3

–

Expérimentations avancées

Introduction

Dans ce chapitre nous allons utiliser d'autres corpus de différentes tailles fournis par la société CASSIDIAN, afin de confirmer les résultats et les décisions prises dans le chapitre précédent concernant la normalisation de *« hamza »*. Par la suite nous allons nous intéresser à l'étude de l'adaptation de nos systèmes de traduction obtenus avec notre corpus de référence, le corpus TRAME.

La suite de cette partie est organisé comme suit : nous donnons en premier lieu une présentation détaillée des corpus utilisés, en deuxième lieu nous comparons les systèmes sans et avec normalisation de *« hamza »*, et enfin dans le but d'optimiser les performances de traduction, nous adaptons nos systèmes avec le corpus TRAME.

1 Chapitre 1 - Présentation des données

Pour réaliser ce projet, l'entreprise CASSIDIAN nous a fournit 4 bitextes de différentes tailles et domaines :

- TRAME : c'est notre corpus de référence spécifique à la tâche de traduction, nous l'avons réparti en deux parties: soit deux heures de transcription pour la partie DEV et quatre-vingt heures pour la partie TRAIN.
- News Commentary AR/FR : ce corpus a été fourni dans le sixième workshop de systèmes de traduction probabiliste qui a eu lieu le 30/07/2011 à l'université Edinburg/UK.
- NIST : Un corpus fourni par *National Institute of Standards and Technology,* qui organise des séries d'évaluation des systèmes de traduction probabilistes. La dernière évaluation a eu lieu en 2009. Et la version avec laquelle nous avons travaillé est NIST08.
- UN : Un corpus disponible en sept langues, construit à partir des documents fournis par l'organisation des Nations unies entre les années 2000 et 2009. Le corpus est nettoyé et aligné au niveau des phrases des phrases.

Une fois qu'on a eu toutes les données, la société Cassidian a décidé que le corpus **NIST** sera notre corpus de test, la partie développement de TRAME (**TRAME-DEV**) servira au développement, et on entrainera des tables de traduction avec le reste de données.

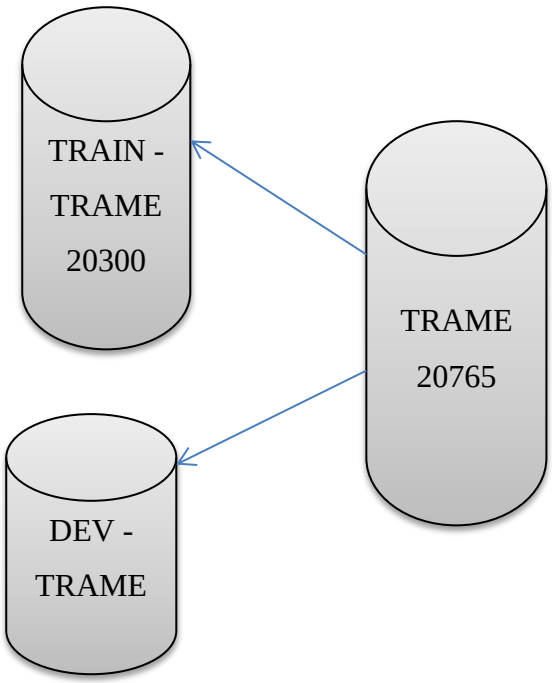

Figure 10– Nouvelle répartition du corpus TRAME

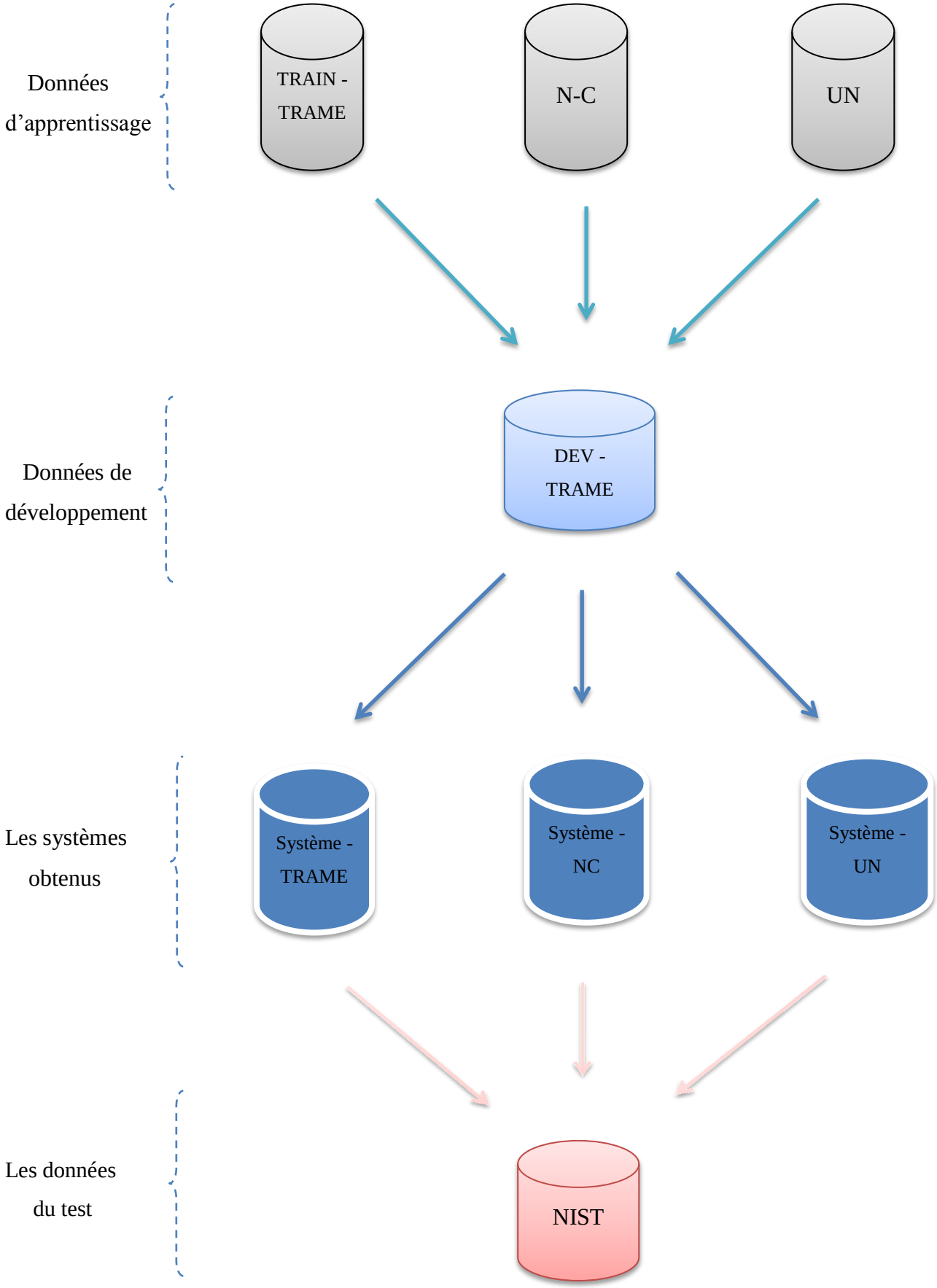

Figure 11–Représentation des données et des systèmes

Pour donner une idée plus précise, le tableau suivant contient plus de détails sur l'ensemble des données utilisées:

		TRAINs			*DEV*	*TEST*
		TRAME-TRAIN	*N-C*	*UN*	*TRAME-DEV*	*NIST*
lignes	*AR* / *FR*	20 300	90 753	7 402 560	459	606
mots	*AR*	542 631	2 180 814	174 978 885	14 231	15 850
	FR	753 208	2 372 649	236 464 034	19 870	23 114
mots différents	*AR*	56 281	147 783	2 174 502	5 931	7 176
	FR	43 902	117 744	869 421	5 071	6 056

Tableau 22– Les tailles des différentes parties du corpus en nombre de phrases

Malgré le fait que la construction de ces corpus soit basée sur la post-édition humaine, ils contiennent tous des données « bruitées » qui ne sont pas appropriées à la création des systèmes de traduction probabilistes. Cela fait du prétraitement une tâche indispensable.

Les données bruitées contiennent généralement des balises ou bien du code, et/ou parfois des commentaires écrites par les post-éditeurs (feedbacks). Par conséquent nous avons commencé par la suppression des balises, des bouts de codes ainsi que des commentaires à l'aide d'expressions régulières. D'autres étapes de préparation et de nettoyage sont également appliquées dans le but d'améliorer la qualité des données comme :

- La correction des caractères mal-encodés, souvent consernant les accents.
- La normalisation de la ponctuation et des caractères spéciaux, i.e. de tout ce qui ne fait pas partie de l'alphabet et des chiffres (points, virgules, guillemets, apostrophes, tirets, crochets, etc.). Nous avons utilisé pour cette tache le script Perl *normalize-punctuation.perl* [7] qui est fourni avec la suite d'outils associée au projet Moses.
- Conversion de toutes les données en UTF-8.

[7] http://www.statmt.org/wmt11/normalize-punctuation.perl

- Tokenisation du côté français : nous avons utilisé le script *tokenizer.perl* [8] qui fait partie de la boîte à outils du projet Moses.
- Transformation de toutes les données en minuscule, ce qui permet de garder l'exactitude de nos modèles résultants, car par exemple, notre modèle attribuera la même probabilité à « Jour » et « jour », où la première forme se trouve au début d'une phrase, tandis que la deuxième se trouve au milieu d'une autre phrase. Pour réaliser cette tâche nous avons utilisé le script de Moses *lowercase.perl* [9].
  - ! Remarque : Cette tâche doit toujours être réalisée après la tokenisation, étant donné que le script utilisé à la tokénisation du côté français se sert de la casse pour identifier les abréviations.
- Suppression des phrases plus longues que 100 mots.

Nos trois nouveaux corpus sont désormais prétraités et nettoyés. L'étape suivante sera de préparer les sous-corpus d'apprentissage, de développement (dev) et d'évaluation (test).

[8] http://www.statmt.org/wmt08/scripts.tgz (scripts/tokenizer.perl)

[9] http://www.statmt.org/wmt08/scripts.tgz (scripts/lowercase.perl)

2 Chapitre 2 - Normalisation du Hamza

2.1 Introduction

On a vu dans le chapitre précédent que le *"hamza"* correspond à un phénomène prosodique, qui ne s'introduit qu'à la prononciation, et de ce fait nous avons eu l'idée de normaliser cette lettre/diacritique dans le corpus TRAME, en rendant chaque أ ou إ ou آ en ا. Les expérimentations effectuées sur corpus TRAME ont montré que la normalisation de la lettre *"hamza"* améliore les performances du système de traduction automatique obtenu. Dans cette partie, nous allons voir l'effet de cette préparation sur d'autres corpus de tailles plus importantes.

2.2 Expérimentations

La normalisation du *"hamza"* a réduit la taille du vocabulaire dans le corpus TRAME, et comme rapporté dans le tableau ci-dessous, elle a eu le même effet sur les différentes nouvelles données :

		TRAINs			***DEV***	***TEST***
		TRAME-TRAIN	***NC***	***UN***	***TRAME-DEV***	***NIST***
nombre de vocabulaire	***Données en Arabe avant normalisation***	57 524	147 783	2 174 502	5 931	7 176
	Données en Arabe après normalisation	56 431	146 329	2 151 748	5 896	6 984
	Données en Français	43 902	117 744	869 421	5 071	6 056

Tableau 23 – La taille du vocabulaire de différentes données

Par la suite, nous avons créé trois systèmes de traduction probabilistes, entrainés respectivement sur **TRAIN-TRAME**, **N-C**, et **UN.** Ces trois systèmes sont développés et testés sur les mêmes données : **TRAME-DEV** et **NIST.**

Les résultats obtenus sont présentés dans le tableau 3 :

	HAMZA non normalisés				***HAMZA normalisés***			
	Test (NIST)			*DEV(TRAME)*	*Test (NIST)*			*DEV(TRAME)*
	BLEU	*TER*	*OOV*	*BLEU*	*BLEU*	*TER*	*OOV*	*BLEU*
TRAIN : TRAME	25,33	0,5855	3,72%	25,04	**26,39**	**0,5742**	**3,64%**	**25,04**
TRAIN : N-C	21,6	0,632	3,34%	18,42	**21,9**	**0,6342**	**3,20%**	**18,483**
TRAIN : UN	29,18	0,5422	0,53%	23,39	**29,51**	**0,54**	**0,52%**	**23,426**

Tableau 24 – Les scores de chaque système avant et après la normalisation

A partir de ces résultats, on peut constater que la normalisation du *"hamza"* du côté arabe réduit considérablement le vocabulaire, mais produit en même temps plusieurs nouvelles séquences de mots car on aura plusieurs traductions possibles en langue cible pour une seule phrase en arabe, ce qui a amélioré, même légèrement, les performances de nos systèmes de traduction.

Cette amélioration est présente dans tous les scores : OOV, TER et BLEU, que ce soit au niveau de **Test** ou de **DEV**.

3 Chapitre 3 - Adaptation de modèle de langage

3.1 Introduction

Dans les expérimentations précédentes, on a utilisé un modèle de langage basé sur notre grand corpus (**UN**), et comme c'est un corpus construit par les documents fournis par les Nations Unies, on peut dire qu'il représente des données d'apprentissages hors-domaine et il n'est pas approprié pour le domaine de notre corpus de développement (corpus référence) le corpus TRAME.

On sait que « les performances de traduction sont en relation directe avec l'homogénéité des données d'apprentissage avec le domaine d'application »[10], l'idée sera alors d'adapter le modèle de langage utilisé « hors domaine » avec le corpus **TRAME**.

L'adaptation consistera à combiner la grande quantité de données « hors-domaine » (corpus **UN**) d'une taille approximative de 7,4 M de phrases, avec la petite quantité de données d'apprentissage « du domaine » (corpus **TRAME**), afin d'améliorer les performances de nos systèmes de traduction dans le domaine 'TRAME'.

3.2 Création de modèles de langages

Nous avons commencé par la création de modèles de langage hors-domaine, basés sur toutes les données en français du corpus **UN**. Nous avons créé cinq modèles d'ordre respectivement deux, trois, quatre, cinq et six à l'aide de l'outil libre SRILM (Stolck, 2002).

Nous avons calculé leurs perplexités sur notre partie de développement **TRAME-DEV**, et le graphe ci-dessous montre les résultats.

[10] Marwen azouzi - Adaptation au domaine de la traduction automatique statistique

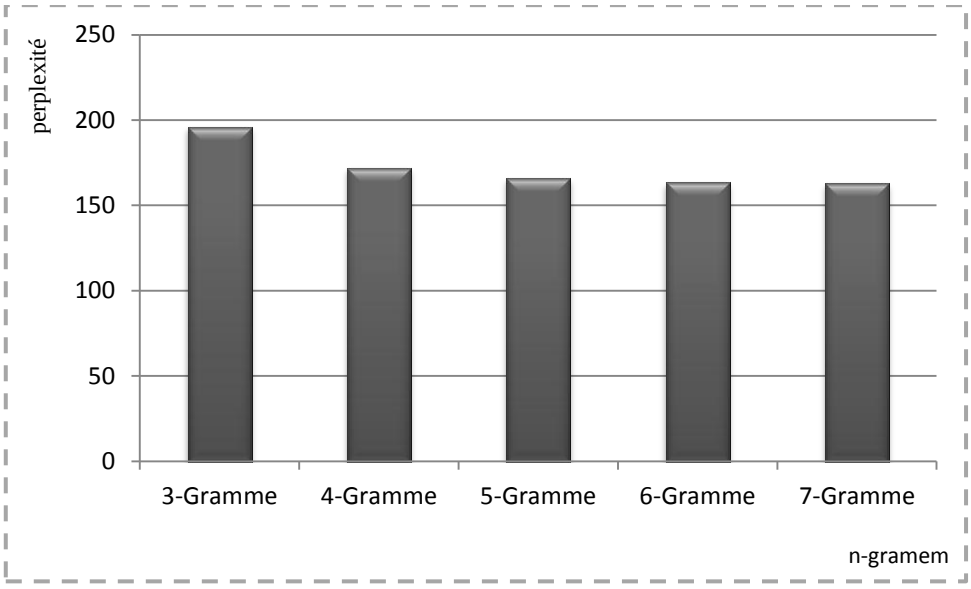

Figure 12- La perplexité calculée de modèle créé sur UN

Par la suite, nous avons créé les modèles de langage « du domaine » qui sont calculés sur le côté français du corpus **TRAME**. La perplexité de ces modèles est représentée ci-dessous.

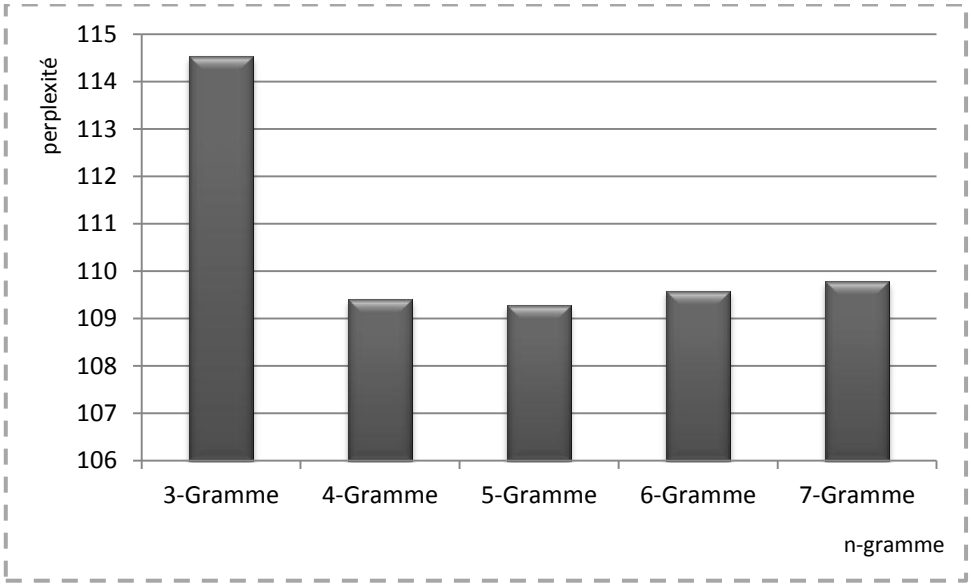

Figure 13- Les perplexités calculées des modèles créés sur TRAME

On constate ici que le modèle « du domaine » à base de 5-gramme semble le plus efficace.

3.3 Vers un modèle interpolé

Le modèle de langage permet d'estimer la probabilité qu'une séquence de mots soit correcte dans la langue cible, c'est pourquoi il aide le système de traduction probabiliste à prendre la bonne décision au moment de traduction. Lorsqu'on est dans le cas où un mot de la langue source possède plusieurs traductions possibles en langue cible, le modèle de traduction (généraliste) va chercher la traduction la plus fréquente dans les données d'apprentissage. Alors qu'en réalité, dans un contexte spécifique, la traduction de ce mot pourrait être différente. C'est donc, à ce stade-là que le modèle de langage spécifique intervient et donne des scores (probabilités) plus élevées aux traductions les plus convenables dans le contexte spécifique.

Dans notre cas, faute d'une taille suffisante, le Corpus TRAME ne permet pas de construire un modèle de langage capable de produire des traductions linguistiquement performantes. C'est pourquoi, on est obligé d'intégrer notre grand corpus UN « hors-domaine » qui portera plus d'informations linguistiques (plus de n-grammes). En même temps, il ne faut pas que ces données « hors-domaine » dominent celles du corpus TRAME.

Dans cette partie, nous allons donc essayer de combiner nos deux modèles de langage issus des deux corpus en-domaine **TRAME** et hors-domaine **UN** afin d'améliorer les performances de nos système de traduction probabilistes.

3.3.1 Interpolation linéaire

C'est une approche permettant de construire un modèle de langage à partir d'autres modèles en affectant des pondérations différentes à chacun de ceux-ci.

Dans notre cas, nous appliquons cette approche grâce à l'outil SRILM (Stolck, 2002), en lui donnant en entrée les modèles créés précédemment, avec un poids de priorité 'lambda' moins élevé pour le modèle « hors-domaine ».

Le tableau ci contre, montre les perplexités de différents modèles construits :

	Perplexité		
	LM-UN	*LM-TRAME*	*INTERPOLATION*
3-Gramme	195,95	114,53	76,25
4-Gramme	171,93	109,21	70,55
5-Gramme	165,72	109,27	**68,53**
6-Gramme	163,52	109,58	78,67
7-Gramme	162,71	109,78	98,17

Tableau 25- Les perplexités des modèles créés par UN, TRAME et leur interpolation

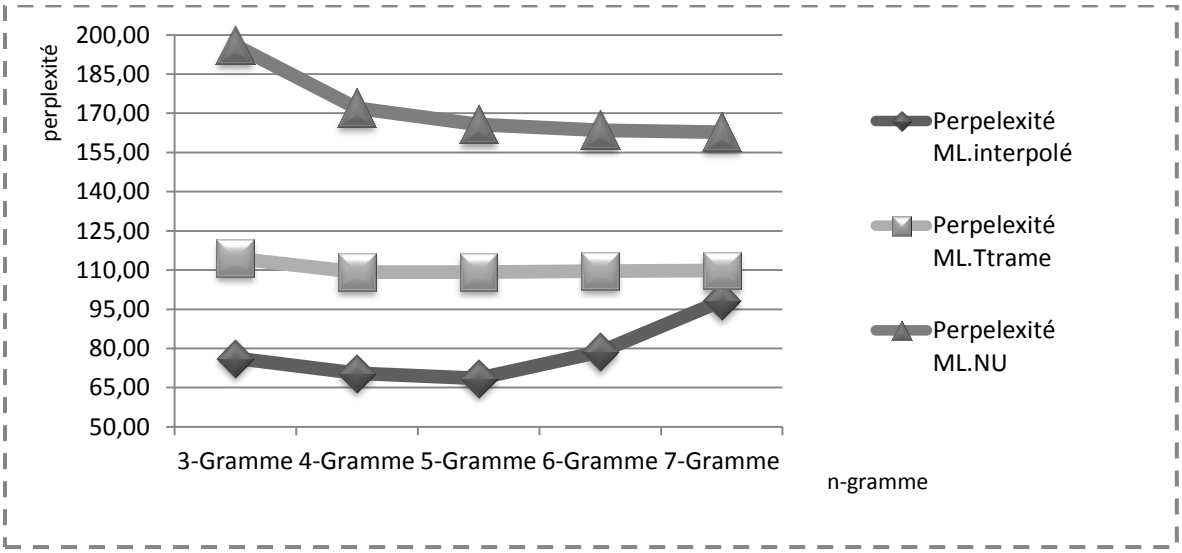

Figure 14- Représentation de l'évolution des perplexités des modèles selon l'ordre (n-gramme)

On note que le modèle de langage ayant la perplexité minimale : 68,53 est obtenue avec l'interpolation de deux modèles de langage d'ordres 4.

3.4 Influence sur les systèmes de traduction

Une fois, que nous avons obtenu un modèle de langage garantissant la fluidité, en produisant des traductions linguistiquement performantes, et spécifique à notre corpus TRAME, nous avons étudié l'influence de ce nouveau modèle de langage sur les systèmes de traduction probabiliste créés précédemment.

Nous avons recréé les trois systèmes, entrainés respectivement sur **TRAIN-TRAME**, **N-C**, et **UN**, et développés et testés sur: **TRAME-DEV** et **NIST**, en utilisant le nouveau modèle de langage : **ML.adapté.**

Une comparaison des résultats obtenus est présentée dans le tableau suivant :

	HAMZA non Normalisés				*HAMZA Normalisés*			
	Test (NIST)			*DEV(TRAME)*	*Test (NIST)*			*DEV(TRAME)*
	BLEU	***TER***	***OOV***	***BLEU***	***BLEU***	***TER***	***OOV***	***BLEU***
TRAIN : TRAME *M-L : UN*	25,33	0,59	0,37	25,04	26,39	0,57	0,36	25,04
TRAIN : TRAME *M-L : ML.adapté*	26,31	0,57	0,37	30,79	26,13	0,58	0,36	30,67
TRAIN : N-C *M-L : UN*	21,60	0,63	0,33	18,42	21,90	0,63	0,32	18,48
TRAIN : N-C *M-L : ML.adapté*	22,13	0,63	0,33	22,24	21,74	6,41	0,32	21,99
TRAIN : UN *M-L : UN*	29,18	0,54	0,53	23,39	29,51	0,54	0,52	23,43
TRAIN : UN *M-L : ML.adapté*	**29,74**	0,54	0,53	**32,29**	**29,83**	0,54	0,52	**32,17**

Tableau 26- Evolution des scores de systèmes avec ML.adapté

Il semble utile d'adapter notre modèle de langage au corpus référence, car on note un gain en score BLEU très appréciable, de plus de 5 points pour le système créé par

corpus TRAME, une amélioration de 3,51 pour le système de N-C, et 9 points de gain pour le système de UN.

Cette amélioration importante est due à la probabilité élevée donnée aux bi-textes du domaine par rapport aux nombreux bi-textes généralistes.

Ici, on peut conclure qu'avec un corpus de très grande taille utilisé pour l'apprentissage du modèle de traduction, et un corpus plus petit mais plus spécialisé on peut avoir une bonne adaptation du modèle de langage au domaine, qui aura un impact avantageux sur le système de traduction probabiliste.

Conclusion

Dans cette partie nous avons étudié l'impact de la normalisation de *"hamza"* sur les performances d'un système de traduction probabiliste développé sur des corpus de taille importantes, et on a vu que cette amélioration a toujours un effet positif.

Par conséquent, nous allons considérer dans les prochaines expérimentations, que la normalisation de *"hamza"* dans nos données sera un prétraitement systématique avant la création de nos systèmes de traduction probabiliste.

Par ailleurs, nous avons appliqué une approche qui utilise deux corpus monolingues en langue cible pour créer un modèle de langage adapté.

Cette technique nous a permis d'obtenir des améliorations considérables du score BLEU et TER dans tous nos systèmes de traduction.

IV. Partie 4

–

Création d'un segmenteur de l'arabe pour un système de traduction arabe/français

Introduction

Comme la plupart des processus de traitement automatique de texte s'appuient sur le niveau du mot, le premier enjeu est donc d'avoir un système capable de reconnaître et d'extraire les mots. La tâche de ce système consiste à découper une phrase en ses plus petites unités linguistiques (atomes). Mais la définition des ces atomes est souvent ambiguë et dépend du contexte de l'application cible ainsi que de la langue traitée. Qu'est ce qu'un mot ? Comment définit-on ses frontières ? La plupart des systèmes de tokénisation qui répondent à ces questions sont basés sur des approches statistiques ou linguistiques.

Les systèmes linguistiques, à base de règles, sont généralement développés sur un corpus donné et pour un domaine spécifique, et dans ce cas ces systèmes manquent de portabilité et de robustesse.

En revanche les systèmes statistiques reposent généralement sur l'approche de 'sac de mots' pour représenter les tokens. Mais si cette approche donne des bonnes performances en termes de reconnaissance de tokens, elle ne permet pas de segmenter précisément les frontières des tokens. Et par conséquent, ces systèmes peuvent produire des mauvais résultats si on les applique à des corpus hétérogènes.

Récemment, des méthodes statistiques à base d'apprentissage supervisé ont été proposées pour la segmentation en mots. Ces méthodes sont utiles pour les langues peu dotées où l'on ne dispose pas d'un algorithme performant purement à base de règles. Par contre, ces méthodes nécessitent un corpus d'entrainement étiqueté manuellement au niveau des mots, et par conséquent leurs performances sont liées principalement à la qualité et la taille du corpus d'apprentissage.

Etant donné que MADA+TOKAN n'est pas utilisable dans des projets commerciaux, dans cette partie nous proposons de créer un système de tokénisation de textes arabes basé sur l'approche d'apprentissage supervisés, où on cherche à répondre aux questions posés précédemment, et à produire automatiquement les règles de segmentation.

Le corpus d'apprentissage de notre système sera segmenté et étiqueté par MADA+TOKAN qui dispose d'un bon score (95%)[11] de segmentation.

La suite de cette partie est répartie comme suit : nous expliquons en premier lieu l'approche proposée pour l'apprentissage d'un segmenteur de textes arabes, en deuxième lieu nous présentons les segmenteurs créés à base de cette approche, par la suite nous

[11] http://www1.ccls.columbia.edu/MADA/

évaluons le meilleur segmenteur obtenu dans des systèmes de traduction probabiliste, et nous comparons les scores avec ceux de MADA+TOKAN.

1 Chapitre 1 - Segmentation avec Maxent

1.1 Introduction à l'approche

De nombreuses tâches en TAL peuvent être considérées comme des problèmes de classification statistique, consistant à estimer la probabilité d'un élément X dans un contexte C. Cette probabilité peut être fiable lorsqu'on la calcule et on estime avec un corpus de grande taille, mais elle n'est pas toujours utilisable en dehors de ce corpus.
Le problème est donc de trouver une méthode pour calculer d'une façon fiable la probabilité p(X, C).
D'après le principe de l'entropie maximale (MAXENT) [Jaynes, 1957, Good, 1963], la distribution correcte de la probabilité p(X, C) est celle qui maximise l'entropie soumise à des contraintes, qui représentent des «preuves».
Autrement dit, avec un modèle statistique, dans le cas de la caractérisation des événements inconnus, nous devrions toujours choisir celui qui a une entropie maximale.
[Jaynes, 1957][12]

> Afin de faire des déductions sur la base d'informations partielles, nous devons utiliser cette distribution de probabilité qui a une entropie maximale pour tout ce qui est connu, C'est la seul méthode non biaisée qu'on peut faire. L'utilisation de tout autre méthode revient à l'hypothèse arbitraire de l'information, qui est, par hypothèse, nous ne l'avons pas.

De façon plus explicite, si *A* et *B* qui représentent respectivement l'ensemble des éléments possibles, et *E* l'ensemble de contextes possibles (de *C*), *P* maximise l'entropie :

$$H(P)= -\sum_{x\,\varepsilon}\ p(x)\ log\ p(x)$$

Équation 10

Avec x = (a, b) a ∈ A, b ∈ B, and E = A * B

12

A partir des années 1997, Maxent a été appliquée avec succès dans les domaines de la vision par ordinateur, de la physique spatiale, du traitement automatique du langage naturel et dans de nombreux autres domaines, puisque l'ordinateur est devenu assez puissant pour traiter des problèmes statistiques complexes comme Maxent.

1.1.1 Modélisation de problème

Maximiser l'entropie est le faite de minimiser le risque, en :

- Modélisant tout ce qui est connu.
- Ne supposant rien sur tout ce qui est inconnu : choisir les distributions les plus uniformes revient à choisir celles qui ont une entropie maximale.

La façon de représenter les contraintes dans cette modélisation est de considérer les faits utiles comme paramètres et d'imposer des contraintes sur les valeurs de ces paramètres.

1.2 Implémentation : OpenNLP

OpenNLP est une boîte à outils de traitement automatique de texte en langue naturel basée sur l'apprentissage automatique. elle traite les tâches les plus courantes en TAL, telles que la segmentation, le balisage de discours, l'extraction d'entités nommées, etc. L'apprentissage de cet outil est basé sur la théorie de l'entropie maximale.

1.3 Création du modèle

Pour créer un modèle d'une application de classification, on doit sélectionner les paramètres qui seront utiles dans la prise de décision, tout en tenant compte du fait que la représentation théorique de ces paramètres n'est pas identique à celle de l'implémentation.
En pratique, nous prenons l'exemple d'un mot dans une phrase sur laquelle on veut appliquer notre tâche de segmentation basée sur Maxent :

طالب الرئيس السلطات السعودية بتسليمه بن علي

Le président a demandé aux autorités saoudiennes l'extraction de Ben Ali.

Si on cherche à savoir si le mot الرئيس(le président) doit subir une tokénisation ou pas, les paramètres peuvent être définis comme suit : [previous = طالب], [current =الرئيس], [next = السلطات], et on peut dire aussi que le mot رئيس a été défini avant comme un nom.

On suppose qu'on possède un modèle de segmentation appris avec *MaxentModel*. Pour demander la segmentation de cette phrase, on envoie une String[] avec tous les paramètres possibles (par exemple ceux qui sont décrits ci-dessus) au modèle en appelant la méthode :

Public double[] eval(String[] context)

Le tableau de double[] qu'on aura en retour contiendra les probabilités des différents résultats que le modèle a assignées.

Si on veut récupérer le résultat le plus probable on appelle la fonction :

Public String getBestOutcome(double[] outcomes)

1.4 Apprentissage d'un modèle

Pour entraîner un modèle, on doit posséder un ensemble de données ayant la forme convenable pour l'apprentissage, dans notre cas, les données doivent être sous la forme suivante:

- Une phrase par ligne.
- Les tokens séparés par l'étiquette <split>.

Format initial	طالب الرئيس السلطات السعودية بتسليمه بن علي
Format d'entré pour apprentissage du modèle	طالب ال<split>رئيس ال<split>سلطات ال<split>سعودية بتسليم<split>ه بن علي
Traduction	Le <split> président a demandé aux <split> autorités saoudiennes l'extraction de Ben Ali.

Tableau 27 – Exemple de préparation de données

Une fois que notre corpus d'apprentissage est prêt, on peut extraire les évènements, qui consistent en l'ensemble des résultats (outcomes) avec leurs contextes.

Exemple :

Outcome : O

Contexte : [previous=طالب], [current =الرئيس], [next = السلطات]

Où *O* est l'objet qui aura la décision de tokénisation de la phrase passée en entrée.

Open NLP.maxent possède une implémentation de l'algorithme GIS (*Generalized Iterative Scaling*) qui cherche la famille exponentielle d'une solution d'entropie maximale, par appel à la méthode :

Public static Maxent trainModel(DataIndexer di, int iterations)

DataIndexer est un objet abstrait qui contient et manipule l'ensemble des événements sur lesquels on fera les itérations. L'entier iterations est le nombre de parcours qui seront appliqués sur les données afin d'extraire les paramètres du modèle.

2 Chapitre 2 - Expérimentations

2.1 Apprentissage du modèle

Les performances d'un système de tokénisation entrainé avec Maxent dépendent essentiellement de la quantité de données et de la qualité de l'annotation des tokens. La meilleure annotation qu'on peut avoir est l'annotation manuelle où, comme notre cas, les balises <split> entre les tokens sont ajoutées par des humains.

Notre système sera entrainé sur des parties du côté arabe du corpus UN qui contiennent en totalité environ 300 million de mots.

Ce corpus a subi une normalisation de ponctuation et du *"hamza"* et une suppression des diacritiques.

Etant donnée l'ampleur de l'annotation manuelle d'un corpus d'une telle taille, et de l'efficacité du toolkit MADA+TOKAN, nous nous somme basé sur ce dernier pour traiter la tâche d'annotation. Après, nous avons amélioré cette annotation en ajoutant les balises <split> qui séparent les tokens et les caractères spéciaux (ponctuation, parenthèse, chiffre…).

2.2 Premières expérimentations

Afin d'étudier d'une façon concrète l'influence de la taille de corpus d'apprentissage sur la précision du système créé, nous avons créé dix systèmes entrainés sur différentes tailles de corpus UN.

Nous avons segmenté un texte de 459 phrases (TRAME-DEV-2H) avec chaque système et nous avons évalué cette segmentation à l'aide de l'outil *opennlp.TokenizerMEEvaluator*, auquel nous donnons comme modèle de référence le modèle de MADA+TOKAN.

Cet outil nous retourne les métriques suivantes :

- Précision : le nombre de séparations correcte qu'un mot a subi rapporté au nombre de séparations proposés par le modèle.
- Rappel : le nombre de séparations correcte qu'un mot a subi au regard du nombre de séparations pertinentes que possède la référence.
- F-mesure : la moyenne harmonique de la précision et du rappel : $F=\frac{2.(précison\ .\ rappel)}{(précison + rappel)}$

Équation 10

Les mesures d'évaluation précision, rappel et F-mesure des systèmes créés sont présentées dans le tableau suivant :

Taille de corpus d'apprentissage en ligne	*rappel*	*précision*	*F-mesure*
100000	0,9691	0,9404	0,9545
500000	0,9752	0,9526	0,9638
1M	0,9749	0,9517	0,9632
2M	0,9765	0,9549	0,9656
3M	0,9859	0,9727	0,9792
4M	0,9897	0,9801	0,9849
5M	0,9905	0,9815	0,9860
6M	0,9912	0,9829	0,9870
7M	0,9916	0,9837	0,9877
7,5M	**0,9917**	**0,9839**	**0,9878**

Tableau 28– Evaluation de segmenteurs

En regardant les textes issus de ces segmenteurs et leurs scores, on constate que la taille du corpus d'apprentissage est un facteur important dans les performances des systèmes de segmentation, vu que les mesures d'évaluation s'améliorent à chaque fois qu'on augmente la taille de notre corpus.

En général plus on possède de données pour l'estimation des paramètres du modèle de segmentation, plus on se rapproche de la bonne probabilité de tokénisation, ce qui conduira bien évidement à une segmentation meilleure.

Pour cette raison on n'utilisera que le tokéniseur entrainé sur la totalité du corpus UN dans toutes les expérimentations à venir.

2.3 Segmentation partielle dans SMT

Comme nous l'avons vu dans la partie 2, la segmentation des données a une influence directe sur les performances d'un système de traduction probabiliste.

Généralement les trois composantes essentielles d'un système de traduction probabiliste (DEV-TEST-TRAIN) sont segmentés par le même tokéniseur et avec les mêmes paramètres afin d'assurer la cohérence et éviter toute confusion de système pendant l'apprentissage de tables de traduction.

Dans cette partie, nous allons étudier l'évolution des performances de nos trois systèmes de traductions (C-TRAME, C-UN, C-NC) lorsqu'on segmente la partie de TRAIN par MADA+TOKAN et les parties DEV et TEST avec le meilleur segmenteur obtenu dans les expérimentations précédentes, vu les contraintes posées sur l'utilisation de MADA+TOKAN qui interdisent son utilisation dans des projets commerciaux.

Le tableau ci-dessous montre l'évolution du nombre de mots dans les trois parties en passant d'une segmentation par MADA+TOKAN a une tokénisation par notre Système O.NLP.

	Nombre de tokens	
	TEST	DEV
Avant Tokénisation	15 850	14 231
MADA	**25 051**	**21 085**
O.NLP	**24 886**	**21 210**

Tableau 29 - Nombre de tokens dans chaque partie

Ces résultats révèlent les mesures rappel et précision, obtenus précédemment qui montrent que malgré un apprentissage sur la totalité du corpus UN, notre segmenteur O.NLP fournit toujours une tokénisation légèrement différente de celle de MADA+TOKAN. En effet en comparant la segmentation de TEST et DEV par MADA à leurs segmentation par O.NLP, on note toujours une différence de ±0,5% mot.

Quelle est l'influence de cette légère différence sur un système de traduction probabiliste ?

Une mesure de cette influence est donnée dans le tableau 4, à travers les variations de score :

		Test : *NIST*			DEV : *TRAME-DEV*
		BLEU	**TER**	**OOV**	**BLEU**
TRAIN : *TRAME-TRAIN*	**MADA**	26,13	0,5757	3,64%	30,669
	O.NLP	23,87	0,79	4,53%	26,15
TRAIN : *N - C*	**MADA**	21,74	0,641	3,20%	21,99
	O.NLP	20,29	0,83	4,14%	19,42
TRAIN : *UN*	**MADA**	29,83	0,539	0,52%	32,17
	O.NLP	27,4	0,758	1,87%	27,91

Tableau 30 - Evaluation de différents systèmes avec MADA et O.NLP

Les résultats de ces expérimentations montrent que la segmentation des parties Test et DEV par un segmenteur différent de celui de la partie TRAIN dégrade considérablement les performances d'un système de traduction probabiliste. En effet on observe au niveau du score BLEU une baisse de 3 points dans S-NC et de 4 points dans S-TRAME et S-UN. Cette baisse est illustrée dans la figure 1 :

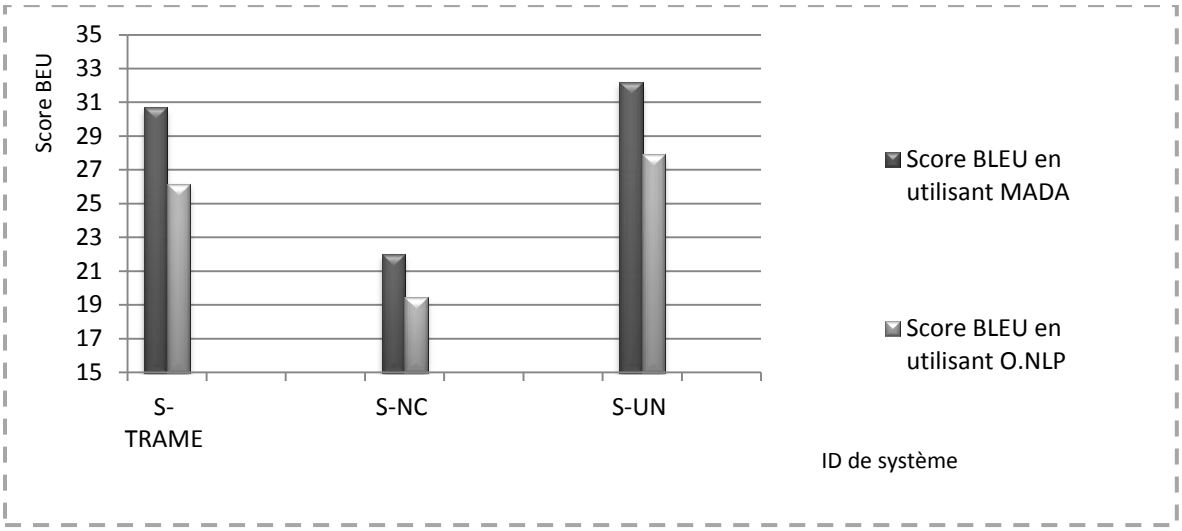

Figure 15- Scores BLEU dans MADA et O.NLP

Il en est de même pour les scores TER et OOV, qui ont connu une augmentation, ce qui montre que nos systèmes sont devenus moins performants.

En examinant les textes issus de deux systèmes de segmentation on remarque que la différence majeure se situe dans le traitement de l'article : ال, article qui doit être tokénisé lorsqu'il s'agit en français de l'article de définition le/la, exemple :

Mot en Arabe	Traduction en Français
الرئيس	Le président

Tableau 31– Exemple de segmentation obligatoire de ال

Alors qu'il ne doit pas subir une séparation d'un lemme dont il fait partie, comme par exemple où l'article ال ne représente pas un article :

Mot en Arabe	Traduction en Français
الى	à
ى ـال	.

Tableau 32- Exemple de segmentation interdite de ال

MADA+TOKAN effectue une analyse morphologique et syntaxique sur les phrases et les mots avant de faire la tokénisation, c'est pourquoi il arrive la plupart des fois à faire

le bon traitement, alors qu'O.NLP se base seulement sur la maximisation de l'entropie et il échoue dans certains cas.

2.4 Segmentation complète dans SMT

Nous avons comparé dans les expérimentations précédentes 2 groupes de système de traduction, un premier groupe dans lequel les systèmes sont basés sur des données segmentées avec l'outil MADA+TOKAN, et un deuxième groupe dans lequel les données sont segmentés par les deux systèmes : MADA+TOKAN et O. NLP :

- Groupe 1 : **TRAIN** -> MADA+TOKAN
  DEV -> MADA+TOKAN
  TEST -> MADA+TOKAN

- Groupe 2 : **TRAIN** -> MADA+TOKAN
  DEV -> O.NLP
  TEST -> O.NLP

Et nous avons vu que les systèmes du Groupe 1 sont plus performants que ceux du groupe 2.

Dans cette partie, nous allons faire des expérimentations afin d'étudier les performances de nos trois systèmes de traduction (S-TRAME, S-UN, S-NC) lorsqu'on segmente les trois parties TRAIN, DEV, Test avec O.NLP.
Ces expérimentations vont nous donner une idée de l'influence de l'utilisation de deux segmenteurs différents (au niveau de l'approche utilisée) dans un système de traduction.
De la sorte, on aura une vision plus claire de l'efficacité de notre système de segmentation.
La segmentation de différentes parties TRAIN avec l'outil O.NLP a fait évoluer leurs tailles en nombre de tokens comme suit :

	Nombre de tokens		
	TRAIN-TRAME	*TRAIN-NC*	*TRAIN-UN*
avant tokénisation	54 2631	2 180 814	174 978 885
MADA	824 744	3 414 409	296 222 564
O.NLP	825 206	3 434 236	297 805 480

Tableau 33– Nombre de tokens dans chaque corpus

Nous rapportons dans le tableau ci-dessous les résultats de l'ensemble des systèmes créés à l'aide de métriques d'évaluation BLEU TER et OOV :

		Test (NIST)			*DEV(TRAME)*
		BLEU	***TER***	***OOV***	***BLEU***
TRAIN : TRAME-TRAIN	***MADA***	26,13	0,5757	3,64%	30,669
	O.NLP	**25,27**	**0,79**	**4,53%**	**26,15**
TRAIN : N - C	***MADA***	21,74	0,641	3,20%	21,99
	O.NLP	**20,45**	**0,83**	**3,48%**	**20,91**
TRAIN : UN	***MADA***	29,83	0,539	0,52%	32,17
	O.NLP	**29,97**	**0,743**	**0,71%**	**31,12**

Tableau 34– Evaluation de différents systèmes avec MADA et O.NLP

Les scores d'évaluation de nos nouveaux systèmes sont très proches de ceux obtenus avec les systèmes segmentés par MADA+TOKAN.

En effet, lorsqu'on teste notre système sur le corpus NIST, on note une baisse de 0,86 en score BLEU dans S-TRAME, 1,29 en S-NC et une augmentation de ce score par 0,14 en C-UN.

Pour les scores TER et OOV on note une petite augmentation dans tous les systèmes.

Le graphe ci-dessous illustre la comparaison des deux systèmes :

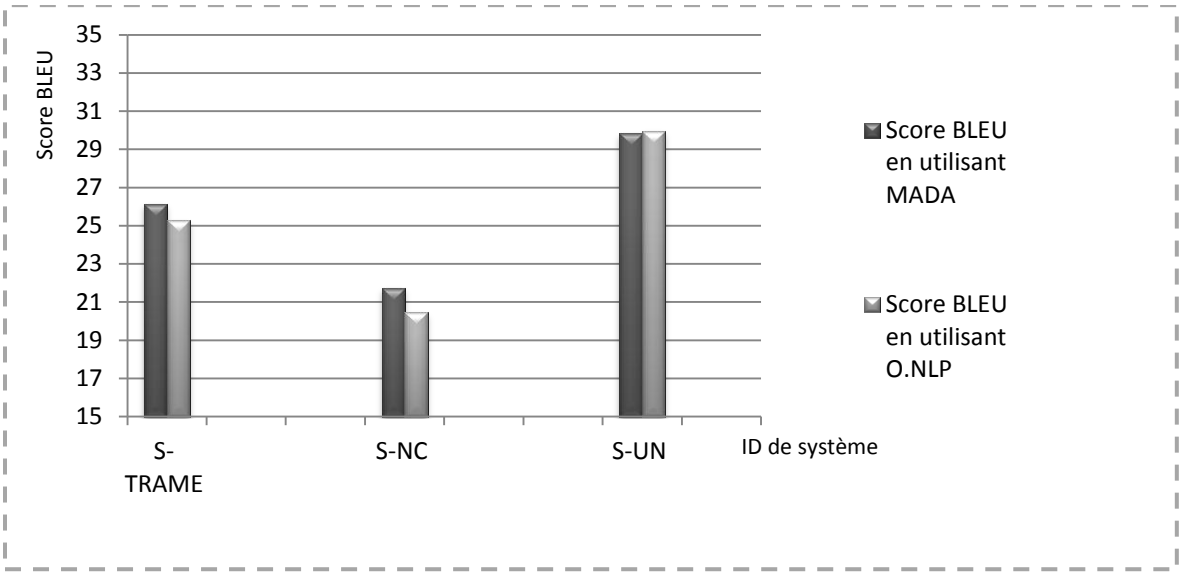

Figure 16– Scores BLEU avec MADA et O.NLP

Conclusion

Dans ce chapitre, nous avons présenté notre approche pour entrainer un segmenteur de textes arabe, en commençant par la présentation de performances de l'existant MADA+TOKAN. Etant donnée l'impossibilité d'utiliser ce dernier dans des projets commerciaux, nous avons décidé de réaliser notre propre segmenteur entrainé sur les résultats de MADA+TOKAN.

En évaluant ce nouveau segmenteur, nous avons trouvé qu'il donne des performances correctes pour pouvoir créer un système de traduction probabiliste de l'arabe vers le français.

Les analyses qualitatives et quantitatives des systèmes de traduction basé sur notre segmenteur ont montré que nous avons bien réussi notre projet de reengineering de l'outil MADA+TOKAN.

Conclusion

Bilan d'étude

L'apparition de modèles statistiques a créé une véritable révolution dans le domaine de la traduction automatique. En se basant sur ces modèles, il est devenu possible, à partir de corpus parallèles alignés, de calculer automatiquement les correspondances significatives les plus probables entre deux langues, et d'élaborer par cela un système de traduction probabiliste. Ce travail a porté sur le couple de langues arabe/français.

En traduction automatique, il y a deux axes principaux de développement : les données d'apprentissage et le calcul. Plusieurs travaux de recherche ont visé à l'amélioration des performances de la traduction par l'optimisation du deuxième axe. Ces recherches ont proposé plusieurs approches pour résoudre des problèmes de modélisation (modèles de langage, modèle de traduction) et des problèmes de décodage (choix de la meilleure traduction).

Au cours de notre stage nous avons consacré notre recherche à des améliorations portant sur le premier axe. En effet nous avons étudié plusieurs hypothèses concernant la préparation des données arabes, nous les avons appliquées sur nos systèmes de traduction, et nous avons examiné leur impact sur les performances de ces systèmes.

Les résultats des expérimentations et de nos évaluations montrent l'efficacité des prétraitements effectués sur le côté source de nos données.

Après avoir choisi la meilleure séquence de prétraitements des données arabe pour créer un traducteur probabiliste vers le français, et étant donné que MADA+TOKAN n'est pas utilisable dans des projets comme le nôtre, nous avons utilisé le principe d'entropie maximale, avec l'outil Maxent, pour construire notre propre système de tokénisation. Notre tokéniseur s'est révélé d'une efficacité très proche de celle de MADA+TOKAN.

Dans le prolongement de nos travaux, outre la validation expérimentale de nos résultats sur une autre langue cible, il serait intéressant de faire une recherche détaillée et plus approfondie sur les principaux raisons de ces résultats.

Concernant les diacritiques, on sait qu'en langue arabe leur suppression diminue les variations morphologiques, et donne à plusieurs mots la forme de leurs lemmes. nous émettons l'hypothèse, que ces deux phénomènes sont la cause de l'amélioration des

performances de la traduction. Un autre point à vérifier, consiste en la réduction de la taille du vocabulaire lorsqu'on diactritise nos données. Cela devrait, théoriquement, améliorer les performances du système de traduction, ce qui contredit les résultats que nous avons obtenus. Nous faisons l'hypothèse que la taille de nos données voyellées n'étaient pas suffisante pour entrainer de manière cohérente un modèle voyellé et désambiguïsé. Il serait très intéressant de confirmer ou d'infirmer cette hypothèse lors d'une recherche spécifique.

Enfin, concernant notre système de segmentation de l'arabe entrainé sur des sorties de MADA+TOKAN, nous pensons qu'une amélioration de ses sorties pourrait rendre notre système encore plus performant : cette piste reste encore à explorer..

Bilan personnel

Ce travail de recherche intégrant aussi une dimension professionnelle grâce au stage, m'a permis de prendre connaissance et d'expérimenter un domaine très intéressant du Traitement du Langage Naturel qui est la traduction automatique probabiliste et le prétraitement des données.

Pendant ce mémoire de recherche, j'ai pu avoir un contact direct tant avec le monde professionnel qu'avec des chercheurs de haut niveau et des ingénieurs spécialisés dans les sujets du TALP (Traitement Automatique de la Langue Ecrite et de la Parole).

Finalement, cette expérience très enrichissante m'a permis d'avoir un aperçu complet du milieu, compte tenu de mon intention de poursuivre ma carrière dans la recherche en informatique.

Bibliographie

Abraham, I. & Roukos, S (2005), *A Maximum Entropy Word Aligner for Arabic-English Machine Translation* Proceeding of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing.

AFLI, H & Besacier, L (2010), *Approche mixte pour la traduction automatique statistique*. Mémoire Master 2 IDL.

Ahmed, H (2012) *Apport de la diacritisation dans l'analyse morphosyntaxique de l'arabe*.

Aljlayl, M. & Frieder, O. (November 2002). *On Arabic Search: Improving the Retrieval Effectiveness*. In 11th International Conference on Information and Knowledge Management (CIKM), Virginia (USA).

Alon, L. & Hassan, A. (2011), *The Impact of Arabic Morphological Segmentation on Broad-Scale Phrase-based SMT*

Alon, L & Denkowski, M. *The METEOR Metric for Automatic Evaluation of Machine Translation*. Machine Translation Journal. 2009

Andreas, E & Yu, C. (2010) MultiUN: *A Multilingual Corpus from United Nation Documents.*

Andreas, S. 2002. SRILM - *an Extensible Lan - guage Modeling Toolkit*. In Proceedings of the International Conference on Spoken Language Processing (ICSLP).

Andreas, Z & Ashish, V & Stephan V. 2006. *Bridging the inflection morphology gap for Arabic statistical machine translaion*.

Arun, A. & Koehn, P. (September 2007). *Online learning methods for discriminative training of phrase based statistical machine translation* ,pages 15–20.

Awdé, A. (2003). *Thèse Comparaison de deux techniques de décodage pour la traduction probabiliste*.

AZOUZI, M & Laurent Besacier (2011), *Adaptation de la traduction automatique statistique,* Mémoire Master 2 IDL.

Bahl, L. R., & Mercer, R. L. (1976). *Part of speech assignment by a statistical decision algorithm*. in IEEE International Symposium on Information Theory, Ronneby , 88-89.

Baloul, S., Alissali, M., Baudry, M., & Boula de Mareüil, P. (24-27 juin 2002*). Interface syntaxique-prosodique dans un système de synthèse de la parole à partir du texte en arabe.*

Barbara Greene, B., & Gerald Rubin, M. (1971). *Automated Grammatical Tagging of English*. Department of Linguistics, Brown University, Providence, Rhode Island.

BEESLEY, R. (2005). Xerox *Arabic Morphological Analysis and Generation Romanization, Transcription and Transliteration*.

Besacier, L. (dec 2007). *De l'utilisation d'unités sous-lexicales pour la traduction automatique de la parole*. Séminaire ATALA.

Besacier, L. (kein Datum). (Oct. 2005/Nov). *Contributions à la traduction de parole arabe dialectal / anglais*. séjour de recherche IBM Watson,.

Besacier, L., & Mahdhaoui, A. (2007). *The LIG Arabic / English Speech translation System at IWSLT07*. pp. 1-2.

Besacier, L., Benyoussef Atef and Blanchon, (October 2008). *H.. The LIG Arabic / English Speech translation System at IWSLT08*. pp.58-62. Hawii

Ben Youssef, A & Besacier, L (2008), *Méthodes Mixtes pour la Traduction Automatique Statistique*. Mémoire Master 2 IDL.

BUCKWALTER, T. (2004). *Buckwalter Arabic Morphological Analyser Version 2.0. Linguistic Data Consortium (LDC) Catalog Number LDC2004L02, ISBN 1-58563-324-0. DEBILI, F. et ACHOUR, H. (1998).* Voyellation automatique de l'arabe. Actes du Workshop on Computational Approaches To Semitic Languages, Université de Montréal.

DEBILI, F. et SOUISSI, E. (1998). *Etiquetage grammatical de l'arabe voyellé ou non*. In Proceedings of the Workshop on Computational Approaches to Semetic Languages, Stroudsburg.

DEBILI, F., ACHOUR, H. et SOUISSI, E. (2002). *La langue arabe et l'ordinateur : de l'étiquetage grammatical à la voyellation automatique*. Correspondances de l'IRMC, N°71, Tunis.

F. Och. 2003. *Minimum Error Rate Training in Statistical Machine Translation*.

Fatiha Sadat (2007), *Introduction à la Traduction Automatique (TA)*

Franz Josef Och, Hermann Ney. *A Systematic Comparison of Various Statistical Alignment Models, Computational Linguistics*.(2003).

GHOUL Dhaou, Olivier Kraif *Outils génériques pour l'étiquetage morphosyntaxique de la langue arabe : segmentation et corpus d'entraînement,* Mémoire Master 2 IDL.

HABASH, N. et OWEN, R. (2005). *Arabic Tokenization, Part-of-speech Tagging and Morphological Disambiguation in One Fell Swoop*. In Proceedings of the Conference of the American Association for Computational Linguistics, New York.

HAJIC, J., SMRZ, F., BUCKWALTER, T. et JIN, H. (2005). *Feature-Based Tagger of Approximations of Functional Arabic Morphology*. Actes de la quatrième conférence sur les Treebanks et les théories linguistiques, Université de Barcelone.

Hassan Al-Haj, Alon Lavie (2007), *The Impact of Arabic Morphological Segmentation on Broad-coverage English-to-Arabic Statistical Machine Translation.*

Holger Schwenk (2010), *Adaptation d'un Système de Traduction Automatique Statistique avec des Ressources monolingues.*

Holger, S & Jean-Baptiste, F & Senellart, J (2008), *First Steps towards a general purpose French/English Statistical Machine Translation System.*

Ibrahim Badr, Rabih Zbib, and James Glass. (2008). Segmentation for english-to-arabic statistical machine translation.

Kevin Knight and Philipp Koehn (2009), *What's New in Statistical Machine Translation.*

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. *BLEU: a Method for Automatic Evaluation of Machine Translation*. In Pro ceedings of the 40th Annual Meeting of the As sociation for Computational Linguistics.

L. Besacier, A. Ben-Youssef, H. Blanchon (2008), *The LIG Arabic / English Speech Translation System at IWSLT08.*

MAAMOURI, M., BIES, A. et BUCKWALTER, T. (2004). *The Pen Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus*. In EMAR Conference on Arabic Language Ressources and Tools, le Caire.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Lin nea Micciulla, and John Makhoul. 2006. *A Study of Translation Edit Rate with Targeted Human Annotation. In Proceedings of the 7th Conference of the Association for Machine Translation in the Americas* (AMTA-2006).

Nizar Habash and Fatiha Sadat. 2006. *Arabic Preprocessing Schemes for Statistical Machine Translation.*

Nizar Habash and Owen Rambow (2005). *Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics* (ACL'05).

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek,John D. Lafferty, Robert L. Mercer, and Paul S. Roossin (1990), *A STATISTICAL APPROACH TO MACHINE TRANSLATION.*

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Robert L. Mercer (1993) *The Mathematics of Statistical Machine Translation: Parameter Estimation.*

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, Evan Herbst, Moses: *Open Source Tool kit for Statistical Machine Translation, Annual Meeting of the Association for Computational Lin guistics (ACL),* demonstration session, Prague, Czech Republic, June 2007.

Philipp Koehn. 2004. *Statistical significance tests for machine translation evaluation. In Proceedings of the Empirical Methods in Natural Language Processing Conference (EMNLP'04)*.

Qin Gao, Stephan Vogel, "*Parallel Implementations of Word Alignment Tool*", Software Engineering, Testing, and Quality Assurance for Natural Language Processing.

Ruhi Sarikaya and Yonggang Deng 2007. *Joint Morphological-Lexical Language Modeling for Machine Translation.*

SERRANO Laurie (2011) *Modélisation d'une ontologie de domaine et des outils d'extraction de l'information associés pour l'anglais et le français.*

Thi-Ngoc-Diep DO, Laurent Besacier (2006), *Extraction de corpus parallèle pour la traduction automatique depuis et vers une langue peu dotée.*

Tim Buckwalter. (2002). Buckwalter Arabic Morphologcal Analyzer. *Linguistic Data Consortium.* (LDC2002L49).

Young-Suk Lee. 2004. *Morphological analysis for statistical machine translation*.

ANNEXES

1 Annexe 1 : Extrait de traduction d'un système diacritsé à 0%

coréens quant كرروا leurs précédentes , qui consiste à حصولهم sur des garanties de ne تهاجمهم des états-unis et de présenter secoure des aides économiques , en contrepartie de geler برامجهم nucléaires , et que l' être , il y a des mesures متزامنة pour mettre fin à la crise en ce qui concerne les américains فيصرون sur la création de la corée du nord de démanteler son programme nucléaire ومنه son programme

2 Annexe 2 : Extrait de traduction d'un système diacritsé à 12%

coréens du nord كرّروا leurs précédentes , qui consiste à حصولهم sur des garanties de ne تهاجمهم des états-unis et de leur fournir de l' aide économique , en contrepartie de geler برامجهم nucléaires et la transparence , il y a des mesures متزامنة pour mettre fin à la crise en ce qui concerne les américains فيصرّون de la création de la corée du nord , absolument démanteler son programme nucléaire

3 Annexe 3 : Extrait de traduction d'un système diacritsé à 25%

coréens du nord كرّروا leurs précédentes , qui consiste à حصولهم sur des garanties de ne تهاجمهم des états-unis et de secoure des aides économiques , en contrepartie de geler برامجهم nucléaires et la transparence , il y a des mesures متزامنة pour mettre fin à la crise . les américains فيصرّون quant à la création de la corée du nord , absolument démanteler son programme nucléaire ومنه son programme لتخصيب

4 Annexe 4 : Extrait de traduction d'un système diacritsé à 37%

les quant كرّروا مطالِبهم précédentes , qui ont fait de حصولِهِم sur des garanties de ne تهاجِمهم les états-unis et de leur fournir de l' aide اِقتِصادِيّة contre le gel برامِجِهِم nucléaires , et il y a des mesures متزامِنة pour mettre fin à la crise , alors que les américains فيصِرّون sur la création de la corée du nord absolument démanteler son programme nucléaire ومِنه programme لِتخصِيبِ الأورانيوم تنفِيهِ pyongyang , à

5 Annexe 5 : Extrait de traduction d'un système diacritsé à 50%

coréens du كرّرُوا مطالِبهُمُ précédentes , qui ont été en dressant حُصُولِهِم sur des garanties de ne تُهاجِمهُمُ les états-unis et de présenter leur des aides اِقتِصادِيّة contre le gel de برامِجِهِم nucléaires , et qu' il y aurait des mesures مُتزامِنة pour mettre fin à la crise , alors que les américains فيُصِرُّون sur la création de la corée du nord , sur le démantèlement son programme nucléaire ومِنهُ برنامجُها لِتخصِيبِ الأُورانيُوم qui تُنفِيهِ pyongyang condition que , comme je l' ai dit , de s' assurer de cela .

6 Annexe 6 : Extrait de traduction d'un système diacritsé à 62%

اَلكُورِيُّونَ nord كَرَّرُوا مَطَالِبَهُمُ السَّابِقَةَ qui , حُصُولِهِم sur des garanties de ne تُهَاجِمَهُمُ les états-unis et de présenter leur des aides اِقتِصَادِيَّة contre le gel بَرَامِجِهِم nucléaires , et qu' il y ait des mesures مُتَزَامِنَة pour mettre fin à la crise , alors que les américains فَيُصِرُّونَ sur la création de la corée du nord sur son programme nucléaire وَمِنهُ بَرنَامَجُهَا لِتَخصِيبِ الأُورَانيُوم تُنفِيهِ wort pyongyang condition de délégations de cela .

7 Annexe 7 : Extrait de traduction d'un système diacritsé à 75%

اَلكُورِيُّونَ du nord كَرَّرُوا مَطَالِبَهُمُ السَّابِقَةَ qui consiste à حُصُولِهِم sur des garanties de ne تُهَاجِمَهُمُ , les états-unis ont , et de présenter des aides اِقتِصَادِيَّة contre le gel بَرَامِجِهِم nucléaires , et ce , pour mettre fin à la crise . il y a des mesures مُتَزَامِنَة , alors que les américains فَيُصِرُّونَ sur la création de la corée du nord sur son programme nucléaire وَمِنهُ بَرنَامَجُهَا لِتَخصِيبِ الأُورَانيُوم تُنفِيهِ pyongyang , à condition qu' il de cela .

8 Annexe 8 : Extrait de traduction d'un système diacritsé à 87%

اَلكُورِيُّونَ quant مَطَالِبَهُمُ السَّابِقَةَ كَرَّرُوا qui , حُصُولِهِم sur des garanties de ne pas تُهَاجِمَهُمُ les états-unis et de présenter des aides اِقتِصَادِيَّة contournantes , en contrepartie de geler بَرَامِجِهِم nucléaires , et qu' il y ait خُطُوَاتٌ مُتَزَامِنَةٌ pour mettre fin à la crise ; d' un autre côté , les américains فَيُصِرُّونَ anniversaire de la création de la corée du nord , sur le démantèlement son programme nucléaire وَمِنهُ بَرنَامَجُهَا لِتَخصِيبِ الأُورَانيُوم تُنفِيهِ pyongyang , à condition qu' il y ait de contrôle .

9 Annexe 9 : Extrait de traduction d'un système dans lequel "HAMZA" est normalisé

Les pays institutions intérêts dans les équipes entre à بيعجبنا et entre en réalité , située présentes قدام des انو je ne peux اقبل et ne que اتصور propos de la région sous miséricorde israël .

10 Annexe 10 : Extrait de traduction d'un système dans lequel "HAMZA" est normalisé

Les pays institutions intérêts en entre à بيعجبنا et entre en réalité , située présentes قدام nous أنو moi , je ne peux أقبل et ne que أتصور propos de la région sous miséricorde d' israël .

11 Annexe 11 : Extrait de traduction du système entrainé sur UN et tokénisé par MADA

l' égypte a connu au cours des trois dernières semaines des mesures de sécurité draconiennes à la station de métro en prévision de la cible de terroristes , et a noté que ces mesures , qui comprenait le contrôle de tous les bagages de clients des stations de métro , a diminué au cours des deux derniers jours .
les dirigeants de l' opposition prétendant pakistanais musharraf à démissionner à londres le 7 , 2007 (a) , en conclusion de la conférence des chefs de l' opposition pakistanais de l' un de ses travaux à londres avec des invitations à la démission du président pakistanais pervez musharraf , et le ministre principal , le retour des ex-combattants بنازير bhutto , de nawaz sharif au pays .

12 Annexe 12 : Extrait de traduction du système entrainé sur UN et tokénisé par notre segmenteur

l' égypte a connu au cours des trois dernières semaines des mesures de sécurité draconiennes sur les stations de métro , en prévision de la cible des éléments terroristes , il a été noté que ces mesures , qui comprenaient la fouille des bagages habitués à toutes les stations de métro , a diminué au cours des deux derniers jours . les dirigeants de l' opposition pakistanaises affirment que musharraf à démissionner londres 9.7 a) en 2007 , le congrès des dirigeants de l' opposition a achevé les travaux de sa pakistanaises dimanche à londres avec des invitations à la démission du président pakistanais pervez musharraf et le retour des anciens premiers ministres : bhutto , nawaz sharif au pays .

Liste des tableaux

Liste des figures

Sommaire

MOTS-CLÉS : Traduction Automatique Probabiliste, traitement automatique de la langue arabe, Prétraitement et segmentation de données, Modèle de langage, Modèles de traduction, Maxent.

RÉSUMÉ

Le domaine du traitement automatique des langues naturelles a connu des évolutions très rapides ces dernières années, et spécialement dans la traduction automatique, c'est pourquoi les demandes en matière de traducteurs automatiques fiables augmentent sans cesse. De ce fait, nous nous sommes intéressés à ce domaine afin de concevoir un traducteur automatique de la langue arabe vers le français, basé sur un modèle probabiliste.

Les performances de traduction des systèmes probabilistes dépendent considérablement de la qualité et de la quantité des données d'apprentissage disponibles. Néanmoins la langue arabe compte encore parmi les langues dites « peu dotées », c'est pourquoi la plupart des travaux sur cette langue sont basés sur les données libre d'accès qui proviennent d'organisations internationales (ONU, etc.).

Nous présentons dans ce travail, une approche d'optimisation des performances d'un système de traduction de l'arabe. Compte tenu du manque de données et d'outils accessibles, nous avons cherché à moindre coût la meilleure combinaison de prétraitements à appliquer sur nos données en arabe pour améliorer la traduction vers le français.

KEYWORDS : Statistical Machine Translation, Preprocessing and Tokenization of data, Language Models, Translation Models, Maxent.

ABSTRACT

In recent years, Natural Language Processing has rapidly evolved, especially in the domain of Statistical Machine Translation causing the need for reliable automatic translations to skyrocket with no sign of slowing. Due to this increased need, we have taken a special interest in this domain with the goal of creating a translation machine capable of translating Arabic into French, based on statistical models.

The performance of Statistical Machine Translation relies heavily on the quality and the quantity of available training data. However, the Arabic language remains one of the languages with the fewest available resources which is why most of the available works in this language are based on open access data from international organizations, such as the U.N.

In this work, we will present our approach to optimizing the performance quality of our Arabic translator. Taking into account the lack of data and available resources, we were able to find a low-cost solution to search for the best pre-processing combinations to apply to our Arabic database in order to obtain the highest quality French translation.