MINISTERE DE L'ENSEIGNEMENT
SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE SAAD DAHLAB - BLIDA 1
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

MEMOIRE DE MASTER

En INFORMATIQUE

Spécialité **:** Traitement Automatique de la Langue

PROPOSITION D'UN SYSTEME DE TRADUCTION AUTOMATIQUE (ANGLAIS-ARABE)

Mémoire présenté par :

Mrs. BENZEGHIOUA Rafiq et SIFI Abdelhak

Mme. N.Toubaline	Président
Mme. M.Mezzi	Promoteur
Mr. M.Abbas	Encadreur
Mme.Nasri	Examinateur

Blida, le 23 juillet 2019

Remerciement

Tout d’abord, nous tenons à remercier le bon Dieu le tout Puissant de nous avoir donné la force et le courage de mener à bien ce modeste travail, également nous remercions infiniment nos parents, qui nous encouragé et aidé à arriver à ce stade de notre formation.

Nous adressons nos remerciements les plus sincères à toutes les personnes qui nous ont permis d'évoluer dans la réflexion et l'élaboration de ce travail. Plus particulièrement, nous tenons à remercier :

Mr ABBAS, encadreur de projet, pour nous avoir accordé sa confiance pour la réalisation de ce projet à distance les unes des autres, et pour nous avoir guidées tout au long de cette étude.

Mme MEZZI, notre promotrice, qui croyait en notre capacité à réaliser quelque chose de merveilleux. Sans aucun doute, cela changera le cours de nos vies pour le mieux et nous aidera dans notre éducation, nous donnant la possibilité de poursuivre et de réaliser ce rêve.

Mr Lichouri, chercheur au Centre de Recherche Scientifique et Technique pour le Développement de la Langue Arabe, auprès duquel nous avons beaucoup appris et soutenu lors du stage CRSTDLA.

Enfin, nous tenons à remercier tous ceux qui nous ont aidés et assistés durant nos études et nous exprimons toute notre gratitude à vous.

Dédicaces

Je dédie ce travail qui n’aura jamais pu voir le jour sans les soutiens indéfectibles et sans limite de mes chers parents qui ne cessent de me donner avec amour le nécessaire pour que je puisse arriver à ce que je suis aujourd’hui. Que dieux vous protège et que la réussite soit toujours à ma portée pour que je puisse vous combler de bonheur.

Je dédie aussi ce travail à mes deux frères Abdelhafid et Mohamed, tous mes amis proches, surtout l'amie qui était avec moi au moment de la détresse, qui était, reste et restera mon soutien dans la vie (Je sais que lorsque vous lisez cette partie, vous sourirez. Ce sourire est le secret de mon bonheur).

Résumé

La plupart d'entre nous, surtout les jeunes, connaissent la traduction automatique lorsque Google a lancé son célèbre service (Google Translator). Mais ce concept existe depuis le milieu du siècle dernier et constitue l'une des tâches les plus célèbres du traitement automatique de la langue.

Lorsque la traduction et la technologie sont mentionnées en même temps, les idées de beaucoup de gens se tournent immédiatement vers la traduction automatique - le processus par lequel le programme informatique traduit le texte d'une langue naturelle à une autre. La traduction automatique existe depuis plus de 50 ans et la qualité de la traduction automatique s'est considérablement améliorée au cours de cette période.

Le développement technologique a conduit à l'émergence de nouvelles méthodes de traduction automatique statistique. Paradoxalement, ces modèles statistiques ont été développés pour la première fois dans les années 1980, mais pas assez de données pour l'apprentissage et obtenir des résultats satisfaisants.

Notre projet vise à réaliser des expériences sur la traduction automatique, à proposer un modèle statistique de traduction automatique (anglais-arabe) basé sur des phrases, à améliorer la traduction par le biais d'observations expérimentales et à proposer des solutions aux problèmes auxquels nous sommes confrontés.

À cet égard, nous proposons l'utilisation de techniques de prétraitement pour la langue arabe en raison de sa complexité morphologique, telle est la segmentation des mots qui vise à séparer les clitiques attachés au mot et séquentiellement la tokenization des mots et des clitiques après la segmentation. Cela a permis de créer un meilleur modèle de langage et de traduction en arabe par rapport à nos expériences précédentes, ce qui a abouti à un modèle avec une traduction relativement bonne. Nous avons également réalisé une amélioration significative de la métrique WER (Word Error Rate) tout en expérimentant l'utilisation de la segmentation des mots dans l'évaluation avec la métrique WER.

Mots-clés : Traduction Automatique, Traduction Automatique Statistique, Traitement Automatique de la Langue, Linguistique Informatique.

Abstract

Most of us, especially young people, knew about machine translation when Google introduced its famous service (Google Translator). But the concept has existed since the middle of the last century and its one of the famous tasks in natural language processing.

When translation and technology are mentioned at the same time, many people's ideas immediately turn to machine translation - the process by which the computer program translates text from one natural language to another. Machine translation has existed for more than 50 years, and the quality of machine translation output has improved significantly during that period.

The technological development has led to the emergence of new ways of statistical machine translation. Paradoxically, these statistical models were developed for the first time in the 1980s, but did not have enough data to train for satisfactory results.

Our project aims to perform experiments on machine translation, propose a phrase-based statistical machine translation model (English-Arabic) and try to improve the translation through observations from experiments, and proposing solutions to the problems that we face.

In this regard we propose the use of preprocessing techniques for the Arabic language because of its morphological complexity, such, is the word segmentation which aim to separate the clitics attached to the word and sequentially the tokenization of both words and clitics after segmentation. This has helped to create a better Arabic language and translation model compared to our previous experiences, resulting in a model with a relatively good translation. Also we achieved a significant improvement in WER (Word Error Rate) metric while experimenting with the use of word segmentation in the evaluation with WER metric.

Keywords: Machine Translation, Statistical Machine Translation, Natural Language Processing, Computational Linguistics, Arabic Morphological Complexity.

ملخص

تعرف معظمنا وخاصة الشباب منا على الترجمة الآلية عندما قدمت جوجل خدمتها الشهورة (جوجل المترجم). لكن المفهوم موجود منذ منتصف القرن الماضي ويعتبر من المهام المشهورة في المعالجة الآلية للغة.

عندما يتم ذكر الترجمة والتكنولوجيا في نفس الوقت ، تتحول أفكار العديد من الأشخاص فورًا إلى الترجمة الآلية - العملية التي يقوم بموجبها برنامج الكمبيوتر بترجمة نص من لغة طبيعية إلى أخرى. الترجمة الآلية موجودة منذ أكثر من خمسين عامًا ، وقد تحسنت جودة مخرجات الترجمة الآلية بشكل كبير خلال تلك الفترة.

وقد أدى التطور التكنولوجي إلى ظهور طرق جديدة للترجمة الآلية الإحصائية ومن المفارقات أن هذه النماذج الإحصائية تم تطويرها لأول مرة في الثمانينات ولكن لم يكن لديها بيانات كافية لتتدرب عليها للحصول على نتائج مرضية.

يهدف مشروعنا إلى إجراء تجارب على الترجمة الآلية واقتراح نموذج ترجمة إحصائي قائم على الجمل (الإنجليزية-العربية) ومحاولة تحسين الترجمة عن طريق الملاحظات من التجارت و إقتراح حلول للمشاكل التي نصادفها.

في هذا الصدد ، نقترح إستخدام تقنيات المعالجة للغة العربية بسبب تعقيدها المورفولوجي ، على سبيل المثال ، تجزئة الكلمة التي تهدف إلى الفصل بين الضمائر و الكلمات ، يليها بعد ذلك التقسيم لكل من الكلمات والضمائر بعد التجزئة. وقد ساعد ذلك في إنشاء نموذج أفضل للترجمة واللغة العربية مقارنةً بتجاربنا السابقة ، مما أدى إلى وجود نموذج يحتوي على ترجمة جيدة نسبياً. لقد حققنا أيضًا تحسنًا كبيرًا في قياس "معدل أخطاء الكلمات" أثناء اختبار استخدام تجزئة الكلمات في التقييم.

الكلمات المفتاحية: الترجمة الآلية ، الترجمة الآلية الإحصائية ، معالجة اللغات الطبيعية ، اللغويات الحاسوبية ، التعقيد الصرفي العربي.

TABLE DES MATIERES

LISTE DES FIGURES

LISTE DES TABLEAUX

INTRODUCTION GENERALE

CONTEXTE GLOBAL

De nos jours, les progrès technologiques modernes en matière de communication ont transformé le monde en un petit village. Il est facile de communiquer par téléphone avec n'importe quelle personne dans n'importe quel lieu géographique. Il est également possible, à l'aide d'appareils mobiles très répandus, de joindre n'importe qui, non seulement à son adresse, mais pratiquement n'importe où. Si deux personnes disposent d'une connexion Internet et peuvent utiliser la messagerie texte et la conversation à l'aide d'appels audio gratuits normaux, elles peuvent également recevoir des appels vidéo gratuits si un appareil photo est installé. Même si beaucoup de gens ont maintenant un téléphone mobile avec un accès 3G ou 4G à Internet, il existe toujours un gros obstacle à la communication entre des personnes de différentes régions du monde. Ce problème est la barrière linguistique entre les personnes parlant différentes langues. De ce fait, l'humanité rêverait d'une technologie fiable capable de surmonter la barrière de la langue et de faciliter la communication entre les gens. Cela peut être une traduction instantanée de pistes audio ou de texte d'une langue étrangère vers notre langue maternelle et inversement.

Au cours de la dernière décennie, la nécessité d'une telle traduction automatique a été motivée par la large diffusion d'Internet et par l'augmentation rapide du contenu Web. De nombreux internautes aimeraient lire et avoir une bonne compréhension des sites Web écrits dans d'autres langues. L'augmentation continue du nombre d'utilisateurs de nombreux services Internet tels que les réseaux sociaux (Facebook et Linkedin, par exemple), les discussions en ligne et les appels audio / vidéo (Whatsapp et Skype, par exemple) a créé un besoin et une activité pour les services de traduction automatique. En effet, la plupart des utilisateurs préfèrent parler, lire et écrire en utilisant leur propre langue maternelle. Si l'utilisateur peut lire dans sa langue maternelle une page Web ou un message sur Facebook écrit dans une autre langue étrangère, cela lui permettrait de communiquer efficacement de manière interactive. Cela signifie également, du point de vue des activités de service, plus de revenus générés par les annonces et une meilleure audience pour les annonces en langue maternelle de l'utilisateur, ce qui signifie davantage de ventes pour l'annonceur. Ces grandes opportunités commerciales étaient intéressantes et ont

permis de collecter davantage de fonds pour la recherche en traduction automatique dans les grandes entreprises Internet. Certaines entreprises ont déjà mis en place un service de traduction automatique gratuit en ligne, tel que Microsoft Bing (prend en charge 51 langues) et Google Translate (prend en charge 90 langues). Facebook a intégré une option permettant à l'utilisateur de traduire sur place tout message écrit dans une autre langue. Ils ont utilisé le service de traduction « Bing » de Microsoft. Une option similaire permettant de traduire le contenu des e-mails est intégrée à Gmail, le service de messagerie largement utilisé de Google. Un autre défi auquel sont confrontés ces services de traduction automatique en ligne gratuits est leur aptitude au transfert et leur fiabilité. En raison de la nature interactive de ces services, les internautes attendent une traduction rapide et un service ininterrompu.

Depuis les débuts de l'informatique, les scientifiques ont essayé de construire des systèmes de traduction automatique. À ce moment-là, ils ont commencé par se concentrer sur les approches linguistiques pour résoudre le problème de la traduction automatique. Ils avaient, avec beaucoup d'optimisme, l'impression qu'une fois le vocabulaire et les règles de grammaire programmés, la traduction automatique deviendrait une tâche facile. Ces approches utilisent l'analyse linguistique et la génération avec une profondeur différente. Plus l'analyse est approfondie, plus la représentation intermédiaire de la phrase source est abstraite, ce qui nécessite également davantage d'efforts pour générer la phrase cible à partir de cette représentation intermédiaire. L'approche linguistique a évolué dans le temps, en partant de la méthode basée sur le transfert, pour atteindre la méthode interlingua.

Une autre meilleure approche qui utilise les traductions extraites de corpus traduits auparavant par des humains est l’approche basée sur les corpus. La traduction automatique statistique (TAS), basée sur des modèles statistiques formés sur des corpus bilingues et monolingues, est un exemple d’approche fondée sur un corpus.

La TAS a été inventé dans le laboratoire de recherche IBM. Fondamentalement, deux modèles probabilistes sont utilisés, un modèle de traduction formé sur des corpus bilingues et un modèle linguistique formé sur des corpus monolingues. TAS a de nombreux avantages, il est indépendant du langage, facile, peu coûteux et rapide à construire. De nombreux outils de formation et de décodage sont

disponibles gratuitement. Les immenses corpus bilingues et monolingues nécessaires à la formation sont également disponibles pour de nombreux couples de langues. L'état actuel de la technologie dans TAS est la traduction automatique statistique basée sur les phrases (PBSMT en Anglais Phrase Based Statistical Machine Translation) car il utilise des unités de traduction plus longues que les modèles initiaux basés sur les mots. De cette manière, le modèle de traduction capture plus d'informations contextuelles, ce qui améliore la qualité de la traduction.

La langue arabe a beaucoup retenu l'attention de la communauté de la traduction automatique au cours de la dernière décennie. C'est la langue officielle de 25 pays et elle est parlée par plus de 295 millions de personnes.

La langue arabe, comme les autres langues sémitiques, se caractérise par une morphologie relativement complexe. Cette complexité morphologique pose un défi à la traduction automatique de l'arabe à l'anglais ou des langues similaires, qui se manifeste différemment dans différents paradigmes. Par exemple Les approches de traduction automatique base sur des données, le défi est que le corpus arabe sera moins volumineux qu'un corpus anglais équivaut, parce que le nombre moyen d'instances observées d'un mot arabe morphologiquement complexe sera plus bas

Dans ce projet, j’ai travaillé sur l’amélioration du système de traduction automatique MOSES de l’Anglais vers l’Arabe dans le cadre du Al Dhakhira Al Arabia au sein du Centre de Recherche Scientifique et Technique pour le Développement de la Langue Arabe (CRSTDLA).

Notre travail vise à mettre en place un système de traduction automatique statistique destiné spécifiquement au couple de langues Anglaise et Arabe. Au terme de notre stage, nous estimons que nous sommes parvenus à des résultats acceptables mettant en évidence la majeure partie de nos contributions, principalement axées sur la collecte, le prétraitement et l’évaluation des données.

PROBLEMATIQUE ET OBJECTIFS

La traduction connait des problèmes d'ordre linguistique et culturel. Les ***problèmes linguistiques*** incluent les différences grammaticales, l'ambiguïté lexicale, et l'ambiguïté sémantique. Les problèmes culturels quant à eux, ils renvoient à des caractéristiques de diverses situations. Nous parlons dans ce cas, du ***contexte*** entourant la tâche de traduction.

En effet, les problèmes majeurs de la traduction peuvent être résumé en la : sur-traduction (ou traduction excessive), sous-traduction (ou traduction pauvre) et la non traductibilité (ou l'impossibilité de traduire certains mots). En addition à ces problèmes classiques, nous nous confrontons au problèmes spécifiquement liés à la langue Arabe, notamment les difficultés d'autres lexical et morphologique.

A ce titre, l'objectif principal des travaux de recherché menés cette année est d'améliorer le système de traduction automatique statistique MOSES et ce, de l'Anglais vers l'Arabe. Les principaux objectifs scientifiques de ce travail sont les suivants :

- Développement et amélioration d'un système de traduction automatique statistique dans le cadre du projet "*Al Dhakhira Al Arabia*".
- Travail sur les Input and Output du système MOSES en améliorant le prétraitement, enrichissant l'apprentissage, et optimisant la phase de décodage afin d'augmenter la pertinence des résultats.

ORGANISATION DU MEMOIRE

Ce mémoire se compose de cinq chapitres qui sont organisés comme suit : Le chapitre 1 est une introduction à la traduction automatique. Le chapitre 2 couvre la notion de Traitement Automatique de la Langue, notamment la langue Arabe. Le chapitre 3 quant à lui expose la conception et modélisation de notre solution proposée. En fin, l'amélioration du système MOSES est présentée et discutée au chapitre 5 avec des mesures de performance que nous avons obtenues et leur comparaison avec d'autres travaux conséquents dans le domaine. Enfin, la conclusion et les perspectives sont abordés dans la conclusion générale de ce mémoire.

CHAPITRE 1
LA TRADUCTION AUTOMATIQUE

1.1 INTRODUCTION

La traduction automatique (TA) est un sous domaine de la Linguistique Computationnelle qui étudie l'utilisation d'un logiciel pour traduire du texte ou des paroles d'une langue source vers une langue cible. Le principal défi consiste à programmer un ordinateur qui comprendra un texte comme le fait une personne et qui crée un nouveau texte dans le langage cible.

Il y a plusieurs raisons pour l'émergence de ce processus. La raison principale en est la pratique car les scientifiques, technologues, ingénieurs, économistes, agronomes, administrateurs, industriels, hommes d'affaires et bien d'autres doivent lire les documents et doivent communiquer dans des langues qu'ils ne connaissent pas et il n'y a tout simplement pas assez de traducteurs pour faire face au volume de plus en plus important des données qui doit être traduit [1].

Les fréquents échanges nationaux et internationaux ont permis à la traduction automatique de s'épanouir, car la traduction humaine ne peut répondre à la demande grandissante d'action de traduction de par le monde. En conséquence, la traduction automatique a énormément progressé depuis sa création en 1940 avec l'émergence de nombreuses architectures et approches. Parmi les approches les plus connues et qui fera l'objet de ce chapitre c'est la traduction automatique statistique.

1.2 LA TRADUCTION AUTOMATIQUE

La traduction automatique [2] est une traduction effectuée par un ordinateur, telle que définie dans le dictionnaire anglais Oxford. C'est un processus référé au traitement automatique de langue, qui utilise un ensemble de données (corpus) et d'autres ressources linguistiques pour créer des modèles de langage et d'expression utilisés pour traduire le texte.

Au niveau de base la traduction automatique effectue une simple substitution de mots dans une langue par des mots dans une autre langue, mais cela ne peut pas produire une bonne traduction d'un texte car il est nécessaire de reconnaître des phrases entières et leurs expressions les plus proches dans la langue cible.

1.2.1 Histoire de Traduction Automatique

L'idée de la traduction automatique remonte au XVIIe siècle [1]. En 1629, René Descartes a proposé une langue universelle, avec des idées équivalentes dans différentes langues partageant un symbole. Le domaine de la "traduction automatique" est apparu grâce au mathématicien américain Warren Weaver qui a été très intéressé par ce domaine. Ce dernier est considéré parmi les pionniers de la traduction automatique tel qu'il compare explicitement le processus de traduction à un processus de chiffrement : un texte traduit en russe peut être vu comme un chiffrage de sa version anglaise à l'aide d'un code particulier. Warren Weaver a écrit un important mémorandum "Translation" en 1949. Par ailleurs, il est à noter que l'idée d'utiliser des ordinateurs pour la traduction de langues naturelles a été proposée dès 1946 par A. D. Booth.

Le premier chercheur sur la traduction automatique, Yehoshua Bar-Hillel, a commencé ses recherches au MIT (1951). Ses recherches ont été poursuivies par une équipe de recherche en Traduction Automatique à l'université de Georgetown en 1951 avec une démonstration publique de son système d'expérience Georgetown-IBM en 1954 sur la machine APEXC du Birkbeck College (Université de Londres) qui présente une traduction rudimentaire de l'anglais en Français.

Les chercheurs ont continué à se joindre au domaine après la création de l'association de Traduction Automatique et de Linguistique Computationnelle aux États-Unis (1962). Par la suite, l'académie nationale des sciences en Amérique a formé le Comité consultatif du traitement automatique du langage (ALPAC) pour étudier la traduction automatique en 1964.

Il y a aussi des programmes de recherche sur la traduction automatique qui sont apparues au Japon et en Russie (1955), et la première conférence sur la traduction automatique s'est tenue à Londres (1956).

Les progrès réels ont toutefois été beaucoup plus lents car d'après le rapport ALPAC (1966), qui estimait que la recherche qui avait duré dix ans n'avait pas répondu aux attentes, et de ce fait, le financement a été considérablement réduit.

La traduction automatique sur le Web a commencé avec SYSTRAN qui permettait traduction gratuite de petits textes (1996), suivi par AltaVista Babel-Fish, qui enregistrait 500 000 demandes par jour (1997).

Parmi les autres innovations au cours des années 2000, citons :

Système	Description	Année	Auteur	Technique utilisé
Moses	System traduction automatique statistique open-source	2007		Modèles statistique
	Service de traduction de Text/SMS pour les téléphones mobiles	2008,2009		
GNMT	Traducteur automatique basé sur les réseaux de neuronnes	2018	Google	les réseaux de neuronnes

1.2.2 Les difficultés de la Traduction Automatique

Dans toute traduction, qu'elle soit humaine ou automatisée, la signification d'un texte dans la langue source doit être intégralement transférée vers son sens équivalent dans la traduction de la langue cible [2]. Bien qu'en surface, cela semble simple, mais la tâche est souvent beaucoup plus complexe car la traduction n'est jamais une simple substitution mot à mot.

En effet, le traducteur humain doit interpréter et analyser tous les éléments du texte et comprendre comment chaque mot peut influencer le contexte du texte. Cela nécessite une grande expertise de la grammaire, de la syntaxe (structure des phrases), de la sémantique (significations), etc., dans les langues source et cible, ainsi que de l'expertise dans le domaine.

La traduction humaine et la traduction automatique ont chacune leur part de défis. Par exemple, aucun traducteur individuel ne produira de traduction identique du même texte dans la même paire de langues et plusieurs révisions peuvent être nécessaires pour répondre aux besoins du client. De plus, les traductions automatisées rencontrent des difficultés pour interpréter les éléments linguistiques, contextuels et culturels d'un texte et la qualité dépend du type de système et de la façon dont il est formé. Les problèmes linguistiques incluent les différences grammaticales, l'ambiguïté lexicale, et l'ambiguïté sémantique. Les problèmes

contextuels et culturels quant à eux, ils renvoient à des caractéristiques de diverses situations.

Dans la section suivante, on va aborder les différentes approches qui ont émergé pour régler ces problèmes.

1.3 LES APPROCHES DE TRADUCTION AUTOMATIQUE

La première démonstration de la traduction automatique s'est tenue en 1954 et elle est connue sous le nom de l'Expérience de Georgetown –IBM » [3]. Avant l'année 1990, les travaux menés sur la traduction automatique ont été typiquement focalisés sur une approche experte et des ressources linguistiques utilisant des analyseurs syntaxiques et sémantiques. Aujourd'hui, les approches statistiques sont fondées sur l'apprentissage automatique à partir de corpus, aussi les approches hybrides sont de plus en plus adoptées par de nombreux chercheurs. La figure 1 ci-dessous synthétise les différentes approches de Traduction Automatique.

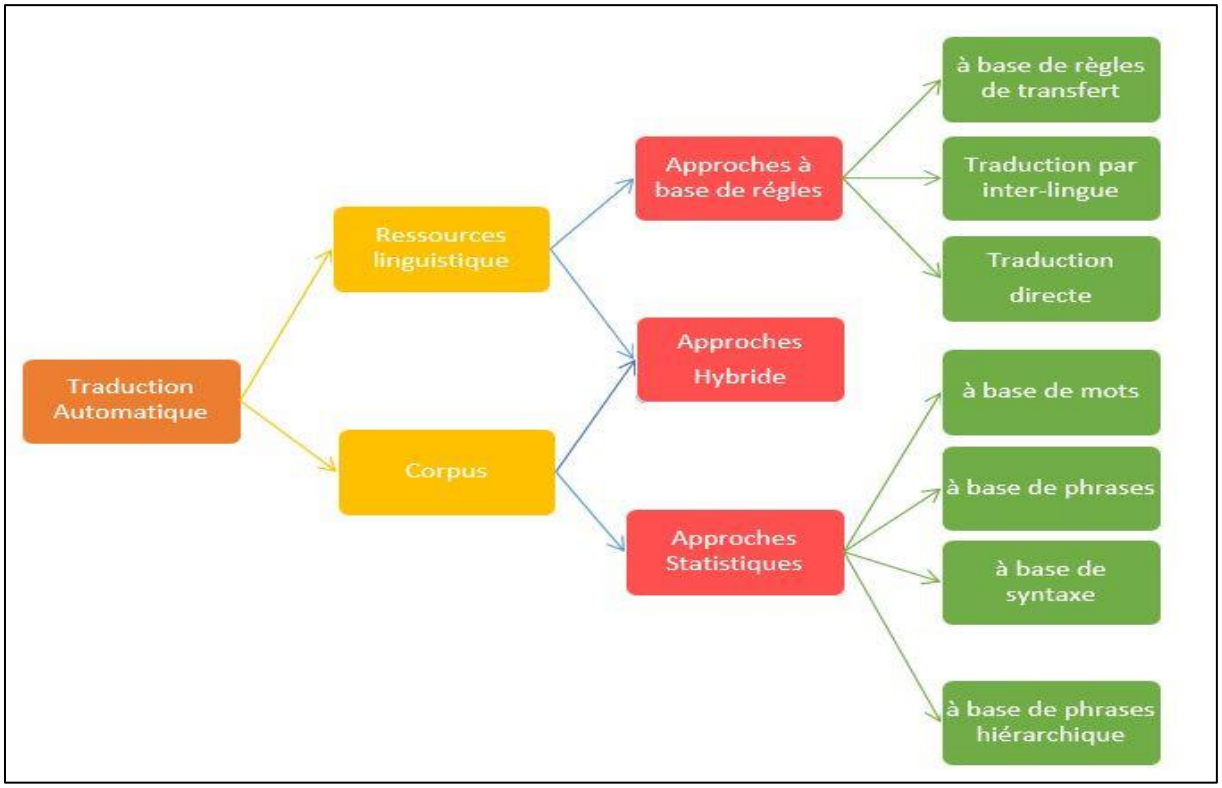


Figure 1: schéma hiérarchique des approches de Traduction Automatique.

De manière générale, nous pouvons décrire le processus de traduction (au sens humain), découpé en trois phases successives :

- **Compréhension :** Ingestion du sens transféré par un texte, au vouloir dire d'un auteur.
- **Dé-verbalisation :** oubli des mots et conservation du sens de texte.
- **Réexpression :** reformulation du vouloir dire en langue cible.

En termes informatiques, la compréhension devient l'**analyse**, la déverbalisation devient le **transfert** et la réexpression devient la **génération**.

1.3.1 L'approche à base de règles

Le processus de traduction automatique linguistique se déroule en trois phases fondamentales [4-5] :

- **L'analyse** : analyser le texte source en des représentations intermédiaires en langue source.
- **Le transfert** : transférer ces représentations intermédiaires vers des représentations intermédiaires en langue cible.
- **La génération** : générer le nouveau texte en langue cible à partir des représentations intermédiaires en langue cible.

Le triangle de Vauquois (voir figure 2) proposé dans [2, 5] décrit les différentes approches à base de règles possibles d'un système de TA. Chaque chemin dans le triangle correspond à une architecture linguistique.

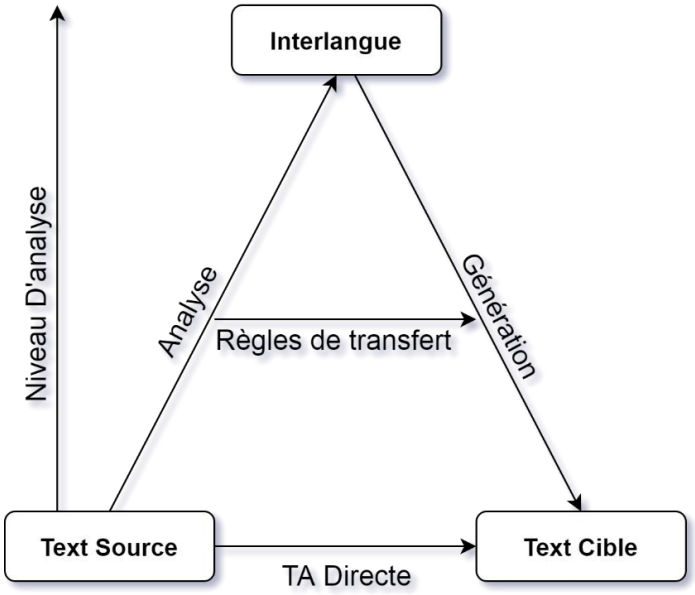


Figure 2: Triangle de Vauquois, représentation des différentes approches à base de règles.

A) Système de traduction automatique direct

La phrase source est segmentée en des termes et le système fait la traduction terme à terme [5]. L'étape de transfert utilise une table bilingue qui associe pour chaque terme ou séquence de termes source un ensemble de règles de traduction et de réarrangement qui permet de traduire et réordonner les mots dans la phrase cible.

B) Système de traduction automatique par transfert

La phrase source est analysée à l'aide d'un analyseur syntaxique et d'une grammaire. Cette phase d'analyse donne lieu à une représentation arborescente qui est ensuite convertie dans la langue cible [3]. Pour assurer le passage de la représentation intermédiaire source à la représentation cible, une table bilingue, contenant les règles de transfert entre les représentations source et cible, est requise. La " Figure 3" ci-dessous, d'illustre un exemple de traduction à base de transfert.

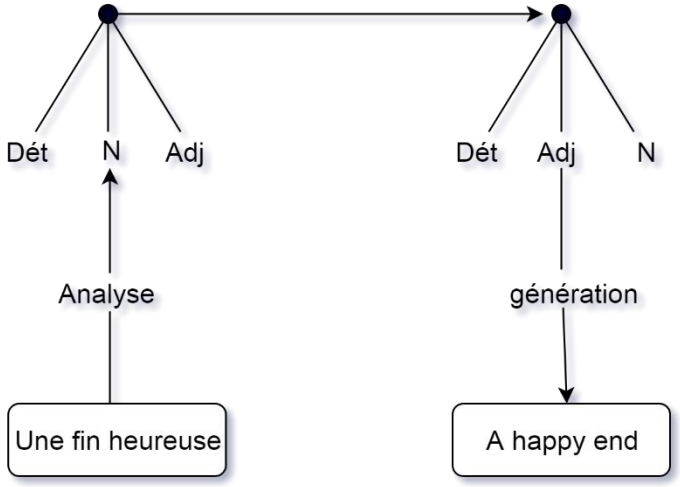


Figure 3: Exemple de traduction d'un groupe nominal en français vers une phrase en anglais [3].

C) Système de traduction automatique par inter-lingue

La traduction de n'importe quelle langue source vers n'importe quelle langue cible est le processus qui consiste à « convertir » la phrase source vers la représentation pivot puis à « convertir » la phrase cible à partir de cette représentation pivot. Le transfert des représentations de la langue source vers celles de la langue cible n'a plus lieu d'être. L'avantage de cette méthode est la possibilité de l'appliquer dans un environnent multilingue. Pour couvrir tous les

sens de traduction entre n langues, nous n'avons besoin que de n modules de conversion dans chacun des sens [6].

Par ailleurs, la complexité de cette méthode réside dans l'obligation de construire un vocabulaire pivot pour représenter tous les concepts possibles de toutes les langues et les liens des concepts entre deux langues [6-5]. La construction peut être basée sur une langue artificielle « logique », sur une langue auxiliaire « naturelle » (comme l'anglais ou l'espéranto), sur un ensemble de concepts primitifs communs à toutes les langues, ou sur un vocabulaire « universel ». UNL (Universal Networking Language) est un exemple de langage pivot.

1.3.2 L'approche statistique

Le domaine de la traduction automatique a été dynamisé par l'apparition de techniques statistiques qui ont permis de rapprocher le rêve de la traduction automatique des langues de la réalité. Ces techniques permettent de traduire un texte d'une langue source vers une langue cible en procédant à l'apprentissage automatique du système en appliquant des calculs mathématiques statistiques sur des corpus [6-7].

L'idée principale de la traduction automatique statistique (TAS) est la suivante : « Etant donné une phrase dans la langue source, pour trouver la traduction dans la langue cible, il faut chercher la proposition ayant la probabilité la plus élevée » [8-9].

Cette approche comprend trois étapes : la modélisation, la formation (entraînement) et le décodage.

La modélisation concerne la définition d'une méthode de calcul de la probabilité qu'une phrase en langage cible ait une phrase du langage source. La phase de formation ou entraînement, consiste à utiliser un corpus pour estimer le paramètre du modèle qui a été défini. A la fin, le décodage se concentre sur la recherche de la phrase présentant la probabilité la plus élevée parmi toutes les traductions candidates [9].

Par exemple, dans le cas où nous traduisons de l'anglais (langue source) vers arabe (langue cible), nous pouvons formuler une séquence de mots dans la langue source :

E=e_1,…,e_j,…,e_J,
/*trouver la séquence la plus probable dans le langage cible :*/
A=a_1,…,a_i,…,a_I,
A^=$argmax_A$ P (A|E) = $argmax_A$ P (A)*P (E|A) … (1)

Telle que **P(E|A)** représente le modèle de traduction et **P(A)** représente le modèle de langage cible. Les traductions optimales sont le produit de leurs probabilités. ***argmax*** **:** c'est l'algorithme de décodage permettant de trouver la phrase qui maximise l'équation (1).

On peut illustrer l'architecture d'un système de traduction statistique avec la "Figure .4".

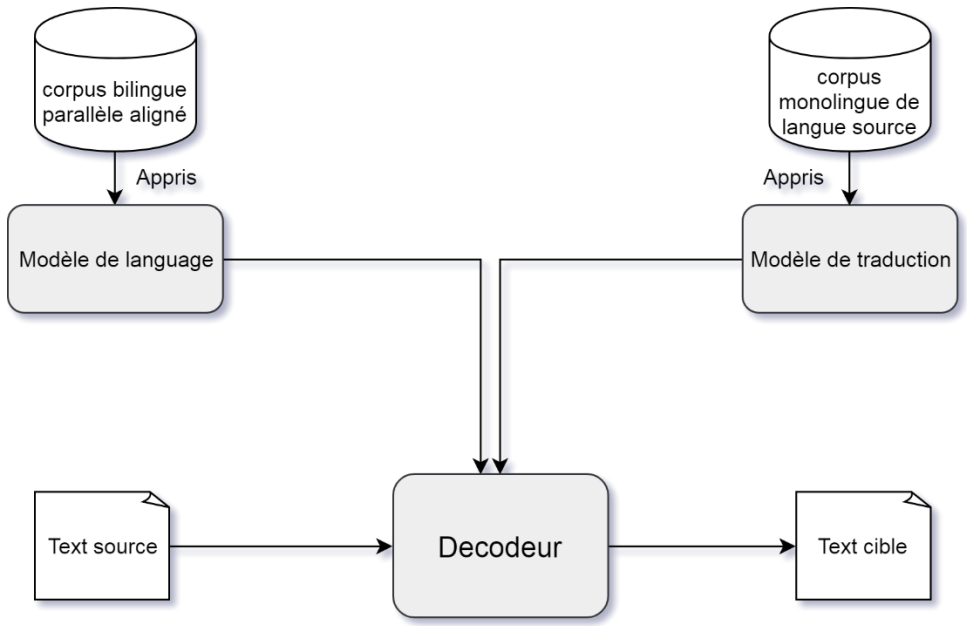


Figure 4: Exemple d'architecture d'un système statistique de traduction automatique.

1.3.3 L'approche hybride

Dans le domaine de la traduction automatique, il existe différentes approches possédant chacune des forces et des faiblesses. Après que les défauts des systèmes conventionnels se soient amplifiés, les chercheurs se sont intéressés de plus en plus

aux approches mixtes qui pourraient tirer profit des forces des approches statistiques et linguistiques. Plus récemment, avec l'émergence de traduction automatique neuronale, une nouvelle version de la traduction automatique hybride est en train de naître. Elle associe les avantages des règles, de la traduction automatique statistique et neurale [5].

1.4 LES MODELES DE TRADUCTION AUTOMATIQUE STATISTIQUE

La Traduction Automatique Statistique (TAS) est une approche basée sur un corpus et sur des modèles statistiques. Elles considèrent le processus de traduction comme une tâche d'apprentissage automatique dans laquelle les modèles du système sont construits à partir de données, généralement de grands corpus bilingues / parallèles et des corpus monolingues. TAS utilise des techniques d'apprentissage automatique tout au long du processus de traduction.

TAS utilise deux modèles probabilistes, un modèle de traduction transformé en un corpus parallèle et un modèle de langage formé sur des corpus monolingues. Le premier étant utilisé pour estimer les distributions de probabilités inverses de la traduction d'une phrase source en une phrase cible, qui est la probabilité qu'une phrase source composée de plusieurs mots soit une traduction de la phrase cible. Tandis que le dernier, formé sur des corpus monolingues, est utilisé pour améliorer la pertinence de la traduction en tant que mécanisme de post-traitement. La combinaison de ces modèles est mathématiquement justifiée par le modèle des canaux bruyants (Noisy Channel) [7].

Certains des principes de la traduction automatique qui ont été établis au tout début restent valables aujourd'hui, tels que le décodage d'une langue étrangère et l'utilisation de techniques de modélisation telles que le modèle canaux bruyants *Noisy-Channel model* utilisé dans la traduction automatique statistique.

On ne peut pas parler d'approches statistiques sans parler de probabilités. La théorie des probabilités est la branche des mathématiques concernée par la probabilité. En tant que fondement mathématique de la statistique, la théorie des probabilités est essentielle pour de nombreuses activités humaines impliquant une analyse quantitative de données [7]. Les méthodes de la théorie des probabilités

s'appliquent également aux descriptions de systèmes complexes pour lesquels on ne dispose que d'une connaissance partielle de leur état ou de la gestion d'événements ayant des résultats incertains.

La traduction, où un mot source peut être traduit en un des nombreux mots possibles dans une langue cible, est un scénario dont l'issue est incertaine [7]. Mathématiquement, une distribution de probabilité est une fonction qui mappe les résultats possibles à une valeur comprise entre 0 et 1. lors de l'analyse d'un événement, nous pouvons découvrir qu'une distribution standard peut être utilisée pour le modéliser. Nous pouvons également collecter des statistiques sur l'événement et estimer les distributions de probabilité par estimation du maximum de vraisemblance.

Dans un sens, en traduction, nous traitons généralement plusieurs événements incertains, car la traduction s'applique généralement à une phrase complète au lieu d'un mot individuel. De manière pratique, les mathématiques de la théorie des probabilités fournissent un ensemble d'outils et de méthodes permettant de traiter des distributions complexes. L'entropie est un concept important du traitement du langage naturel et de l'apprentissage automatique : elle permet d'évaluer l'incertitude des résultats possibles et des probabilités d'un événement.

1.4.1 Modèle à base de mots (Word-Based Models)

Les modèles basés sur les mots sont issus des travaux originaux sur la TAS réalisés par le projet *IBM Candide* à la fin des années 80 et au début des années 90 [7]. Bien que cette approche ne soit plus à la pointe de la technologie, ses principes et méthodes sont toujours d'actualité.

Le paradigme de la TAS implique l'utilisation de statistiques. Si nous examinons la tâche de traduction d'un mot d'une langue source vers une langue cible dans un dictionnaire bilingue, nous pouvons voir qu'un mot peut avoir plusieurs traductions mais nous nous intéressons à la meilleure traduction c'est à dire la traduction la plus probable de ce mot dans ce cas. Dans une traduction humaine, on peut déduire de la familiarité avec la langue ou de la recherche du mot le plus approprié pour décider de la meilleure traduction. Dans une tâche de traduction automatique, nous devons

toutefois collecter des statistiques sur les traductions pour permettre à la machine de décider quelle est la meilleure traduction.

La collecte de statistiques est nécessaire pour estimer une distribution de probabilités qui est l'essence de la traduction automatique statistique. Le processus de collecte des statistiques nécessite des ressources sous forme de collections de textes dans la langue source et leur traduction dans la langue cible. Toutefois, dans les modèles basés sur les mots, la collecte de statistiques est effectuée en décomposant les phrases en une séquence de mots au lieu de phrases en raison de leur faible densité. Cette phase de collecte de données est nécessaire pour établir une distribution de probabilité qui sera utilisée pour trouver la traduction la plus probable d'un mot.

Le processus d'estimation d'une distribution de probabilités à partir des statistiques collectées. C'est-à-dire, trouver la probabilité d'une traduction d'un mot donné pour chaque traduction possible dans la langue cible, ce qui peut être formalisé en une fonction qui renvoie la probabilité de traduction d'un mot à chacune des traductions possibles dans la langue cible. La fonction doit renvoyer une valeur comprise entre 0 et 1 indiquant la probabilité de traduction avec la valeur 0; comme impossible et plus la valeur est élevée, plus la probabilité est élevée. Les modèles IBM proposent des algorithmes d'estimation des probabilités de traduction [3]permettant d'aligner les mots d'une phrase cible avec les mots correspondants de la phrase source, le processus de traduction des mots individuels produisant une séquence de mots pouvant être mal alignés ou, en d'autres termes, mal alignés dans la langue cible.

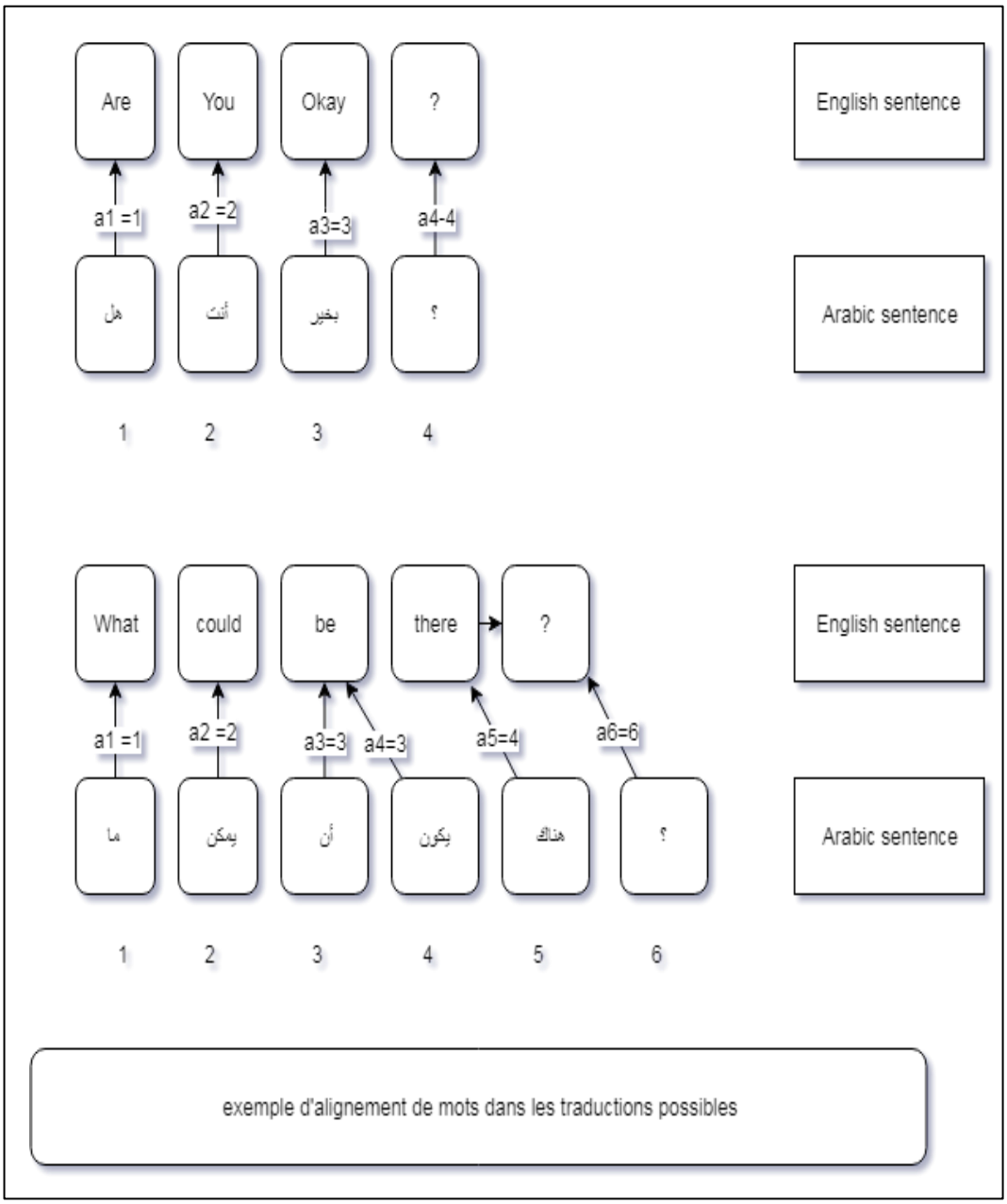


Figure 5: Alignement des mots et les relations d'alignement.

L'alignement est un mappage de mots implicite de la langue source à la langue cible qui capture la correspondance (niveau des mots) entre le mot source et le mot cible. Un alignement peut être formalisé avec une fonction d'alignement ***a*** qui mappe les mots correspondants entre source à une position ***i*** et langue cible à une position ***j*** [7].

$$a : i \rightarrow j$$

Le mappage d'une séquence de mots formant une phrase serait noté comme un vecteur de positions d'alignement a_i pour chaque mot t_i dans t.

$$a : \{i_1 \rightarrow j_1 , \ldots\ldots i_n \rightarrow j_n\}$$

La probabilité conditionnelle ***p(t/s)*** est exprimée sous la forme de la somme des probabilités des alignements ***a*** entre la source ***s*** et la cible ***t*** [3]

$$P(t/s) = \sum_a P(t, a|s)$$

Certaines langues ont un ordre de mots similaire ou très proche comme l'allemand et l'anglais [7], tandis que d'autres langues sont relativement plus différentes dans leur ordre de mots. Dans certaines langues, le nombre de mots nécessaires pour exprimer le même concept peut différer cas où il n'y a pas de mots équivalents clairs entre une paire de langues, ce qui entraîne la suppression de tels mots pendant la traduction.

La fonction d'alignement mappe chaque mot source à un mot cible. Toutefois, comme indiqué précédemment, certains mots source nécessitent plus d'un mot cible pour exprimer le même concept, ce qui donne une relation un à plusieurs.

Un autre cas est l'absence de toute relation entre les mots d'entrée et les mots de sortie. Pour y remédier, le concept d'insertion de mots pendant la traduction est ajouté ou un « *NULL TOKEN* » a été introduit (on peut l'assimiler à l'epsilon dans les automates à états finis) car la fonction d'alignement nécessite un mappage entre chaque paire de mots du processus de traduction, le jeton nul peut être traité comme un mot pour définir complètement la fonction d'alignement. Notez que la fonction d'alignement dans l'autre sens (c'est-à-dire, Cible → Source) ne peut mapper qu'un seul mot de sortie sur un mot d'entrée, y compris le jeton nul.

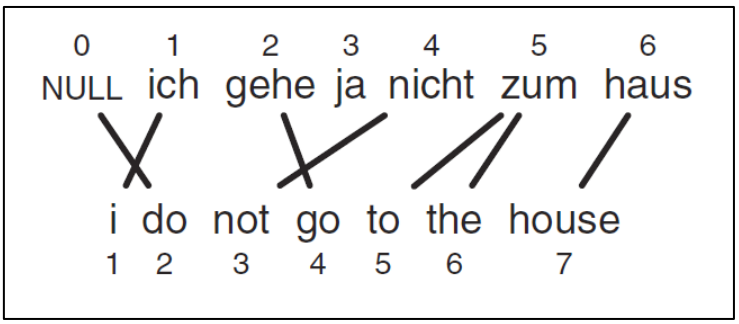


Figure 6: Insertion de mots pendant la traduction à l'aide du jeton NULL [7].

1.4.2 Les modèles IBM

La méthode de modélisation basée sur les mots consiste à estimer la distribution de probabilités pour la traduction de mots individuels en décomposant le processus de modélisation d'une distribution de probabilités de traduction de phrases complètes en chaînes de mots plus petites et en collectant suffisamment de statistiques pour la traduction de mots individuels [7]. Les petites chaînes de mots ainsi générées seront à leur tour modélisées avec des distributions de probabilité, puis combinées de manière cohérente, appelée modélisation générative.

Jusqu'ici, ce que nous avons vu, les probabilités de traduction lexicale et la notion d'alignement permettent de définir un modèle générant plusieurs traductions pour une phrase, chacune avec une probabilité différente. Ce modèle s'appelle modèle IBM 1. Brown et al. [1990] ont proposé cinq modèles génératifs nommés séquentiellement IBM modèle 1 jusqu'au modèle IBM 5. Chaque modèle a amélioré son prédécesseur.

1.4.3 Modèles basés sur les phrases (Phrase-Based Models)

Les modèles basés sur les mots considèrent les mots comme la plus petite unité de traduction. Parfois, les systèmes implémentant ce type de modèles échouent lorsque le mot cible et le mot source ne se trouvent pas dans une relation mappée un à un. Un nouveau type de modèle est basé sur les phrases et considère une séquence de mots (phrase - unité multiple comme convention - non motivée par la linguistique) car cette unité de traduction semble mieux convenir à la traduction.

Les modèles basés sur les phrases alimentent des systèmes de traduction automatique statistiques plus performants que ceux utilisant des modèles basés sur des mots [7]. Ces modèles utilisent des unités de traduction plus longues, à savoir traduire de petites séquences de mots à la fois permet de capturer plus d'informations contextuelles dans la traduction, ce qui permet une meilleure sélection des mots parmi différentes traductions possibles. Les modèles basés sur les phrases segmentent la phrase d'entrée en unités de plusieurs mots (voire segment de phrase). Ensuite, chaque segment est traduit et mappé un à un en fonction d'un tableau de traduction de phrases et peut être réorganisée après traduction en fonction de la paire de langues.

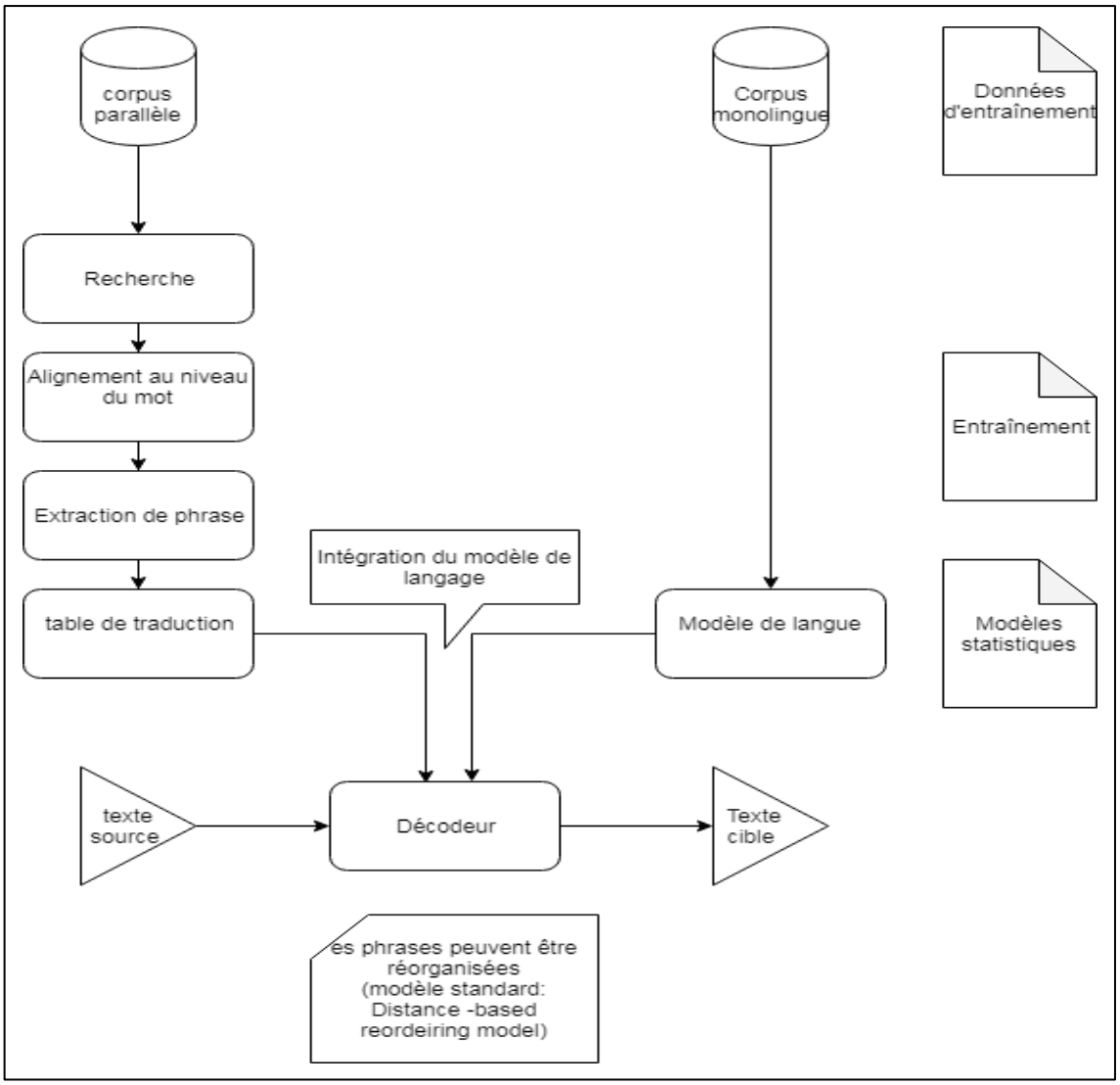


Figure 7: Schéma d'illustration le modelé à base de phrase.

La structure de données fondamentale dans les modèles basés sur les expressions est un tableau de paires d'expressions avec des scores associés pouvant provenir d'une distribution de probabilités [14]. Le plus souvent, ce tableau est déduit en alignements de mots qui énumèrent de manière exhaustive toutes les phrases compatibles avec l'alignement selon une longueur prédéfinie. Comme nous l'avons vu, les modèles IBM ont pu évaluer le bon alignement des mots, ce qui joue un rôle important dans les modèles de traduction basés sur les phrases.

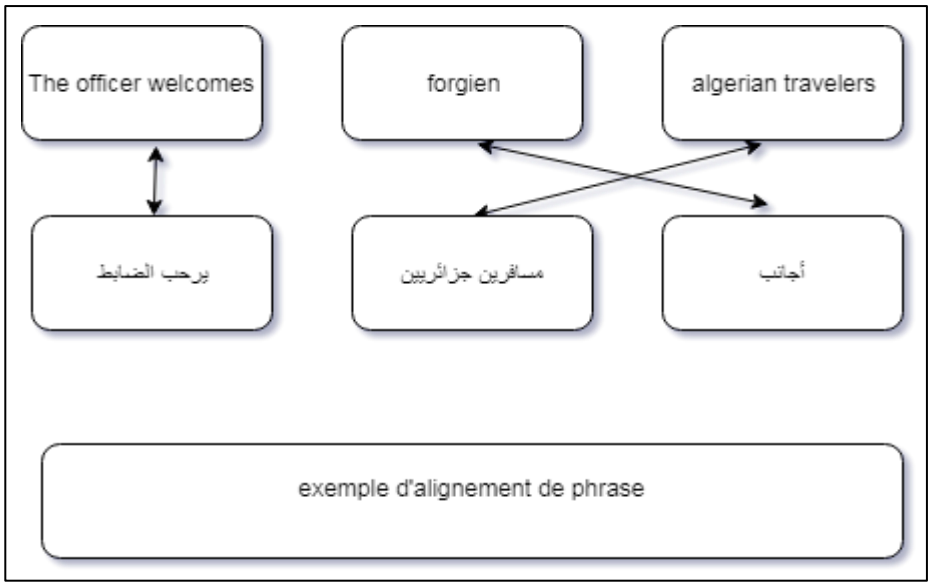


Figure 9: Alignement des phrases en (segments) de phrase.

La table de traduction de phrases est un élément fondamental de la traduction basée sur les phrases. L'acquisition de cette table nécessite des données d'alignement de mots établies à l'aide de la formation de maximisation d'attente (*Expectation Mazimization (EM)*) qui désigne l'alignement des mots entre chaque paire de phrases du corpus parallèle. L'étape suivante consiste à extraire des paires de phrases cohérentes avec les données d'alignement des mots.

	what	could	be	there	?
ما					
يمكن					
أن					
يكون					
هناك					
؟					

Figure 8: Représentation graphique de l'alignement des mots.

L'extraction des paires de phrases repose sur l'alignement des mots dans la paire de phrases. C'est-à-dire que l'alignement des phrases est l'alignement cohérent de leurs mots constitutifs. La cohérence ici est que tous les mots d'une phrase source qui ont des points d'alignement dans un alignement ont également des alignements avec les mots de la phrase cible et inversement [Fig. 10]. Le processus d'extraction de phrases consiste à vérifier toutes les phrases possibles pour une phrase cible et à rechercher les phrases sources minimales qui y correspondent, avec peu de contraintes :

- Les paires de phrases ne peuvent pas être extraites si la phrase source correspondante a plus de points d'alignement en dehors de l'espace de phrase cible.
- Les phrases cibles ne contenant que des mots non alignés ne doivent pas être associées.
- Si une phrase source autre que la phrase à correspondance minimale comporte des mots non alignés aux bords, mais correspondants et cohérents avec la phrase cible. Elle est également ajoutée comme une traduction possible en l'étendant à ces mots.

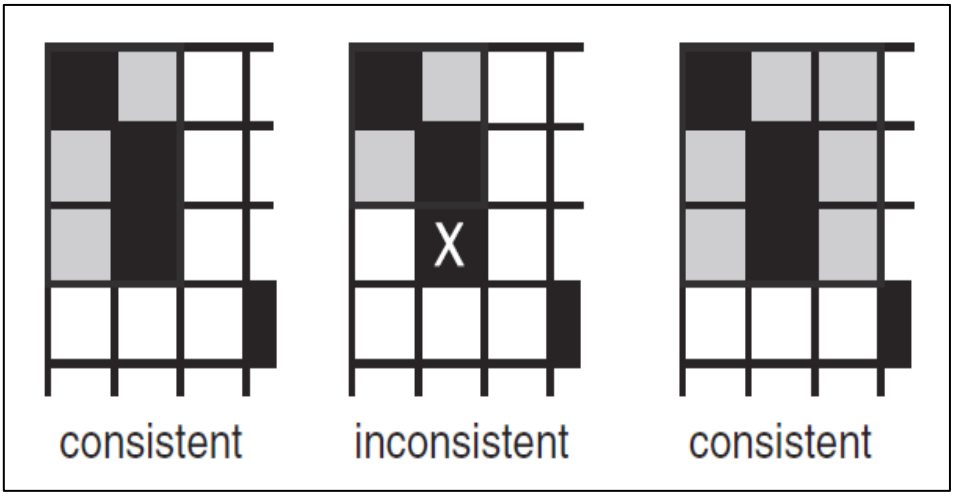


Figure 10: Table de traduction - phrases cohérentes et incohérentes.

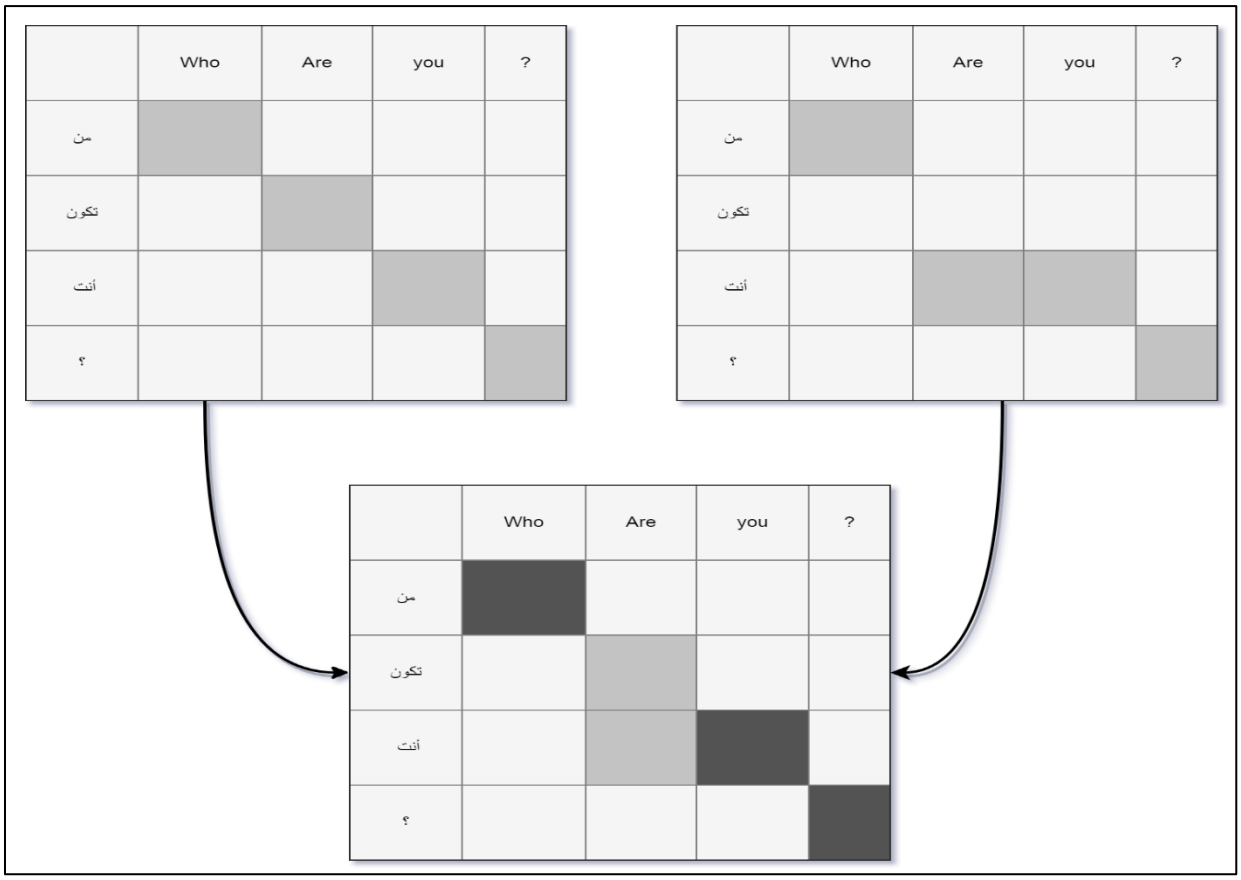


Figure 11: Alignement (IBM) en combinant l'alignement des deux directions; source à cible et cible à source pour obtenir un alignement plus précis.

Le modèle de traduction standard basé sur les phrases sert de base à d'autres variations pouvant être considérées comme une extension de ce modèle. Le modèle standard utilise les règles de Bayes (modèle de canal bruyant) pour inverser le sens de la traduction et intégrer un modèle de langage de la même manière que les modèles basés sur les mots, mais la probabilité est ensuite décomposée en modèle basé sur l'expression où la phrase source est divisée en segments (unités de plusieurs mots). Chaque segment de la phrase source est traduit dans un segment de la langue cible tandis que la probabilité de traduction indique la traduction de la langue cible vers la langue source en raison de l'inversion du sens de la traduction dans le canal bruyant (règle de Bayes).

La réorganisation dans le modèle standard est effectuée par un modèle basé sur la distance, dans lequel la réorganisation est relative à la phrase précédente. La distance de réorganisation est le nombre de mots ignorés.

Formellement, nous pouvons noter : De la même manière que les modèles basés sur les mots, trouver la meilleure traduction t_{best} en utilisant les règles de bayes pour inverser la traduction et intégrer un modèle de langage p_{lm} est définie comme suit :

$$t_{best} = argmax_t\, p(t / s)$$
$$= argmax\ t\ p(s / t)\ plm^{(t)}$$

Cependant, pour le modèle basé sur les segments de phrases, le modèle *p(s|t)* est divisé en ***I*** segments et la traduction inverse est définie comme suit :

$$\mathrm{P(s|t)} = \prod_{\mathrm{i=1}}^{\mathrm{I}} \emptyset(s_i|t_i)\mathrm{d}(\mathrm{a_i^s} - \mathrm{a_{i-1}^e} - 1)$$

La première partie $\emptyset(s_i|t_i)$ est la probabilité de traduction que la phrase s_i soit la traduction de la phrase s_i cible. Cette probabilité est obtenue en extrayant la paire de phrases pour chaque paire de phrases. Puis calculez la fréquence relative du nombre d'extractions d'un couple de phrases particulier par rapport au nombre de phrases extraites. Défini comme :

$$\emptyset(s|t) = \frac{count(t|s)}{\sum_{t_i} count(t|s_i)}$$

La deuxième partie est un modèle de réorganisation basé sur la distance. Tel que $\mathrm{a_i^s}$ (début) est le premier mot de la phrase source et $\mathrm{a_{i-1}^e}$ est la position du dernier mot.

La distance de ré-ordonnancement est calculée en tant que $\mathrm{d}(\mathrm{a_i^s} - \mathrm{a_{i-1}^e} - 1)$. C'est le nombre de mots sautés en avant ou en sens inverse [1] lors de la prise de mots source hors séquence[1].

Le modèle de traduction standard basé sur les segments de phrases est la version la plus simple, mais offre généralement une qualité de traduction supérieure à celle des modèles IBM statistiques basés sur des mots. Ce modèle a été étendu et

[1] Pour plus de détail, veuillez consulter la référence [7].

amélioré, ce qui a amélioré les performances de traduction. Cependant, dans ce chapitre, nous n'entrerons pas dans les nombreuses extensions du modèle de traduction.

1.4.4 Modèles basés sur les syntaxes (Syntax-Based Models)

Alors que les modèles de TAS basés sur des phrases dominaient le marché, les approches basées sur la syntaxe se sont révélées être une alternative et une meilleure solution aux nombreux inconvénients qu'ils présentaient.

L'objectif des modèles de traduction automatique basés sur la syntaxe est d'intégrer une représentation de la syntaxe dans les systèmes statistiques afin d'obtenir de meilleurs résultats et de minimiser les efforts humains.

La nécessité d'utiliser la syntaxe dans la traduction est que de nombreux problèmes de traduction peuvent être mieux contournés (ou pris en charge) par la syntaxe. La syntaxe encourage également une sortie grammaticalement cohérente et ouvre la voie à des modèles plus motivés par la linguistique qui utilisent la sémantique.

La traduction basée sur la syntaxe alimente des systèmes TAS très puissants pour certaines paires de langues.

D'après ce que nous avons vu, le processus de traduction consiste à trouver une phrase cible ayant la probabilité la plus élevée étant donnée une phrase source correspondant à une distribution de probabilités. Pour ce faire, la modélisation d'une distribution de probabilités et les paramètres d'apprentissage des modèles, puis la séquence cible appelée « décodage » sont des étapes séquentielles.

Dans la section suivante, nous allons décrire ces étapes fondamentales du TAS et souligner la différence entre les techniques utilisées par les modèles basés sur la syntaxe. Nous allons principalement parler de l'une des approches les plus largement utilisées qui est la traduction automatique basée sur des phrases Hiérarchique.

1.4.5 Modèles basée sur des phrases Hiérarchique (Hierarchical phrase based MT)

La première étape consiste à modéliser une distribution de probabilités. Dans TA, il utilise presque la même technique que le model basée sur les phrases; C'est-à-dire utiliser des paires de phrases mais avec des imbrications modélisées à l'aide de la grammaire synchrone sans contexte (SCFG), qui permettent de générer simultanément une paire de chaînes liées. Autrement dit, spécifier la structure de deux phrases en même temps ; une dans la langue source et l'autre dans la langue cible. SCFG étant une grammaire sans contexte, traduit des phrases en les analysant, mais de manière synchrone en analysant Bitext.

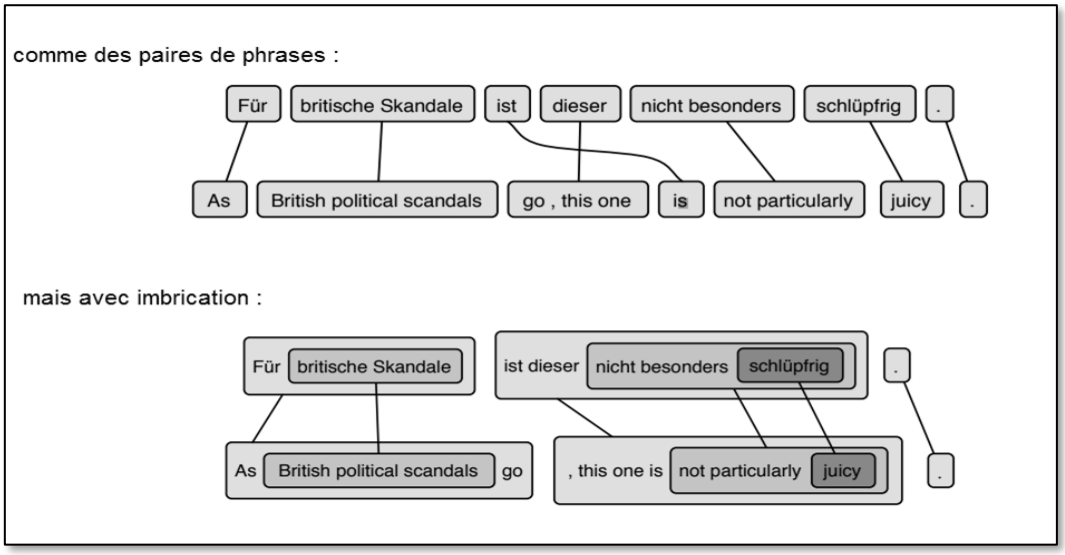


Figure 12: paires de phrases mais avec des imbrications modélisées à l'aide de la grammaire synchrone sans contexte [14].

Dans la grammaire synchrone sans contexte [14] :

- Il faut réécrire les règles de la forme (A, B) → (α, β, ∼).
- A et B sont des non-terminaux source et cible, respectivement.
- α et β sont des chaînes de terminaux et de non-terminaux pour les côtés source et cible, respectivement.
- ∼ est une correspondance un à un entre les non-terminaux source et cible.

La plupart des modèles TAS utilisent la formulation log-linéaire d'Och et Ney (2002) pour la probabilité de modélisation. Une reformulation en termes de dérivations SCFG est définie comme :

$$t^* = argmax_t \sum_{m=1}^{M} \lambda_m h_m(t|s)$$

$$= argmax_t \sum_{d} \sum_{m=1}^{M} \lambda_m h_m(t|s|d)$$

Où d ϵ D qui est l'ensemble des dérivations synchrones avec la source ***s*** et la sortie ***t***. Dans la pratique, t^* est approché avec la recherche de la meilleure dérivation d^* définie comme suit:

$$d^* = argmax_d \sum_{m=1}^{M} \lambda_m h_m(t|s|d)$$

L'apprentissage des 'paramètres' pour le modèle dans ce contexte est l'apprentissage du GSSC (grammaire synchrone sans contexte) qui consiste à extraire des règles à partir de textes parallèles fonctionne sur de grandes collections de paires de phrases alignées par mots. Ce sont des paires de traductions de phrases qui ont été automatiquement alignées avec des outils tels que GIZA ++ [4].

Le processus d'extraction de règles commence par identifier les paires de phrases initiales qui sont un sous-ensemble de toutes les paires de phrases possibles. Cet ensemble est :

- Cohérent.
- Contient au moins un point d'alignement positif.
- Ne contient pas de mots non alignés au bord des phrases.

Il est ensuite possible de créer des phrases hiérarchiques selon un processus de soustraction de phrases, dans lequel de courtes paires de phrases initiales incorporées dans des paires de phrases plus longues sont supprimées et remplacées par des non-terminaux. Les phrases supprimées deviennent des points de substitution étiquetés par X. ce processus est répété, ce qui dérive une grande grammaire en raison de la génération de multiples non-terminaux. Chiang (2007) fournit les contraintes à suivre :

- Une longueur maximale des paires de phrases initiales fixée à dix ;

- Du côté source, il peut y avoir au maximum cinq symboles ;
- Enfin, les mots non alignés ne sont pas autorisés sur les bords de la paire de phrases initiale.

Ces contraintes limitent la complexité de l'analyse. Notez que parallèlement à l'extraction des règles, un nombre important d'entités (features) sont extraites, associées à chaque règle. Ces caractéristiques sont utilisées dans le modèle du décodeur pour guider la recherche du décodeur vers de bonnes dérivations. Les règles extraites peuvent ensuite être utilisées pour traduire de nouvelles phrases avec un algorithme de décodage.

1.5 Evaluation de la Traduction Automatique

L'évaluation est un processus essentiel dans le domaine de la traduction automatique, à la fois pour déterminer l'efficacité des systèmes de traduction automatique existants et pour optimiser la performance de ces systèmes.

Idéalement, sans aucune contrainte de temps ni d'argent, les humains pourraient juger de la qualité de la traduction afin de donner une idée des performances du système. Évidemment, ce n'est pas le cas, car il est nécessaire de disposer d'un moyen rapide et peu coûteux d'évaluer les systèmes de traduction automatique.

L'idée de l'évaluation automatique est de comparer la sortie d'un système de traduction automatique à une traduction de référence (généralement humaine). On répond alors à la question : « A quel point la sortie du système est-elle proche de la traduction de référence ? »

Pour évaluer les performances de la traduction, on utilise des *unités de mesure de performance* connues dans le domaine de la traduction automatique. Ces mesures sont présentées ci-dessous.

1.5.1 Taux d'erreur de mots (WER)

C'est une dérivée de la *distance de Levenshtein,* en travaillant au niveau des mots au lieu des caractères. WER est calculée comme la somme des insertions, substitutions et suppressions, normalisée par la longueur de la phrase de référence. La formule pour la calculer est la suivante :

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C}$$

- S : est le nombre de substitutions.
- D: est le nombre de suppressions.
- I : est le nombre d'insertions.
- C: est le nombre de mots corrects.
- N: est le nombre de mots de la référence (N = S + D + C).

Plus le taux est faible (minimum 0.0) plus la traduction est bonne. Le taux maximum n'est pas borné et peut dépasser 1.0 en cas de très mauvaises traductions s'il y a beaucoup d'insertions.

1.5.2 Bilingual Evaluation Understudy (BLEU)

C'est un algorithme d'évaluation de la qualité d'un texte traduit automatiquement d'une langue vers une autre. Pour calculer le score BLEU entre une traduction candidate ***c*** et une traduction de référence ***r,*** (Papineni et al, 2002) d'IBM proposent la formule suivante [10] :

$$BLEU = \min\left(1, \frac{output-lenght}{reference-lenght}\right) * \left(\prod_{i=1}^{4} précison_i\right)^{\frac{1}{4}}$$

Ci-dessous un exemple pour illustrer la formule [10] :

SYSTEM A: Algerian officials responsibility of airport safety.

SYSTEM B: airport security algerian officials are responsible.

REFERENCE: algerian officials are responsible for airport security

Tableau 1: Exemple de l'évaluation d'une traduction selon BLEU score.

Mesure	SYSTEM A	SYSTEM B
Précision (1 gram)	3/6	6/6
Précision (2 gram)	1/5	4/5
Précision (3 gram)	0/4	2/4
Précision (4 gram)	0/3	1/3
Brevity penalty	6/7	6/7
BLUE Score	0=0%	0.52=52%

Le score BLEU est normalisé entre 0 et 1 et exprimé souvent en pourcentage.

1.5.3 Métrique d'évaluation de la traduction METEOR

C'est une mesure pour l'évaluation de la sortie de traduction automatique. La mesure est basée sur la moyenne harmonique de précision et de rappel uni-gramme, avec un rappel pondéré supérieur à la précision. Elle comporte également plusieurs fonctionnalités que l'on ne retrouve pas dans d'autres métriques, telles que le suivi et la correspondance de synonymie, ainsi que la correspondance de mots exacte [11].

Précision uni-gramme : fraction de mots dans la sortie du système qui apparaissent dans la référence.

Rappel uni-gramme : fraction des mots de la traduction de référence qui apparaissent dans la sortie du système.

ci-dessous un exemple pour illustrer le calcul de cette mesure [11] :

- Reference: "the Iraqi weapons are to be handed over to the army within two weeks".
- Output: "in two weeks Iraq's weapons will give army".

L'appariement :

- Reference: Iraqi weapons army two weeks.
- Output: two weeks Iraq's weapons army.

Score METEOR calculé comme un score Fmean réduit (paramètres d'origine α= 0.9 β= 3.0 γ= 0.5) :

- Facteur d'actualisation: DF = γ * (frag ^β)
- Fmean= P*R/ (α*P+ (1-α)*R)
- Résultat finale: Fmean * (1-DF)
- **Précision** (P)= 5/8 =0.625, **Rappel** (R)= 5/14 = 0.357
- Fmean= 10*P*R/(9P+R) = 0.3731
- Fragmentation: 3 frag de 5 mots = (3-1) / (5-1) = 0,50
- DF = 0.5 * (frag**3) = 0.0625
- Résultat finale = Fmean * (1-DF) = 0.3731*(1-0.0625) =0.3498

1.6 CONCLUSION

Au début de ce chapitre, on a étudié l'historique, les concepts et les approches classiques de la traduction automatique (TA), dont la plupart s'adressent principalement au domaine dans son sens le plus large. Et puis nous avons abordé les modèles de traduction automatique statistique (TAS) et leurs extensions et on a parlé aussi sur l'évaluation automatique de TA les Systèmes de TA.

Les Systèmes de TA actuels sont associés à de nombreux problèmes, à cause de l'incapacité du système à traiter les textes longs et à présenter des résultats adéquats.

CHAPITRE 2
TRAITEMENT AUTOMATIQUE DE LA LANGUE ARABE

2.1 INTRODUCTION

L'Arabe est l'une des principales langues du monde et se classe au cinquième rang pour le nombre de locuteurs natifs[2]. Elle est parlée par près de 500 millions de personnes dans le monde et c'est l'une des six langues officielles de l'ONU.

Au cours des dernières années, l'intérêt croissant pour la langue Arabe a donné lieu à des projets visant à développer une Traduction Automatique Arabe-Anglais rapide et précise ainsi que d'autres application de Traitement Automatique de la Langue (TAL) Arabe.

La langue Arabe présente des défis sérieux pour les chercheurs et développeurs d'applications de TAL pour le texte et le discours en Arabe. Parmi ces challenges, on trouve la Traduction Automatique. La Traduction Automatique entre deux langues est un problème très complexe, et l'Arabe le rend plus compliqué car elle diffère des autres langues par ses règles et ses propriétés linguistiques.

Dans ce chapitre on va aborder le traitement automatique de langue Arabe en mettant l'accent sur la Traduction Automatique de et vers l'Arabe.

2.2 TRAITEMENT AUTOMATIQUE DE LA LANGUE ARABE

La langue Arabe est à la fois difficile et intéressante. Intéressante en raison de son histoire [5, 6], l'importance stratégique de ses habitants et de la région qu'ils occupent, ainsi que son patrimoine culturel et littéraire [7]. C'est aussi un défi en raison de sa structure linguistique complexe.

L'Arabe est une langue sémitique parlée par plus de 500 million de personnes en tant que langue maternelle, dans une zone s'étendant du golfe Arabo-Persique à l'est jusqu'à l'océan Atlantique à l'ouest. L'Arabe est une langue très structurée et dérivée où la morphologie joue un rôle très important [5].

Les applications de TAL en Arabe doivent traiter plusieurs problèmes complexes liés à la nature et à la structure de la langue Arabe [5]. Par exemple, l'Arabe s'écrit

[2] https://www.fluentin3months.com/most-spoken-languages/#

de droite à gauche. Comme le chinois, le japonais et le coréen, il n'y a pas de capitalisation en Arabe. De plus, les lettres Arabes changent de forme en fonction de leur position dans le mot.

L'Arabe est une langue unique en termes d'histoire, de structure interne, de lien indissociable avec l'islam, et de culture et identité Arabes. Le défi que la langue Arabe pose aux chercheurs ne se limite pas aux aspects sociaux de la langue, mais s'applique également à sa structure linguistique inhérente [5].

Dans la suite de cette section nous aborderons les problèmes auxquels nous sommes confrontés lorsque nous traitons avec la langue Arabe.

2.2.1 L'écrit arabe

L'une des propriétés linguistiques clés de la langue Arabe qui pose un défi au traitement automatique de cette langue est l'écriture elle-même. Bien que l'Arabe soit une langue phonétique, c'est-à-dire Il y a un mappage un à un entre les caractères de la langue et les sons associés, mais l'Arabe est loin d'être une langue facile à lire en raison du manque de lettres dédiées pour représenter les voyelles courtes, changements dans la forme de la lettre en fonction de sa place dans le mot, et l'absence de majuscule et de ponctuation minimale [5].

Bien que l'écriture Arabe ne comporte pas des lettres dédiées pour représenter les voyelles courtes dans la langue, les voyelles courtes ont été représentées par des signes diacritiques qui sont des marques situées au-dessus ou au-dessous des lettres. Il n'en reste pas moins que cet absence dans les textes "Arabe standard moderne" rend difficile l'apprentissage de la langue pour les locuteurs dont l'Arabe n'est pas la langue native et présente des défis au traitement l'automatique de l'Arabe.

Les lettres Arabes ont des formes différentes selon leur position dans le mot. Par exemple, la lettre (ع) "ain" a une forme initiale (عـ), une forme médiane (ـعـ), une forme de connexion finale (ـع) et une forme finale sans connexion (ع). Le choix de la forme correcte par rapport à sa position dans le mot est régi par des règles.

De plus, l'outil de traitement morphologique doit gérer certaines formes. Par exemple, la lettre « ء » Hamza est remplacée par d'autres formes au cours des générations morphologiques et syntaxiques du mot infléchi. Par exemple, l'utilisation de la lettre "ي " (Yeh pour indiquer la

possession), avec le pluriel irrégulier “أصدقاء ” (amis) produit “أصدقائي ” (mes amis) au lieu de “أصدقاء ي ”.

En Anglais et dans d'autres langues utilisant des scripts latins, la plupart des phrases commencent par une lettre majuscule et se terminent par un point. Dans les applications TAL telles que la Traduction Automatique, la Recherche d’Informations et le Résume de textes, il est nécessaire de diviser un texte correctement en phrases, mais des scripts tels que l'Arabe, le chinois et le coréen n'ont pas de majuscule ni de règles strictes de ponctuation et leur absence rend le travail de prétraitement d'un texte plus difficile.

Alors pour un linguiste informaticien développant des applications TAL pour des langues telles que l'Arabe, où le script ne permet pas la capitalisation et ne suit pas des règles de ponctuation strictes, il est important de comprendre la structure et la syntaxe de la langue Arabe pour identifier des modèles en l'absence de ces règles [8].

2.2.2 La Normalisation de l’écrit Arabe

Les chercheurs et les développeurs de la linguistique informatique Arabe se heurtent à un autre défi : le dilemme de la normalisation. Le problème se pose du fait de l’incohérence dans l’utilisation des marques diacritiques et de certaines lettres dans les textes Arabes contemporains.

Certaines lettres Arabes ont la même forme et ne sont marquées qu'en ajoutant certaines marques, telles qu'un point, Hamza ou maddah placé au-dessus ou au-dessous de la lettre.

Par exemple, “alif” en Arabe (ا) peut être composé de trois lettres différentes selon qu’il ait une hamza en haut comme dans (أ) ou une hamza en bas comme en (إ) ou une maddah en haut comme (آ). Reconnaître ces marques au-dessus ou au-dessous d'une lettre est essentiel pour pouvoir distinguer des lettres similaires [5].

Mais les textes écrits en “Arabe standard moderne” n'intègrent souvent pas la voyelle, comme mentionné précédemment, et n'adhèrent pas à l'inclusion “correcte” de marques au-dessus ou au-dessous de certaines lettres Arabes. Pour gérer ce problème, la technique courante dans les systèmes de TAL Arabe consiste à normaliser le texte saisi [9].

Par exemple, afin de gérer les différentes variations de l'écriture Arabe, [9] remplacent alif hamza situé au-dessus ou au-dessous initial par un simple alif. Ils normalisent également l'alif maddah avec un simple alif.

L'analyseur statistique de Stanford Arabic conçu par le Groupe de traitement du langage naturel de Stanford met en œuvre une stratégie de normalisation similaire pour les textes Arabes [5].

Le système de Traduction Automatique SYSTRAN Arabe-Anglais [10] a également incorporé la normalisation. Mais il est vite apparu que, bien que la normalisation améliore la reconnaissance en résolvant la variabilité des entrées, elle augmente la probabilité d'ambiguïté [11].

Par exemple, normaliser un alif initial avec une hamza au-dessus ou au-dessous de celui-ci supprime une distinction importante entre (أن) ann et (إن) inn. Le premier se traduisant par "That" et doit être suivi d'une phrase nominale et le second pourrait se traduire par "to" qui indique l'infinitif Anglais, mais dont la traduction n'a pas de sens si elle est suivie d'un nom. Par conséquent, Bien que la normalisation résolve les problèmes de reconnaissance, elle crée l'effet involontaire d'une ambiguïté [1].

Tant que nous avons parlé d'ambiguïté qui est considérée comme un problème complexe dans toutes les langues et en particulier l'Arabe, allons plus en profondeur dans cette notion.

2.2.3 L'ambiguïté dans la Langue Arabe

Bien que l'ambiguïté soit principalement à cause de l'absence de voyelles courtes, les chercheurs de SYSTRAN ont constaté que l'ambiguïté en Arabe était présente à d'autres niveaux [5]. Malheureusement, l'ambiguïté de ces niveaux est plus complexe pour pouvoir être exprimée dans une petite section. Toujours est-il, nous donnons dans ce qui suit certains exemples de l'ambiguïté juste pour illustrer le problème :

- *Les homographes :* Un mot appartenant à plus d'une partie du discours, tel que (قدم) qui pourrait être un verbe qui signifie ***"to introduce"*** ou un verbe qui signifie ***"to arrive from"*** ou un nom qui signifie ***"foot"***. Certaines ambiguïtés de l'homographe peuvent être résolues par des règles contextuelles [5].

- *L'ambiguïté syntaxique :* Comme dans le cas d'un attachement prépositionnel, par exemple "قابلت مدير البنك الجديد " ce qui pourrait signifier **"I met with the new bank manager"** ou **"I met with the manager of the new bank"**.
- *L'ambiguïté sémantique :* Les phrases peuvent être interprétées de différentes manières, par exemple "يحب علي أحمد أكثرمن إبراهيم" Est-ce que cela signifie que Ali aime Ahmed plus que Ibrahim ? ou Ali et Ibrahim aiment-ils Ahmed, mais Ali aime Ahmed plus que Ibrahim aime Ahmed ?
- *L'ambiguïté des limites du constituant* : Par exemple, "مدير البنك الجديد" **"the new manager of the bank"** ou **"the manager of the new bank"**, en fonction de la limite de l'adjectif dans cette construction de noms.
- *L'ambiguïté anaphorique :* Comme dans, "قال علي أنه نجح" en Anglais "**Ali said that he succeeded**". Cette phrase est ambiguë en Anglais et en Arabe. La question qui se pose ici est : est-ce que "**he**" se réfère à Ali ou à quelqu'un d'autre ?

Dans cette section nous avons parlé sur le domaine du traitement automatique de la langue en particulier l'Arabe et les défis auxquels les chercheurs font habituellement face. Dans la section qui suit, nous allons aborder une tâche spécifique du TAL qui est la Traduction Automatique de et vers la langue Arabe.

2.3 TRADUCTION AUTOMATIQUE DE LA LANGUE ARABE

L'intérêt pour la langue Arabe a augmenté à mesure qu'elle a gagné en notoriété pour des raisons politiques, stratégiques et commerciales. Cet intérêt a donné lieu à des projets visant à développer une Traduction Automatique Arabe-Anglais rapide et précise. Ces efforts ont principalement été menés par le ministère Américain de la Défense, des leaders industriels tels que Google ou IBM, ainsi que par quelques petites entreprises en Europe, aux États-Unis et en Égypte. Nombre de ces efforts ont été consacrés à la traduction de l'Arabe vers l'Anglais pour des raisons stratégiques, mais les deux directions de traduction posent des problèmes essentiellement différents en raison des différences morphologiques des deux langues [12].

2.3.1 Les défis de traduction Arabe

La langue Arabe, comme les autres langues sémitiques, se caractérise par une morphologie relativement complexe [12]. Cette complexité morphologique pose un

défi à la Traduction Automatique de l'Arabe à l'Anglais ou des langues similaires, qui se manifeste différemment dans différents paradigmes. Par exemple Les approches de Traduction Automatique basées sur des données. Le défi est que le corpus Arabe sera moins volumineux qu'un corpus Anglais équivalent, parce que le nombre moyen d'instances observées d'un mot Arabe morphologiquement complexe sera plus bas.

Dans certains cas, le système de Traduction Automatique pourrait rencontrer une nouvelle forme d'un mot, et ne pas être capable de la traduire même s'il avait été entraîné sur des données contenant une forme morphologique différente de ce mot.

L'absence de signes diacritiques dans le texte Arabe crée une ambiguïté dans la prononciation et l'ambiguïté lexicale, puisque deux mots différents, avec des significations différentes, peuvent partager la même orthographe non diacritisée. De plus, l'ambiguïté de la source, un phénomène commun à toutes les langues humaines, est exacerbée par le manque de diacritiques dans le cas de l'Arabe. Les humains s'appuient sur le contexte et sur leurs connaissances du lexique pour résoudre cette ambiguïté lors de la lecture d'un texte Arabe. Mais cette capacité est loin d'être facilement disponible pour un système de Traduction Automatique.

Certains systèmes de Traduction Automatique, en particulier les systèmes à base de connaissances, diacritent automatiquement le texte Arabe dans le cadre de l'analyse qu'ils effectuent. D'autres systèmes, y compris ceux basés sur des exemples et des statistiques, traitent implicitement de l'ambiguïté lexicale. Plutôt que d'essayer de résoudre explicitement l'ambiguïté, ils permettent au contexte environnant, entre autres facteurs, d'affecter la décision de traduction.

Les différences considérables entre la syntaxe de l'Arabe et celle de l'Anglais et des autres langues européennes sont une autre source de difficulté pour la Traduction Automatique en Arabe. Par exemple la différence dans la structure du verbe, la position des adjectifs par rapport aux noms qu'ils décrivent. Toutes ces difficultés rendent la tâche de traduction loin d'être simple et parfaite.

2.3.2 Ressources linguistiques en Arabe

La Traduction Automatique basée sur des exemples et sur les statistiques dépend de l'existence de corpus de texte [12]. Des corpus bilingues de texte parallèle forment la base pour la construction de systèmes de Traduction Automatique dans ces deux paradigmes axés sur les données. Et les corpus monolingues, en particulier dans la langue cible, sont également importants pour la construction de modèles de langage, spécialement dans la Traduction Automatique statistique.

Les ressources linguistiques en Arabe ont été développées par le Consortium de Données Linguistiques (LDC) au cours des deux dernières décennies. Ces ressources ont permis une grande partie de la recherche sur le traitement de la langue naturelle Arabe et la Traduction Automatique Arabe.

Ces corpus couvrent différents types de données : paroles, textes, vidéos et lexiques. Ils couvrent également différentes variétés de l'Arabe : l'Arabe moderne standard ainsi que les dialectes égyptiens, du golfe et levantin. Ces corpus de phrases parallèles entre Arabe et Anglais ont été particulièrement pertinents pour la recherche en Traduction Automatique.

LDC a développé de grands corpus parallèles, qui ont été utilisés pour la formation des modèles de traduction, ainsi que des données pour les évaluations des systèmes de Traduction Automatique open source du NIST.

The Penn Arabic Treebank (ATB), est un corpus de phrases en Arabe avec des arbres syntaxiques complets a également été important pour la Traduction Automatique en Arabe basée sur la syntaxe.

The Standard Arabic Morphological Analyzer (SAMA) et son prédécesseur, Buckwalter Arabic Morphological Analyzer (BAMA), sont des outils d'analyse morphologique - avec lexique correspondant - qui déterminent les décompositions morphologiques de mots Arabes.

2.3.3 Les approches connus de Traduction Automatique de et vers l'Arabe

Comme on a vu précédemment, il y a plusieurs approches de Traduction Automatique et presque chaque approche couvre plusieurs modèles. Le but de ces modèles est de résoudre certains problèmes qui diffèrent d'une langue à l'autre, Par conséquent, les résultats d'évaluation d'un modèle diffèrent d'une langue à l'autre.

Dans cette section on va parler des approches qui ont fait la preuve de leur efficacité à traduire de et vers l'Arabe en commençant par l'approche basé sur des exemples.

i. La Traduction Automatique basé sur des exemples

La Traduction Automatique basée sur des exemples [12], proposée pour la première fois par (Nagao 1984), est une approche de la Traduction Automatique basée sur les données qui tente d'effectuer la traduction par "analogie ". Un grand corpus d'exemples de phrases traduits est utilisé pour produire une traduction pour une nouvelle phrase.

Comme nous l'avons vu dans l'approche statistique, le corpus de phrases traduites (corpus bilingue) est utilisé pour créer un modèle de traduction et les probabilités associées au moment de la formation (entrainement), qui sont ensuite

utilisés au moment de l'exécution (ou du décodage) pour générer des traductions pour les nouvelles phrases.

Par ailleurs, dans le cas de modèles basés sur des exemples, les phrases exemplaires sont utilisées directement pour générer les nouvelles traductions, en faisant d'abord correspondre la phrase d'entrée avec des exemples de phrases complètes ou des fragments de phrases, puis en composant les fragments appariés et en les adaptant de manière appropriée pour produire la traduction de la nouvelle phrase.

Le défi de Traduction Automatique basée sur des exemples consiste à élargir les phrases d'entrée qui peuvent correspondre à la traduction dans le corpus d'exemples. Parce que lors de la traduction de l'Arabe, la chance de correspondance exacte entre la forme de l'entrée et les exemples de phrase sont considérablement réduites par la complexité morphologique de la source Arabe.

Pour surmonter ce défi Cavalli-Sforza et Phillips utilisent une analyse morphologique de l'entrée Arabe pour améliorer la qualité de la Traduction Automatique Arabe-Anglais basée sur des exemples. Ils proposent de généraliser des formes morphologiques observées dans la phrase d'entrée aux nouvelles formes morphologiques à l'aide d'analyses morphologiques des mots d'entrée. Ils montrent ainsi une amélioration significative de la qualité de la traduction, en particulier avec de petites quantités d'exemples de données, où le problème de rareté est plus grave.

Un autre modèle par Bar et Dershowitz propose d'utiliser l'équivalence sémantique entre les mots Arabes pour élargir l'espace des phrases correspondantes. L'effet de l'utilisation de synonymes pour les noms et les verbes sur les performances d'un système de Traduction Automatique basé sur des exemples est étudié. Ce travail utilise également les informations morphologiques de " The Buckwalter Morphological Analyzer " pour faire correspondre les mots à des niveaux de représentation plus profonds, tels que stems et lemmes.

ii. La Traduction Automatique Hybride

Les approches hybrides de la Traduction Automatique combinent des méthodes issues de différents paradigmes, en tirant parti de la force de chacun. La motivation

pour développer des systèmes de Traduction Automatique hybride découle de l'incapacité d'une technique donnée à atteindre un niveau de précision satisfaisant.

Parmi les modèles hybrides pour la Traduction Automatique de l'Arabe on a la méthode propose par Shaalan et Hossny [12]. Elle consiste à apprendre les règles de traduction en utilisant les langages de programmation inductifs (IPL), une technique d'apprentissage automatique qui utilise la programmation logique pour apprendre des hypothèses à partir d'exemples.

Le modèle de traduction IPL dans un système de traduction à base de règles. Contient d'abord, une phase d'apprentissage des règles de l'analyse morphologique en utilisant IPL, et aussi les règles de transfert entre des fragments de mots du côté Arabe et leur traduction Anglaise correspondante.

Le modèle peut apprendre un nombre considérable de règles uniques à partir d'un petit nombre de phrases parallèles, ce qui est l'un des principaux avantages par rapport aux approches basées sur des corpus nécessitant de très grands corpus parallèles.

iii. La Traduction Automatique neurale pour l'Arabe

La Traduction Automatique neurale (TAN) est devenue une méthode hautement privilégiée et est considérée comme étant meilleure que les modèles traditionnels de Traduction Automatique statistique (TAS) [13].

TAN est conçu pour imiter les neurones du cerveau humain. Les neurones peuvent créer des liens, apprendre de nouvelles informations et évaluer les entrées dans leur ensemble plutôt que partie par partie.

Ce processus de Traduction Automatique neuronale ne génère pas de phrases dans une langue cible, contrairement à TAS, TAN effectue plutôt une analyse en deux étapes, codage et décodage. Durant la phase de codage, le texte en langue source est envoyé à la machine, puis trié en une série de vecteurs linguistiques. Durant la phase de décodage, ces vecteurs sont transformés dans la langue cible sans étape de génération.

Bentivogli et Luisa ont élaboré les résultats expérimentaux des comparaisons entre les modèles TAS et TAN et ont fait le constat que dans de nombreux cas, les résultats mis en évidence par le biais du TAN sont supérieurs à ceux obtenus à l'aide des modèles TAS. La caractéristique du TA N qui le rend unique et qui en fait un favori des autres modèles est sa puissance en différents dialectes Arabe, il a donné des résultats étonnants par rapport aux autres modèles.

Par exemple dans le modèle proposé par [13] qui étudie le problème de l'utilisation d'un modèle de Traduction Automatique neurale pour traduire les dialectes Arabes en Arabe standard moderne, les auteurs proposent le développement d'un modèle d'apprentissage multitâche qui partage un décodeur entre des paires de langues, et chaque langue source possède un encodeur séparé.

Le modèle proposé peut être appliqué à des volumes de données limités ainsi qu'à des quantités de données importantes. Les expériences effectuées ont montré que le modèle d'apprentissage multitâche proposé peut assurer une traduction de meilleure qualité que les modèle à apprentissage individuel.

Dans cette section, nous avons présenté quelques exemples généraux sur les déférentes approches et modèles qui tentent d'améliorer la Traduction Automatique de et vers l'Arabe, Il y a ceux qui peuvent analyser la morphologie ou la sémantique, d'autres sont des modèles combinés. Pour finir, nous avons vu également la Traduction Automatique neuronale, qui est devenue extrêmement compétitive même s'il s'agit d'une nouvelle approche. Dans la section suivante on va aborder les modèles inspirés par l'approche de Traduction Automatique statistique.

2.4 LA TRADUCTION AUTOMATIQUE STATISTIQUE POUR L'ARABE

La Traduction Automatique de et vers l'Arabe rencontre les mêmes problèmes que la traduction entre toutes les langues, ainsi que certaines problématiques spécifiques à la langue Arabe (diacritiques ou voyelles courtes).

Selon [Arnold et al., 1993] [14]. les défis et les difficultés de la TA en général peuvent être classés en trois catégories principales :

- Problèmes d'ambiguïté.

- Problèmes liés aux différences structurelles et lexicales entre les langues.
- Unités multi-mots comme les idiomes et les collocations semblables à la traduction humaine.

La similitude entre un couple de langues influence généralement la facilité de la traduction, où plus les deux langues sont similaires, plus la traduction est facile. Dans le cas de pair de langues non similaires Anglais-Arabe, de nombreux défis et difficultés peuvent subsistent. Ces défis peuvent être classés dans quatre catégories: les problèmes d'ambiguïté, la similarité des paires de langues, les problèmes et les défis liés à les approches de la Traduction Automatique ainsi que les problèmes humains .

2.4.1 Problèmes et défis

4 Problèmes d'ambiguïté

- *Ambiguïté lexicale :* L'ambiguïté lexicale se produit lorsqu'un mot peut avoir plus d'un sens, du côté Anglais un cas d'ambiguïté lexicale lorsque le mot comporte deux catégories lexicales ou plus (par exemple, *Tank* en tant que nom. *Tank* en tant que verbe). Le nom fait référence à un véhicule militaire et le verbe n'a que peu d'interprétations différentes mais proches selon le contexte. Dans ce cas, la désambiguïsation peut être réalisée en utilisant un Part-of-Speech (POS) tagger [15]. Un autre cas peut se présenter lorsqu'un mot possède une ou plusieurs significations dans la même catégorie lexicale (par exemple, une *banque* en tant qu'institution financière. Une *banque* comme dans une rivière).
  Du côté de la langue Arabe, l'omission des voyelles courtes (diacritiques) et parfois des points sur les lettres augmente l'ambiguïté lexicale, cependant, les locuteurs natifs peuvent toujours comprendre le sens correct du contexte. La langue Arabe a peu de zones qui génèrent de l'ambiguïté :
  - Quand un mot a deux catégories lexicales ou plus, il peut causer des ambiguïtés lexicales. Exemple (1) illustre ce phénomène

    ذهب أحمد *(1)*

    Dans l'exemple (1), le mot ذهب peut faire référence au nom *or* (***Gold*** en Anglais) ou au verbe *aller* (to ***go*** en Anglais) de sorte que les traductions possibles seraient : « Ahmad est allé / est parti » ou bien « L'or d'Ahmad ».

Dans ce cas, le processus de traduction devra résoudre ces problèmes en utilisant des techniques de désambiguïsation du sens des mots, de manière implicite ou explicite, un moyen de résoudre ce problème dans le phrase-based TAS consiste à traduire une grande phrase pour capturer plus de contexte,

Cependant, le système ne pourra pas générer la traduction correcte si la phrase source n'a pas été vue auparavant dans les données d'apprentissage.

- Le signe d'emphase (Shadda)

En Arabe, le signe d'emphase Shadda équivaut à écrire deux fois la même lettre, le premier contenant « skoon » et le second une des voyelles courtes « harakat », qui signifie littéralement « mouvements ». Ce sont des marques utilisées comme guides phonétiques {'Fathah', 'Kasrah', 'dammah'}. Par exemple, le mot فَضَّلَ est en fait فَضْضَلَ. Mais au lieu d'écrire ض deux fois, il est remplacé par un ض avec « shadda » [27]. La shadda change le sens du mot, en prenant l'exemple précédent فَضَلَ sans shadda signifie «be in excess (être en excès)», «remain (reste)», «be surplus (sois en surplus)» . La shaddah peut introduire une ambiguïté et, en même temps, peut être utilisée pour lever une ambiguïté en distinguant la catégorie lexicale si la shadda est prise en compte. Par exemple, le mot قبل (before) est doublement ambigu (nom ou verbe), mais après l'insertion de Shaddah, l'ambiguïté diminue à une catégorie (verbe قبّل) (kissed)

- Hamza

Hamza, bien qu'elle ne fait pas partie des 28 lettres de l'alphabet, la hamza peut être considérée comme une lettre dans l'aphabet Arabe. La présence de hamza dans un mot réduit les ambiguïtés et réduit la catégorie lexicale du mot [16], par exemple, le mot « ان » peut avoir plusieurs significations et il est aussi doublement ambigu (verbe et nom), mais si nous ajoutons la hamza, nous réduisons le nombre de catégories lexicales et de même les ambiguïtés "ان"→ " إن " [26].

Il existe d'autres types d'ambiguïtés lexicales courantes, telles que l'agglutination, où, en langue Arabe, des particules, des prépositions et des pronoms peuvent être attachés aux adjectifs, noms, verbes et particules auxquels ils se rapportent. Par exemple, le caractère «ف» dans le mot «فصـل» (saison) est un caractère original, tandis que dans le mot «فصل» (*il prie alors*), il s'agit plutôt d'un proclitique. Un autre type courant est celui des mots composés où un type supplémentaire d'ambiguïté est introduit. A savoir l'adjonction entre les mots et l'interprétation de la lecture. Par exemple "الشّاشـة الصّـغيرة" est un nom composé et chaque mot est une unité distincte mais sera traité comme une unité minimale en tant que telle, il devra contenir des informations syntaxiques pertinentes telles que la forme et le type de complément de cette unité. Un autre exemple est le suivant :

- استعمل الحاسوب المحمول في العربة I use a laptop in the vehicle
- استعمل الحاسوب المحمـول فـي العربـة I use the computer, which is portable in the vehicle.

Dans cet exemple, le mot composé est lié à la souplesse de lecture et à la compréhension de la phrase. Il en résulte deux interprétations différentes où nous trouvons une lecture stricte puisque le mot composé pris en compte est «الحاسـوب المحمـول» mais que le second prend «المحمـول فـي العربـة» en tant que nom composé qui peut introduire une ambiguïté en fonction du sens recherché.

Un autre problème lexical qui génère une ambiguïté est la divergence lexicale ,où un mot pourrait se traduire par un autre mot ayant un sens différent, par exemple le mot Anglais « *Watch* » pourrait se traduire en Arabe par : « مشاهدة », « ساعة يد», « راقب ».

La traduction nécessite souvent de résoudre les mêmes problèmes que la désambiguïsation du sens des mots.

5 Similarité de langages

Les caractéristiques de langage peuvent être utilisées pour déterminer la similarité entre un couple de langues, nous allons maintenant passer en revue les caractéristiques morphologiques et syntaxiques.

- ***Morphologie*** : L'Arabe est une langue morphologiquement riche alors que d'autres langues ont une morphologie relativement plus simple comme l'Anglais. Les caractéristiques morphologiques peuvent tirer peu de différences entre les langues telles que le nombre de morphèmes par mot où certaines langues comme l'Arabe, chaque mot ont plusieurs morphèmes. Ces langues sont appelées langues polysynthétiques [Jurafsky et Martin, 2000] [17]. En outre, la langue Arabe étant une langue agglutinante dans laquelle un seul affixe peut associer plusieurs morphèmes, il est difficile de segmenter le mot en morphèmes, ce qui est un processus important du prétraitement visant à réduire les différences morphologiques entre les langues source et cible. En plus des inflexions morphologiques complexes, le traitement de l'Arabe partage les mêmes problèmes que le langage riche en morphologie.
- ***Syntaxe*** : Les langues peuvent avoir différentes structures de phrases comme l'Arabe et l'Anglais, alors qu'en Anglais la structure de phrase est Sujet-Verbe-Objet, en arabe nous trouvons un ordre syntaxique plus souple qui pourrait être SVO ou VSO ou VOS ou même S-Predicate (P) [14]. Différentes structures syntaxiques introduisent des difficultés lors de la traduction, nécessitant davantage de réorganisation de la traduction pour correspondre à la structure syntaxique. En tant que tel, plus la structure syntaxique de la paire de langues est différente, plus la traduction nécessite des efforts.

De plus, les différences idiosyncratiques font également partie des défis de la traduction entre paires de langue.

Les ressources constituent un autre défi majeur pour la Traduction Automatique qui relève des défis liés à l'homme. De nos jours, le Web est l'une des sources communes des corpus et la diversité du contenu et des utilisateurs d'Internet ayant des antécédents et des niveaux d'éducation différents. Ceci a une influence directe sur le comportement en écriture et introduisent des erreurs orthographiques parfois pas de simples lettres manquantes, comme dans des langues comme l'Anglais, mais

en substituant certaines lettres à d'autres proches qui sont proches de la prononciation ou en raison du contexte de l'utilisateur ou même des limitations technologiques telles que l'absence d'un clavier Arabe complet. Un autre phénomène consiste à répéter une lettre pour souligner un mot exprimant une action ou des émotions.

Pour la Traduction Automatique statistique, de telles erreurs auront un impact sur l'alignement des mots, la traduction et les modèles linguistiques. Dans un TAS pour résoudre ce problème, nous pouvons soit former notre système sur de telles données et lui permettre d'accepter et de traduire des mots erronés, soit effectuer une phase de prétraitement et une normalisation à chaque étape pour corriger les fautes d'orthographe.

Enfin, il existe des problèmes spécifiques à l'approche de la Traduction Automatique , par exemple dans les approches basées sur des corpus, nous utilisons des corpus bilingues et monolingues et donc un vocabulaire fermé qui conduit à plusieurs problèmes : certains mots sources ne seront pas traduits car ils sont inconnus du modèle de traduction. De même, certains mots cibles inconnus du modèle de langage et peut-être existe-t-il un décalage entre les corpus d'entraînement bilingue et monolingue et la tâche de traduction où les données utilisées par exemple, c'est l'Arabe standard mais utilisé pour traduire un texte dialectal ou informel tel qu'un dialecte algérien. L'ordre des mots est également un problème, car plusieurs ordres peuvent être corrects, mais la flexion peut être différente dans chaque ordre. Les ressources de données limitées causent un problème de dispersion des données, la fréquence d'occurrence du mot dans les données d'apprentissage est en corrélation avec la qualité de la traduction, où une fréquence basse peut poser des problèmes pour la modélisation statistique et si le mot ne se produit jamais, cela pose un problème de manque de vocabulaire qui est traité en utilisant plus de données.

Ceci, conclus notre aperçu des défis de la traduction Arabe. Dans la suite, nous aborderons quelques techniques différentes utilisées dans TAS qui visent à améliorer la qualité de la traduction Anglais-Arabe dans un système de Traduction Automatique statistique.

Précédemment, nous avons parcouru le pipeline de Traduction Automatique statistique, où le processus de traduction pour différentes paires de langues est identique, mais certaines paires de langues nécessitent un peu de modifications du système TAS pour améliorer la qualité de la traduction. Dans la section suivante, nous allons passer en revue de telles modifications qui ont été effectuées dans TAS pour le couple de langues Anglais-Arabe.

2.4.2 Modifications du Système Traduction Automatique

Un certain nombre de chercheurs ont tenté d'améliorer la traduction de la paire de langues Anglais-Arabe dans un système de Traduction Automatique statistique en introduisant des modifications dans le pipeline TAS tant sur le prétraitement et post-traitement des deux côtés du corpus parallèle. Ainsi que la modification dans le modèle de langage.

Comme nous l'avons vu dans la section précédente concernant les défis de la Traduction Automatique Arabe, les techniques de prétraitement et de post-traitement visent à résoudre divers problèmes liés à la langue Arabe. De même, la langue Anglaise subit un prétraitement et un post-traitement afin d'améliorer la traduction du couple de langues [16]. Les techniques actuelles de prétraitement de l'Arabe sont les suivantes : normalisation orthographique, tokenisation/décomposition morphologique et réorganisation syntaxique. Tandis que les techniques actuelles de post-traitement de l'Arabe sont : enrichissement orthographique et détokénisation / recombinaison morphologique. Les techniques actuelles de prétraitement en Anglais sont : mise au rebut, séparation de la ponctuation des mots et séparation ('s). Les améliorations en cours de processus se trouvent dans le modèle de langage. Ils ont utilisé le modèle de langage factorisé (FLM) qui intègre des fonctionnalités supplémentaires à chaque mot : Part-of-speech (POS), nombre, sexe, classe sémantique, etc.

i. Prétraitement

Une grande partie des travaux sur la TAS à partir de langages riches en morphologie a montré que la segmentation morphologique et la normalisation orthographique aident à améliorer la qualité du TAS en raison de la réduction de leur faible densité.

- *Normalisation orthographique* : Badr et al [18]. Dans ont nommé ce processus en tant que processus de normalisation. Cependant, El Kholy et Habash dans [19] l'ont appelé une normalisation orthographique. La normalisation orthographique a deux formes : Enrichie (ENR) et Réduite (RED). La forme "RED" est un processus réducteur qui convertit toutes les formes "Hamzated Alif" (إ ، أ ، à nu Alif (ا) et Ya sans points ou Alif Maqsura (ى) en pointillés Ya (ي). Le formulaire ENR présente une différence par rapport au formulaire "RED" qui consiste à sélectionner la forme appropriée de l'Alif. El Kholy et Habash dans [19] ont proposé les deux formes et les ont nommées. Mais la forme "RED" a également été utilisée par [18]. Il est bien connu que l'Arabe ENR est la forme à générer et à évaluer.
- *Tokenisation morphologique* : La segmentation du mot Arabe en morphèmes est un processus important avant la formation des données. Le marquage en texte Arabe a pour but de réduire la rareté et le nombre de mots hors vocabulaire (OOV). Badr et al., dans [18] ont nommé ce processus « Segmentation du texte ».
- *Réorganisation syntaxique* : Pour améliorer l'alignement entre l'Anglais et l'Arabe dans la traduction de l'Anglais vers l'Arabe, Badr et al., dans [18] ont proposé un ensemble de règles sur la source Anglaise pour mieux s'aligner sur la traduction Arabe. L'implémentation a été réalisée en utilisant un arbre d'analyse.

Du côté Anglais, le texte source suit plusieurs étapes, comme illustré à la *figure 13.*

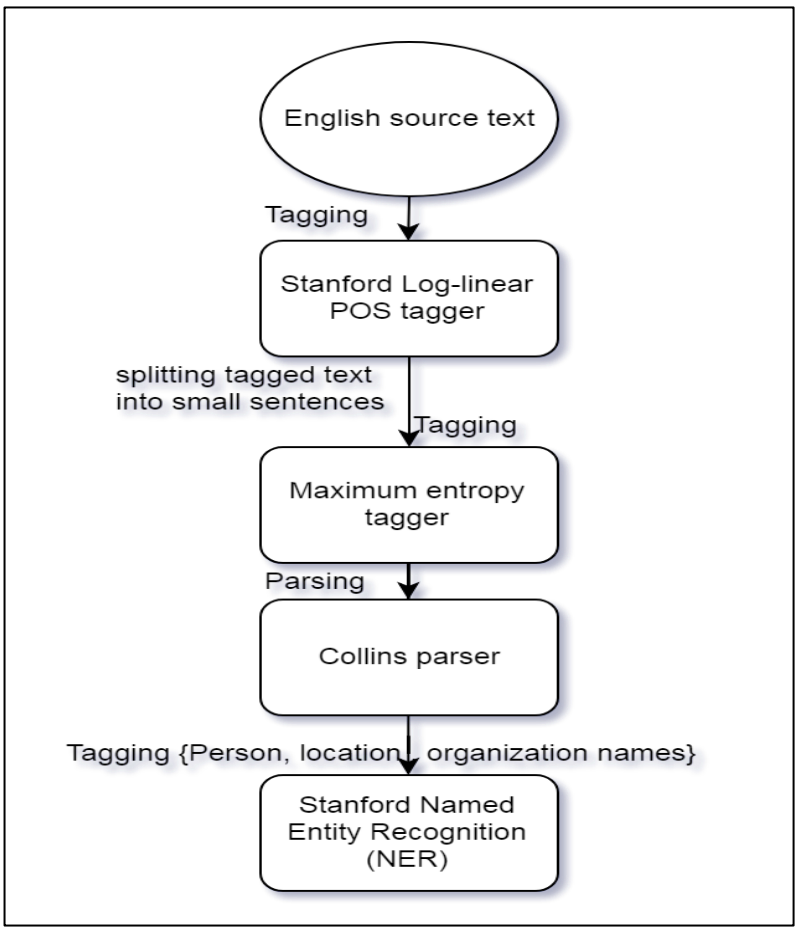


Figure 13: Étapes de prétraitement du texte source anglais.

Du côté Arabe, ils normalisent le texte, et le segmentent ensuite à l'aide de *Morphological Analysis and Disambiguation for Arabic (MADA) toolkit.*

Badr et al [18] ont prouvé que les règles proposées font un gain significatif. De plus, ils ont intégré le réordonnancement syntaxique avec des techniques de décomposition morphologique et de recombinaison. Habash dans [19] a réalisé un essai de réorganisation syntaxique de la traduction Arabe-Anglais et de réorganisation de la langue source qui est ici l'Arabe. Il a montré que si la qualité de l'analyse n'est pas bonne, elle reflétera la traduction aussi. C'est parce que l'analyse Arabe a un travail limité par rapport à l'analyse syntaxique en Anglais

ii. Post-traitement

- *Détokénisation morphologique* : Cela semble être le processus inverse de la tokenisation morphologique. Badr et al. dans [18] proposent quatre schémas pour recombiner le texte traduit en Arabe: Simple(S), Rule-based(R), Table-based (T) and Table+Rule(T+R). Ils appellent le processus recombinaison morphologique El Kholy et Habash dans [19] l'ont nommé Détokénisation morphologique. Tandis qu'El Kholy et Habash dans [19] ont ajouté deux autres schémas de détokénisation / recombinaison pour le texte traduit en Arabe à savoir Table+Language Modeling (T+LM) and Table+Rule Language Modeling (T+R+LM).. Badr et al. dans [18] ont rapporté que la technique (T + R) était la plus performante dans les expériences. De plus, El Kholy et Habash dans [19] ont rapporté que la technique (T + R + LM) était la meilleure dans toutes les conditions.

- *Enrichissement orthographique et détokénisation* : El Kholy et Habash dans [19] ont proposé deux techniques d'enrichissement orthographique et de détokénisation.

La production en tokens réduite doit être enrichie et décomposée pour produire un Arabe correct. La première technique consiste à utiliser la boîte à outils MADA (Analyse Morphologique et Ambiguïté en Arabe) [20] pour enrichir le texte réduit détokénisé (MADA-ENR). L'autre technique consiste à décomposer et à enrichir en une seule étape (Joint-DETOK-ENR).

La technique conjointe de [19] était préférable à la réalisation des deux tâches en deux étapes distinctes. La meilleure configuration pour le MT dans son ensemble dans les expériences de [19] porte sur le texte RED et applique ensuite la technique conjointe ; enrichir et décomposer en une étape.

- *Modèle de langage pondéré* : Sarikaya et Deng dans [21] ont proposé une modélisation conjointe du langage morphologique-lexicale (JMLLM) pour la Traduction Automatique. Le processus commence par une segmentation morphologique du texte Arabe. Ils ont proposé une structure arborescente appelée Morphological-Lexical Parse Tree (MLPT) pour combiner les informations morphologiques avec les informations lexicales dans un seul fichier JMLLM, comme illustré à la figure 14. L'idée est de scinder le mot en segments pour former une unité lexicale significative. Ci-dessous, un exemple d'arbre d'analyse morphologique-lexical.

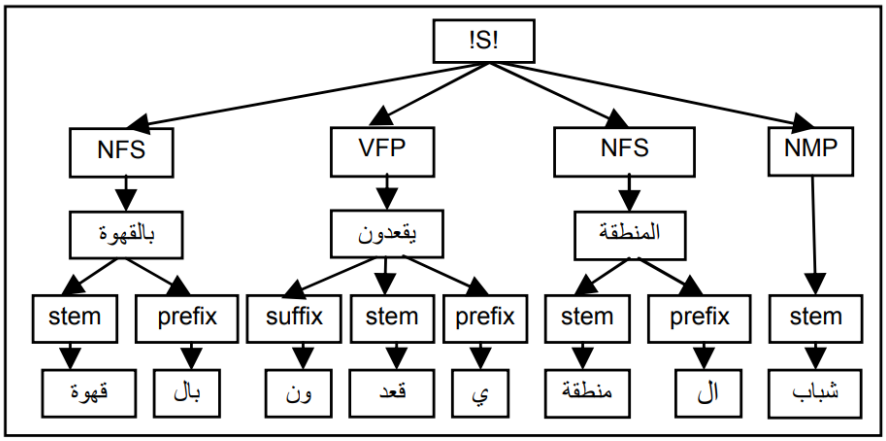


Figure 14: Arbre d'analyse morpho-lexicale.

Nous n'entrerons pas dans les détails concernant les techniques mentionnées ci-dessus car ça rentre au-delà de la portée et l'objectif de cette section. Qu'il doit fournir un aperçu des modifications introduites dans la Traduction Automatique statistique afin d'améliorer la traduction du couple de langues Anglais-Arabe.

Enfin, il convient de mentionner quelques ressources et outils, car TAS utilise à la fois des corpus monolingues et parallèles pour les modèles de langage et de traduction. Eisele et al. dans [22] décrivent l'acquisition d'un corpus multilingue extrait des documents officiels des Nations Unies (ONU). Le corpus multilingue qu'ils ont nommé multiUN est composé de six langues: Arabe, chinois, Anglais, français, russe et espagnol. Ils l'ont mis à la disposition de la communauté des chercheurs sur le site Web du projet EuroMatrixPlus.

La boîte à outils MADA (MorphologicalAnalysis and Disambiguation forArabic) [30] est largement utilisée dans les documents examinés précédemment pour la production de diverses formes enrichies et de schémas de jetons.

Pasha et al. dans [23] a présenté la boîte à outils MADAMIRA; c'est une intégration des fonctionnalités de MADA et AMIRA et il est public pour une utilisation en recherche.

Pour le traitement de l'Anglais, les outils les plus couramment utilisés sont les suivants : Reconnaissance d'entité nommée Stanford, étiqueteur POS Log-Linear de Stanford [24] et analyseur inspiré par entropie maximale [25].

Pour le pipeline TAS, le toolkit SRILM est largement utilisé pour la modélisation de langage [26]. GIZA ++ [4] est largement utilisé dans l'alignement des mots et le décodage est effectué

à l'aide du système open source TAS basé sur les phrases [27]. Il est presque utilisé pour la réalisation d'expériences dans les articles précédemment examinés.

En conclusion, la plupart des travaux portant sur la Traduction Automatique Arabe utilisent l'approche statistique axée sur la traduction de l'Arabe vers l'Anglais et d'autres langues. Une enquête littéraire sur les articles de TAS a été faite pour permettre de classer par approche et récapitulatif des langues source / cible presente que la majorité des contributions de la recherche sont axées sur une approche basée sur la phrase sensée être traduite de l'Arabe à l'Anglais.

2.5 CONCLUSION

Au début de ce chapitre, on a étudié le traitement automatique de la langue et en particulier l'Arabe, on a commencé par l'écriture Arabe et sa différence par rapport aux autres langues, ce qui a créé une difficulté à traiter par la machine. On a également parler du prétraitement, la normalisation et l'ambigüité, qui sont des défis qui permettent à la pertinence d'augmenter ou de diminuer en fonction de la tâche effectuée.

Parmi les tâches connues du traitement automatique de la langue, il y a la Traduction Automatique qui fait l'objet de nos recherches. Dans cette section, on a approfondi les défis de la traduction de et vers l'Arabe en particulier, les ressources linguistiques pour les tâches du traitement automatique de la langue Arabe en générale et la Traduction Automatique en particulier. Nous avons par la suite, enrichi cette section avec des modèles appartenant à différentes approches qui tentent de fournir une bonne traduction de et vers l'Arabe.

Enfin, nous sommes passés de la traduction Arabe vers Traduction Automatique statistique, où, nous avons d'abord, abordé les défis intrinsèques auxquels sont confrontées les langues riches sur le plan morphologique comme l'Arabe dans un contexte général, puis dans un contexte de Traduction Automatique statistique. Dans la dernière section, nous avons énuméré quelques de modifications introduites par les chercheurs sur le système TAS dans le but d'atténuer les problèmes et de surmonter les défis auxquels est confronté l'AT en langue Arabe.

Nous avons conclu ensuite ce chapitre en mentionnant quelques outils et ressources utilisés dans la recherche et en prenant note de la densité de recherche centrée sur chaque modèle de TAS, ce qui peut aider à décider de l'orientation des prochaines phases de ce travail.

CHAPITRE 3
CONCEPTION ET MODELISATION DE LA SOLUTION PROPOSEE

3.1 INTRODUCTION

Dans ce chapitre, on va présenter la modélisation de notre solution proposée. Ce procédé a pour objectif de permettre la formalisation des étapes préliminaires du développement d'un système afin de rendre ce développement plus fidèle aux besoins du client. On va commencer par un rappel sur la problématique et les objectifs de notre projet ainsi qu'une description du centre où on a passé notre stage. Par la suite, nous allons présenter l'architecture globale de notre système et la décrive brièvement dans un premier temps. Après, nous allons définir l'utilité de chaque sous système, module et composant du système en détaillant ses fonctionnalités. A la fin, nous allons introduire brièvement la démarche de tests et validation du système.

3.2 PROBLÉMATIQUE

La traduction connait des problèmes d'ordre linguistique et culturel. Les problèmes linguistiques incluent les différences grammaticales, l'ambiguïté lexicale, et l'ambiguïté sémantique. Les problèmes culturels quant à eux, ils renvoient à des caractéristiques de diverses situations.

Parmi les approches robustes qui tentent de surmonter ces difficultés figurent les modèles statistiques. Dans notre cas, nous nous intéressons particulièrement aux modèles statistiques basés sur les phrases et dont le principe est de traduire les phrases en unités.

3.3 RAPPEL DES OBJECTIFS DU PROJET

Le travail visé dans ce projet consiste à faire des expériences sur la traduction automatique en proposant un système de Traduction Automatique Statistique basé sur les phrases accessible via une interface WEB. Le corpus parallèle de tests est un corpus fourni par l'encadreur et le sens de la traduction choisi est de l'anglais (langue source) vers l'arabe (langue cible).

Ce projet rentre dans le cadre du projet international Trésor ou base de données culturelles et scientifiques « Dhakhira Arabia » proposé par le professeur Hadj Salah président de l'Académie Algérienne de la Langue Arabe et également président du Haut comité de ce projet.

Le projet est une initiative de la Ligue Arabe. C'est en septembre 2004, que le projet a été adopté à l'unanimité par le Conseil ministériel de la Ligue Arabe. Un projet d'organigramme et de règlement intérieur a été alors soumis à la Ligue, fondé sur le principe de la participation des institutions scientifiques et culturelles dans chaque pays sur la base d'une aide matérielle de l'Etat [37].

Parmi les centres de recherche scientifiques impliqués dans ce projet se trouve le Centre de Recherche Scientifique et Technique pour le Développement de la Langue Arabe (CRSTDLA). Les tâches principales du centre consistent à mener des recherches théoriques et pratiques sur le développement de la langue et de la linguistique arabe en coopération avec des organismes et institutions concernés par la réalisation de projets de recherche dans les domaines de la linguistique et de ses techniques appliquées à la langue arabe. Et ce, en utilisant les technologies informatique, électronique, acoustiques et phonétiques, ainsi que de toute méthodologie technique répondant aux besoins de la recherche interdisciplinaire moderne [38].

En ce qui concerne notre projet, le stage a été effectué dans le Département de linguistique informatique sous l'encadrement du Docteur Mourad Abbas dont la principale mission est réaliser des études scientifiques sur les méthodes de traduction et adapter les techniques de traduction arabe aux besoins des traducteurs de livres scientifiques.

3.4 SOLUTION PROPOSÉE

La TAS tente de générer des traductions à l'aide de méthodes statistiques basées sur des corpus parallèles [39] comme le Multi-UN[3] ou encore le corpus OpenSubtitles[4]. Ces ressources fournissent plusieurs corpus parallèles avec paires de langues différentes. Lorsque de tels corpus sont disponibles, la traduction de textes similaires peut donner de bons résultats, mais ces corpus sont encore rares pour de nombreux couples de langues.

[3] Multi-UN est une collection de documents traduits des nations unies.
[4] OpenSubtitles est un ensemble de documents collectés à partir de films.

En ce qui concerne la formule que nous avons adopté pour le cas de la traduction de l'anglais (langue source) en arabe (langue cible), nous pouvons formuler une séquence de mots dans la langue source : E=e_1,…,e_j,…,e_J , et trouver ensuite la séquence la plus probable dans le langage cible : A=a_1,…,a_i,…,a_I,

$$A^{\wedge} = argmax_A\, P(A|E) = P(A) * P(E|EA)$$

Ici, P (E|A) représente le modèle de traduction et P (A) représente le modèle de langage. Le modèle de traduction joue le rôle de garant de la fidélité de la traduction et le modèle de langue permet d'assurer la fluidité de la traduction [27].

Les composantes de l'approche statistique sont illustrées dans Figure-15.

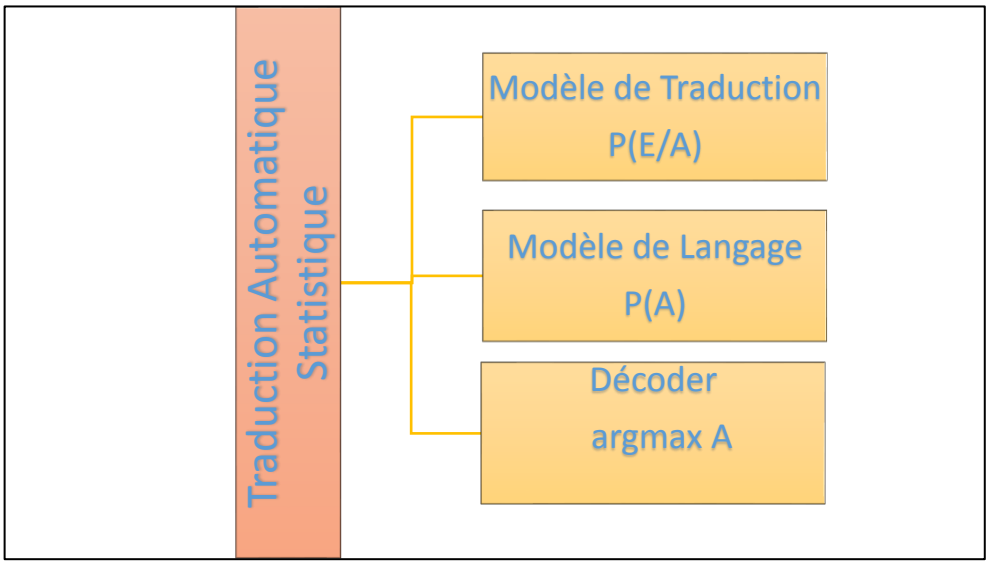


Figure 15: Composants d'un système de TAS.

Pour construire toutes ces composantes, nous allons procéder en plusieurs étapes. Nous avons schématisé le processus global de traduction automatique statistique basé sur les phrases afin d'illustrer le fonctionnement de notre solution depuis la préparation des données à l'évaluation de la qualité de la traduction et l'estimation du système.

On peut résumer les étapes principales de construction du système par les taches suivantes ; en commençant par le prétraitement qui est considéré comme la pierre angulaire de toutes les applications de traitement automatique du langage, Suivi par l'apprentissage qui est divisé en deux étapes (modèle de traduction et de langage) et

on termine par le décodage qui permet de trouver la phrase la plus probable, tout cela est montré dans la Figure-16 ci-dessous.

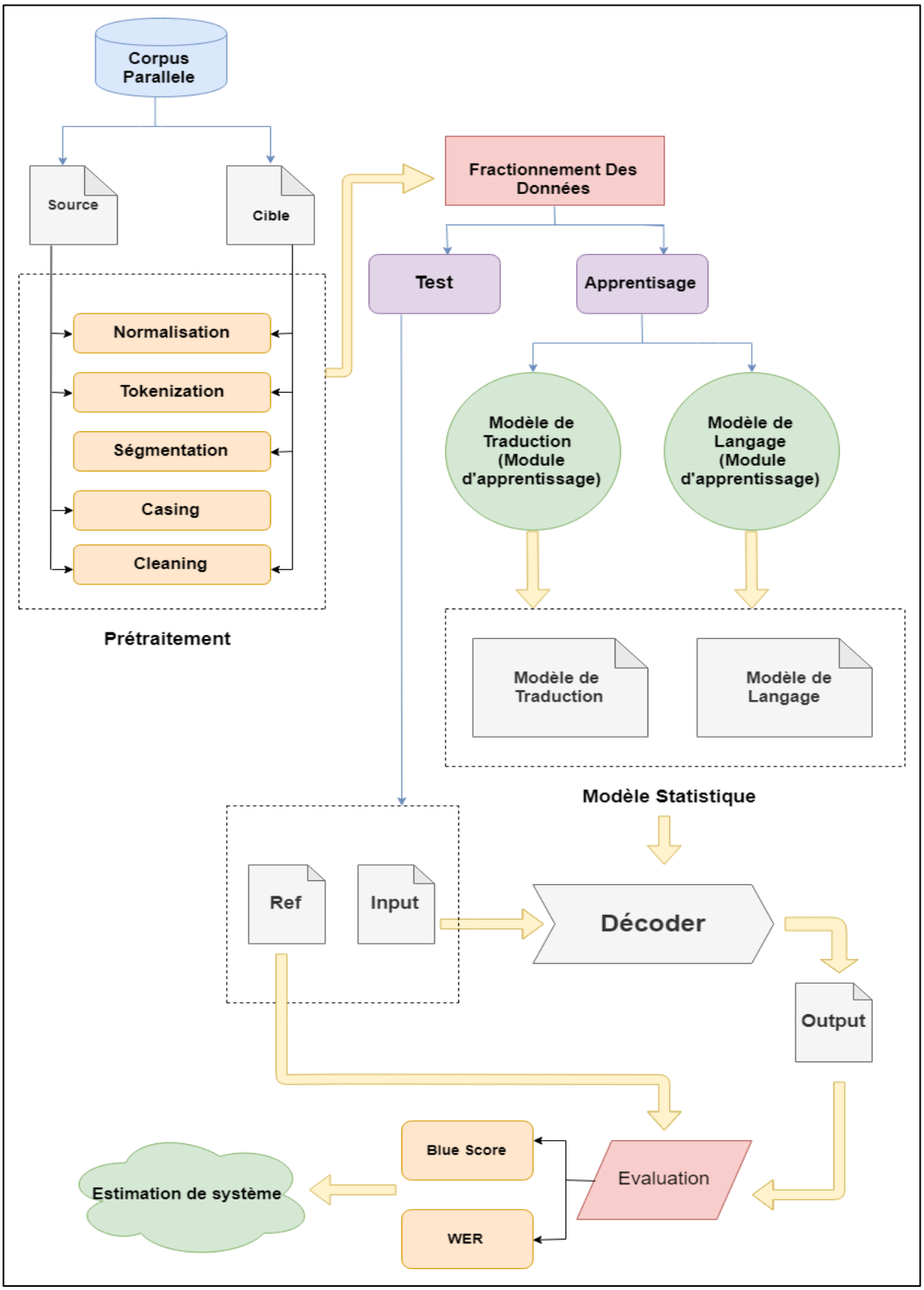


Figure 16: Fonctionnement d'un système de traduction automatique statistique basé sur les phrases.

3.4.1 Prétraitement

Les données bilingues et monolingues sont pré-traitées avant la préparation des modèles de traduction et des modèles de langages. Le prétraitement décrit dans notre cas est pour deux langues anglais et l'arabe, mais à cause des grandes différences morphologique entre les deux langues on va utliser des techniques différentes pour essayer d'obtenir des modèles clairs.

On va commencer par la langue source (l'anglais) qui est considéré comme l'une des langues les plus courantes dans le domaine de traitement automatique de la langue et les techniques que nous allons utiliser dans ce cas sont comme suit : la normalisation, la tokenization, casing et nettoyage (cleaning).On a schématisé le processus par un diagramme de flux illustré dans la Figure-3.

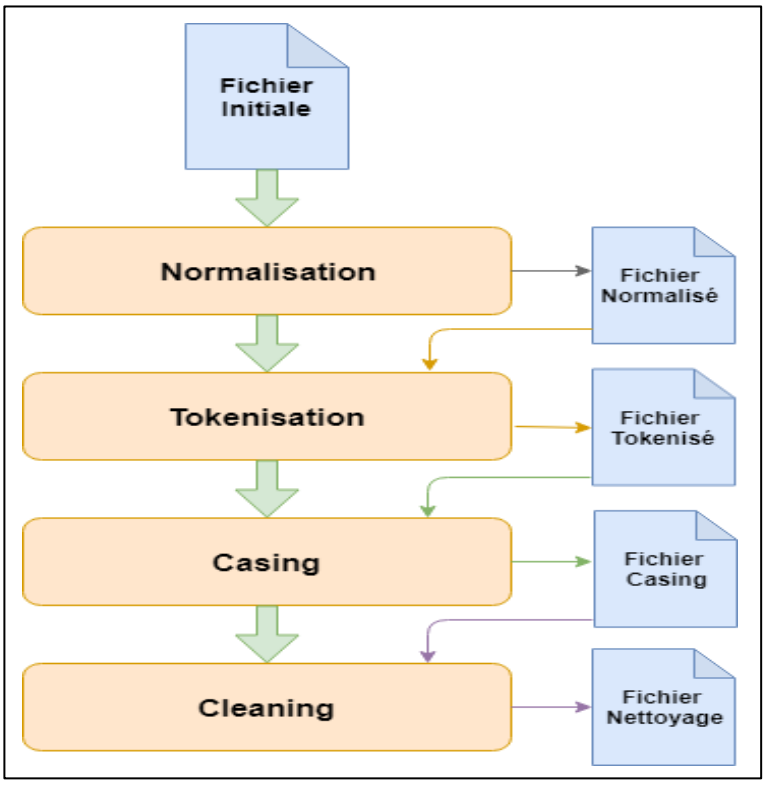


Figure 17: Etapes de prétraitement de la langue Anglaise.

Dans ce qui suit, nous aborderons chaque sous processus en détail en commençant par la normalisation.

i. Normalisation pour l'anglais

C'est un processus morphologique permettant de regrouper des mot lexicalement similaire mais avec des diacritics differents et dans ce cas pour l'anglais ces mots sont sémantiquement similaire , ou pour être plus clair c'est un processus qui rend ces mots plus uniformes dans l'ensemble de données [6].

Par exemple suppression des accents et des signes diacritiques (naïve → naive, Dubaï → Dubai) et suppression des espaces blancs ou replacmeent par un seul blanc, par exemple (hello my friend→ hello my friend), suppression des points consécutifs au cas où ils n'auraient aucun sens ou leur remplacement par un seul point ou une virgule, par exemple (hello my...friend…how are you…→hello my friend, how are you.).

ii. Tokenisation pour l'anglais

C'est un processus de conversion d'une séquence de caractères en une séquence de tokens, qui correspondent approximativement à des "mots". Voici un exemple:

Input: "Hello sir, is the director here? "

Output: |Hello| |sir| |, | |is| |the| |director| |here| |? |.

On peut exprimer les instructions de ce processus par l'algorithme suivant :

Algorithme Tokenization

```
Chaine Input = "Text..." ; Chaine Output=[] ; Chaine mot = "" ;
Debut
Pour toutes les characters 0...Length( Input ) Faire
    Si character Non vide et Non isSpecialChar Alors
        concaténer mot , character
    Sinon Si mot Non vide alors
        Ajouter mot , Output
        tronquer mot
        Fin si
    Fin Sinon
    Fin Si
Fin pour
Fin
```

Tels que :

- ***Input:*** représente le texte sur lequel nous allons implémenter le processus.
- ***Output***: c'est le tableau ou liste des mots (tokens) après le traitement.

On a défini trois fonctions : **isSpecialChar** une fonction booléenne qui renvoie « true » si le paramètre est un caractère spécial, sinon la fonction renvoie « false », la fonction **Add** ajoute une chaîne au tableau et la fonction **lenght** renvoie la taille de chaine.

iii. Casing

Le casing est une fonctionnalité spécifique à des certains langues. L'anglais utilise la capitalisation, alors que l'arabe ne possède pas cette fonctionnalité. En anglais, la majuscule est utilisée au début des phrases pour indiquer une entité nommée ou un nom propre.

Cette étape peut aider à faciliter le marquage d'une partie de la parole et la reconnaissance d'entité nommée [40]. Cependant, la capitalisation peut dégrader les performances de la traduction automatique statistique, l'apparition d'un mot avec et sans majuscule étant traitée comme deux mots différents. Par exemple, si on a la phrase suivante comme entrée : *" HEllO mY FrienD. " nous allons avoir en sortie* : *"* Hello my friend. *"*

iv. Nettoyage (Cleaning)

Une tâche commune pour la traduction automatique statistique, est le nettoyage des fichiers du corpus. Son rôle est de nettoyer les longues phrases et remplacer les caractères non autorisés avant d'alimenter les fichiers du corpus pour l'apprentissage [41]. De plus, nous nettoyons /remplaçons les chaînes de caractères par une expression régulière conformément à certaines règles afin d'améliorer le modèle de traduction.

Cette étape est effectuée sur les deux langues (source, cible), c'est la dernière étape avant l'apprentissage et son impact sur l'alignement des mots.

Après avoir terminé le prétraitement de l'Anglais (la langue source) nous nous tournons vers l'Arabe (la langue cible) qui est relativement difficile par rapport à son

prédécesseur à cause de sa complexité morphologique et différentes spécifications par rapport à d'autres langues.

Le processus de prétraitement de la langue est illustré par le diagramme de flux suivant :

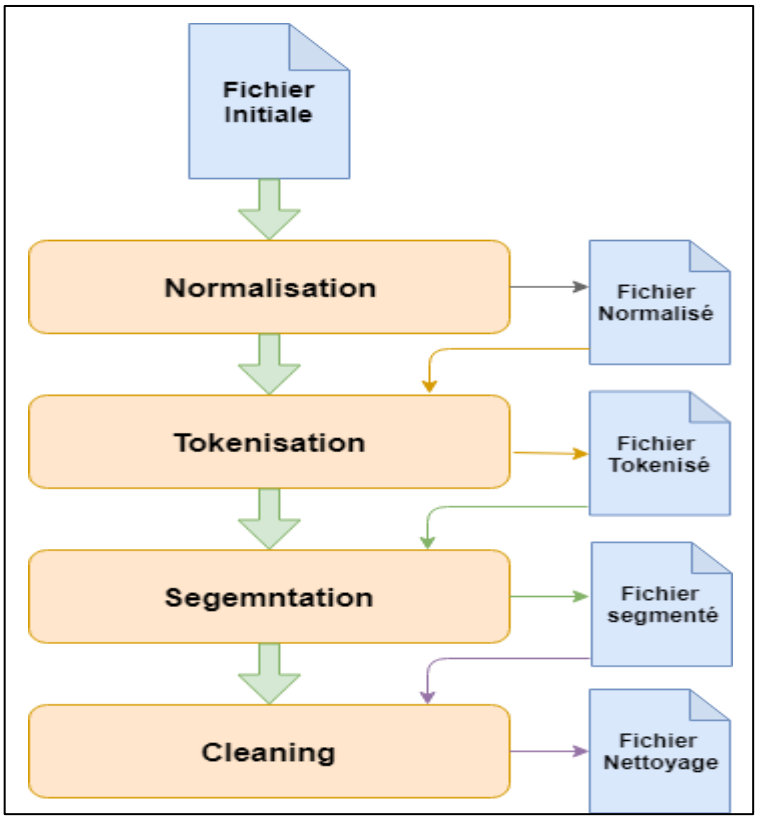


Figure 18: Etapes de prétraitement de la langue Arabe.

v. Normalisation pour l'arabe

Comme nous l'avons vu concernant la normalisation pour l'Anglais, nous avons des problèmes encore plus complexes pour l'Arabe du fait de l'incohérence dans l'utilisation des marques diacritiques et de certaines lettres dans les textes Arabes contemporains.

Certaines lettres arabes ont la même forme et ne sont différenciées qu'en ajoutant certaines marques, telles qu'un point, Hamza ou maddah placé au-dessus ou au-dessous de la lettre [42].

Par exemple, "alif" en arabe (ا) peut être composé de trois lettres différentes selon qu'il ait une hamza en haut comme dans (أ) ou en bas comme dans (إ) ou une maddah en haut comme dans (آ).

Un autre exemple : بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ → بسم الله الرحمان االرحيم

vi. Tokenisation pour l'arabe

La Tokenisation pour l'Arabe est la même que pour l'Anglais. Elle consiste à découper les espaces blancs et les caractères spéciaux [42]. Par exemple :

Input : "مرحبا يا سيدي ، هو المدير هنا؟"

Output : | ? | | هنا | | المدير | | هو | | ء| |سيدي| | يا | | مرحبا |.

vii. Segmentation pour l'arabe

C'est le processus de division du texte écrit en unités significatives. C'est-à-dire subdivision des images de texte en parties constitutives (des lignes des composants ou des mots et des caractères individuels). C'est le premier module de conception du système de reconnaissance de caractères.

Dans la littérature, diverses méthodes de segmentation du texte arabe ont été proposées. Ces techniques de segmentation sont décrites brièvement pour la segmentation des lignes, des composants (ou des mots) et des caractères [43].

Des exemples sur la langue arabe :

Input : الجزائر → output : الجزائر

Input : الأصدقاء → output : |أصدقاء| |ال|

Input : أصدقاءه → output : |ه| |أصدقاء|

Input : كهرومنزلية → output : |منزلية| |كهرو|

Après toutes ces étapes, il faut nettoyer les données, cette étape est une étape commune et cruciale pour de préparer les données d'apprentissage que nous aborderons en détail dans la section suivante.

3.4.2 Processus d'Apprentissage du Système

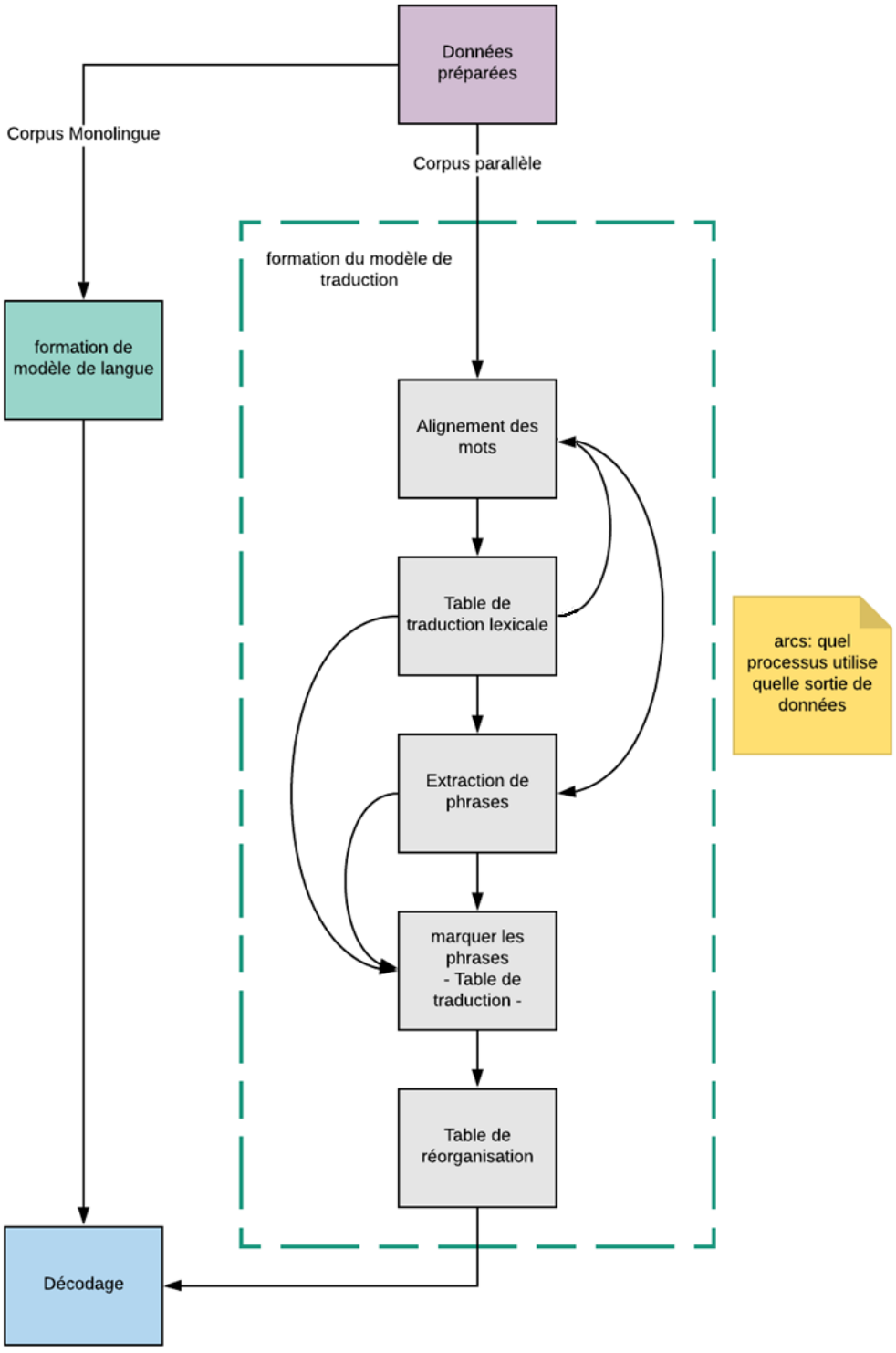


Figure 19: Processus d'apprentissage du système.

Le processus d'apprentissage illustré dans la figure précédente peut être simplifié selon les points suivants :

i. Préparation des données;
ii. Alignement des mots;
iii. Tableau de traduction;
iv. Extraction de phrases;
v. Notation des phrases;
vi. Tableau de réorganisation;
vii. Modèle de langage.

Les deux composants principaux du système de traduction automatique sont *l'apprentissage* et le *décodage*.

Le processus d'apprentissage comprend différentes étapes. D'abord, il prend des données prétraitées qui sont des corpus parallèles et monolingues et les transforme en un modèle de traduction automatique. Après, le modèle doit être donné à un décodeur avec une collection de phrases en langue source (ensemble de test). Ainsi, les phrases sources seront traduites vers la langue cible.

Vous trouverez ci-après un bref aperçu de chacune des étapes du processus d'apprentissage énumérées ci-dessus.

i. Préparation des données

Les données doivent généralement être préparées avant d'être utilisées dans l'apprentissage. Il est nécessaire de collecter les données et de les définir dans un format utilisable. Le processus de prétraitement est ensuite exécuté sur les données, comme indiqué dans la section précédente.

Les données d'apprentissage doivent être fournies sous forme de phrases alignées, une phrase par ligne (les lignes vides n'étant pas autorisées) dans deux fichiers différents. L'un pour les phrases en langue source et l'autre pour les phrases en langue cible.

ii. Alignement des mots

L'alignement des mots consiste à établir des liens d'alignement entre les mots source et cible dans un corpus parallèle bilingue qui sont des traductions les uns des autres.

L'alignement des mots est la première étape et peut-être la tâche la plus critique dans toute approche de la traduction automatique basée sur les données (ou basée sur un corpus), car les modèles successifs sont approximés en fonction d'associations de mots source-cible établies au cours de l'étape d'alignement des mots. C'est la première étape pour découvrir la connaissance de la traduction cachée dans le corps parallèle.

De plus, dans le système de traduction automatique statistique basé sur des phrases, des paires de phrases sont extraites sur la base des connaissances contenues dans la table d'alignement de mots.

Il existe différentes méthodes pour obtenir un alignement de mots, pour le système que nous proposons, nous avons opté pour une méthode utilisant l'intersection de deux alignements de la langue cible et source pour obtenir un alignement global des mots. Cette méthode ajoute également des points d'alignement supplémentaires comme présenté dans le premier chapitre dans la section alignement.

Il a été prouvé que cette méthode améliore la qualité de l'alignement et permet une meilleure extraction des phrases et un meilleur tableau de traduction.

La sortie de cette étape contient des informations d'alignement, un point d'alignement à la fois, sous la forme de la position de la paire de mots des langues source et cible.

Bien que les informations d'alignement constituent les premières données établies dans le processus, il est nécessaire de savoir que la probabilité de traduction lexicale est calculée pour chaque mot des corpus au cours de l'alignement. la probabilité de traduction lexicale sera compilée ultérieurement dans la table de traduction lexicale.

iii. Tableau de traduction

Etant donné les informations d'alignement, l'établissement de la table de traduction lexicale sera une tâche simple en utilisant une estimation du maximum de vraisemblance de la traduction.

Dans cette étape, nous estimons la table de traduction de mots $\boldsymbol{\omega(a|e)}$ et son inverse $\boldsymbol{\omega(e|a)}$. Le but de cette étape est d'obtenir un tableau de traduction lexical sous la forme d'une paire de mots avec une valeur d'estimation de la traduction pour chaque paire.

$$\boldsymbol{S_{word}} \quad \boldsymbol{T_{word}} \quad \boldsymbol{\omega(a|e)}$$

où S_{word} est le mot source et T_{word} le mot cible, tandis que $\boldsymbol{\omega(a|e)}$ est la probabilité de traduction .

iv. Extraction de phrases

À partir des informations d'alignement, nous pouvons extraire des phrases compatibles avec l'alignement des mots. Illustrons le processus à partir de l'exemple que on a déjà vu dans le premier chapitre :

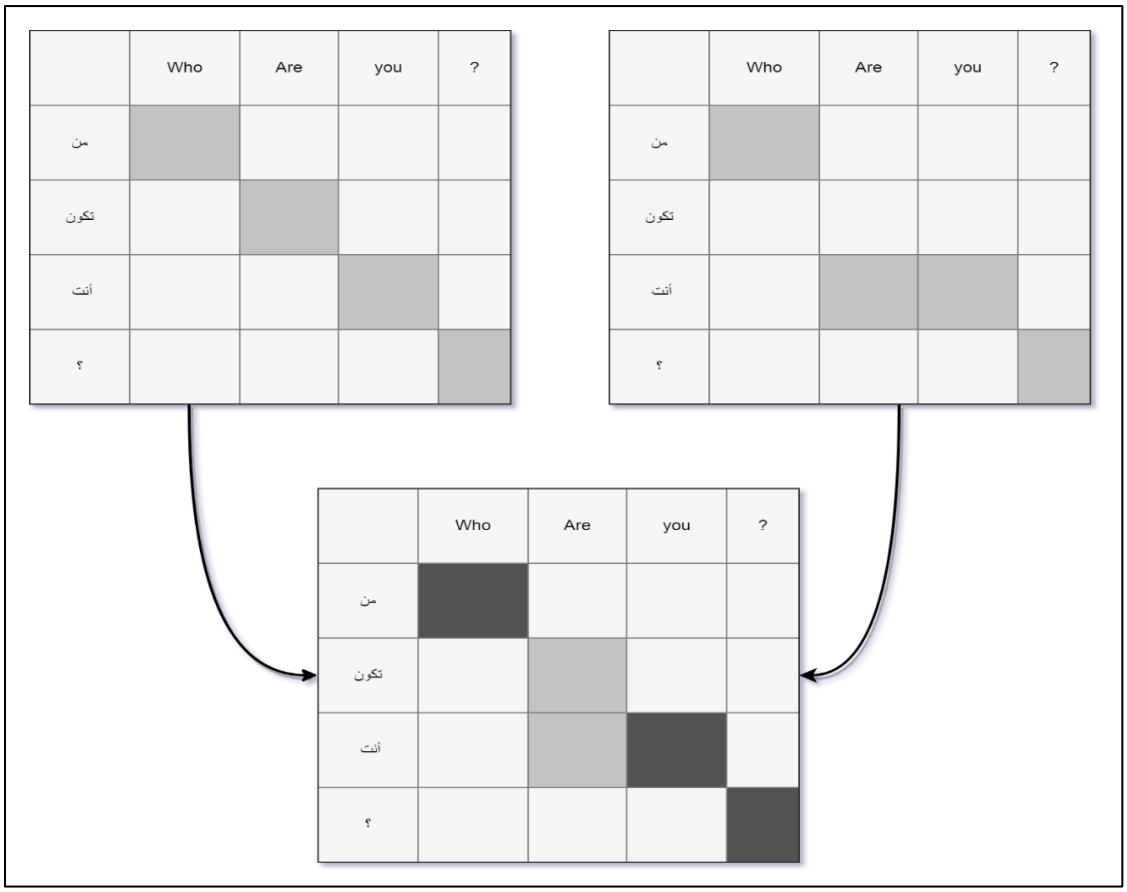


Figure 20: Extraction de phrases.

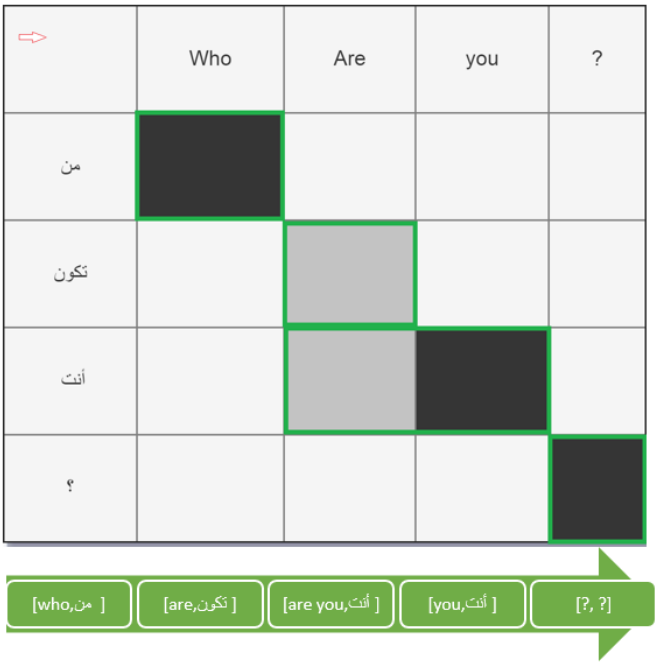
Les informations d'alignement dans les deux alignements sont combinées pour créer les dernières informations d'alignement qui seront utilisées lors de l'extraction de phrases.

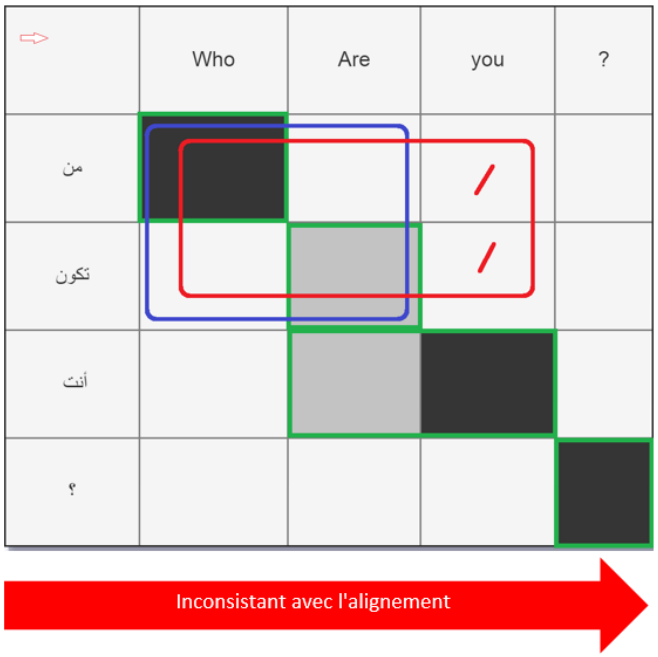
Le processus implique la collecte de phrases courtes et de phrases longues qui seront bénéfiques pour capturer un contexte plus local, aidant ainsi à traduire des phrases plus grandes.

Le processus de collecte de phrases est basé sur la cohérence des mots alignés dans les phrases. Formellement :

$$(E, F) \text{ consistent with } A \equiv$$
$$\forall e_i \in E : (e_i, f_j) \in A \rightarrow f_j \in F$$
$$\cap \quad \forall f_j \in F : (e_i, f_j) \in A \rightarrow e_i \in E$$
$$\cap \quad \exists e_i \in E, f_j \in F : (e_i, f_j) \in A$$

À des fins d'illustration, nous pouvons prendre exemple présentant le processus d'extraction des phrases en fonction des informations d'alignement. Ici l'alignement est représenté dans une matrice où chaque cellule représente un point d'alignement.





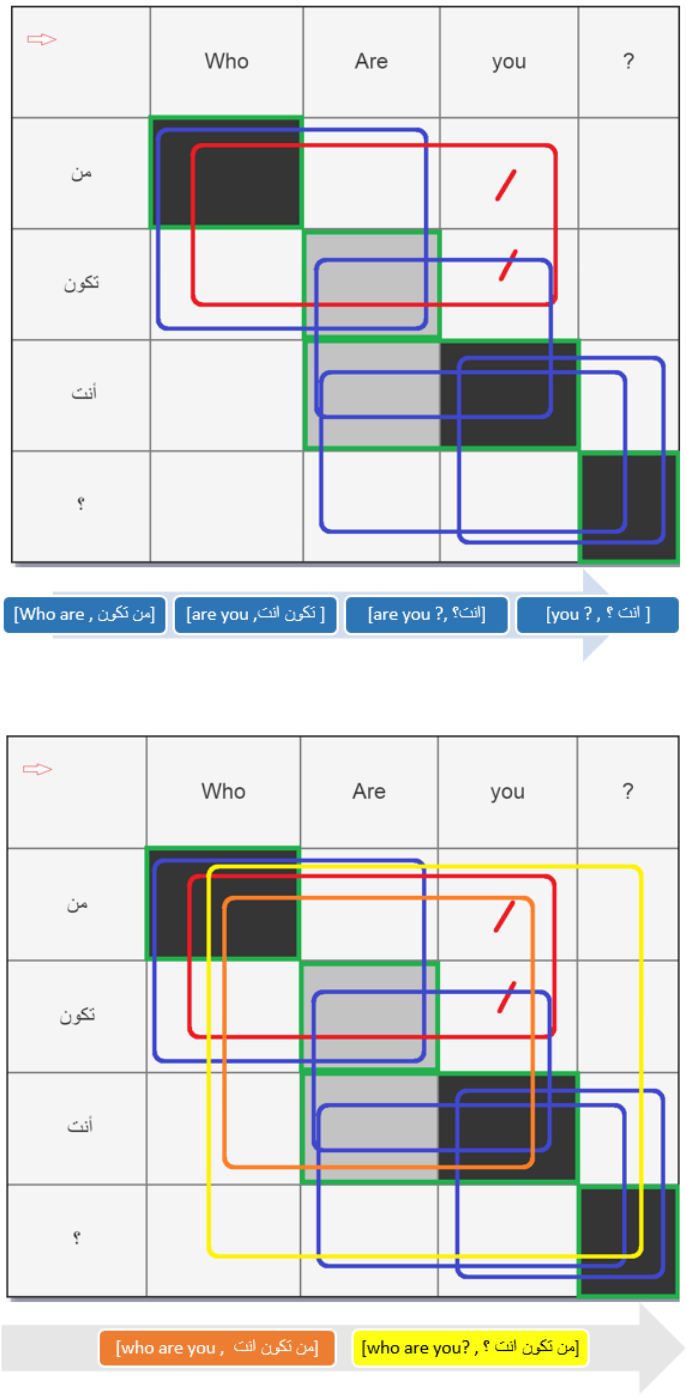


Figure 21 : Etapes d'extraction des phrases

v. La Notation des Phrases

Un tableau de traduction est créé à partir des paires de traduction de phrases que nous avons établies aux étapes précédentes. Le processus d'estimation de la probabilité de traduction $\emptyset(a|e)$ d'une phrase étrangère nécessite l'ensemble des traductions de la langue cible.

À cette fin, toutes les traductions de la langue cible d'une phrase étrangère sont placées les unes à côté des autres en triant le tableau de traduction de phrases.

Une fois que les traductions sont regroupées, les traductions sont notées en collectant des nombres et en calculant $\emptyset(a|e)$ phrase par phrase. Le même processus est exécuté pour estimer la traduction inverse $\emptyset(e|a)$.

En plus de la distribution de probabilité de traduction $\emptyset(e|f)$, on calcule la traduction inverse $\emptyset(f|e)$, la pondération lexicale doit également être calculée pour chaque $\emptyset(e|f)$ nécessaire pour tenir compte de la surestimation de la phrase rare, dans le cas d'une occurrence unique qui aurait comme résultat $\emptyset(e|f)$= 1.

$$Foreign_{phrase} \quad Target_{phrase} \quad \emptyset(e|f) \quad weight(e|f)$$

vi. Tableau de réorganisation

La langue source et la langue cible ont un ordre différent pour les mots d'une phrase. Par conséquent, toute tentative de traduction avec une table de traduction uniquement renvoie une traduction non ordonnée dans laquelle les mots ne sont pas dans le bon ordre. Par exemple :

japan invested million dollars in the research	*(1)*
استثمرت اليابان مليون دولار في البحث	*(2)*
اليابان استثمرت مليون دولار في البحث	*(3)*

Comme on peut le voir, la traduction de la phrase source (1) à l'aide d'un tableau de traduction n'a abouti qu'à une traduction non ordonnée, où (3) correspond à la traduction correcte. Dans l'exemple, un seul mot n'était pas ordonné et, si l'on considère l'expression référence, le sens est toujours correctement transmis, mais dans la plupart des cas, les phrases non ordonnées sont très déformées.

the quick brown fox jumps	*(4)*
يقفز الثعلب البني السريع	*(5)*
السريع البني الثعلب يقفز	*(6)*

De la même manière, la phrase non ordonnée (5) bien qu'elle soit écrite et lue correctement de gauche à droite, la sortie réelle est (6).

Comme nous pouvons voir, certaines phrases conservent le sens dans les cas où la position d'un mot est interchangeable, ce n'est pas toujours le cas pour toutes les phrases d'une langue. Par conséquent, une technique de réorganisation doit être utilisée pour produire une traduction correcte des phrases.

Un modèle de réorganisation peut être considéré comme faisant partie du modèle de traduction. Un modèle basé sur la distance est le modèle standard par défaut pour la traduction automatique statistique basée sur les phrases, qui est uniquement la fonction de la distance de mouvement. Cela suffit pour une paire de langues anglais-arabe en raison de la simple réorganisation locale requise dans la traduction.

Jusqu'à présent, nous avons terminé avec le modèle de traduction qui est considéré comme la première partie de l'apprentissage et nous passerons à la deuxième partie, le modèle de langage.

vii. Modèle de langage

C'est un composant essentiel de tout système de traduction automatique statistique, le modèle de langue est responsable de la fluidité de la traduction traduite. Il fait son apprentissage sur un corpus monolingue afin de pouvoir estimer la probabilité d'une suite de mots.

Un modèle de langage est une fonction qui prend une phrase et renvoie la probabilité qu'elle ait été produite par un locuteur de cette langue. Il est plus probable que le locuteur prononce la phrase "the house is small" (المنـزل صـغير) que la phrase "small the is house" (صـغير هو المنزل).

Par conséquent, un bon modèle de langage P_{LM} attribue une probabilité plus élevée à la première phrase. Cette préférence du modèle de langue aide un système statistique de traduction automatique à trouver le bon ordre des mots.

De plus, le modèle de langage aide au choix des mots, par exemple si un mot étranger (tel que le mot Anglais *house*) possède plusieurs traductions en arabe (منـزل, بيـت, دار.مسـكن. السـكن. سـكن), les probabilités de traduction lexicale donnent déjà la préférence à la traduction la plus courante (house). Mais dans des contextes spécifiques, d'autres traductions peuvent être préférées.

En bref, le modèle de langage donne une probabilité plus élevée au choix de mot le plus naturel en contexte, par exemple :

$$P_{LM}\ (I\ am\ going\ home)\ >\ P_{LM}\ (I\ am\ going\ house)$$

$$P_{LM}\ (\text{أنا ذاهب إلى المنزل})\ >\ P_{LM}\ (\text{أنا ذاهب إلى السكن})$$

L'estimation du modèle commence par la collecte de n-grammes et de leurs comptes de fréquence. Ensuite, les paramètres sont estimés pour chaque niveau de n-grammes afin de résoudre le problème de dispersion des données. Le modèle est créé à partir des ensembles de chaque niveau de n-grammes contenant des n-grammes avec des probabilités et des poids.

Le modèle de langage le plus couramment utilisé est le modèle de langage n-gram qui est basé sur des statistiques de la probabilité des mots qui doivent se suivre.

- ***Modèle de langage N-gramme :*** Formellement, le but de la modélisation du langage est de calculer la probabilité de la chaîne de mots $W = w_1, w_2, \dots\dots\dots, w_n$, L'approche typique consiste à collecter des comptes de W. Cependant, la rareté des données pose un problème, car la plupart des longues séquences de mots ne se produiront pas du tout, ou peuvent avoir une seule occurrence qui peut entraîner un score biaisé. Pour atténuer ce problème, le calcul de $P(W)$ doit être décomposé en étapes plus petites. Pour lesquels des statistiques pourraient être collectées pour estimer la distribution de probabilité P.
- ***La chaîne de Markov:*** À l'aide de la règle de la chaîne, le processus d'estimation de la probabilité est décomposé en mots au lieu de phrases.

$$p(w_1, w_2, \dots, w_n)\ =\ p(w_1)\ p(w_2|w_1) \dots p(w_n|w_1, w_2, \dots, w_{n-1})$$

La probabilité de modèle de langage $P(w_1, w_2, \dots\dots, w_n)$de W est le produit des probabilités de mots étant donné l'historique des mots précédents.

$$P(w_1, \dots\dots, w_n)\ =\ \prod_{i=1}^{n} P(w_i|w_1, \dots\dots, w_{i-1})$$

Pour estimer les distributions de probabilités de mots avec une historique limité m.

$$p(w_n|w_1, w_2, \dots, w_{n-1}) \cong p(w_n|w_{n-m}, \dots \dots, w_{n-2}, w_{n-1})$$

Encore une fois, est le produit de probabilités de mots avec l'historique limité m.

$$P(w_1, \dots \dots, w_n) = \prod_{i=1}^{n} P(w_i|w_{n-m}, \dots \dots, w_{i-1})$$

Ce qui considère le premier uni-gramme comme sa propre historique précédente. Alors que ce qui suit ne prend en compte que l'historique actuelle.

$$P(w_1, \dots \dots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-(n-1)}, \dots \dots, w_{i-1})$$

Décomposer une séquence où nous parcourons une séquence de mots en considérant une historique limitée de mots. Ce type de modèle s'appelle une chaîne de Markov, où l'ordre n du modèle est l'ordre de l'historique $m + 1$.

Par exemple, un ordre de modèle $P(w3|w1, w2)$ dans un modèle de langage à trois grammes est donc 3, et donc l'ordre d'historique est 2, ce qui correspond à $w1, w2$. Comme dans un modèle de trigramme. Considérons une historique de deux mots pour estimer le troisième $w3$.

La probabilité de n-grammes dans le modèle de langage est estimée à l'aide de l'estimation du maximum de vraisemblance en collectant les occurrences et fréquences du mot apparaissant après une séquence, par opposition à d'autres mots.

$$P(w_i|w_{n-m}, \dots \dots, w_{i-1}) = \frac{Fcount(w_{n-m}, \dots \dots, w_{i-1}, w_i)}{\sum_w Fcount(w_{n-m}, \dots \dots, w_{i-1})}$$

Grâce à la formule précédente, nous obtenons le compte de fréquence de $w3$ après la séquence $w1, w2$ par opposition à d'autres mots. En fonction de l'estimation du maximum de vraisemblance. Nous calculons :

$$P(w3|w1,w2) = \frac{Fcount(w1,w2,w3)}{\sum_w Fcount(w1,w2,w)}$$

Finalement, le problème majeur de l'estimation du n-gramme est que beaucoup de n-grammes ne se produisent pas ou se produisent une ou plusieurs fois, mais pas suffisamment pour attribuer un score approprié.

Les n-grammes non observés recevront un score de 0 tandis que les n-grammes avec un compte de fréquence très basse recevront un score élevé biaisé .ce phenomene s'aggravera et devenir plus fréquent avec des n-grammes d'ordre supérieur.

Pour ce problème, différentes méthodes pour ajuster les comptes de fréquences existent. Ces techniques partagent le même objectif : résoudre le problème de la rareté des données [27].

Jusqu'ici, nous avons présenté les étapes de notre processus jusqu'à la phase d'apprentissage, nous avons décrit la plupart des composants du système de traduction. Le dernier composant est le décodeur qui permet à trouver la traduction la plus probable en fonction d'un ensemble de paramètres précédemment appris.

viii. Le décodeur

Le décodeur est chargé de trouver la meilleure phrase cible T_{best} qui maximise la probabilité de traduction $p(t|s)$.

$$T_{best} = argmax_t\ p(t|s)$$

Notre système est basé sur les phrases, dans lequel la formule du modèle se base sur une phrase :

$$T_{best} = argmax_t \prod_{i=1}^{I} p(s_i|t_i)\ d(start_i - end_{i-1} - 1\ )\ P_{Lm(s)}$$

Plusieurs composants contribuent à la note globale, la probabilité de traduction de phrase p, le modèle de réorganisation d et le modèle de langage p_{Lm} , phrases d'Entrée données s_i et phrases de sortie t_i et leurs positions $start_i$ et end_i , la probabilité de traduction peut être calculée.

Le processus de traduction d'une phrase est effectué sur des phrases comme hypothèses de morceaux de la phrase et à chaque traduction partielle les scores, les trois composantes du modèle P, d, p_{Lm} sont prises en compte :

- *Modèle de traduction* : avec chaque phrase de l'expression, la table de traduction de phrase P est consultée pour chercher la probabilité de traduction pour la paire de phrase.
- *Réorganisation du modèle* : compte tenu de l'expression position de début et de fin $start_i$ et end_i, la distribution de probabilité de distorsion d est consultée pour obtenir la réorganisation pour la phrase actuelle.
- *Modèle de langage* : les probabilités de chaîner le mot suivant en fonction de l'historique des mots précédents dans la phrase extraite de p_{Lm} ont une incidence sur la traduction.

Plusieurs décodeurs sont disponibles pour le public, notre système est un système basé sur les phrases dont l'algorithme de décodage le plus couramment utilisé dans ce modèle est le beam-search stack decoding où le décodeur commence par rechercher toutes les traductions possibles dans la table de phrases.

Cela inclut les traductions possibles de toutes les phrases possibles d'une phrase source donnée. Il en résulte un graphique de recherche avec de nombreuses options de traduction où la meilleure traduction est le meilleur chemin à travers le graphique de recherche avec la meilleure traduction de notation.

Le décodage d'une phrase source commence par une hypothèse initiale, puis les hypothèses de sortie sont construites de gauche à droite. Les hypothèses sont développées en choisissant les options de traduction disponibles à partir de la traduction possible de la phrase source actuelle.

he does not go home | هو لا يذهب إلى البيت

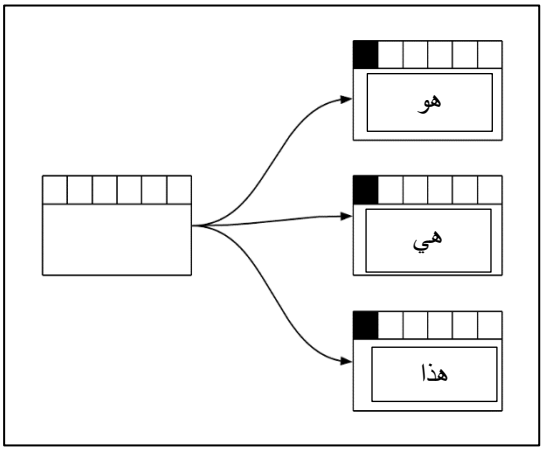


Figure 22 hypothèse initiale et debut d'expansion

Le processus d'expansion continue jusqu'à ce que tous les mots source soient couverts, les hypothèses couvertes sont appelées des « hypothèses complètes », une fois qu'il n'y a plus d'hypothèses incomplètes le décodeur sélectionne les hypothèses avec la plus forte probabilité à partir des hypothèses complétées comme traduction la plus probable T_{best}. Une illustration du processus de décodage est donnée dans la figure suivante :

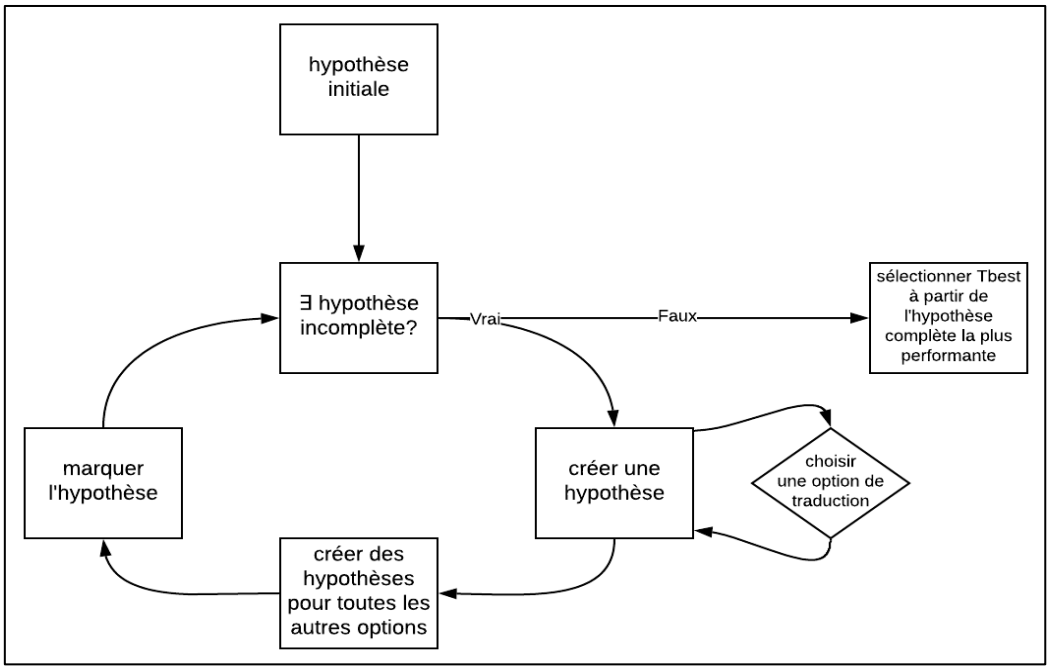


Figure 23: Processus de décodage.

Cependant, l'établissement d'un tel graphe avec une phrase longue avec de nombreuses options de traduction pose une énorme complexité d'espace de recherche, ainsi des hypothèses similaires sont recombinées pour réduire l'espace de

recherche. Cependant, bien que le problème ait peut-être été atténué ainsi, mais il reste présent.

Le décodage de pile *stack-decoding* dans la recherche *beam-search* organise les hypothèses en piles d'hypothèses. Si les piles deviennent trop volumineuses, nous devons éliminer les pires hypothèses de la pile. Une façon d'organiser les piles d'hypothèses est basée sur le nombre de mots étrangers traduits. Ainsi, l'algorithme suivant explique ce pr:

```
1. Place l'hypothèse vide dans la pile
2. Pour toutes les piles 0 ... n-1 Faire
3.     Pour toutes les hypothèses en pile Faire
4.         Pour toutes les options de traduction Faire
5.             Si applicable Alors
6.                 Créer une nouvelle hypothèse
7.                 Placer dans la pile
8.                 Recombiner avec l'hypothèse existante Si possible
9.                 Taillez la pile Si elle est trop grosse
10.                Fin Si
11.            Fin Pour
12.        Fin Pour
13.    Fin Pour
```

Cette organisation est un moyen d'organiser les piles en fonction des mots traduits. Toutes les hypothèses avec le même nombre de mots traduits sont placées dans la même pile. Chaque option de traduction est appliquée à une hypothèse et déposée dans la pile.

L'élagage des hypothèses se fait selon un algorithme d'élagage, telle que la plus mauvaise hypothèse de la pile en fonction de son score partiel est élaguée.

Le décodage de pile de recherche de faisceau réduit la complexité de l'espace en utilisant le décodage de pile pour élaguer une mauvaise hypothèse. Ceci produit une heuristique compacte réduisant ainsi la complexité et les erreurs de traduction lors du Backtracking pour trouver la meilleure traduction.

L'étape de décodage est l'étape qui prend la phrase d'entrée et produit une traduction dans la langue cible. Cependant, le résultat doit passer une évaluation pour être considéré comme une traduction correcte. Ainsi, dans la section suivante, nous établissons un processus d'évaluation de la traduction.

3.5 L'ÉVALUATION

Après toutes ces étapes franchies par le système pour fournir la traduction qui lui semble la plus optimale, il faut passer par une étape l'évaluation qui n'est pas moindre afin de fournir les mesures d'évaluation de la performance du système. Si nous revenons un peu en arrière, nous constatons que les données sont divisées en deux parties : la première étant pour l'apprentissage et la seconde pour les tests.

L'ensemble du test lui-même est divisé en deux parties, la partie en langue source (anglais) qui entrera dans le décodeur et nous fournira le fichier de sortie contenant les phrases traduites par le système qui sont considérées comme des phrases d'hypothèse. Ainsi qu'une partie de la langue cible (arabe) qui contient des phrases de référence qui seront comparées à l'ensemble de l'hypothèse.

Pour évaluer les performances de la traduction, nous utilisons quelques unités de mesure dans le domaine de la traduction automatique. Parmi ces mesures, on trouve :

3.5.1 Bleu Score

Le bleu score en anglais c'est l'abréviation de "Bilingual Evaluation Understudy" c'est-à-dire en français la compréhension de l'évaluation bilingue. C'est un score permettant de comparer une traduction de texte candidate à une ou plusieurs traductions de référence [44].

Bien que développé pour la traduction, il peut être utilisé pour évaluer le texte généré pour une suite de tâches de traitement de langage naturel.

Cette mesure se calcule comme suit :

$$\text{BLUE}=\min\left(1,\frac{hypothesis-lenght}{reference-lenght}\right) * \left(\prod_{i=1}^{4} précison_i\right)^{\frac{1}{4}}$$

Dans ce qui suit, nous donnons un exemple présentant l'utilisation de cette formule.

REF : يا له من يوم مشرق

HYP : يا له من يوم

On va commencer par le calcul de la précision de chaque N-gram (n= 1 jusqu'à n=4), c'est à dire on calcul les N-gram commune entre la phrase référence et la phrase hypothèse et on divise sur le nombre de N-gram dans la phrase hypothèse. Le calcul de précision est résumé dans le tableau suivant :

N-gram	Output
Précision (1 gram)	4/4
Précision (2 gram)	3/3
Précision (3 gram)	2/2
Précision (4 gram)	1/1

Tableau 2: Exemple de calcul des précisions pour la mesure BLEU score.

Après le calcul de la précision, on calcule le bleu score :

$$\text{Bleu score} = \min\left(1, \frac{4}{5}\right) * \left(\frac{4}{4} * \frac{3}{3} * \frac{2}{2} * \frac{1}{1}\right)^{\left(\frac{1}{4}\right)}$$

$$= \min(1, 0.8) * (1)^{(0.25)}$$

$$= 0.8 * 1 = 0.8 * 100 = \mathbf{80\%}$$

3.5.2 Le taux d'erreur des mots

Le taux d'erreur sur les mots (en abrégé : WER pour Word Error Rate en Anglais) est à la base un moyen de mesurer les performances des reconnaissances automatiques de la parole, mais il est également utilisé en traduction automatique [45]. Comme mentionné dans le premier chapitre le WER est juste la distance de Levenshtein pour les mots. Il compare une référence à une hypothèse et est défini comme suit :

$$\text{WER} = \frac{S+D+I}{N} \quad \text{Où :}$$

S : est le nombre de substitutions,

D : est le nombre de suppressions,

I : est le nombre d'insertions et

N : est le nombre de mots dans la référence.

Par exemple :

REF: يا له من يوم مشرق

HYP: يا له من يوم

Dans ce cas, les actions suivantes se produisent :

- Une suppression est survenue. " مشرق " a été supprimé par le système
- Une insertion s'est produite. " مشرق " a été inséré par le système
- Une substitution s'est produite. " مشرق " a été remplacé par le système

Ci-dessous, nous présentons un morceau de pseudocode pour faciliter la compréhension de cet algorithme :

```
function (References r, Hypotheses h) faire:
  int D [|r+1|] [|h+1|] ;
  Pour (int i=0 ; i<= |r| ; i++) faire :
    Pour (int j=0 ; i<=|h| ; j++) faire :
      Si i==0 faire :
        D[0][j] =j ;
      Sinon Si j == 0 faire :
        D[i][0] =i ;
  Fin Si ; Fin Pour ; Fin Pour ;
  Pour (int i=1; i<= |r|; i++) faire:
    Pour (int j=1 ; i<=|h| ; j++) faire :
      Si r[i-1] ==h[i-1] alors
          D[i][j] =D[i-1] [j-1];
      Sinon
          sub=D[i-1] [j-1] +1;
        ins=D[i][j-1] +1;
        del=D[i-1] [j]+1;
        D[i][j] =min (sub, ins, del);
  Fin Si ; Fin Pour ; Fin Pour ;
```

```
    Return D[|r|] [|h|];
Fin function;
```

3.6 CONCLUSION

Dans ce chapitre, nous avons présenté la conception de notre solution et nous avons modélisé chaque sous processus du système de traduction automatique statistique basé sur les phrases. Dans la première section, nous avons décrit l'architecture du système et la théorie impliquée, ainsi que ses composants. Ensuite, dans la section suivante, nous avons abordé la phase de préparation des données, processus indispensable dans le processus global de traduction, nous y avons passé en revue quelques techniques visant spécifiquement cette paire de langues et fourni des algorithmes de traitement de données simples pour répondre aux problèmes de langue arabe auxquels le système peut être confronté dans le processus de traduction.

Après la préparation des données, nous avons abordé le processus de traduction étape par étape, de l'apprentissage au décodage, en décrivant chaque composant nécessaire ainsi que le traitement des données dans chaque phase de l'apprentissage du système jusqu'à la phase de décodage où se produit la traduction. L'objectif étant de prendre une phrase dans la langue source et produire une traduction dans la langue cible.

A la fin, nous avons mis en place un pipeline d'évaluation pour estimer la traduction avec BLEU score afin de comparer la sortie de la traduction automatique à sa contrepartie traduite par l'homme bien que la notation ne soit pas vraiment adaptée à la langue arabe en raison de sa nature. Par ailleurs, nous avons présenté une autre technique d'évaluation qui est le WER pour faire une comparaison entre les deux mesures.

Dans le chapitre suivant, nous allons présenter les tests d'implémentation et de validation de notre système ainsi que les langages de programmation et outils utilisé pour réaliser ce projet en détail.

CHAPITRE 4
TESTS ET VALIDATION DU SYSTEME

4.1 INTRODUCTION

Dans ce chapitre, nous allons présenter l'implémentation de notre système. Nous commençons tout d'abord par la présentation des langages de programmation et environnements de développement, en détaillant les différents outils utilisés dans chaque étape, dès le prétraitement jusqu'à l'apprentissage en terminant par le décodage. Puis nous expliquons le déroulement de l'application, et enfin nous interprétons et commentons les résultats obtenus.

4.2 ENVIRONNEMENT DE DEVELOPPEMENT

Nous présentons dans cette section, le langage de programmation Python utilisé et son environnement de développement PyCharm. En plus, le Bash Scripting que nous avons utilisé avec les autres outils pour la construction des composants de notre système et qui ont été cité dans la conception.

4.1.1 Python

Python est un langage de programmation interprété de haut niveau qui a été initialement conçu par Guido van Rossum à la fin des années 1980 en tant que membre de l'Institut national de recherche en mathématiques et en informatique. Bien entendu, Python, à l'instar d'autres langages, a connu plusieurs versions. Python 0.9.0 a été publié pour la première fois en 1991. Outre la gestion des exceptions, Python incluait des classes, des listes et des chaînes.

En 2000, Python 2.0 a été publié. Cette version était plutôt un projet à source ouverte émanant de membres de l'Institut national de recherche en mathématiques et en informatique. Cette version de Python incluait des connaissances de liste, un collecteur d'ordures complet et la prise en charge de l'Unicode.

Python 3.0 était la version suivante et a été publié en décembre 2008 (la dernière version de Python est 3.8). Bien que Python 2 et 3 soient similaires, il existe des différences subtiles. Le fonctionnement de l'instruction « print » est peut-être celui qui convient le mieux. Comme dans Python 3.0, l'instruction « print » a été remplacée par une fonction « print () » [46].

Python fournit de nombreuses fonctionnalités faciles à apprendre et à utiliser. C'est un langage plus expressif, ce qui signifie qu'il est plus compréhensible et lisible comparé à d'autres langages [47].

- Python est un langage interprété, c'est-à-dire que l'interprète exécute le code ligne par ligne à la fois. Cela facilite le debugging et convient donc aux débutants.
- Il est multi plates-formes : les programmes tournent sans modification sur tous les environnements où Python existe (Windows, Unix et Mac).

4.1.2 PyCharm

PyCharm [48], est un environnement de développement intégré (EDI), utilisé en programmation informatique, spécialement pour le langage Python. Il est développé par la société tchèque JetBrains[5]. Il fournit une analyse de code, un débogueur graphique. PyCharm est multiplateforme, supporté sur les versions de Windows, MacOs et Linux. L'édition communautaire est publiée sous la licence Apache.

Il existe également une édition professionnelle avec des fonctionnalités supplémentaires - publiée sous une licence propriétaire.

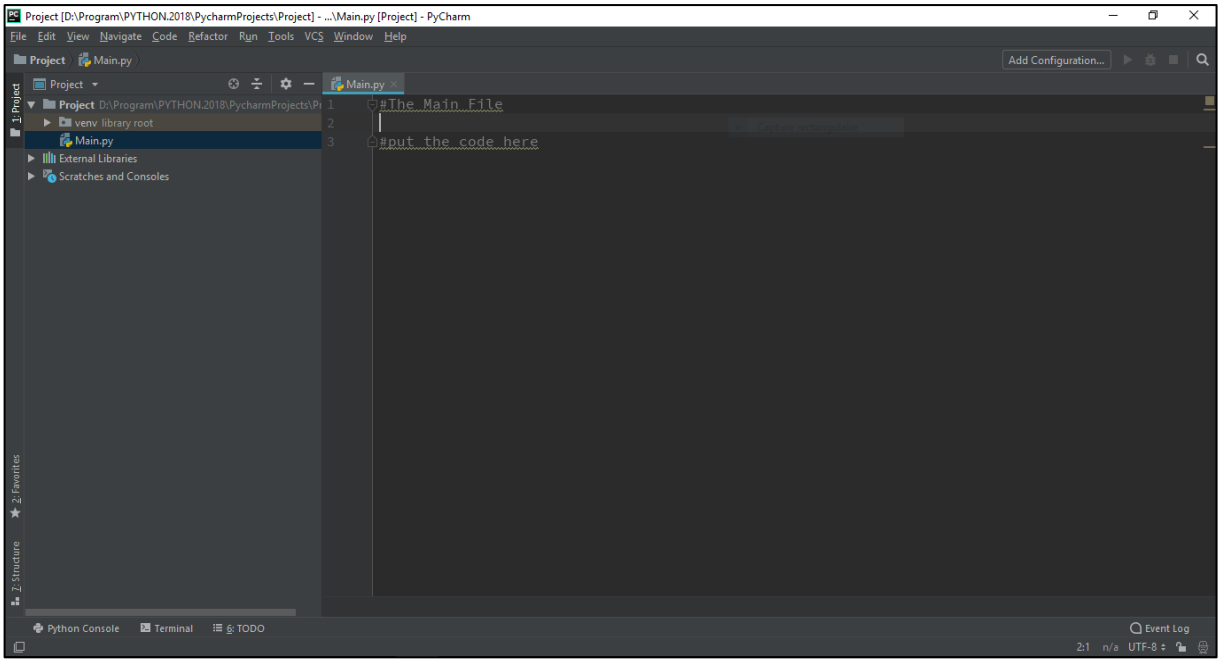


Figure 24: Environnement PyCharm.

[5] https://www.jetbrains.com/

4.1.3 Bash Scripting

Bash est un Shell Unix [49], sous forme d'une interface de ligne de commandes permettant d'interagir avec un système d'exploitation. Toute commande que vous pouvez exécuter à partir de la ligne de commande peut être utilisée dans un script Bash. Les scripts sont utilisés pour exécuter une série de commandes. Bash est disponible par défaut sur les systèmes d'exploitation Linux et MacOs.

Dans notre projet on a utilisé le système Ubuntu[6] comme un système d'exploitation qui est considérer une distribution de Linux.

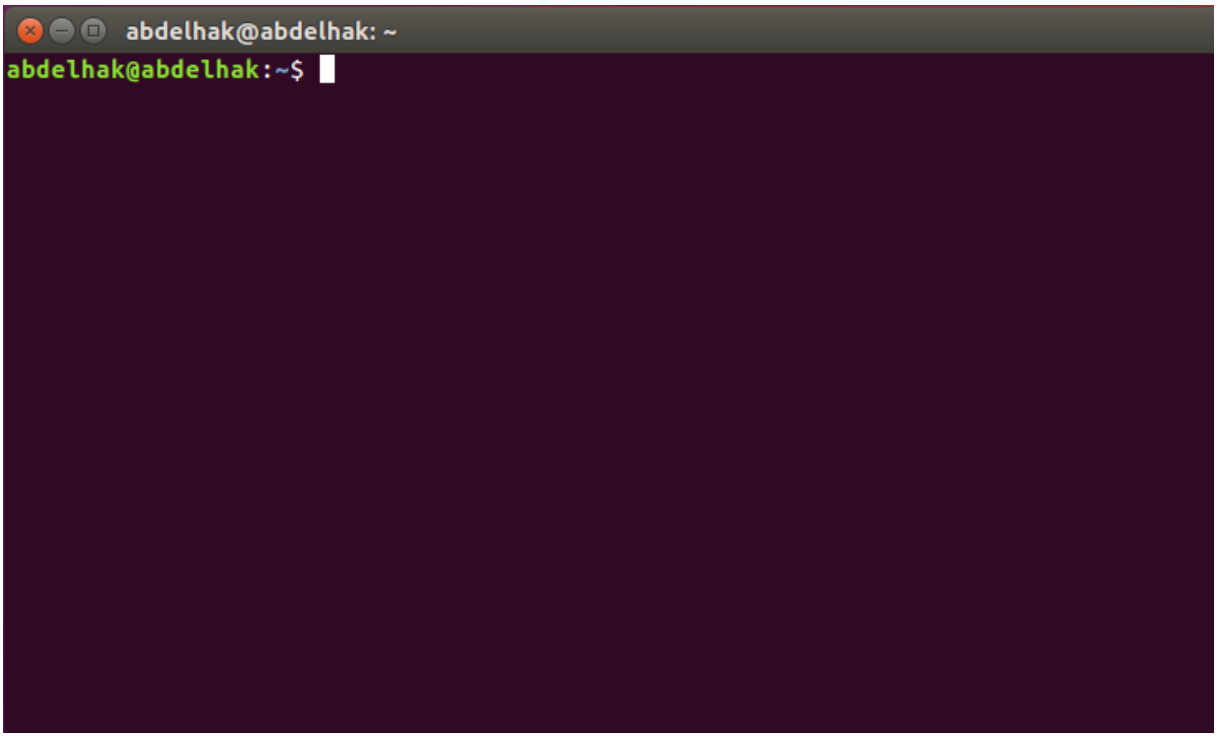


Figure 25: Interface en lignes de commandes (bash).

4.3 LES OUTILS UTILISES

Pour la construction de nos composants, nous avons utilisé un certain nombre d'outils. Que nous allons brièvement définir ci-dessous :

4.3.1 Moses

Le cœur de notre système est Moses. MOSES est une implémentation open source de l'approche statistique de la traduction automatique [50]. Cet outil contient deux composants

[6] https://ubuntu.com/

principaux qui font partie de l'apprentissage et du décodeur. Où la partie d'apprentissage est composée d'un ensemble d'outils et le décodeur est une application unique qui prend une entrée (texte) ainsi qu'un modèle de langage et un modèle de traduction. Il est en mesure de produire une traduction de l'entrée.

Moses et sa collection de composants fournissent la mise en œuvre de sous-tâches et d'algorithmes dans la traduction automatique statistique ainsi que des outils pouvant être utilisés pour configurer un système de pointe complet.

4.3.2 IRSTLM

Le IRST Language Modeling Toolkit [50] contient des algorithmes et des structures de données permettant d'estimer, de stocker et d'accéder à de très grands modèles de langage n-gram. Notre système repose sur le modèle de langage n-gramme, comme indiqué dans le chapitre précédent. IRSTLM est compatible avec Moses. De ce fait, nous l'avons utilisé pour la modélisation de la langue au lieu de la boîte à outils intégrée de modélisation du langage KenLm présente dans Moses. Cet outil est utilisé dans de nombreux systèmes de traduction automatique tels que l'ONU, la Commission européenne, Translated Srl, Apptek Inc et d'autres systèmes de traduction assistée par ordinateur.

4.3.3 GIZA++

La première étape du processus consiste à établir des alignements de mots. Pour cette tâche, nous utilisons Giza ++ qui est un outil de traduction automatique statistique ; extension du programme GIZA (faisant partie d'outils SMT EGYPTE) [50]. Cet outil est utilisé dans ce projet pour obtenir l'alignement des mots entre l'Arabe et l'Anglais. Il est basé sur les modèles d'alignement de mots décrits dans le chapitre précédent et intègre de nombreuses fonctionnalités. GIZA ++ est écrit en C ++ d'où l'appellation ++.

Nous avons également utilisé la bibliothèque Boost qui fournit des sources portables C ++ gratuites. Cette bibliothèque est nécessaire pour construire Moses et Giza ++.

4.4 DESCRIPTION DU SYSTEME

Notre système contient trois packages principaux qui sont : *modèle traduction*, *modèle de langage*, et *le décodeur*. Pour développer ces packages, nous avons utilisé plusieurs outils mais avant, nous avons procédé par un prétraitement pour les textes Anglais et Arabe.

Le système est construit sur des logiciels existants. le travail d'ingénierie comprend le choix et la compilation des systèmes logiciels et des bibliothèques ainsi que la sélection, le formatage et la collecte des corpus. Le travail de développement logiciel permettant de combiner et de coordonner différents systèmes logiciels et l'application de méthodes d'évaluation automatiques de la TA ainsi que des méthodes d'évaluation personnalisées et enfin de construire une interface utilisateur pour interagir avec le système. L'un des problèmes rencontrés a été l'utilisation de grandes quantités de données lors de l'apprentissage, ainsi que le test du système dans un délai raisonnable. Les tâches étaient souvent effectuées et reproduites parallèlement et les expirations étaient effectuées progressivement.

4.4.1 Prétraitement des données

Dans cette étape, nous avons implémenté notre propre algorithme depré-traitement avec python et nous avons utilisé différentes bibliothèques pour chaque sous-tâche. Nous commençons tout d'abord par le prétraitement des textes Anglais où la première étape est la normalisation (dans cette étape, il est question de réduire toutes les lettres en minuscules et supprimer toutes les lignes vides entre autre). La figure suivante montre le résultat d'un fragment de texte Anglais après normalisation.

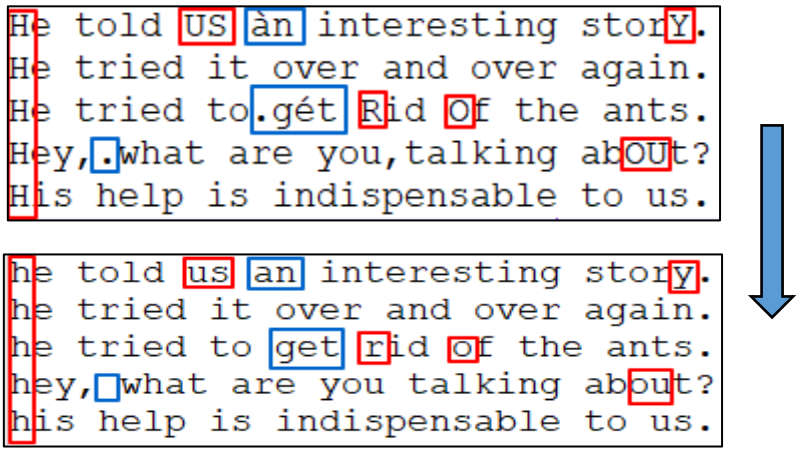


Figure 26: Exemple de normalisation d'un texte Anglais.

L'étape qui suit la normalisation est la tokenization et son but est d'identifier les mots et les signes spéciaux (les ponctuations, &, %, …etc.), Les différences dans les textes initiaux et les textes après la normalisation et tokenization sont illustrés dans la figure qui suit :

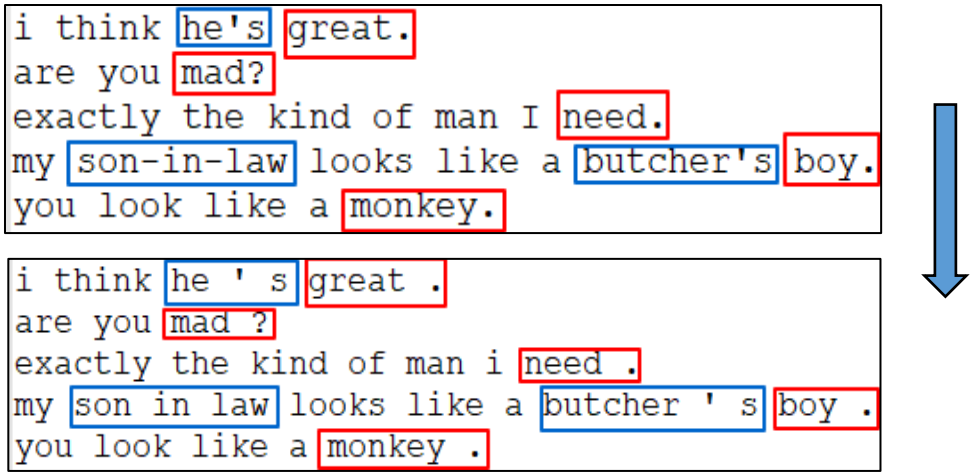


Figure 27: Exemple de tokenisation d'un texte Anglais.

Après la normalisation et la tokenization, nous procédons par un casing qui est une sous-tâche particulière pour les textes Anglais. Pour cette tâche, nous utilisons des scripts intégrés dans MOSES (les scripts sont écris en langage Perl[7]). Nous exécutons ce code avec le Bash-Scripting comme le montre la suivante :

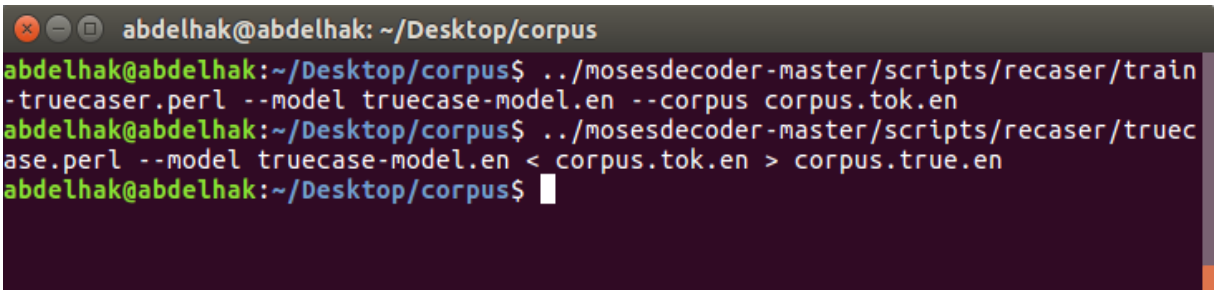


Figure 28: Exécution des scripts de casing.

Comme le montre la figure, il y a deux commandes, la première pour créer le model de casing c'est-à-dire de déterminer la capitalisation appropriée des mots (on prend le fichier qui contient des textes normalisé et tokenisé comme input et on retourne le fichier qui contient le casing de chaque mot). La deuxième commande concerne la création d'un fichier final qui contient les textes prétraités (normalisation, tokenization, casing).

[7] https://www.perl.org/

Idem pour la langue Arabe, nous devons nettoyer, mais avant cela, les textes en Arabe doivent être prétraités car le processus de nettoyage se déroulera en parallèle pour les textes en Arabe et en Anglais.

Pareil, on commence par la normalisation des textes Arabe. Pour cette sous-tâche nous avons utilisé notre propre code python et les résultats sont montrés dans la Figure 28 .

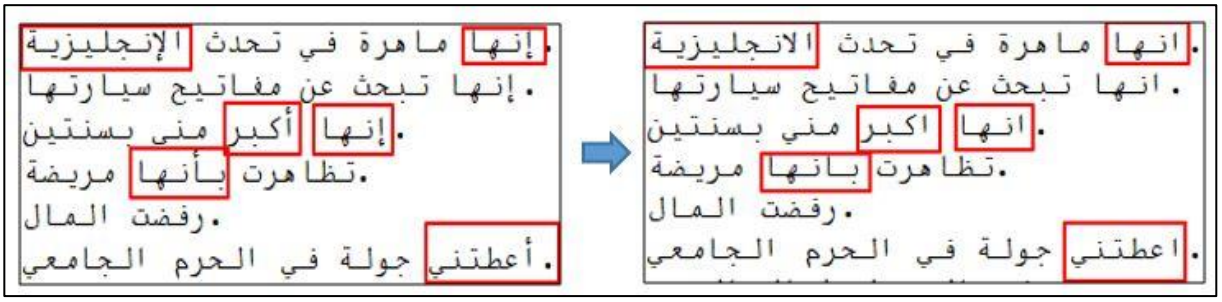


Figure 29: Exemple de normalisation d'un texte Arabe.

Nous passons maintenant à la tokenization et à la segmentation. Pour cette étape, nous avons utilisé les outils développés par le groupe de traitement du langage naturel de Stanford[8], à savoir Stanford-Segmenter et Stanford-Tokenizer, des bibliothèques développées en Java. Nous les connectons à l'aide de python via la bibliothèque NLTK et les résultats obtenus sont montrés dans la figure 29 :

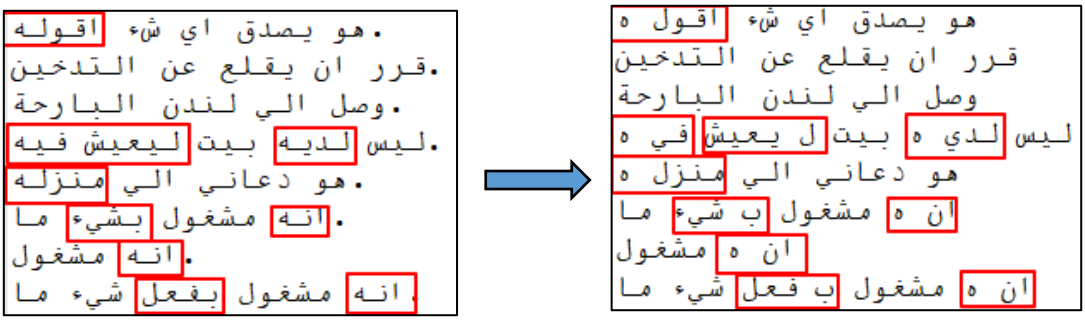


Figure 30: Exemple de tokenisation d'un texte Arabe.

Finalement, la dernière étape de prétraitement qui est le Casing pour l'Anglais la segmentation pour l'Arabe alors on peut maintenant entamer l'étape de nettoyage.

[8] https://nlp.stanford.edu/

Dans cette étape, nous avons utilisé des scripts intégrés dans MOSES (les scripts sont écris en langage Perl), on prend les deux fichiers finaux générés après les trois sous-tâches précédentes comme input et précisons les paramètres (1 à 80), c'est-à-dire la longueur minimale et maximale de la phrase.

Nous lançons l'exécution des scripts dans le bash comme montré dans la figure 30 ci-dessous :

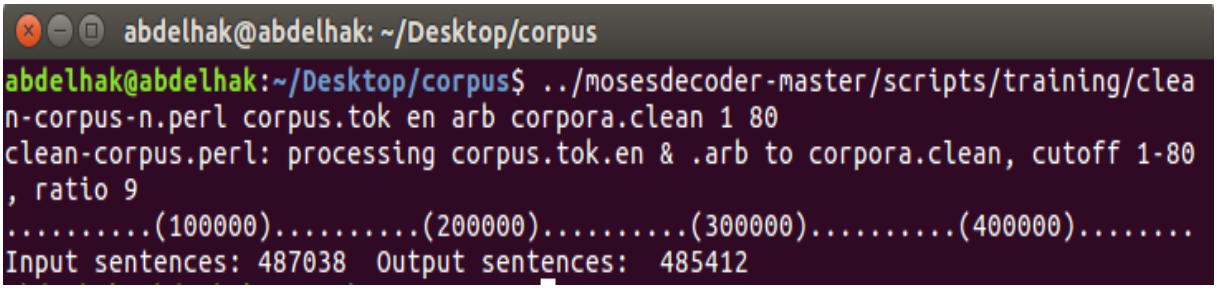


Figure 31: Exécution du script de nettoyage.

Nous remarquons qu'après le nettoyage le nombre de phrases diminue en raison de la suppression de phrases contenant plus de 80 mots.

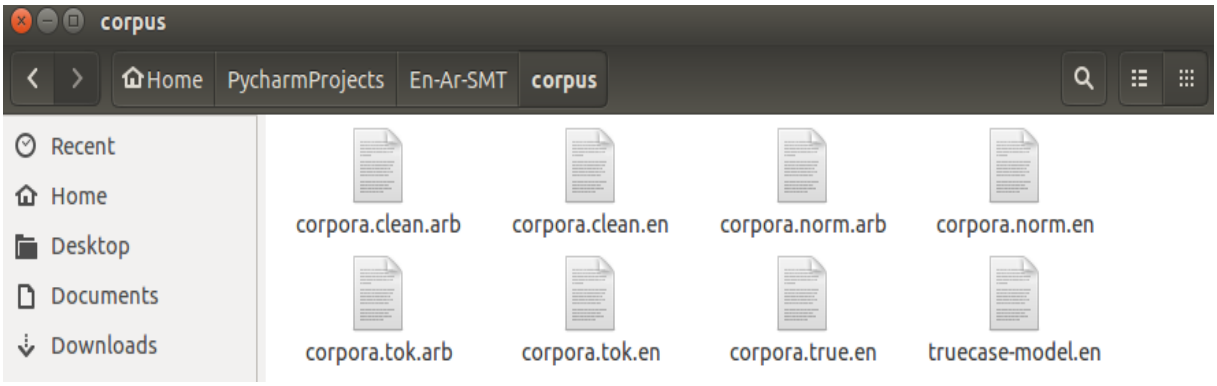


Figure 32: Le dossier Corpus contenant tous les fichiers traités.

4.4.2 L'apprentissage du Système

i. Modèle de Langage

Un modèle de langage est un modèle qui attribue une probabilité à une phrase, qui a une séquence de mots arbitraire. Le modèle de langage le plus largement utilisé est de loin le modèle de langage n-gram, qui décompose une phrase en séquences de mots plus petites (n-grammes) et calcule la probabilité sur la base de probabilités individuelles de n-grammes. Étant donné le

grand corpus de texte brut, nous avons voulu former un modèle de langage n-gramme et estimer la probabilité d'une phrase arbitraire.

Dans notre cas, nous allons utiliser IRSTLM Toolkit. Après la configuration de cet outil, nous créons un répertoire nommé LM qui contiendra tous les fichiers de notre modèle de langage.

Durant cette étape, nous allons utiliser des scripts intégrés dans IRSTLM (les scripts sont des fichiers SH qui sont des scripts programmé pour bash). La première instruction prend en entrée le fichier final obtenu après le prétraitement des textes Arabe et retourne un fichier qui contient des phrases marquées par le début et fin, comme il est montré dans la figure 10 :

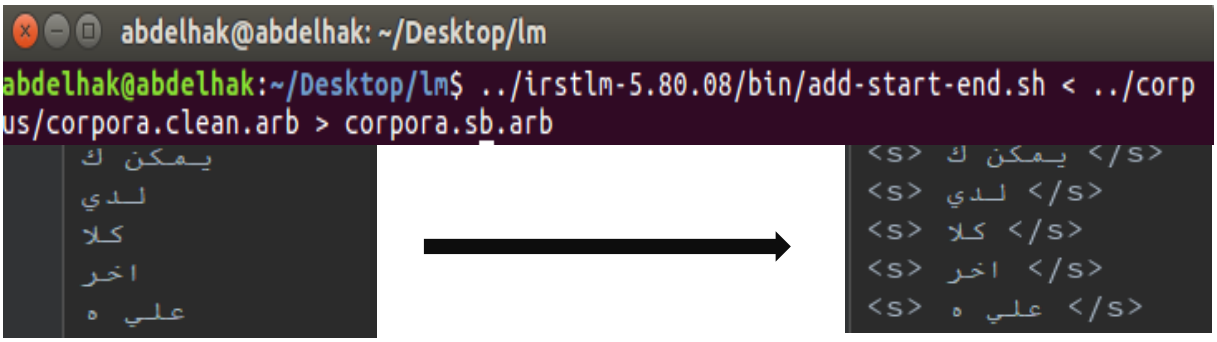


Figure 33:Exécution des scripts de début et de fin avec résultats avant/après.

La deuxième étape repose sur un script d'apprentissage qui utilise peu la mémoire de l'ordinateur et implémente la méthode de lissage Witten-Bell (Une approximation de la méthode de lissage de Kneser-Ney modifiée est également disponible). La commande est montrée dans la figure 33 :

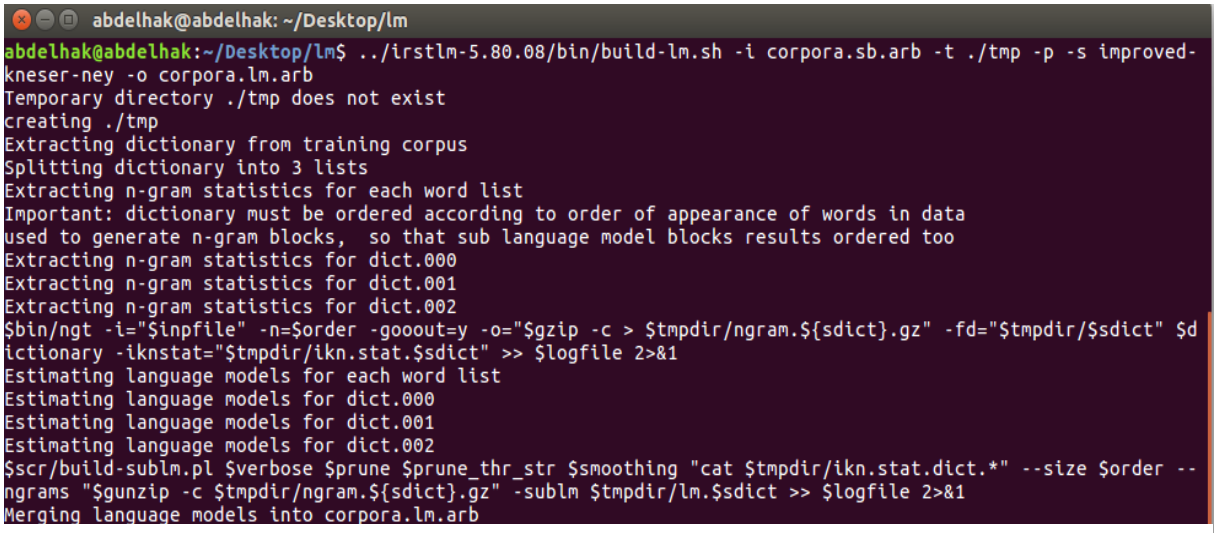


Figure 34: Script permettant de construire le modèle de langue.

Le script crée un modèle de langage de 3 grammes à partir de la commande d'entrée spécifiée (-i). Le modèle de langage sera enregistré dans le fichier de sortie (-o) corpora.lm.arb.gz avec un format ARPA intermédiaire. Ce format peut être correctement géré via la commande compile-lm afin de produire une version compilée ou une version ARPA standard du modèle de langage.

La prochaine instruction consiste à compiler le fichier corpora.lm.arb.gz et l'enregistrer dans un fichier texte nommé corpora.arpa.arb (appelé format ARPA) ce format convient bien pour l'interopérabilité entre les packages, la commande est montrée dans la [figure 34]

Figure 35: Script permettant de compiler les fichiers.

Cependant, ce format n'est pas aussi efficace que les formats binaires. Il est donc préférable de convertir ARPA en binaire pour une meilleure production. La commande est montrée dans la [figure 35] :

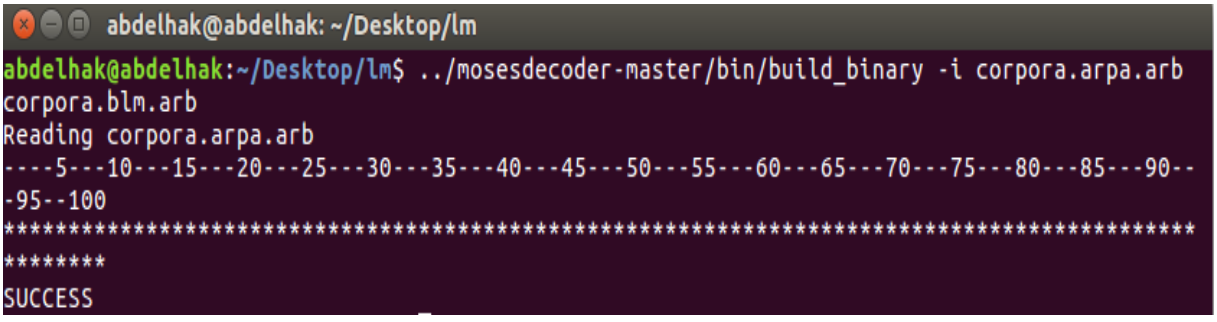


Figure 36: Conversion du fichier ARPA en binaire.

ii. Modèle de Traduction

Le processus d'apprentissage du modèle de traduction est séquentiel, car les données de chaque étape sont nécessaires à l'étape suivante, à l'exception du modèle de langage qui peut être exécuté en parallèle. Référençant le chapitre précédent, nous suivrons le même pipeline dans la formation du système.

a) *Préparation des données de d'apprentissage :* Les données d'entraînement doivent être alignées phrase par phrase, une phrase par ligne dans deux fichiers, une pour les phrases en Anglais et une pour les phrases en Arabe. Les données d'apprentissage passent par une phase de prétraitement, comme indiqué dans la sous-section précédente.

Les fichiers résultants sont alors prêts pour le processus d'apprentissage.

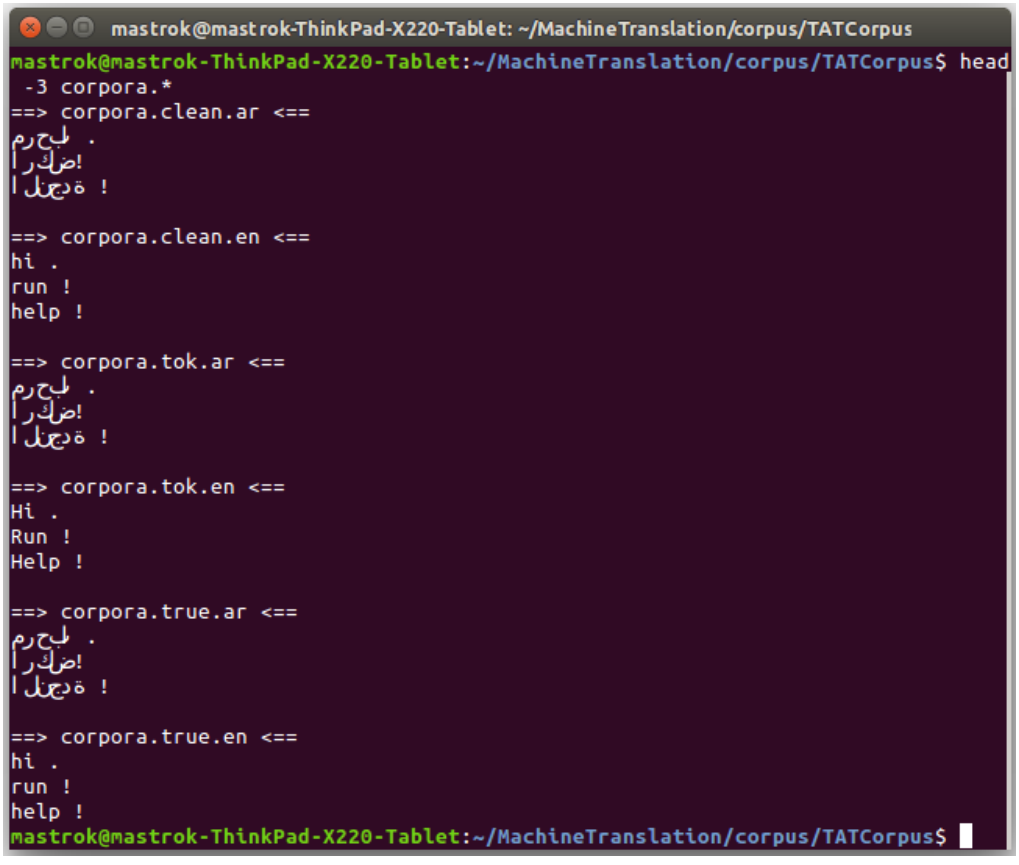


Figure 37 les fichiers nécessaires à la formation

b) *Alignement des mots*: La première étape de la formation du système consiste à établir des alignements de mots. Nos alignements de mots proviennent de l'intersection des informations d'alignement bidirectionnel de Giza++ et de

quelques points d'alignement supplémentaires issus de l'union des deux exécutions.

Le corpus parallèle doit être converti dans un format compatible à la boîte à outils GIZA++. Deux fichiers de vocabulaire sont générés et le corpus est numéroté.

```
mastrok@mastrok-ThinkPad-X220-Tablet: ~/MachineTranslation/working/TATWorking
mastrok@mastrok-ThinkPad-X220-Tablet:~/MachineTranslation/working/TATWorking$ he
ad -10 train/corpus/*.vcb
==> train/corpus/ar.vcb <==
1       UNK     0
2       و       157017
3       ان      119431
4       ل       118324
5       ب       105819
6       ،       90508
7       ه       87405
8       في      84499
9       من      79701
10      ؟       67583

==> train/corpus/en.vcb <==
1       UNK     0
2       ,       167254
3       the     163332
4       &apos;  114650
5       to      104562
6       you     103400
7       I       101760
8       and     100324
9       a       87836
10      of      82664
mastrok@mastrok-ThinkPad-X220-Tablet:~/MachineTranslation/working/TATWorking$
```

Figure 38 fichiers de vocabulaire

Les fichiers de vocabulaire contiennent des mots, des identificateurs de mots entiers et des informations sur le nombre de mots. GIZA++ exige également que les mots soient placés dans des classes de mots. Cela se fait automatiquement en appelant l'outil MKCLS.

Disney.com	51	2	-ابدا
Disneyland	38	35	-ابداوا
Disorder	12	7	-ابوك
Disordered	35	33	-اتعتقد
Dispatch	18	12	-اتعرف
Dispersion	35	33	-اتمني
Dispositive	35	26	-اتيت
Disposses	51	31	-اثنان
Dissipating	35	44	-اثنتان
Dissuaded	35	13	-اجتهد
Dissuasion	39	44	-اجل
Distel	7	44	-اجلسي
Distilleries	12	14	-احب
Distinguished	12	8	-احذية

Figure 39 : Les classes des mots Arabes et Anglais.

GIZA++ construit deux alignements, un dans chaque direction, produisant des informations sur l'alignement des mots.

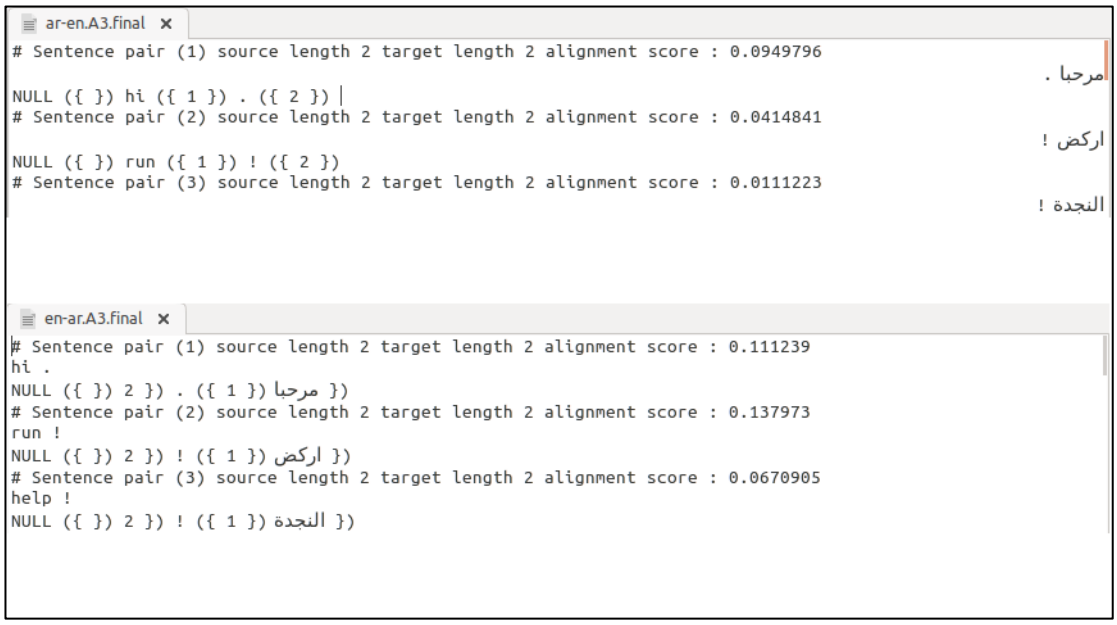


Figure 40 : Information d'alignement des mots dans les deux sens.

Après quelques informations statistiques et la phrase Arabe, la phrase Anglaise est listée mot par mot, avec des références à des mots Arabes alignés.

Chaque mot Anglais peut être aligné sur plusieurs mots Arabes, mais chaque mot Arabe ne peut être aligné que sur au plus un mot Anglais. Cette restriction un-à-plusieurs est inversée dans les informations d'alignement inverses GIZA++, comme indiqué sur la figure.

A partir de cette information d'alignement GIZA++ construit un fichier d'informations d'alignement utilisable, un point d'alignement à la fois, sous forme de position du mot Arabe et Anglais.

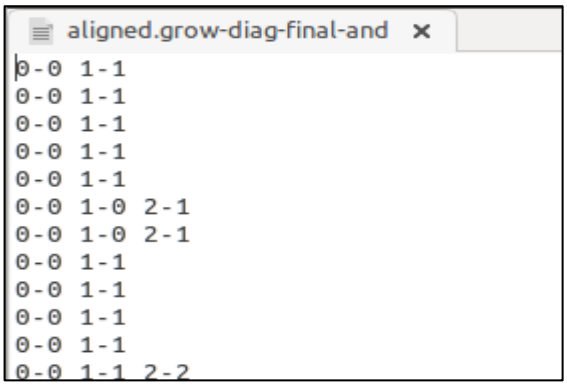


Figure 41 Iinformations d'alignement

c) *Table de traduction lexicale:* Moses propose des scripts de formation qui fonctionnent dans un ordre séquentiel étant donné cet alignement. Nous pouvons commencer le processus de formation en utilisant nos informations d'alignement de mots.

```
mastrok@mastrok-ThinkPad-X220-Tablet: ~/MachineTranslation/working/TATWorking
mastrok@mastrok-ThinkPad-X220-Tablet:~/MachineTranslation/working/TATWorking$ no
hup nice ../../mosesdecoder-master/scripts/training/train-model.perl -root-dir t
rain -corpus  ../../corpus/corpora.clean -f en -e ar -alignment grow-diag-final-
and -reordering msd-bidirectional-fe -lm 0:3:/home/mastrok/MachineTranslation/lm
/TATLm/corpora.blm.ar:8 -external-bin-dir ../../mosesdecoder-master/tools >& tra
ining.out &
[1] 10248
mastrok@mastrok-ThinkPad-X220-Tablet:~/MachineTranslation/working/TATWorking$ ps
  PID TTY          TIME CMD
 9537 pts/0    00:00:00 bash
10248 pts/0    00:00:00 perl
10258 pts/0    00:00:03 mkcls
10259 pts/0    00:00:00 ps
mastrok@mastrok-ThinkPad-X220-Tablet:~/MachineTranslation/working/TATWorking$ ps
```

Il est assez simple d'estimer une table de traduction lexicale à vraisemblance maximale. Nous estimons la table de traduction de mots $w(e|a)$ et son inverse $w(a|e)$.

broke 0.0294118 اخلف	broke 0.3333333 اخلف
broke 0.0294118 فعليهم	broke 0.2000000 فعليهم
broke 0.0294118 كسرها	broke 0.5000000 كسرها
broke 0.1176471 اندلعت	broke 0.3636364 اندلعت
broke 0.0588235 تعطلت	broke 0.5000000 تعطلت
broke 0.0294118 منكسرا	broke 0.3333333 منكسرا
broke 0.0294118 حريق	broke 0.3333333 حريق
broke 0.1470588 لقد	broke 0.0202429 لقد
broke 0.0588235 خالفت	broke 0.5000000 خالفت
broke 0.1470588 كسر	broke 0.6250000 كسر
broke 0.0294118 اقتحم	broke 0.1666667 اقتحم
broke 0.2058824 كسرت	broke 0.5833333 كسرت
broke 0.0294118 وكسر	broke 0.2500000 وكسر
fighting 0.2000000 يقاتل	fighting 1.0000000 يقاتل
fighting 0.2000000 استمروا	fighting 0.1428571 استمروا
fighting 0.2000000 بمكافحة	fighting 0.5000000 بمكافحة
fighting 0.4000000 يحل	fighting 0.2500000 يحل
golden 0.5000000 الصمت	golden 0.2000000 الصمت
golden 0.5000000 ذهب	golden 0.0312500 ذهب

Figure 42 : Table de traduction lexicale.

d) *Extraction des phrases:* Le processus de formation est séquentiel. Dans l'étape d'extraction de phrases, toutes les phrases sont placées dans un seul gros fichier par le module Extraction de phrases de Moses.

Le contenu de ce fichier est le suivant, pour chaque ligne :

Phrase Arabe, phrase Anglaise et points d'alignement.

Les points d'alignement sont des paires (Arabe, Anglais). De plus, des informations d'alignement inversées sont générées.

Figure 43 Extraction des Phrases

e) *Notation des phrases:* Ensuite, une table de traduction est créée à partir des paires de traduction de phrase stockées.

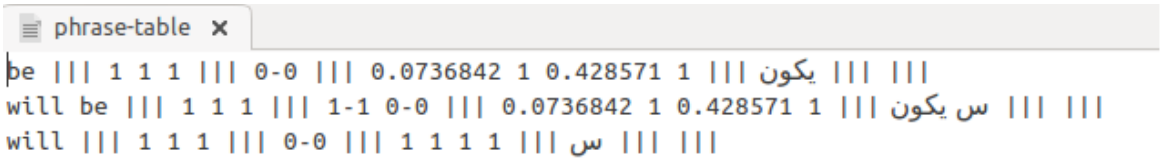


Figure 44 Table de traduction.

Quatre scores de traduction de phrases différents sont alors calculés:

1. probabilité de traduction de la phrase inverse φ (a | e)
2. pondération lexicale inverse lex (a | e)
3. probabilité de traduction de la phrase directe φ (e | a)
4. pondération lexicale directe lex (e | a)

f) *Le modèle de réorganisation:* Moses implémente un modèle de réorganisation basé sur la distance par défaut et accepte les paramètres. Pour notre système, nous avons utilisé la configuration par défaut pour un SMT basé sur une phrase.

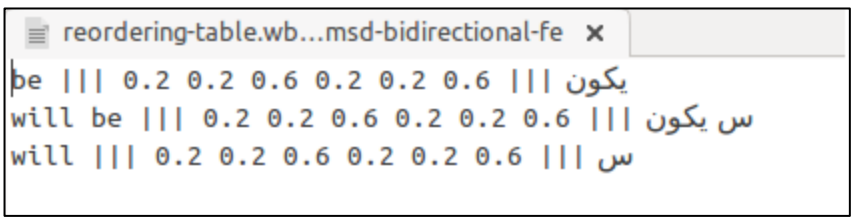


Figure 45 : La table de réorganisation

g) *Fichier de configuration*: Enfin, un fichier de configuration pour le décodeur est généré avec tous les chemins corrects pour le modèle généré et un certain nombre de paramètres par défaut. Ce fichier s'appelle moses.ini. Le fichier de configuration contient les chemins pour la table de phrases et la table de réorganisation, ainsi que des fonctions telles que Pénalité de mots inconnus, Pénalité de mots, Pénalité de phrase. Le fichier de configuration contient également le chemin d'accès et la configuration du modèle de langage.

Le fichier de configuration est défini par défaut sur l'utilisation de SRILM en tant que kit d'outils de modèle de langage. Cependant, nous avons utilisé la boîte à outils IRSTLM pour notre modèle de langue. En tant que tel, nous mettons à jour le fichier de configuration pour utiliser notre modèle de langage construit avec IRSTLM.

$$IRSTLM\ order = 3\ factor = 0\ path = \sim/corpora.blm.ar\ oov - feature = 1$$

Enfin, dans le fichier de configuration, nous pouvons trouver les pondérations pour les fonctions, le modèle de langage, le modèle de traduction ainsi que la réorganisation lexicale. Les poids peuvent être ajustés à l'aide de la fonctionnalité de réglage pour perfectionner le système.

h) *Binarisation des tables lexicales et reorganization:* À ce point, le système est prêt à effectuer la traduction mais, en raison de la surcharge des données chargées en mémoire, la traduction est lente. Nous effectuons donc la binarisation de la table de traduction ainsi que de la table de réorganisation lexicale et les sauvegardons sur disque.

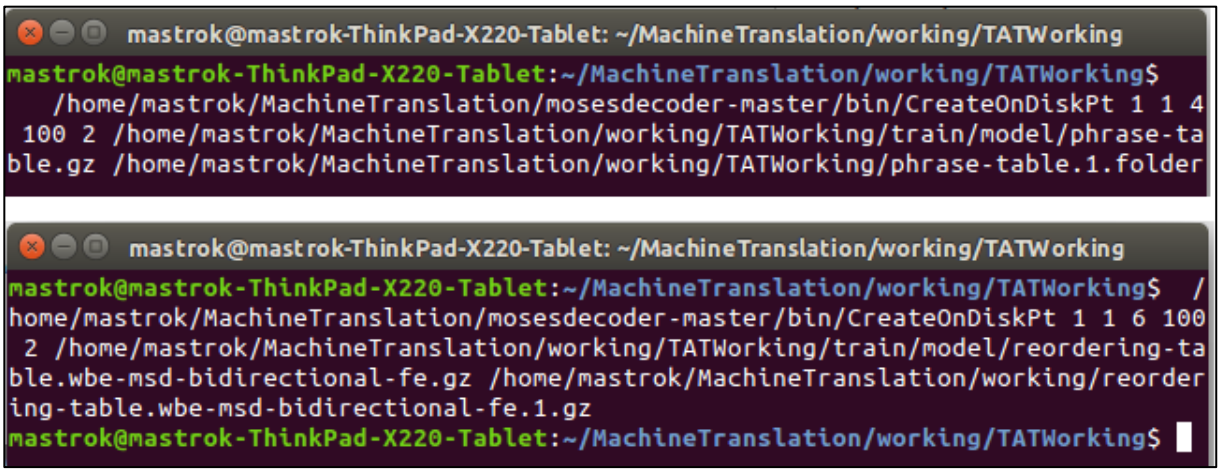


Figure 46 : Binarisation de la table de traduction et de la table de réorganisation lexicale, respectivement.

Une fois encore, le fichier de configuration doit être mis à jour pour utiliser les données binaires. Ceci conclut le processus de formation. Dans la section suivante, nous testons notre système de base et effectuons des expérimentations évaluant ainsi les résultats obtenus.

4.5 EXPERIMENTATIONS

Avec le système de base terminé, nous pouvons effectuer des expérimentations pour tester notre système.

4.5.1 Système de base

Dans la section précédente, nous avons construit et formé notre premier modèle. Dans ce document, nous testons et évaluons ce modèle en tant que système de base que nous pourrons ensuite utiliser comme référence de mesure et données statistiques pour comparer et évaluer avec les autres modèles issus des expérimentations.

4.5.2 Données d'apprentissage

Pour ce modèle, nous avons utilisé le corpus parallèle Anglais-Arabe de Tatoeba[9] avec 13,000 phrases. Nous avons utilisé un prétraitement simple pour nettoyer et tokenizer le corpus parallèle uniquement pour le pipeline de formation et de test.

4.5.3 Ensemble de test

L'ensemble de test a été extrait à partir des données de formation. En effet, nous avons sélectionné 10% du corpus parallèle comme notre ensemble de jeu d'essai. La majeure partie de l'ensemble de test consistait en des moyennes à longues phrases du corpus parallèle. L'ensemble de tests ne comprend que les phrases Anglaises en entrée du système de traduction. Tandis que les phrases en Arabe sont utilisées comme référence pour l'évaluation.

[9] https://tatoeba.org/fra/

4.5.4 Résultats

Pour évaluer le modèle, nous avons utilisé l'évaluateur de score Interactive BLEU de *letsmt*[10] pour effectuer des évaluations comparatives de la qualité des fichiers traduits avec notre système SMT. L'évaluateur nous permet de comparer la sortie de traduction de notre système à la traduction humaine, qui correspond aux phrases Arabes de l'ensemble de tests.

Entrée de test	948 phrases en Anglais de l'ensemble de test	BLEU = 4.08
Référence d'évaluation	948 phrases en Arabe de l'ensemble de test	

Tableau 3: Résultats de l'évaluation avec BLEU.

4.5.5 Analyse et Observations

Après avoir évalué la qualité de la traduction avec l'évaluateur de scores BLEU interactif, nous avons effectué, à l'aide des données statistiques de l'évaluateur de scores Bleu interactif, une inspection manuelle de l'évaluation de la sortie de notre système de traduction. Nous avons observé de nombreux problèmes avec la sortie de traduction et analysé l'impact et les implications possibles sur la qualité de la traduction qui sont présentés dans le tableau suivant :

Observation	Cause	Impact
Apparition de mots Anglais dans la traduction Arabe	La fragmentation des données. Le corpus de formation est petit et contient des phrases uniques.	Une observation insuffisante des mots respectifs dans l'entraînement entraîne l'apparition de mots non traduits. Réduction considérable de la qualité de la traduction ou les cas de non-production de traduction.
Evaluation négative de mots lexicalement identiques.	Différents signes diacritiques ornant les mêmes mots affectent l'évaluation	Évaluation négative non discriminante. faux négatifs. évaluation de traduction inférieure.

Tableau 4: Observations de l'évaluation BLUE score.

Sur la base de ces observations, nous avons cherché à résoudre les problèmes en appliquant des techniques de prétraitement afin de résoudre le problème des signes

[10] https://www.letsmt.eu/Bleu.aspx

diacritiques grâce à la normalisation des données en texte Arabe. Nous avons temporairement ignoré la taille du corpus et la fragmentation des données, que nous avons résolus par la suite en augmentant la taille de données de formation.

Nous avons effectué l'évaluation à nouveau après avoir normalisé les données textuelles Arabes à l'aide du même évaluateur de scores BLEU interactif et de la même sélection d'ensemble de tests.

Entrée de test	948 phrases en Anglais de l'ensemble de test, normalisées	BLUE = 8.2
Référence d'évaluation	948 phrases en Arabe de l'ensemble de test, normalisées	

Tableau 5: Résultats de l'évaluation BLEU score après normalisation.

Une deuxième analyse de ces résultats, nous nous a permis de noter les nouvelles observations présentées dans le tableau ci-dessous :

Observation	Cause	Impact
Problèmes de traduction et d'évaluation avec quelques mots Arabes.	Les mots Arabes avec plusieurs morphèmes affectent l'alignement des mots et ne sont pas suffisamment observés lors de la formation. le système d'évaluation attribue souvent des notes négatives à ces mots pour la même raison.	Il en résulte de nombreuses traductions incorrectes de certains mots car il existe un conflit lorsque ces types de mots et des mots simples similaires ont une traduction identique. Faux positifs fréquents et faux négatifs lors de l'évaluation.

Tableau 6: Observations de l'évaluation BLEU score après normalisation.

Pour résoudre ce nouveau problème, nous avons implémenté une autre technique de prétraitement pour segmenter le texte Arabe, ce qui a amélioré considérablement l'alignement des mots dans la formation ainsi que dans l'évaluation. Après une autre itération de formation et de tests des nouvelles données, nous avons réalisé une amélioration significative de l'évaluation ainsi que du résultat de la traduction.

Entrée de test	948 phrases en Anglais de l'ensemble de test, normalisées et segmentées	BLUE = 18.36
Référence d'évaluation	948 phrases en Arabe de l'ensemble de test, normalisées et segmentées.	

Tableau 7: Résultats de l'évaluation BLEU score après normalisation et segmentation.

Grâce aux expériences itératives sur le système de base, nous avons résolu en partie le problème et obtenu une qualité de traduction raisonnable, ce qui nous donnait la confiance nécessaire pour tenter de résoudre le problème de la fragmentation des données en augmentant le volume de données de formation.

4.6 MODELES

Les données sont l'un des principaux défis que nous avons dû relever pour établir un modèle de traduction offrant une qualité de traduction décente. La sélection et la collecte de corpus ainsi que le temps nécessaire au traitement des données sont liés au volume croissant de données.

4.6.1 Données d'apprentissage

La collecte d'un corpus parallèle prend beaucoup de temps et, compte tenu du fait que la qualité des données est étroitement liée à la qualité de la traduction, la tâche est plus ardue.

En raison des contraintes de temps, nous avons opté pour des corpus existants dans tous les modèles, à l'exception de l'un d'entre eux, qui consistait à compiler manuellement un corpus parallèle à partir de différentes sources afin d'atteindre un certain volume dans le cadre des restrictions.

Les corpus utilisés au cours des expériences avec différents modèles proviennent principalement d'OPUS[11], une collection grandissante de textes traduits sur le Web. les données compilées proviennent de sources gratuites et la plupart des corpus sont alignés. Ci-dessous dans le tableau sont les corpus utilisés :

[11] http://opus.nlpl.eu

Corpus	Doc.	Phrases	Ar. Tokens	An. Tokens	Parallèle	modèle
MultiUN v1	67617	10.6 M	263.1 M	289.6 M	Oui	[2]
MultiUN v2	1	47.1K	3.3M	3.7M	Oui	[2]
TED 2013 v1.1	1	0.2 M	2.4 M	2.6 M	Oui	[5]
GNOME v1	1313	0.5 M	2.4 M	2.6 M	Oui	[5]
Tatoeba	1	13 K	90.1K	3.6M	Oui	[1,5]
GlobalVoices v2017q3	3693	52.8k	1.2 M	1.7M	Oui	[5]
OpenSubtitles v2018	40979	31.9 M	7.9 M	5.6 M	Oui	[3, 4, 5]
Dictionray App DB	1	70k	UNK	UNK	Oui	[5]

Tableau 8: tableau des différents corpus utilisés, leurs caractéristiques et, dans quel modèle, les corpus ont été utilisés.

Naturellement, nous n'avons utilisé qu'une petite partie des grands corpus en raison du temps nécessaire au traitement et à la formation du système, qui utilisait un très grand volume de données, des contraintes matérielles et, enfin, la portée de ce travail.

Le tableau ci-dessous présente les données utilisées dans chaque modèle :

modèle	Copus	Phra.	Observation
Baseline [Exp 1]	Tatoeba	13k	La qualité du corpus est assez bonne.
Model 2 [Exp 2]	MultiUN[v1 , v2]	25k	Traduction avec perte et phrases manquantes.
Model 3 [Exp 3]	OpenSubtitles v2018	100k	Domaine, termes et qualité de la traduction variables.
Model 4 [Exp 4]	OpenSubtitles v2018	400k	Domaine, termes et qualité de la traduction variables.
Model 5 [Exp 5]	TED,Gnome, GV,Tatoeba,OpenSubs,DADB	500k	Corpus collecté manuellement de la collection. [200k,OpenSubs] ,[100,Gnome], [10k,GV]

Tableau 9: tableau montrant les données collectées utilisées dans chaque modèle, ses sources et observations sur la qualité des données.

Le modèle 5, qui est la dernière expérience que nous avons effectuée, utilise un corpus que nous avons collecté manuellement à partir de la collection de corpus.

4.6.2 Évaluation

En suivant la même méthode que la dernière itération du système de base. nous avons mis en œuvre de nombreux scripts de nettoyage pour supprimer le bruit des corpus, tels que des caractères spéciaux, des tabulations, des lignes vides, des caractères mal codés et bien d'autres types de bruit, ainsi que des techniques de prétraitement appropriées pour résoudre les problèmes auxquels nous étions confrontés avec le système de base.

Notant que nous avons utilisé un ratio de 20% des corpus d'entraînement comme jeu d'essais pour la plupart des modèles, ce qui est un très grand ensemble d'essais considéré comme exhaustif et non optimal pour évaluer correctement le système. D'autant plus qu'il introduit le problème de la fragmentation des données. Le tableau ci-dessous montre les résultats de l'évaluation pour chaque modèle.

Model	**Test set sents.**	**BLEU**	**1-gram**	**2-gram**	**3-gram**	**4-gram**	**Observation**
Baseline [Exp 1]	948	18.36	58.33	47.24	36.81	18.36	[1]
Model 2 [Exp 2]	5k	12.08	38.73	25.19	17.28	12.08	[2]
Model 3 [Exp 3]	20k	11.02	36.41	24.64	17.19	11.02	[3]
Model 4 [Exp 4]	40k	14.80	42.71	28.12	19.08	14.80	[4]
Model 5 [Exp 5]	50k	18.11	49.10	37.01	27.83	18.10	[5]

Tableau 10: tableau montrant les résultats d'évaluation de chaque modèle et jeu de données de test utilisé et observations

[1] : le corpus est assez correct mais l'évaluation a été grandement compromise par la rareté des données (Data Sparsity).

[2] : le corpus souffre d'une traduction avec pertes et de morceaux de phrases manquants.

[3] : le corpus présente des domaines, des termes et une qualité de traduction variables. L'ensemble de tests provient également d'un domaine différent de celui observé dans les données d'apprentissage, ce qui entraîne un problème de parcimonie des données.

[4] : Augmentation du volume de données. Problème de faible densité de données sensiblement atténué, mais pas suffisamment pour contrer les problèmes au sein du corpus.

[5] : le corpus est collecté à partir de différentes sources et fusionné. L'objectif était d'introduire des phrases plus longues progressivement dans le corpus. Nous avons d'abord ajouté une traduction mot à mot unique à partir d'une base de données d'une application de traduction afin d'accroître le vocabulaire du système, puis une traduction à deux mots et une incrémentation progressive de la longueur des phrases.

Nous avons ainsi constaté que cette approche améliorait grandement l'évaluation par rapport aux modèles précédents.

4.6.3 Discussion

En utilisant le modèle 5, nous avons effectué une autre évaluation en utilisant un ensemble d'essais plus petit, numéroté 1600, extrait d'OpenSubtitles. Constatant que le corpus posait de nombreux problèmes et qu'il était spécifique à un domaine et que la traduction était de qualité et norme variables, nous avons obtenu les résultats suivants :

modèle	**Phr. D'ensemble de test**	**BLEU**	**1-gram**	**2-gram**	**3-gram**	**4-gram**
modèle 5 [Exp 5]	1600	22.29	50.38	38.16	29.04	22.29

Tableau 11: Résultats de l'évaluation du modèle 5 de BLUE en utilisant un ensemble de données de test plus petit avec des domaines mixtes.

Cependant, nous avons toujours rencontré un problème de parcimonie des données et avons ainsi rassemblé un petit corpus de tests de 1600 phrases non spécifiques à un domaine mais plus générales. De nombreuses phrases ont été ajoutées manuellement tandis que le reste a été rassemblé et édité pour être plus général que spécifique à un domaine.

Nous avons à nouveau évalué le modèle 5 avec le nouvel ensemble de tests et avons obtenu les résultats suivants:

Modèle	Phr. D'ensemble de test	BLUE	1-gram	2-gram	3-gram	4-gram
modèle 5 [Exp 5]	1600, general	26.90	55.92	43.49	34.08	26.90

Tableau 12: Résultats de l'évaluation du modèle 5 de BLUE à l'aide d'un ensemble de données de test plus petit, général et non spécifique à un domaine

En conclusion, les expérimentations incrémentielles nous ont permis de parcourir les résultats et l'évaluation de la traduction, d'observer les problèmes et de les résoudre ce qui a conduit progressivement à une amélioration significative de la qualité de la traduction. Les expérimentations nous ont montré que le prétraitement et le traitement du problème de fragmentation des données entraînaient une amélioration importante de la qualité de la traduction, et que la qualité initiale des données était étroitement liée à la qualité de la traduction. Un autre problème est la compatibilité entre les données d'apprentissage et de test. Plus elles sont proches, meilleure est la qualité de la traduction, et inversement. En outre, nous sommes convaincus que d'autres améliorations pourront être apportées à cet égard, que nous prévoyons d'explorer dans les travaux futurs.

CONCLUSION GENERALE

Au cours des dernières décennies, la traduction automatique a considérablement évolué en raison du développement technologique en particulier dans le domaine de l'Informatique et aussi la facilité d'accès à Internet, qui ont fait du monde un petit village. Cela a conduit à la convergence de cultures et de langues différentes, d'où un intérêt accru pour la traduction automatique afin d'améliorer la communication entre les personnes.

Notre travail visait à mettre en place un système de traduction automatique fonctionnel, un système de traduction automatique statistique destiné spécifiquement au couple de langues Anglaise et Arabe, qui est l'objectif principal de ce travail. Cependant, au cours de cette tâche, nous avons rencontré quelques difficultés affectant négativement la qualité de la traduction, mais que nous sommes parvenus à résoudre grâce à des observations, des analyses et des expérimentations. Nous estimons qu'au terme de notre stage, nous sommes parvenus à des résultats acceptables mettant en évidence la majeure partie de nos contributions, principalement axées sur la collecte, le prétraitement et l'évaluation des données.

La première difficulté que nous avons rencontrée dans notre projet est le défi de la traduction de documents qui ne ressemblent pas au contenu du corpus d'apprentissage (problème courant dans les modèles statistiques). Bien que notre système puisse exceller avec les textes définis par le corpus d'apprentissage, tels que des textes techniques écrits dans un style simple, le système aura des difficultés si nous lui fournissons un texte contenant de l'argot, des idiomes ou un style qu'il n'a pas appris.

Dans ces cas, la qualité de notre système diminue considérablement. Par conséquent, les corpus doivent être personnalisés pour qu'un style spécifique soit le plus efficace possible. Même dans ce cas, le système est incapable de traduire les idiomes et le matériel marketing. Pour être plus clair son utilisation pour un style informel entraîne une qualité de traduction médiocre.

Pour améliorer la qualité de traduction on a essayé plusieurs expériences avec différents corpus en termes de type et de taille jusqu'à ce que nous sommes arrivés au modèle final qui traite différents domaines élargir la portée

Il y a aussi une autre difficulté que nous avons rencontré qui est la différence morphologique entre l'Anglais et l'Arabe. La complexité morphologique de la langue arabe a conduit à l'utilisation de certaines techniques de prétraitement, dont la segmentation. Cette technique nous a permis de séparer les pronoms attachés aux mots ce qui a considérablement amélioré la traduction par a rapport à d'autres modèles.

Nous avons aussi utilisé cette technique dans l'ensemble de test de référence dans l'étape d'évaluation, avec la mesure WER et les résultats étaient bons par rapport à l'ensemble de tests de références non segmenté.

Grâce aux travaux menés dans ce projet, nous avons acquis une connaissance satisfaisante dans le domaine de la traduction automatique et une connaissance approfondie de la traduction automatique statistique. Allant de l'histoire à la théorie en passant par le concept, en la mettant en pratique et en approfondissant les détails du fonctionnement interne d'un système de traduction statistique. Cependant, nous avons également acquis une bonne compréhension de certaines des difficultés auxquelles se heurtent la traduction automatique en général et la traduction automatique statistique en particulier, car nous avons appris qu'une partie de ces difficultés provenait de l'histoire, de la théorie et de la pratique, et que nous avons rencontré au cours de ce processus. Nous avons cherché à résoudre ces difficultés telles que l'acquisition et la collecte de données, les données ayant un impact direct sur la qualité de la traduction. Un autre problème est lié aux limites de la technologie, telles que l'accessibilité aux logiciels existants et au matériel de hautes performances, qui sont nécessaires pour traiter de gros volumes de données et atténuer le problème de temps de traitement.

Naturellement, ce travail est sévèrement affecté par un temps limité, ce dont ce type de travail a grandement besoin. Cette contrainte ainsi que le fait que ce travail soit une nouveauté pour nous ont joué un très grand rôle nous empêchant d'explorer, d'expérimenter ainsi que de mettre en œuvre des techniques agréés ou prouvées qui amélioreraient considérablement la qualité de la traduction.

Cependant, la traduction automatique statistique a ses limites, comme nous l'avons appris et expérimenté. Ainsi, tirant parti de nos connaissances et de notre

expérience nouvellement acquise, Notre travail suivant porterait sur l'exploration de systèmes de traduction automatique hybride et la tentative d'introduire de nouvelles contributions de ce que nous pensons, selon nos connaissances actuelles, être bénéfique pour la traduction automatique de et vers la langue arabe.

REFERENCES BIBLIOGRAPHIQUES

[1] «Machine Translation past present future,» John Hutchins, 1985.

[2] «Machine Translation,» [En ligne]. Available: kantanmt.com/machinetranslation.html. [Accès le 13 Mars 2019].

[3] P. F. C. J. D.-P. S. A. e. a. Brown, «A statistical approach to machine translation.,» chez *Computational Linguistics*, 1990.

[4] F. J. O. a. H. Ney, «A systematic comparison of various statistical alignment models,» chez *Computational linguistics*, 2003.

[5] A. F. K. Shaalan, «Arabic Natural Language Processing: Challenges and Solutions,» chez *ACMTransactions on Asian Language Information Processing,*, Village, Block 17, Dubai United Arab Emirates (UAE), P.O. Box 502216, Dubai, 2009.

[6] K. VERSTEEGH, «The Arabic Language,» chez *Columbia University Press,* New York., 1997.

[7] M. H. BAKALLA, «Arabic Language Through Its Language and Literature,» chez *Kegan Paul*, London, 2002.

[8] K. A. R. H. SHAALAN, «Arabic named entity recognition from diverse text types,» chez *6th International Conference on Natural Language Processing (GoTAL'08)*, B. Nordstr¨om, and A. Ranta, Eds, 2008.

[9] L. A. C. M. E. LARKEY, «Arabic information retrieval,» chez *10th Text Retrieval Conference (TREC'01)*, The University of Massachusetts Amherst, 2001.

[10] A. A. S. J. FARGHALY, «Intuitive coding of the Arabic lexicon.,» chez *the MT Summit IX*, Americas, 2003.

[11] A. FARGHALY, «Introduction in Arabic computational linguistics,» chez *CSLI Publications*, Stanford, CA., 2010.

[12] A. F. G. N. R. Z. Soudi Abdelhadi, Challenges for Arabic Machine Translation, Amsterdam/Philadelphia: John Benjamins Publishing Company, 2012.

[13] S. P. S.-B. P. Laith H. Baniata, «A Neural Machine Translation Model for Arabic Dialects That Utilizes Multitask Learning (MTL),» *Research Article,* pp. 1-12, 10 Décembre 2018.

[14] L. B. S. M. R. H. a. L. S. D.J. Arnold, «Machine Translation: an Introductory Guide,» London, 1993.

[15] W. Aransa, «Statistical Machine Translation of the Arabic Language,» 2016.

[16] N. &. H. K. Ghezaiel, «Study and Resolution of Arabic Lexical Ambiguity Through Transduction on Text Automaton,» *Communications in Computer and Information Science,* 2016.

[17] D. J. a. J. H. Martin., Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition (Prentice Hall Series in Artificial Intelligence), Prentice Hall, 2000.

[18] R. Zbib et I. and Badr., «Preprocessing for English-to-Arabic statistical machine translation,» chez *Challenges for Arabic Machine Translation*, Amesterdam, 2012.

[19] A. El Kholy et N. and Habash, «Orthographic and morphological processing for English-Arabic statistical machine translation,» chez *Machine Translation*, 2012.

[20] N. Habash, «Syntactic preprocessing for statistical machine translation,» chez *MT Summit XI*, 2007.

[21] R. D. Y. Sarikaya, «Joint morphological-lexical language modeling for machine translation,» chez *Human Language Technologies : The Conference of the North American Chapter of the Association for Computational Linguistics*, 2007.

[22] A. C. Y. Eisele, «Multiun: A multilingual corpus from united nation documents,» chez *LREC*, 2010.

[23] A. A.-B. M. K. A. E. R. D. M. H. N. P. M. R. O. R. R. Pasha, «Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic.,» chez *Proceedings of the 9th International Conference on Language Resources and Evaluation*, Reykjavik, Iceland, 2014.

[24] K. K. D. M. C. S. Y. Toutanova, «Feature-rich part-ofspeech tagging with a cyclic dependency network.,» chez *the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, vol. 1, pp. 173–180. Association for Computational Linguistics*, 2003.

[25] E. Charniak, «A maximum-entropy-inspired parser,» chez *the 1st North American chapter of the Association for Computational Linguistics*

Conference, pp. 132–139. Association for Computational Linguistics, 2002.

[26] S. e. a. Andreas, «Srilm-an extensible language modeling toolkit,» chez *INTERSPEECH*, 2002.

[27] P. koehn, Statistical Machine Translation, Cambridge: Cambridge University Press, 2009.

[28] F. Sadat et N. and Habash, «Combination of Arabic preprocessing schemes for statistical machine translation,» , *In Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL, New York, USA: Association for Computational Linguistics,* pp. 49-52, 2006.

[29] S. Mansour, «MorphTagger : HMM-based Arabic segmentation for statistical machine translation,» *Proceedings of the 7th Internationa Workshop on Spoken Language Translation, Paris, France,,* pp. 321-327, 2010.

[30] A. El Kholy et N. and Habash, «Orthographic and morphological processing for English-Arabic statistical machine translation. Machine Translation,» pp. 25-45, 2012.

[31] P. Koehn, A. Axelrod, A. Mayne, C. Callison-Burch, M. Osborne et D. and Talbot, «Edinburgh system description for the 2005 IWSLT speech translation evaluation,» *Proceedings of the International Workshop on Spoken Language Translation,* pp. 68-75, 2005.

[32] T. Alkhouli, A. Guta et H. and Ney, «Vector space models for phrasebased machine translation, In Proceedings of SSST-8,,» *Eighth*

Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar: Association for Computational Linguistics,, 2014.

[33] Y. Marton, D. Chiang et P. and Resnik, «Soft syntactic constraints for Arabic-English hierarchical phrase-based translation. Machine Translation,,» pp. 137-157, 2012.

[34] M. Huck, D. Vilar, D. Stein et H. and Ney, «Advancements in Arabicto-English hierarchical machine translation,» chez *Preoceeding of the 15th Conference of the European Association for Machine Translation,*, Leuven,Belgium: European Association for Machine Translation, 2011.

[35] M. Khalilov et J. and Fonollosa, «Syntax-based reordering for statistical machine translation. Computer Speech and Language,» 2011.

[36] M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang et I. and Thayer, «Scalable inference and training of context-rich syntactic translation models,» chez *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, Association for Computational Linguistics*, 2006.

[37] ««Al Dhakhira Al Arabiya» est un projet ouvert sur l'avenir,» 23 Novembre 2010. [En ligne]. Available: http://www.elmoudjahid.com/fr/actualites/5730. [Accès le 10 Juin 2019].

[38] «Présentation du centre,» CRSTDLA, [En ligne]. Available: http://www.crstdla.dz. [Accès le 10 Juin 2019].

[39] «Opus,» Open source Coprora Collection 01 06 2019. [En ligne]. Available: http://opus.nlpl.eu/.

[40] S. S. A. Karunesh Kumar Arora, «Pre-Processing of Corpus for Statistical Machine Translation,» 06 juillet 2017. [En ligne]. Available: http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1405-55462017000400725&lng=pt&nrm=iso#aff2. [Accès le 10 Juin 2019].

[41] «Preparing Training Data,» Moses-SMT, 14 juillet 2006. [En ligne]. Available: http://www.statmt.org/moses/?n=FactoredTraining.PrepareTraining. [Accès le 10 06 2019].

[42] F. S. Nizar Habash, «Arabic Preprocessing Schemes for Statistical Machine Translation».

[43] C. M. Spence Green, «Arabic Natural Language Processing,» The Stanford Natural Language Processing Group, [En ligne]. Available: https://nlp.stanford.edu/projects/arabic.shtml. [Accès le 10 Juin 2019].

[44] J. Brownlee, «A Gentle Introduction to Calculating the BLEU Score for Text,» 20 Novemebre 2017. [En ligne]. Available: https://machinelearningmastery.com/calculate-bleu-score-for-text-python/. [Accès le 15 Juin 15].

[45] M. Thoma, «Word Error Rate Calculation,» 15 Novembre 2013 . [En ligne]. Available: https://martin-thoma.com/word-error-rate-calculation/. [Accès le 15 Juin 2019].

[46] J. Wolfe, «A Brief History of Python,» 05 Mars 2018. [En ligne]. Available: https://medium.com/@johnwolfe820/a-brief-history-of-python-ca2fa1f2e99e. [Accès le 24 06 2019].

[47] «Python Features,» 2018. [En ligne]. Available: https://www.javatpoint.com/python-features. [Accès le 24 juin 2019].

[48] Michael Kennedy, «PyCharm IDE,» JetBrains, [En ligne]. Available: https://www.jetbrains.com/pycharm/. [Accès le 24 Juin 2019].

[49] T. Rascia, «How to Create and Use Bash Scripts,» 29 Mai 2018. [En ligne]. Available: https://www.taniarascia.com/how-to-create-and-use-bash-scripts/. [Accès le 24 juin 2019].

[50] «Moses,» 24 juin 2018. [En ligne]. Available: http://www.statmt.org/moses/. [Accès le 07 07 2019].

[51] S. B. Richard Sproat, «Text Normalization,» September 2011. [En ligne]. Available: http://www.csee.ogi.edu/~sproatr/Courses/TextNorm/. [Accès le 10 Juin 2019].